RESEARCH ARTICLE

A practical authentication protocol with anonymity for wireless access networks

Yen-Cheng Chen1*, Shu-Chuan Chuang1, Lo-Yao Yeh2 and Jiun-Long Huang2

1 Department of Information Management, National Chi Nan University, Puli, NanTou 545, Taiwan
2 Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan

ABSTRACT

The use of anonymous channel tickets was proposed for authentication in wireless environments to provide user anonymity and to probably reduce the overhead of re-authentications. Recently, Yang et al. proposed a secure and efficient authentication protocol for anonymous channel in wireless systems without employing asymmetric cryptosystems. In this paper, we will show that Yang et al.’s scheme is vulnerable to guessing attacks performed by malicious visited networks, which can easily obtain the secret keys of the users. We propose a new practical authentication scheme not only reserving the merits of Yang et al.’s scheme, but also extending some additional merits including: no verification table in the home network, free of time synchronization between mobile stations and visited networks, and without obsolete anonymous tickets left in visited networks. The proposed scheme is developed based on a secure one-way hash function and simple operations, a feature which is extremely fit for mobile devices. We provide the soundness of the authentication protocol by using VO logic.

1. INTRODUCTION

Due to the popularity of wireless communications, there is an increasing demand for secure access to wireless networks via mobile devices. A mobile user usually accesses a wireless network via an association with the nearest network access point. These wireless associations should be authenticated for preventing unauthorized access from wireless networks. Since mobile users may move freely and may occasionally access networks when in need, the associations among users and access points may change dynamically. Due to unstable wireless signals or the temporary power-saving sleep of mobile devices, associations may also be discontinued and resumed frequently within a short period of time. If re-authentication is required whenever a suspended association is to be resumed, much overhead will be incurred for authentication. In addition, when a mobile user roams to a far visited network, the visited network will spend a longer round trip time in authenticating the user through the authentication server located in the home network of the user. Therefore, these practical issues are usually taken into account in the development of authentication schemes for wireless environments. Another vital security issue in wireless networks is the protection of user privacy. During an authentication process, it is usually required to present the identity of a user in authentication messages. The user identity may reveal the current location of a certain user. This information may intrigue malicious intruders. Therefore, for use in wireless networks, authentication schemes preserving user anonymity are preferred.

In recent years, many authentication schemes have been developed to consider security issues particular to wireless networks [1–10]. Park-Go-Kim’s authentication and key agreement protocol [3] used the temporary identity (TID) of a mobile user instead of its real one for providing user anonymity. Lin and Jan [4] proposed an authentication scheme with user anonymity based on the use of hash functions and
smart cards. Their scheme achieves user anonymity by using a hashed token to protect user’s identity. However, we find that all users of the same home agent share the same hashed token. A malicious user can make use of the hashed token to get the identities of other users. Thus, Zhu-Ma’s authentication scheme fails to preserve user anonymity. Besides, Zhu-Ma’s scheme also fails to prevent ticket copies for malicious users intentionally setting same session keys in each session. Recently, Yang et al. [10] proposed a secure and efficient protocol for anonymous channel in wireless systems without employing asymmetric cryptosystems. Yang et al.’s scheme achieves user anonymity by using symmetric cryptosystem for authenticating each user. Another merit of this scheme is the prevention of ticket copy. In this paper, we will show that Yang et al.’s scheme is vulnerable to guessing attacks performed by malicious visited networks. A visited network can successfully guess the secret keys of the users who are visiting it. To withstand the proposed attack, we will develop a new authentication scheme preserving the same merits of Yang et al.’s scheme. The proposed scheme also provides several enhancements. In the proposed scheme, it is not necessary to store a verification table in the home network, which is the perfect solution to the stolen verifier problem. Moreover, different from previous approaches using timestamps, the proposed scheme uses nonces to prevent replay attacks, because in practice it is difficult for a mobile station to present a synchronized timestamp before the mobile station is granted to access a visited network. The proposed scheme also provides anonymous tickets but limits the use of anonymous tickets by expired times, instead of the number of logins. This can further prevent storing obsolete tickets in visited networks.

The rest of this paper is organized as follows. Section 2 is a brief overview of Yang et al.’s scheme. In the end of Section 2, we will present attacks and comments on Yang et al.’s scheme. Then, a new practical authentication scheme is proposed in Section 3. In Section 4, we show the security and performance analysis of the proposed scheme. A conclusion is given in Section 5. Finally, we will further prove the correctness of the proposed scheme by the logic of authentication in Appendix.

2. REVIEW OF YANG ET AL.’S SCHEME

In this section, Yang et al.’s protocol is reviewed. Then, we present our attacks and comments on Yang et al.’s scheme. The notations used in Yang et al.’s protocol are as follows. Three entities are involved in the protocol: a mobile station MS, a visited network VN, and a home network HN. IDi, IDVN, and IDHN denote the identity of MS, VN, and HN, respectively. It is assumed that HN and VN share a secret key kIDi,j, and HN and MS share a secret key kIDi,A: “X → Y: M” denotes that X sends message M to Y. (M)k denotes that ciphertext of the message M encrypted with the symmetric key k, and “⊕” indicates the bit-wise XOR operation. And, p is a large prime, Q is a prime factor of p−1 and g is an element of order Q in Z∗p. Yang et al.’s protocol consists of two phases, described as follows.

2.1. Yang et al.’s scheme

2.1.1. The anonymous channel issuing phase.

In this phase, MS purchases an anonymous ticket from HN via VN before MS is granted to access VN.

Step 1. MS → VN: IDHN, kIDi,j mod p, (IDi, T1, A, B, C)h,i

MS selects three random numbers a, b, and c to compute A = g^a mod p, B = g^b mod p, and C = g^c mod p. MS then uses kIDi,j to encrypt message (IDi, T1, A, B, C), where T1 is a timestamp. The encrypted message along with IDHN and kIDi,j mod p is then sent to VN.

Step 2. VN → HN: IDVN, kIDi,j mod p, (IDi, T1, A, B, C)h,i, (IDVN, T2, D, E, F)h,a

VN selects three random numbers d, e, and f to compute D = g^d mod p, E = g^e mod p, and F = g^f mod p. VN then uses kIDi,j to encrypt message (IDVN, T2, D, E, F), where T2 is a timestamp. Then VN sends IDVN, kIDi,j mod p, (IDi, T1, A, B, C)h,i, and (IDVN, T2, D, E, F)h,a to HN.

Step 3. HN → VN: kIDi,j ⊕ (T2, A, B, C, Texpire), kIDi,j ⊕ (T1, D, E, F, Texpire)

HN first records the received time T3 and accords kIDi,j mod p to find IDi and kIDi,A. HN then uses kIDi,j and kIDi,B to extract (IDi, T1, A, B, C) and (IDVN, T2, D, E, F), respectively. HN then checks whether T1 − T2 ≤ ΔT and IDHN is valid, where ΔT denotes a valid time interval. If yes, HN successfully authenticates VN. Similarly, HN authenticates MS by checking whether T2 − T1 ≤ ΔT and IDi is valid. Then, HN sends kIDi,j ⊕ (T2, A, B, C, Texpire) and kIDi,j ⊕ (T1, D, E, F, Texpire) to VN, where Texpire denotes the maximum login numbers of MS.

Step 4. VN → MS: kIDi,j ⊕ (T1, D, E, F, Texpire)

From the received message kIDi,j ⊕ (T1, D, E, F, Texpire), VN uses kIDi,j to extract (T1, A, B, C, Texpire) and authenticates VN by verifying T1. Then, VN computes A’ = A^D = g^D mod p, B’ = B^E = g^E mod p, and C = C’ = g^F mod p, and stores (A’, B’, C’, Texpire) in a ticket database. Message kIDi,j ⊕ (T1, D, E, F, Texpire) is then forwarded to MS.

After receiving kIDi,j ⊕ (T1, D, E, F, Texpire) in Step 4, MS extracts (T1, D, E, F, Texpire) with shared key kIDi,j and authenticates VN by verifying T1. MS then computes A1 = D’ = g^D mod p, B1 = E’ = g^E mod p, and C1 = F’ = g^F mod p, and stores them with Texpire in its device storage.
MS presents $A_1$ and $T_{\text{expire}}$ to declare the ownership of an anonymous ticket.

Step 2. VN $\rightarrow$ MS: $(B', g^{a'}\mod p, g^{e'}\mod p)_{C_1}$

According to $A_1$, VN finds the corresponding $(A', B', C', T_{\text{expire}})$ in the ticket database and checks whether $T_{\text{expire}} > 0$. If yes, VN selects two random numbers $a'$ and $e'$ and computes $g^{a'}\mod p$ and $g^{e'}\mod p$. Then VN encrypts $(B', g^{a'}\mod p, g^{e'}\mod p)$ with key $B'$ and sends it to MS.

Step 3. MS $\rightarrow$ VN: $(C_1, g^{a'}\mod p, g^{e'}\mod p)_{C_1}$

MS extracts $(B', g^{a'}\mod p, g^{e'}\mod p)$ with key $B'$. If $B' = B_1$, MS believes the VN is authentic. MS selects two random numbers $a'$ and $b'$ and compute $g^{a'}\mod p$ and $g^{b'}\mod p$ for the next authentication. Then sends $(C_1, g^{a'}\mod p, g^{b'}\mod p)_{C_1}$ to VN. VN receives the message and extracts $(C_1, g^{a'}\mod p, g^{b'}\mod p)$ with key $C_1$. If the obtained $C_1$ is the same as $C'$, VN successfully authenticates MS. VN then updates $(A', B', C', T_{\text{expire}})$ in its ticket database by setting $A' = C'$, $B' = g^{a'} \cdot g^{b'} \mod p$, $C' = (g^{b'})^{e'} \mod p$, and $T_{\text{expire}} = T_{\text{expire}} - 1$.

2.2. Attacks and comments on Yang et al.’s scheme

The major merits of Yang et al.’s scheme are user anonymity, mutual authentication, and secure anonymous tickets. In Yang et al.’s scheme, each home network has to store $k^{ID}_{h,i}$ mod $p$, $ID_h$ and $k_{h,i}$ for each of its registered users. Obviously, these values should be securely protected in order to prevent possible stolen verifier attacks. In addition, we find that Yang et al.’s scheme is vulnerable to guessing attacks performed by visited networks. In addition, we indicate a few potential deficiencies in the implementation of Yang et al.’s scheme.

2.2.1. Guessing attack.

Recall that VN receives $k_{h,i} \oplus (T_2, A, B, C, T_{\text{expire}})$ and $k_{h,i} \oplus (T_1, D, E, F, T_{\text{expire}})$ from HN in Step 3 of the anonymous channel ticket issuing phase. VN attempts to guess the value of $k_{h,i}$ from message $k_{h,i} \oplus (T_1, D, E, F, T_{\text{expire}})$. First, VN uses $k_{h,i}$ to extract $(T_2, A, B, C, T_{\text{expire}})$. VN is also aware of the values of $D, E, F$, and $T_{\text{expire}}$, but does not know the value of $T_1$. However, since $T_2 - T_1 \leq \Delta T$ and $\Delta T$ is usually a very small value, VN can easily list all the possible values of $T_1$. Then, for each possible value of $T_1$, denoted by $T'$, VN gets $k'_{h,i}$ by computing $(k_{h,i} \oplus (T_1, D, E, F, T_{\text{expire}})) \oplus (T', D, E, F, T_{\text{expire}})$, and then checks whether message $(ID_h, T_1, A, B, C)_{k'_{h,i}}$, received in Step 1, can be successfully decrypted by $k'_{h,i}$. If the decryption succeeds, VN successfully obtains secret key $k_{h,i}$ and also gets $ID_h$ from the decrypted message. Therefore, the proposed guessing attack can lead to the disclosure of users’ secret keys. Yang et al.’s scheme is insecure.

2.2.2. Time synchronization problem.

Yang et al.’s scheme uses timestamps to confirm the freshness of authentication messages. To ensure precise timestamps, time synchronization is required among MSs, VN, and HNs. Time synchronization is usually performed via a time synchronization protocol, e.g., Network Time Protocol (NTP). Thus, to synchronize clock times, MSs have to be already in the network. However, each MS is asked to present a synchronized timestamp before the MS is granted to access the wireless network. This raises a chicken-or-egg dilemma in the implementation of time synchronization. Therefore, we recommend that timestamps be used only within wired or fixed wireless networks.

2.2.3. Key length problem.

Yang et al.’s scheme uses shared keys to perform bitwise XOR operations on messages sent in Steps 3 and 4 of the anonymous channel issuing phase. Basically, the lengths of symmetric keys are short. In practice, these messages to be XORed may be longer than the shared keys. Thus, the latter part of an XORed message is subject to being revealed and modified.

2.2.4. Obsolete tickets left in VNs.

Yang et al.’s scheme allows each issued anonymous ticket to be used at most $T_{\text{expire}}$ times. Each VN is responsible for storing all the tickets whose $T_{\text{expire}}$ values are not reached yet. Since each MS may travel everywhere and stay in a visited network just for a short period of time, it will be very possible that a lot of tickets are maintained in a visited network and these tickets will not be used anymore. Yang et al.’s scheme did not address the obsolete ticket issue.

3. THE PROPOSED SCHEME

In this section, we propose a practical authentication protocol with user anonymity for wireless environments. Considering the implementation issues aforementioned, the proposed scheme uses random nonces in message exchanges within the wireless network, and restricts the use of anonymous tickets within a time period. Moreover, for better scalability, we do not adopt shared key schemes between MSs and HNs. Instead, we assume that the home agent, denoted by HA, in the home network has a secret key $x$, only known by the HA itself. It will be shown that HA can successfully authenticate each MS without the use of any verification table. On the other hand, in each visited network, there is a foreign agent, denoted by FA, responsible for authenticating anonymous tickets. Each FA and HA shares a secret key $k_{h,f}$. For less computation cost, most computations on authentication messages are based on a secure one-way hash function, denoted by $h()$, and string concatenation operations, denoted by ‘||’.

For better protection, we assume that each MS uses a smart card to store information for authentication. The proposed scheme consists of three phases: the registration phase, the ticket issuing phase, and the ticket authentication phase, described as follows.
3.1. The registration phase

MS first submits its identity $ID_{MS}$ and password $PW_{MS}$ to its HA for an initial registration. HA selects a random number $R_{MS}$ and computes $h(ID_{MS} \oplus x) \oplus h(PW_{MS})$ and $h(R_{MS} \oplus ID_{HA} \oplus x)$ with a secret number $x$. HA stores $ID_{HA}$, $R_{MS}$, $h(ID_{MS} \oplus x) \oplus h(PW_{MS})$, and $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$ in a smart card. The smart card is then issued to MS.

3.2. The ticket issuing phase

Step 1. MS $\rightarrow$ FA: $ID_{HA}$, $R_{MS}$, $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$, $h(ID_{MS} \oplus x) \oplus N_{MS}$

MS selects a random number $a$ and generates $N_{MS} = g^a \mod p$. Then, MS retrieves $h(ID_{MS} \oplus x)$ from the smart card and computes $h(ID_{MS} \oplus x) \oplus N_{MS}$. Finally, MS sends $ID_{HA}$, $R_{MS}$, $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$, and $h(ID_{MS} \oplus x) \oplus N_{MS}$ to FA.

Step 2. FA $\rightarrow$ HA: $R_{MS}$, $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$, $h(ID_{MS} \oplus x) \oplus N_{MS}$, $ID_{FA}$, $(ID_{FA}, T_1, N_{FA})_{h_{bf}}$

FA selects a random number $b$ and generates $N_{FA} = g^b \mod p$. Then, $(ID_{FA}, T_1, N_{FA})_{h_{bf}}$ is computed, where $T_1$ is the current timestamp. Finally, FA sends $R_{MS}$, $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$, $h(ID_{MS} \oplus x) \oplus N_{MS}$, $ID_{FA}$, and $(ID_{FA}, T_1, N_{FA})_{h_{bf}}$ to HA.

Step 3. HA $\rightarrow$ FA: $M_1$, $M_2$, $(T_3, N_{MS}', M_3)_{h_{bf}}$,

where $M_1 = h(ID_{MS} \oplus x) \oplus N_{MS} \oplus N_{FA}'$, $M_2 = h(ID_{MS} \oplus x) \oplus N_{MS}$, and $M_3 = h(ID_{MS} \oplus x) \oplus N_{MS} + 1 \oplus N_{FA}' + 1$.

Upon receiving messages at time $T_3$, HA computes $h(R_{MS} \oplus ID_{HA} \oplus x)$, gets $ID_{MS}$ from message $ID_{MS} \oplus h(R_{MS} \oplus ID_{HA} \oplus x)$, and gets $N_{MS}$ from message $h(ID_{MS} \oplus x) \oplus N_{MS}$. After that, HA decrypts $(ID_{FA}, T_1, N_{FA})_{h_{bf}}$ to get $ID_{FA}$, $T_1$, and $N_{FA}$. If $T_2 - T_1 \leq \Delta T$, where $\Delta T$ denotes a valid time interval, and $ID_{FA}$ is as expected, HA chooses a random number $c$ and computes $N_{FA}' = g^c \mod p$. Finally, HA prepares messages $M_1$, $M_2$, and $M_3$, and sends messages $M_1$, $M_2$, and $(T_3, N_{MS}', M_3)_{h_{bf}}$ to FA, where $T_3$ is the current time.

Step 4. FA $\rightarrow$ MS: $M_1$, $M_2$, $T_{ID}$, $T_{exp}$

FA decrypts $(T_3, N_{MS}', M_3)_{h_{bf}}$ and verifies its freshness from timestamp $T_3$. Finally, FA takes $N_{MS}'$ and $M_3$, and forwards $M_1$, $M_2$, $T_{ID}$, and $T_{exp}$ to MS.

Step 5. MS $\rightarrow$ FA: $M_1'$

MS computes $h(h(ID_{MS} \oplus x) \oplus N_{MS})$ and retrieves $N_{FA}'$ by calculating $M_1(h(ID_{MS} \oplus x) \oplus N_{MS})$. Then, message $M_2$ is verified. After a successful verification, MS computes $M_3 = h(ID_{MS} \oplus x) \oplus N_{MS} + 1 \oplus N_{FA}' + 1$, and sends it to FA. Upon receiving $M_3'$, FA authenticates MS by verifying whether $M_3' = M_3$.

After step 5, MS and FA have successfully authenticated each other, and have obtained $N_{FA}'$ and $N_{MS}$ respectively. Based on the Diffie-Hellman key agreement scheme, MS and FA determine a session key $SK_1 = (N_{FA}')^x \mod p = (N_{MS})^y \mod p = g^{abc} \mod p$. $SK_1$ will be used to encrypt all the messages delivered in the ongoing session. Figure 1 illustrates the ticket issuing phase of the proposed scheme.

3.3. The ticket authentication phase

After obtaining an anonymous ticket, MS can use this ticket to access a visited network before the ticket is expired. Each anonymous ticket should be authenticated before a secure and anonymous session is started. In addition, a new session key will be negotiated for use in the next session. Suppose $SK_i$ is the session key of the $i$th session. The following describes the ticket authentication phase in the $i$th session.

Step 1. MS $\rightarrow$ FA: $T_{ID}$, $(SK_i, N_{MS}'_{SK_i})$

MS selects a random number $a'$ and computes $N_{MS}' = g^{a'} \mod p$. MS then uses $SK_i$ to encrypt message $(SK_i, N_{MS}')$ and sends the encrypted message to FA.

Step 2. FA $\rightarrow$ MS: $(SK_i + 1, N_{FA}')$

According to $T_{ID}$, FA finds the corresponding ticket entry $(T_{ID}, T_{exp}, SK_i)$ in the ticket table. $T_{exp}$ is first used to validate whether the ticket is overdue. In fact, it is possible that...
4.1. Security analysis

The security of the proposed scheme is analyzed with respect to some well known attacks.

4.1.1. Replay attack.

Our protocol uses nonces and timestamps to withstand the replay attacks. Since both nonces $N_{MS}$ and $N_{FA}$ are generated independently, attacks by just replaying messages of previous sessions will fail.

4.1.2. Stolen-verifier attack.

An attacker may try to steal or modify the verification table. Our scheme does not store any verifiers in HAs. No stolen-verifier attack can be applied.

4.1.3. Impersonation attack.

An attacker may attempt to masquerades a legal entity involved in the scheme. However, the attacker has no way of knowing $h(ID_{MS} || x)$ and nonce values to generate proper authentication messages. Furthermore, in Step 2 and Step 3 of the ticket issuing phase, the shared key $k_{p,f}$ is only known between FA and HA. No one can correctly send forged messages without knowing $k_{p,f}$.

4.1.4. Guessing attack.

All of the delivered messages are protected by a secure one-way hash function and nonce values to withstand guessing attack. Hence, the attacker cannot verify his guessing from the eavesdropped data.

4.1.5. Known-key security.

Known-key security refers that if the session key is disclosed, it will not cause the compromise of any future session key. Each session key $SK_i$ is constructed based on nonces and the Diffie-Hellman key agreement scheme. Knowing the current session key is unable to derive other session keys.

4.1.6. Forward secrecy.

Forward secrecy in our scheme means that a compromise of the secret key $x$ held in HA does not cause the compromise of any session key. If secret $x$ is disclosed, $ID_{MS}$, $N_{MS}$, and $N_{c,MS}$ will be also disclosed in the ticket issuing phase. However, the proposed scheme adopts Diffie-Hellman key agreement algorithm to construct session keys. Perfect forward secrecy is ensured.

4.1.7. User anonymity preservation.

During messages delivered in our scheme, $ID_{MS}$ is protected by $h(w || ID_{HA} || x)$, which is only available in HA. Therefore, any other entity, including FA, cannot obtain any identity information about MS.

4.2. Performance analysis

The performance of the proposed scheme is evaluated by comparing it with Yang et al.‘s scheme. The following notations are used in the performance comparison.

$T_{exp}$: the time for computing modular exponentiation.

$T_{sym}$: the time of computing symmetric key cryptography.

$T_{hash}$: the time of computing one-way hash function

$T_{XOR}$: the time of computing XOR operation.

To achieve better performance, Yang et al.‘s scheme adopts pre-computations in the preparation of messages required in their scheme. It is claimed that, with pre-computations, step 1 and step 4 of the ticket issuing phase,

4. SECURITY AND PERFORMANCE ANALYSIS

To confirm the correctness of the proposed scheme, we use VO logic [11] to prove our protocol. VO is an extension of BAN logic [12] developed to analyze authentication protocols with key agreements. The detailed proof is described in the Appendix. In this section, we present the security and performance analysis of the proposed scheme.

The ticket with ticket identifier $T_{ID}$ is overdue and has been deleted automatically from the ticket table. In this case, no ticket will be found and the ticket authentication request will be rejected. If the ticket is still valid, FA uses $SK_i$ to decrypt $(SK_i, N_{MS})_{SK}$ and verifies whether the decrypted $SK_i$ is as expected. If yes, FA successfully authenticates an anonymous ticket. After that, FA selects a random number $b'$ and computes $N_{FA} = g^{b'} \mod p$. Then, FA uses $SK_i$ to encrypt message $(SK_i + 1, N_{FA})$ and sends the encrypted message to MS. MS receives the message, and decrypts it using $SK_i$. MS will successfully authenticate FA if the decrypted $SK_i + 1$ is as expected. After mutual authentication, MS and FA negotiate a new session key $SK_{i+1} = (N_{FA})^{b''} \mod p = (N_{MS})^{b''} \mod p = g^{a'b''} \mod p$. The $SK_{i+1}$, stored in MS’s smart card and FA’s ticket table, is thus replaced with $SK_{i+1}$. The ticket authentication phase is summarized in Figure 2.

Figure 2. The ticket authentication phase of the proposed scheme.
that pre-computations have been done. In this phase and a round of the authentication phase assuming that the parameters used in the messages. Indeed, pre-computations are needed to store pre-computed messages and corresponding hash values.

Based on the above estimated times, the computational cost of our scheme is 5.2941 s, while Yang et al.’s scheme requires 10.5096 s. Our scheme reduces about 50% in computational cost. In terms of overall computation time, our scheme is only 1 s. Most of our computation time is spent in the ticket issuing phase. Nevertheless, compared with Yang et al.’s scheme, our scheme takes the same computation time in the authentication phase, which will be performed more frequently than ticket issue phase. Table I summarizes the performance comparisons of our scheme with Yang et al.’s one. In summary, our scheme takes more computation time in the ticket issuing phase, but achieves the same performance in the authentication phase. Our scheme outperforms Yang et al.’s approach in terms of the computational cost. In addition, our scheme provides improvements in security protection, time synchronization, use of verification tables, and ticket maintenance.

<table>
<thead>
<tr>
<th>Table I. Performance comparison.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Computation time*</td>
</tr>
<tr>
<td>Ticket issuing phase</td>
</tr>
<tr>
<td>Authentication phase</td>
</tr>
<tr>
<td>Overall</td>
</tr>
<tr>
<td>Estimated time (s)</td>
</tr>
<tr>
<td>Computational cost</td>
</tr>
<tr>
<td>Ticket issuing phase</td>
</tr>
<tr>
<td>Authentication phase</td>
</tr>
<tr>
<td>Overall</td>
</tr>
<tr>
<td>Estimated time (s)</td>
</tr>
<tr>
<td>User anonymity</td>
</tr>
<tr>
<td>Mutual authentication</td>
</tr>
<tr>
<td>Resistant to guessing attack</td>
</tr>
<tr>
<td>Free of time synchronization in MS</td>
</tr>
<tr>
<td>Free of user verification table in HN (HA)</td>
</tr>
<tr>
<td>Ticket expiration</td>
</tr>
</tbody>
</table>

*aWith pre-computations.

as well as the entire authentication phase, all take zero computational time. Accordingly, the computation time of Yang et al.’s scheme is only $1T_{\text{sym}} + 2T_{\text{XOR}}$. We find that the evaluated computation time is incorrect, since it doesn’t include the computations required for decrypting and extracting received messages. In addition, there are the following problems in those pre-computations: (1) The symmetric encryption in step 1 of the ticket issuing phase cannot be pre-computed until timestamp $T_1$ is determined; (2) Most pre-computations for the next session should be performed in the current session; and (3) Additional storage space is needed to store pre-computed messages and corresponding parameters used in the messages. Indeed, pre-computations can reduce the running time of the scheme. However, it doesn’t imply that not any computational cost is incurred in pre-computations. Therefore, a performance analysis in terms of computational cost is also required. In the following performance analysis, the computational cost of a scheme is evaluated according to all the computations required in the scheme, and the computation time is estimated by the elapsed time for running the ticket issuing phase and a round of the authentication phase assuming that pre-computations have been done.

By our performance analysis, Yang et al.’s scheme takes $4T_{\text{sym}} + 4T_{\text{XOR}}$ in terms of computation time, and requires $20T_{\text{exp}} + 8T_{\text{sym}} + 4T_{\text{XOR}}$ in computational cost. Our scheme takes $2T_{\text{exp}} + 6T_{\text{sym}} + 6T_{\text{hash}} + 4T_{\text{XOR}}$ in terms of computation time, and spends $10T_{\text{exp}} + 8T_{\text{sym}} + 9T_{\text{hash}} + 6T_{\text{XOR}}$ in computational cost. Obviously, Yang et al.’s scheme gains better performance from the pre-computations, but incurs more computational cost than our scheme does. More precisely, as indicated in Reference [13], a one-way hashing operation takes about 0.0005 s and a symmetric encryption/decryption requires 0.0087 s. An exponential operation is approximately equal to 60 symmetric encryption/decryptions. Therefore, an exponential computation takes about 0.522 s. The computational cost of XOR operations can be ignored compared to the other computations. Based on the above estimated times, the computational cost of our scheme is 5.2941 s, while Yang et al.’s scheme requires 10.5096 s. Our scheme reduces about 50% in computational cost. In terms of overall computation time, our scheme is slower than Yang et al.’s scheme by 1 s. Most of our computation time is spent in the ticket issuing phase.

5. CONCLUSION

We have successfully presented guessing attacks on Yang et al.’s scheme, and also indicated potential drawbacks in the implementation of their scheme. A new secure and practical authentication scheme is thus proposed. In the proposed scheme, nonces are used for both message protection and key agreement between MS and FA. To protect the secrecy of MS from FA, we carefully use nonces in the scheme such that FA can issue an anonymous ticket based on nonces, but cannot learn any information about MS. In summary, the proposed scheme provides the following merits: user anonymity, mutual authentication, ticket copy prevention, lower computational cost, free of time synchronization in
Authentication for wireless access networks

Y. Chen et al.

wireless clients, and free of verification tables in HAs. We further prove the correctness of the proposed scheme by VO logic analysis.

Our study is a theoretical approach for the authentication in wireless access networks, and several practical issues have been considered in the development of our scheme. The security issues of wireless access networks become more crucial for contemporary mobile applications. In the future, we will take into consideration the application of our scheme in current wireless and mobile networks, e.g., IEEE 802.11 wireless LANs, 3G, and WiMAX networks.

APPENDIX: LOGIC ANALYSIS

In the VO logic, the original protocol must be first transformed to an idealized form, and write assumptions about the initial state of the protocol, and then use the logic to derive the beliefs held by protocol principals. $P \models X$ denotes $P$ believes that the statement $X$ is true. $P|\equiv X$ and $P|\neg X$ denote $P$ says $X$ and $P$ said $X$ to discriminate the tense, i.e., $P$ sends or once sent a message including $X$. $\{X\}_P$ represents $X$ encrypted with the key $K$. $<X>_P$ denotes $X$ combined with the formula $Y$. $PK_E(A,K)$ denotes the public signature verification key associated with principal $A$, and the corresponding public signature verification key is $PK_E^{-1}(A)$. $PK_A(A,K)$ denotes the public key-agreement key associated with principal $A$, and the corresponding good private key-agreement key is $PK_E^{-1}(A)$. $X \models Y$ denotes that the current knowledge of $X$ can be demonstrated to $Y$, $f(k)$ is a key agreement function $f$(private_info, public_info).

Goals

The soundness of our protocol is proven if the following six generic authentication goals for party MS, similar to FA, can be finally achieved via the VO logic analysis:

G1. Far-end operative: $MS|\equiv FA|\equiv Y$
MS believes FA recently sent a message $Y$. This implies that FA is currently operational.

G2. Targeted entity authentication: $MS|\equiv FA|\equiv (Y, R(G(C_MS)), Y)$
MS believes a message $Y$ sent by FA in response to the specific challenge $C_MS$. It provides authentication of FA to MS in the sense that the response is from a corroborated operational entity, and is targeted in response to a challenge from MS.

G3. Secure key establishment: $MS|\equiv MS \leftrightarrow FA$
MS believes that the key $K$ is shared with no party other than party FA.

G4. Key confirmation: $MS|\equiv MS \leftrightarrow FA$
MS believes the key $K$ is shared with FA alone, and FA has provided evidence of knowledge of the key to MS.

G5. Key freshness: $MS|\equiv \#(K)$
MS believes the key $K$ is fresh.

G6. Mutual belief in shared secret: $MS|\equiv FA|\equiv FA \leftrightarrow MS$
MS believes the target entity FA also believes $K$ is an unconfirmed secret suitable for use with MS.

Idealization

We transform the proposed protocol to the following idealized form suitable for further logic manipulation:

combined with the formula $Y$. $PK_E(A,K)$ denotes the public signature verification key associated with principal $A$, and the corresponding public signature verification key is $PK_E^{-1}(A)$. $PK_A(A,K)$ denotes the public key-agreement key associated with principal $A$, and the corresponding good private key-agreement key is $PK_E^{-1}(A)$.

Step 1. $MS \rightarrow FA: \{ID_MS, R_MS, \langle ID_MS, h(ID_MS)\rangle, \langle N_MS, h(ID_MS)\rangle, \langle N_MS, h(ID_MS)\rangle, ID_FA, \langle ID_FA, T_1, N_FA \rangle, k_{h,f} \}$

Step 2. $FA \rightarrow HA: \{R_MS, \langle ID_MS, h(ID_MS)\rangle, \langle N_MS, h(ID_MS)\rangle, ID_FA, \langle ID_FA, T_1, N_FA \rangle, k_{h,f} \}$

Step 3. $HA \rightarrow FA: \{T_1, N_MS, \langle h(ID_MS)\rangle, \langle N_MS, h(ID_MS)\rangle, N_FA, k_{h,f} \}$

Step 4. $FA \rightarrow MS: \{N_FA, k_{h,f} \}$

Step 5. $MS \rightarrow FA: \{h(h(ID_MS)\langle ID_MS, h(ID_MS)\rangle), N_MS + 1, N_FA + 1 \}$

Step 6. $MS \rightarrow FA: \{K, N_MS \}$

Step 7. $FA \rightarrow MS: \{K + 1, N_FA \}$

Assumptions

The formal assumptions required for party MS are listed as follows. Similar assumptions are also required for FA.

A1. $MS|\equiv HA|\Rightarrow PK_k(FA, K_FA)$, where $PK_k(FA, K_FA) = N_FA$

A2. $MS|\equiv PK_E^{-1}(MS)$

A3. $MS|\equiv PK_E^{-1}(FA)$

A4. $MS|\equiv \#(N_MS)$

A5. $MS\models (HA\models PK_k(fa, T_FA)\langle fa, T_FA \rangle)$$MS\models (HA\models PK_k(fa, T_FA)\langle fa, T_FA \rangle)$

Proofs

We prove the proposed protocol in six lemmas corresponding to the above six generic goals.

Lemma 1. The proposed scheme provides secure key establishment, i.e., goal (G3) $MS|\equiv MS \leftrightarrow FA$ is achieved.
Proof:
1. MS sees \( N_{FA}^c \leftrightarrow h(\langle ID_{MS} | x | N_{MS} \rangle) \)
   By Step 4

2. MS sees \( N_{FA}^c \leftrightarrow MS \) has \( N_{FA}^c \), where \( N_{FA}^c = PK_{x}(FA) \)
   (S1)
   By brief conjunctatenation

3. MS has \( K \), where \( K = f(PK_{x}^{-1}(MS), PK_{x}(FA)) \)
   (S2)
   By unqualified key-agreement, (S1), and (A2)

4. MS| ≡ \( PK_{x}(FA, K_{FA}) \) (S3)
   By jurisdiction, (A1) and (AS)

5. MS| ≡ MS \( \leftrightarrow \leftrightarrow \) FA, where \( K = f(PK_{x}^{-1}(MS), PK_{x}(FA)) \)
   By qualified key-agreement, (S3), (A2), and (A3)
   That is, MS believes \( K \) is shared with no party other than FA. Implicitly, MS also now possesses this key.

Lemma 2. The proposed scheme provides key confirmation, i.e., goal (G4) MS| ≡ MS \( \leftrightarrow \leftrightarrow \) FA is achieved.

Proof.
We require two additional formal assumptions:

1. MS| ≡ \( \phi(N_{MS}^c) \) (S4)
   That is, MS believes that \( N_{MS}^c \) generated by MS itself is fresh and recognizable using GNY constructs.

2. MS sees \( \phi(N_{MS}^c) \)
   By recognizability rule, (S2), and (S4)(S6)

3. MS sees \( \phi(N_{MS}^c) \leftrightarrow confirm(K) \)
   By Confirmation Axiom, (S4), and (S5)

4. MS| ≡ MS \( \leftrightarrow \leftrightarrow \) FA, where \( K \leftrightarrow \leftrightarrow \) is the session key \( SK \) of the proposed scheme.

By key confirmation, (S7), and Lemma 1
That is, upon a successful completion of the protocol, MS believes that the session key \( K \) is shared with only FA, and FA has provided the evidence of knowledge of this key to MS.

Q.E.D.

Lemma 3. The proposed scheme provides key freshness, i.e., goal (G5) MS| ≡ \#(K) is achieved.

Proof.
1. MS| ≡ \#(\( (N_{MS}^c)^h \)), where \( (N_{MS}^c)^h = g^{dh} \mod p \)
   By freshness propagation, and (A4)
   For non-zero \( a, b, \) and \( c \), the entropy \( K = (N_{FA}^c)^h = (N_{MS}^c)^h \) is large. Therefore, the freshness conjunctatenation rule over this exponentiation provides freshness of the key \( K \).

Q.E.D.

Lemma 4. The proposed scheme establishes that the far-end party is operative, i.e., goal (G1) MS| ≡ FA| \( \approx \) Y is achieved.

Proof.
1. MS sees \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \) (S8)
   By Step 4

2. For shared secrets, we postulate
   MS| \( \leftrightarrow \leftrightarrow \) HA
   (S9)

3. MS| ≡ HA| \( \approx \) \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \) (S10)
   By message meaning, (S8) and (S9)

4. MS| ≡ \#(\( (N_{FA}^c)^h \)) \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \)
   (S11)
   By freshness propagation, (S10) and (A4)

5. MS| \( \approx \) HA| \( \approx \) \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \)
   By nonce-verification, (S10) and (S11)

6. MS| \( \approx \) HA| \( \approx \) \( (N_{FA}^c)^h \) \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \)
   (S12)
   By freshness propagation
   Thus, MS believes that HA recently said message
   \( (N_{FA}^c)^h \) \( h(h(ID_{MS} | x) | N_{MS} | N_{FA}^c) \) \( N_{FA}^c \)
   (S13)
can be deduced that FA. After a successful completion of the protocol, HA and FA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

By Lemma 4, and Step 4

We break the concatenation and derive $h(h(ID_{MS}||x)||N_{MS})$ which provides authentication evidence of FA and HA to MS in the sense that the response is from the corroborated operational entity HA and FA, and it is targeted to response to the challenge from MS in Step 1. Furthermore, since $N_{MS} = g^a$, provided that MS does not intentionally re-choose a random number $a$ to generate the nonce in the current epoch using an appreciate random number generator, the nonce will not be a duplicate of a previous nonce. Thus, upon a successful completion of the protocol, MS believes that FA conveyed

\[
\langle h(h(ID_{MS}||x)||N_{MS}||N_{FA}^r)||N_{FA}^r, T_{ID}, T_{exp}\rangle
\]

**AUTHORS’ BIOGRAPHIES**

**Yen-Cheng Chen** received the Ph.D. degree in Computer Science from the National Tsing Hua University, Taiwan, in 1992. He was an Associate Researcher of the ChungHwa Telecom Labs. from 1992 to 1998. From 1998 to 2001, he was an Assistant Professor of the Department of Information Management, Ming Chuan University, Taiwan. Currently, he is an Associate Professor of the Department of Information Management, National Chi Nan University, Taiwan. His current research interests are network management, wireless networks, and security.

**Shu-Chuan Chuang** received the M.S. degree in the Department of Information Management from National Chi-Nan University in 2006. Currently, she is a computer technician of the department of Information Management in Kaohsiung Veterans General Hospital, in charge of the applications development and information security audit. Her interests include Internet technology and network security.

**Lo-Yao Yeh** received the B.S. degree in Information Management from Da Yeh University, Taiwan, in 2003. He got the M.S. degree in the Department of Information Management from National Chi Nan University in 2005. Now, he is a Ph.D. candidate in the Department of Computer Science in National Chiao Tung University. He was a visiting scholar in UC Berkeley. His current research interests include network security and overlay networks security, and sensor networks.

**Jiun-Long Huang** received the B.S. and M.S. degrees from the Department of Computer Science and Information Engineering at National Chiao Tung University in 1997 and 1999, respectively, and the Ph.D. degree from the Department of Electrical Engineering at National Taiwan University in 2003. Currently, he is an Assistant Professor in the Department of Computer Science at National Chiao Tung University. His research interests include mobile computing, mobile data management, wireless access networks, and Internet technology.