Efficient GSM Authentication and Key Agreement Protocols with Robust User Privacy Protection

Jing-Lin Wu, Wen-Shenq Juang and Sian-Teng Chen
Department of Information Management
Shih Hsin University
No. 1, Lane 17, Sec. 1, Muja Rd., Wenshan Chiu,
Taipei, Taiwan, 116, R.O.C
wsjuang@cc.shu.edu.tw

Abstract

Nowadays, GSM is used widely by people around the world. However, there are also some problems of GSM authentication to be found. In 2004, Choi et al. proposed an authentication scheme with user privacy protection in GSM. They claimed that their scheme can improve some drawbacks of GSM authentication and achieve an ability of user privacy protection. But we point out that Choi et al.'s scheme is not able to achieve privacy and is not able to resist some well-known attacks completely. Hence, we propose a more efficient GSM authentication protocol with robust identity privacy protection. Our scheme also can remedy all drawbacks of GSM authentication mentioned by Choi et al. and resist to well-known attacks.

Keywords: Identity privacy, Mutual authentication, Network security, Mobile security

1. Introduction

Recently, the wireless networking becomes more popular and convenient to us. No matter where people travel to, he can use services provided by service providers. Radio interface and wireless network access are two territories, where the same level of protection as wired networks must be provided, in wireless communications. When an illegal user enters the territory of the radio interface of a network provider, he is easy to intercept the transferred messages over this radio interface and the sensitive information of the legal user could be exposed to the adversary. Besides, the adversary can pretend a legal user to access wireless network services. These problems may cause bill controversy among a mobile user, network providers and service providers. In order to avoid these problems, a secure authentication protocol must be established before users use services.

Since the Global System for Mobile communications (GSM), known as second-generation digital cellular system (2G), was proposed in several years ago, it has been widely utilized around the world so far [5]. GSM authentication key agreement (GSM AKA) is based on a challenge-response mechanism, but this mechanism can not achieve mutual authentication. VLR can easily authenticate MS by the assistance of HLR, but MS can not authenticate VLR since the random challenge is only generated by HLR. In addition to mutual authentication, some drawbacks of GSM AKA were pointed out and many improved schemes [3][4][9][10] have been proposed to overcome weaknesses of GSM AKA. Choi *et al*. proposed GSM AKA scheme to overcome some problems of GSM AKA in 2004 [4], but we point out that some security problems are still not solved in Choi *et al.*'s scheme. In this paper, we propose efficient and robust identity privacy GSM AKA schemes to improve Choi *et al.*'s scheme.

The remainder of this paper is organized as follows. In section 2, we review Choi *et al*.'s GSM AKA protocol. In section 3, we describe some problems of Choi *et al.*'s AKA scheme. In section 4, we show our proposed GSM AKA scheme with robust identity privacy protection. In section 5, we make a comparison of the efficiency and security among our scheme and the other related schemes. In section 6, we make a discussion. Finally, we make a conclusion.

2. Related works

Before illustrating Choi *et al.*'s authentication protocol [5], the notations must be demonstrated. HLR and VLR represent the home location register and the visitor location register, respectively. IMSI and TMSI represent the international mobile subscriber identity and the temporary mobile subscriber identity. LAI represents location area identifier. *A*3(), *A*5() and *A*8() are three main cryptographic algorithms [5], where *A*3() indicates an authentication algorithm, *A*5() indicates an encryption/decryption algorithm and *A*8() indicates a cipher key generation algorithm. $E_k()$ denotes the encryption function with the symmetric key *k* via the *A*5 algorithm. $D_k()$ denotes the decryption function with the symmetric key k via the *A*5 algorithm. ID_{HLR} denotes a unique identification of HLR, ID_{VLR} denotes a unique identification of VLR, and *f*() denotes a one-way hash function. *K* represents the common shared key between HLR and MS and *Kc* denotes the cipher key. "||" represents the string concatenation symbol and "$\oplus$" denotes the bitwise exclusive-or operation.

Choi *et al.*'s proposed the GSM AKA scheme with privacy is based on the Alias (*AL*), where *AL* is a unique identity of MS for traveling and assigned to the IMSI of MS one-by-one. HLR assigns this *AL* to a user when

MS registration. The relationship between the alias and the real identity should be kept secretly by HLR. This scheme occurs while MS receiving an identity request at location updating. We demonstrate this scheme as follows.

Step 1: When MS receiving an identity request from VLR, MS will extract *AL* from its database instead of IMSI, chooses a random number *RAND*, computes a response *SRES*1=*A*3(*RAND*||*K*), another encryption key K_u=*f*(IMSI||ID_{HLR}||*K*) and an encrypted message E_{K_u}(*RAND*) via the *A*5 as the encryption algorithm. Then, MS sends *AL*, ID_{HLR}, *SRES*1 and E_{K_u}(*RAND*) to VLR.

Step 2: After receiving *AL*, ID_{HLR}, *SRES*1 and E_{K_u}(*RAND*), VLR identifies the identity of HLR, derives the corresponding shared key *VH* and generates another encrypted message E_{VH}(*AL*). Then, VLR sends ID_{VLR}, E_{VH}(*AL*) and E_{K_u}(*RAND*) to HLR.

Step 3: While receiving ID_{VLR}, E_{VH}(*AL*) and E_{K_u}(*RAND*), HLR identifies the identity of VLR and knows the shared key *VH*. HLR decrypts E_{VH}(*AL*) and gets the alias *AL* of MS. HLR can use the alias *AL* to find the corresponding IMSI in its key table and obtain shared key *K*. Then, HLR calculates the symmetric decryption key K_u=*f*(IMSI||ID_{HLR}||*K*) and decrypts E_{K_u}(*RAND*) by using the *A*5 as the decryption algorithm. HLR computes another response *SRES*2= *A*3(*RAND*||*K*) and an encrypted message E_{VH}(*RAND*|| *TK*||IMSI). Next, HLR sends *SRES*2, ID_{HLR} and E_{VH}(*RAND*|| *TK*||IMSI) back to VLR.

Step 4: Upon receiving *SRES*2, ID_{HLR} and E_{VH}(*RAND*||*TK*||IMSI), VLR first verifies if *SRES*1=*SRES*2. If the equation is satisfied, VLR decrypts E_{VH}(*RAND*||*TK*||IMSI) using the shared key *VH* and stores the temporary key *TK* in its database. Subsequently, VLR sends the random number *RAND* and encrypted message E_{TK}(TMSI$_{new}$) to MS, where E_{TK}() is the encryption algorithm by using the *A*5 algorithm with the temporary key *TK*.

Step 5: Once MS receives *RAND* and E_{TK}(TMSI$_{new}$), MS will verify the random number *RAND* is chosen by itself. If yes, MS computes the temporary key *TK*=*A*8(*RAND*||*K*) and decrypts E_{TK}(TMSI$_{new}$) by using the *A*5 algorithm and the temporary key *TK*. The process of authentication is accomplished.

3. Some problems of Choi *et al.*'s GSM AKA protocol

In this section, we point out that some drawbacks of Choi *et al*.'s GSM authentication protocol as follows.

(i) The redirection attack is to redirect MS's traffic to another domain or base station. Assume that an attacker is manipulating a device having the capability of a base station, called the false base station, and this attacker's device can also imitate the capability of a mobile station. In order to carry out functionality of two different devices, the special device can be used, *i.e*., IMSI catcher [12]. The attacker can imitate a base station and allure a legal mobile station to camp on the radio channels of the false base station impersonated by the attacker. On the other hand, the attacker can also imitate a mobile station and creates the connection with a pure base station. We point also out that Choi *et al.*'s GSM AKA protocol is easily attacked by the redirection attack problem [15] since the HLR can not check whether any information is sent from visited VLR by MS actually or sent from a false VLR. Besides, if one of networks is corrupted, the security of all networks will be jeopardized. The adversary can implement this attack, called the corrupted network attack, leading to a large damage for networks. According to [15], the redirection attack and the corrupted network attack can be solved by checking the validity of the authenticator and checking if the identity of VLR truly visited by MS is embedded in that authenticator.

(ii) Even though Choi *et al*.'s scheme uses alias *AL* to conceal the MS's identity, the adversary can still know which location a MS travels to since the adversary can send an identity request to many MS in different times or various locations. If the adversary wants to know MS's location and what he does, the adversary can easily know that by recoding the *AL* of MS even if the identity IMSI of MS is not exposed. So Choi *et al.*'s scheme is achieving weak identity privacy for MS.

(iii) The modification attack is to disturb the normal communications between both ends. Choi *et al*.'s scheme is vulnerable to the modification attack while VLR sends an encrypted message E_{TK}(IMSI$_{new}$) and a random number *RAND* to MS. If the correctness of *RAND* is checked by MS, MS will compute the temporary key *TK* and decrypt the encrypted message E_{TK}(IMSI$_{new}$). In this way, MS only checks the correctness of the random number *RAND* and MS then accepts the new temporary identity TMSI$_{new}$. The attacker can easily replay the same random number *RAND* and forge an encrypted message $E_{TK'}$(TMSI$_{new}$'). Then, the adversary can send the random number *RAND* and $E_{TK'}$(TMSI$_{new}$') to MS and MS will believes that TMSI$_{new}$' is produced by a genuine VLR for communications.

4. Our proposed scheme with robust user privacy protection

The system architecture of our scheme is the same with that of the original GSM authentication and key agreement protocol and also based on three cryptographic algorithms *A*3(), *A*5() and *A*8(), where *A*3() is an authentication algorithm, *A*5() is an encryption algorithm and *A*8() is an cipher key

generation algorithm [5]. Let x be a master secret key kept secretly by the HLR.

There are two situations for authentication when a MS wants to use the service including the normal case and the case while a MS receiving an identity request. Due to space consideration, we only describe our modified scheme for the case while a MS receiving an identity request in Figure 1. The normal case will be appeared in the full version of this paper. Now, we will describe our scheme as follows.

Step 1. VLR sends identity request including a random number N_3 to MS through the downlink channel.

Step 2. While MS receives the identity request and N_3, it firstly extracts the secret token $w_i=A3(x||r_i)$ and the random number r_i. Then, MS selects a random number *RAND*, and computes the authentication tag $VAC=f(K||ID_{VLR}||RAND||N_3||w_i)$ and an expected response $SRES1=A3(K||RAND)$. Then MS generates an encrypted IMSI by computing $P_i=\text{IMSI}\oplus w_i$. Next, MS sends P_i, ID_{HLR}, *SRES*1, *VAC*, *RAND* and r_i to VLR through the uplink channel.

Step 3. Upon receiving P_i, ID_{HLR}, *SRES*1, *VAC*, *RAND* and r_i, VLR stores *SRES*1 in its database and sends ID_{VLR}, P_i, *VAC*, *RAND*, N_3 and r_i to HLR via a secure channel.

Step 4. When receiving ID_{VLR}, P_i, *VAC*, *RAND*, N_3 and r_i, HLR firstly checks if the nonces *RAND* and N_3 are fresh. HLR can keep a recently used nonces table for checking freshness. If they are fresh, HLR calculates the secret token $w_i=A3(x||r_i)$ and gets IMSI by computing $P_i\oplus w_i=(\text{IMSI}\oplus w_i)\oplus w_i=\text{IMSI}$. Then, HLR can derive the shared key K and verifies if $VAC=f(K||ID_{VLR}||RAND||N_3||w_i)$. If they are not identical, HLR halts this connection. Otherwise, HLR computes a temporary key $TK=A8(K||RAND)$ and an expected response $SRES2=A3(K||RAND)$. HLR then selects a random number N_4 and another random number r_{i+1} for being used next time, and generates a secret token $w_{i+1}=A3(x||r_{i+1})$ to be used next time and a message of concealed secret token $T_{i+1}=w_{i+1}\oplus A3(K||r_{i+1})$. Next, HLR generates an authenticator $MAC=f(K||N_4||VAC||w_{i+1})$ and an encrypted message $E_{VH}(\text{IMSI}||TK||SRES2||T_{i+1}||MAC||r_{i+1})$, where *VH* is shared key with VLR. HLR sends N_4, $E_{VH}(\text{IMSI}||TK||SRES2||T_{i+1}||MAC||N_4||r_{i+1})$ to VLR.

Step 5. While receiving N_4, $E_{VH}(\text{IMSI}||TK||SRES2||T_{i+1}||MAC||N_4||r_{i+1})$, VLR decrypts the message by using the shared key *VH*. Then, it verifies if *SRES*1=*SRES*2. If they are not match, VLR aborts this connection. Otherwise, VLR keeps the temporary key *TK* in its database and computes another authenticator $AUTH=f(TK||\text{TMSI}_{new}||MAC)$, where TMSI_{new} is a new assigned temporary identity by VLR. Then, VLR sends *AUTH*, T_{i+1}, r_{i+1}, *MAC*, N_4 and an encrypted message $E_{TK}(\text{TMSI}_{new})$, where $E_{TK}()$ is using the *A*5 as the encryption algorithm with the temporary key *TK*.

Step 6. After receiving *AUTH*, T_{i+1}, r_{i+1}, *MAC*, N_4, and $E_{TK}(\text{TMSI}_{new})$, MS computes the secret token $w_{i+1}=T_{i+1}\oplus A3(K||r_{i+1})=w_{i+1}\oplus A3(K||r_{i+1}))\oplus A3(K||r_{i+1})$ to be used next time and verifies if $MAC=f(K||N_4||VAC||w_{i+1})$. If they are not match, MS halts this connection. Otherwise, MS stores the secret token w_{i+1} and the random number r_{i+1} for being used next time. If it is valid, MS computes the temporary key $TK=A8(K||RAND)$ and decrypts $E_{TK}(\text{TMSI}_{new})$ and gets TMSI_{new}. Next, MS verifies authenticators $MAC=f(K||N_4||VAC||w_{i+1})$ and $AUTH=f(TK||TMSI||MAC)$. If they are identical, HLR and VLR are authenticated by MS and the process of authentication is accomplished.

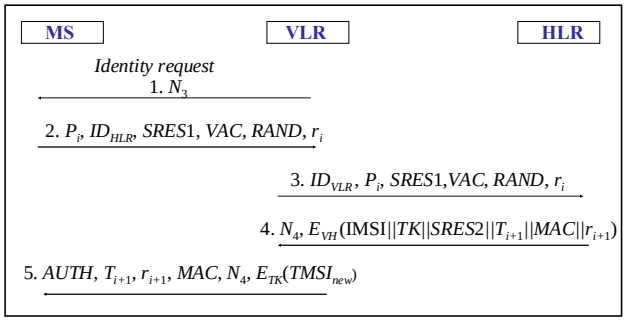

Figure 1. Our proposed GSM AKA authentication while MS receiving an identity request

5. Security analysis and performance consideration

5.1 Identity privacy

In [4], Choi *et al* claimed that their proposed GSM AKA scheme can achieve identity privacy. But the adversary can still know wherever location the same MS is while sending an identity request to the same MS via the alias *AL* during various times. Hence, we propose a more strong GSM AKA scheme with identity privacy protection by using the secret token $w_i=A3(x||r_i)$ and the random number r_i to protect IMSI. We generates the concealed message $P_i=\text{IMSI}\oplus w_i$ for achieving location privacy. Nobody, except HLR, can generate the secret token $w_i=A3(x||r_i)$. If the adversary eavesdrops the concealed message P_i and the random number r_i, the adversary is also impossible to obtain IMSI without the master key x kept secretly by HLR. Thus, compared with [4], our schemes provide more robust location privacy by using the secret token $w_i=A3(x||r_i)$.

5.2 Mutual authentication

The goal of mutual authentication is that MS and VLR establish an agreed temporary key *TK* and MS and VLR can authenticate each other with the assistance of HLR. In our scheme, we assume that the temporary key *TK* is a kind of session keys to be used for a valid period. Let $A \overset{TK}{\longleftrightarrow} B$ denote that A and B share a common session key *TK*. The mutual authentication is accomplished between A and B if there exists an *TK* such that A believes $A \overset{TK}{\longleftrightarrow} B$ and B believes $A \overset{TK}{\longleftrightarrow} B$ for the transaction [1][6][7][8]. A strong mutual authentication should include the following

statement [1][6][7][8]: A believes B believes $A \xleftrightarrow{TK} B$ and B believes A believes $A \xleftrightarrow{TK} B$. We can illustrate that our scheme can achieve strong mutual authentication between VLR and MS as follows.

After Step 4 of our proposed GSM AKA scheme in section 4.1, VLR receives the message N_4 and $E_{VH}(\text{IMSI}||TK||SRES2||T_{i+1}||MAC||N_4||r_{i+1})$ from HLR, VLR can decrypt this message and verify if $SRES1=SRES2$. If yes, VLR believes VLR $\xleftrightarrow{TK}$ MS. Since the random number N_3 is chosen by VLR, and confirms the freshness of N_3 embedded in *VAC*, also embedded in *MAC*, with HLR's assistance, VLR believes MS believes VLR $\xleftrightarrow{TK}$ MS.

After of Step 5 of our proposed scheme in section 4.1, MS receives *AUTH*, N_4 and $E_{TK}(\text{TMSI}_{new})$, MS computes the temporary key $TK=A8(K||RAND)$, decrypts $E_{TK}(\text{TMSI}_{new})$ and checks if $AUTH=f(TK||\text{TMSI}||MAC)$ is valid. If yes, MS will believes VLR $\xleftrightarrow{TK}$ MS since the random number *RAND* selected by MS is embedded in *VAC*, also embedded in *MAC*. Since MS can also confirm if the random number *RAND* is fresh, and $AUTH=f(TK||\text{TMSI}||MAC)$ can only be calculated by HLR and sent to VLR, MS believes VLR believes VLR $\xleftrightarrow{TK}$ MS.

5.3 Secret token protection

For achieving identity privacy, our scheme uses the secret token $w_i=A3(x||r_i)$ to conceal IMSI as a message $P_i=\text{IMSI}\oplus w_i$ and MS sends this concealed message to HLR for anonymous authentication. If an adversary wants to know IMSI of MS, it must firstly get the secret token $w_i=A3(x||r_i)$. Even though the random number r_i and the concealed message $P_i=\text{IMSI}\oplus w_i$ are known by the attacker, the secret token $w_i=A3(x||r_i)$ is impossible to derive since the master secret key x is only kept secretly by HLR, and only HLR can generate the secret token $w_i=A3(x||r_i)$. The secret token w_i is computed as $w_i=A3(x||r_i)$, where r_i is ith random number selected by HLR.

After HLR accepts the request of the anonymous authentication by using the secret token mechanism, HLR will generate a new secret token $w_{i+1}=A3(x||r_{i+1})$ to be used next time and the random number r_{i+1} to be used next time. Later, the new secret token w_{i+1} will be sent secretly to MS by using another concealed message $T_{i+1}=w_{i+1}\oplus A3(K||r_{i+1})$, where K is shared key between HLR and MS. As soon as MS receives the conceal message T_{i+1} and the random number r_{i+1}, MS can derive the new secret token $w_{i+1}=T_{i+1}\oplus A3(K||r_{i+1})$ to be used next time, and store this new secret token w_{i+1} and the random number r_{i+1} for next time use. Except HLR, nobody can calculate the new secret token $w_{i+1}=T_{i+1}\oplus A3(K||r_{i+1})$. Even if the adversary taps the concealed message P_{i+1} and the random number r_{i+1} while location updating phase happens again, it is still difficult to compute the new secret token w_{i+1} since the adversary does not has the shared key K.

5.4 Withstanding attacks

(i) Man-in-middle attack [11]

The man-in-middle attack means that the attacker tries to modify the content of communications and is not to be observed by both ends of communications. The adversary tries to get the transmitting messages between both ends, and replaces a modified message with the transmitting message. This attack can be resisted in our scheme since both ends can verify whether a message is modified or not by checking the authenticator. If the message is modified, the receiver will reject it immediately.

(ii) Dictionary attack [2]

For calculating the temporary key *TK*, the adversary must know the random number *RAND* and the shared key *K*. Even if the random number *RAND* is transmitted on plain text, and it is got by the adversary. It is impossible to calculate the temporary key *TK* since the shared key *K* is kept secretly by HLR and MS. Only MS and HLR have the shared key *K*. The adversary can not compute the temporary key *TK* since the high entropy of the shared key *K*.

(iii) Replay attack [13]

For prevent this attack, our scheme uses nonces N_3, N_4 and *RAND* to resist to the replay attack. HLR can check if the nonce is used in its recently used nonce table. If the message is replayed by the adversary, the receiver can observe the replayed message and reject it.

(iv) Modification attack [14]

The modification attack is to disturb normal communication between both ends. Our scheme can resist this pixilated attack since our schemes use authenticators to verify if the message is modified by the attack. Even if the message is altered by the attacker, the receiver will check the correctness of the authenticator. If it not correct, the receiver will reject it.

5.5 Efficiency comparisons

In this section, we make efficiency comparison with related schemes. According to [5], there are three cryptographic algorithms, *A*3(), *A*5() and *A*8() used in GSM authentication. *A*3() is the authentication algorithm to generate message authentication code and the output parameters of *A*3() is 32 bits. *A*8() is the cipher key generation algorithm and the out parameters of *A*8() is 64 bits. *A*5() is the encryption algorithm. We also assume that the master secret key x and random number *RAND* is 128 bits, the shared key *VH* is 64 bits, secret token $w_i=A3(x||r_{i+1})$ is 32 bits and the temporary key $TK=A8(K||RAND)$ is 64 bits. Beside, we also assume that *A*3() algorithm and *A*8() algorithm are similar to hash operation and *A*5() algorithm resemble a symmetric encryption/decryption operation. Also, we assume that the new assigned TMSI is encrypted by VLR and sent to HLR. The computation cost of encryption and decryption for the new assigned TMSI is also considered.

Efficiency comparison of our scheme and related schemes [3][4][5][10] while MS receiving an identity

request is shown in Table 1. In our scheme, the memory needed for MS is 352 bits. In [3] and [10], the memory needed for MS is 128 bits and 832 bits, respectively. The memory needed for MS in [4] and [5] is 192 bits. The memory needed for VLR is 160 bits in our scheme and [4]. In [3] and [10], the memory needed for VLR is 146 bits. The memory required for VLR is (224×*n*) bits in [5]. In our scheme, the memory needed for HLR is 320 bits. In [3] and [5], the memory needed for HLR is 128 bits. In [4], the memory needed for HLR is 192 bits. The memory needed for HLR is 640 bits in [10].

The computation cost for MS is one decryption operation, six hash operations and two exclusive-or operations in our scheme. The computation cost of MS is two encryption operations and two hash operations in [3]. In [4], the computation cost of MS is one encryption operation, one decryption operation and three hash operations. The computation cost for MS in [10] is one exponential operation, one encryption and three hash operations. The computation cost for MS in [5] is one decryption operation and two hash operations. In our scheme, the computation cost of VLR is one encryption operation, one decryption operation and one hash operation. Two encryption operations and one decryption operation is required for VLR in [4]. In [3] and [10], the computation cost for VLR is two encryptions and one encryption operation, respectively. Only one encryption operation is needed for VLR in [5]. The computation cost on HLR in our scheme is one encryption operation, seven hash operations and two exclusive-or operations. The computation cost of HLR in [3] is two hash operations. However, in [4], the computation cost for HLR is one encryption operation, two decryption operations and three hash operations. In [10], one exponential operation and two hash operations are needed for HLR. However, (2×*n*) hash operations are required for HLR in [5].

Table 1. Efficiency comparison among our 3GPP AKA scheme and the other related schemes while MS receiving an identity request

	E1	E2	E3	E4	E5	E6
Our scheme	352 bits	160 bits	320 bits	1Sym+6H+2 XOR	2Sym+1H	1Sym+7H +2XOR
Chang *et al.* [3]	128 bits	146 bits	128 bits	2Sym+2H	2Sym	2H
Choi *et al.* [4]	192 bits	160 bits	192 bits	2Sym+3H	3Sym	3Sym+3H
Peinado [10]	832 bits	146 bits	640 bits	1Exp+1Sym +3H	1Sym	1Exp+2H
GSM [5]	192 bits	(224 × *n*) bits	128 bits	1 Sym + 2 H	1Sym	(2 × *n*)H

E1: Memory needed in MS; E2: Memory needed in VLR; E3: Memory needed in HLR; E4: Computation cost for MS; E5: Computation cost for VLR; E6: Computation cost for HLR; Exp: Exponential operation; Sym: Symmetric encryption/decryption operation; H: Hash operation; XOR: Exclusive-or operation; *n*: numbers of authentication vectors.

Note that the encryption of TMSI using the encryption key Kc_i through the *A*5 algorithm on the VLR's side and the decryption of TMSI using the decryption key Kc_i through the *A*5 algorithm on the MS's side are included in the computation cost in the related GSM AKA protocols for the comparison.

We summarize the functionality of our scheme and the related schemes in Table 2. In compared to Chang *et al.*'s scheme [3], it is not able to provide identity privacy, and not able to withstand the redirection attack and the corrupted network attack. Besides, the modification attack on Chang *et al.*'s scheme is not available (N/A) since their scheme assumes that the new TMSI is already assigned to MS before the authentication phase. Compared with Choi *et al.*'s scheme [4], Choi *et al.*'s scheme is not able to prevent the redirection attack, the corrupted network attack, the replay attack and the modification attack. In addition, Choi *et al.*'s scheme only achieves weak identity privacy. Compared with [10], even though their scheme provides identity privacy but their scheme has not great performance since using public key cryptography. Beside, Peinado's scheme [10] can not resist to the redirection attack and the corrupted network attack, and have time synchronization problem. Note that the modification attack on encrypted TMSI is not available (N/A) since Peinado's scheme uses an encrypted ticket from HLR to replace with a new assigned TMSI by the visited VLR. Regarding to computation or communication cost, since using the temporary key mechanism, our scheme and other related schemes [3][4] are more lower than the schemes [5][10]. Our scheme satisfies all property of the listed and has relatively great performance.

Table 2. Functionality comparison among our GSM AKA and the other related GSM AKA schemes

	Our scheme	Chang *et al.* [3]	Choi *et al.* [4]	Peinado [10]	GSM [5]
C1	Low	Low	Low	High	Low
C2	Low	Low	Low	High	High
C3	Low	Low	Low	Low	Low
C4	Low	Low	Low	Low	High
C5	Low	Low	Low	Low	High
S1	Yes	No	Partial	Yes	No
S2	Yes	Yes	Yes	Yes	No
S3	Yes	Yes	No	Yes	No
S4	Yes	No	No	No	No
S5	Yes	No	No	No	No
S6	Yes	N/A	No	N/A	No
S7	Yes	Yes	Yes	No	Yes

C1: The computation cost for MS; C2: The computation cost for HLR; C3: The computation cost for VLR; C4: The communication cost between HLR and VLR; C5: The space overhead for VLR; S1: Identity privacy; S2: Mutual authentication between MS and VLR; S3: Preventing the replay attack S4: Preventing the redirection attack; S5: Preventing the corrupted network attack; S6: Preventing the modification attack while VLR assigns a new TMSI; S7: No time synchronization problem.

6. Discussion

In this section, we discuss our proposed schemes for more detailed considerations in advance. Instead of using the exclusive-or operation, we also provide more robust identity privacy of MS in GSM AKA protocols. Besides, we focus on the lifetime of the temporary key and make more detailed demonstrations.

In our proposed GSM AKA protocol, we use the secret token $w_i = A3(x||r_i)$ to protect IMSI of MS by computing the message P_i=IMSI$\oplus w_i$. First of all, for convenience to demonstrate, we assume that the approach of checking the authenticator is ignored. Instead of using P_i=IMSI$\oplus w_i$, MS generates an secret token w_i=$A3(x||r_i)$ as a symmetric encryption key and an encrypted message $P_i = E_{w_i}$ (IMSI), where $E_y()$ is a

symmetric decryption function and y is a symmetric encryption key as an input. Then, MS sends P_i and r_i to HLR, Once HLR receives P_i and r_i, HLR generates the symmetric decryption key w_i and decrypts the identity IMSI= $D_{w_i}(P_i)$, where $D_y()$ is a symmetric decryption function and y is the symmetric decryption key.

For protecting the new transferred secret token $w_{i+1}=A3(x||r_{i+1})$, HLR will generates another symmetric encryption key $K_{st}=A3(K||r_{i+1})$ and encrypts $T_{i+1}= E_{K_{st}}(w_{i+1})$, where K is shared key of MS and HLR. Then, HLR sends the random number r_{i+1} to be used the next time and the secret token T_{i+1} to be used the next time to MS. After receiving r_{i+1} and T_{i+1}, MS can obtain w_{i+1} by decrypting $D_{K_{st}}(T_{i+1}) = D_{A3(K||r_{i+1})}(T_{i+1})$. Since only HLR can generate the correct secret token $w_{i+1}=A3(x||r_{i+1})$ for the next time use and MS will believes the new received secret token w_{i+1} is valid. MS will store r_{i+1} and w_{i+1} in its memory, and use them while MS receiving an identity request next time. Either using the exclusive-or operation or a symmetric cryptosystem to be applied in our proposed scheme, the identity privacy can be implemented in our proposed scheme. Note that using symmetric cryptosystems have higher overheads than exclusive-or operations in MS and HLR. For the performance consideration, our proposed scheme adopts the exclusive-or operation to enforce identity privacy. For achieving more robust identity privacy, TMSI can also apply the above mentioned approaches to be concealed but space overhead and computation cost will be increasing in MS and VLR.

In our proposed GSM scheme, the lifetime of the generated temporary key *TK* is not considered. For specifying the lifetime of each generated *TK*, HLR can send the lifetime $LifeTime_{TK}$ of *TK* to VLR. Also, VLR will forward the $LifeTime_{TK}$ of *TK* to MS. For simply illustration, we only specify the time life of the temporary key used in our GSM AKA protocol at normal case. The authenticator $MAC=f(K||N_2||VAC||w_{i+1})$ be replaced with $MAC=f(K||N_2||VAC||w_{i+1}||TimeLife_{TK})$ in Step 4 of Section 4.1.

7. Conclusion

In this paper, we have proposed a GSM authentication scheme with robust identity privacy protection. In order to achieve the user's identity privacy, only using alias, used in Choi *et al.*'s scheme, is not enough since the MS's location is still easy to be exposed by the location privacy attack. Hence, we use the exclusive-or operation to produce dynamically a secret token for achieving robust identity privacy protection. Besides, we also use the temporary key mechanism for reducing bandwidth consumption. Our proposed scheme can also resist well-known attacks.

Acknowledgment. This work was supported in part by the National Science Council of the Republic of China under the Grant NSC 95-2221-E-128-004-MY2, and by the Taiwan Information Security Center (TWISC), National Science Council under the Grants NSC 95-3114-P-001-001-Y02 and NSC 94-3114-P-011-001.

References

[1] M. Burrow, M. Abadi and R. Needham, "A Logic of Authentication," *ACM Trans. Comput. Syst.*, Vol. 8, pp. 18-36, 1990.

[2] S. Bellovin and M. Merritt, "Encrypted Key Exchange: Password-Based Protocol Secure Against Dictionary Attacks," Research in Security and Privacy, Proceedings IEEE Computer Society Symposium, pp. 72-84, 1992.

[3] C. Chang, J. Lee and Y. Chang, "Efficient Authentication Protocols of GSM," *Computer Communications*, Vol. 28, pp. 921-928, 2005.

[4] Y. Choi and S. Kim, "An Improvement on Privacy and Authentication in GSM," Proceedings of Workshop on Information Security Applications (WISA'2004), pp. 14-26, 2004.

[5] Digital Cellular Telecommunications System (Phase 2+), Security Related Network Functions (GSM 03.20 version 8.1.0 Release 1999), ESTI TS 100 929 V8.1.0, 2001.

[6] W. Juang, "Efficient Multi-server Password Authenticated Key Agreement Using Smart Cards," *IEEE Trans. on Consumer Electronics*, Vol. 50, No. 1, pp. 251-255, 2004.

[7] W. Juang, "Efficient Password Authentication Key Agreement Using Smart Cards," *Computer and Security*, Vol. 23, pp. 167-173, 2004.

[8] W. Juang, "Efficient User Authentication and Key Agreement in Ubiquitous Computing," In Proceeding of the 2006 International Conference Computational Science and is Applications, Lecture Notes in Computer Science 3983, pp. 396-405, Springer-Verlag Press, German, 2006.

[9] C. Lee, M. Hwang and W. Yang, "Extension of Authentication Protocol for GSM," *IEE Proceedings of Communications*, Vol. 150, pp. 91-95, 2003.

[10] A. Peinado, "Privacy and Authentication Protocol Providing Anonymous Channels in GSM," *Computer Communications*, Vol. 27, pp. 1709-1715, 2004.

[11] D. Seo and P. Sweeney, "Simple Authenticated Key Agreement Algorithm," *Electronics Letters*, Vol. 35, pp. 1073-1074, 1999.

[12] J. Qurike, "Security in the GSM System," AusMobile, http://www.ausmobile.com.

[13] P. Syverson, "A Taxonomy of Replay Attacks," Computer Security Foundations Workshop VII, CSFW 7. Proceedings 14-16, pp. 187-191, 1994.

[14] C. Yang, T. Chang and M. Hwang, "Cryptanalysis of Simple Authenticated Key Agreement Protocols," *IEICE Trans. Fundamentals*, Vol. E87-A, No. 8, pp. 2174-2176, 2004.

[15] M. Zhang and Y. Fang, "Security Analysis and Enhanced of 3GPP Authentication and Key Agreement Protocol," *IEEE Trans. on Wireless Commun.*, Vol. 4, No. 2, pp. 734-742, 2005.