EUROPEAN TRANSACTIONS ON TELECOMMUNICATIONS
Euro. Trans. Telecomms. 2004; **15**:1–9 (DOI: 10.1002/ett.957)

Communication Networks

New mutual authentication and key exchange protocol with balanced computational power for wireless settings

Chou-Chen Yang[1,*], Jian-Wei Li[2] and Min-Shiang Hwang[1]

[1]*Department of Management Information Systems, National Chung Hsing University, 250 Kuo Kuang Road, 402 Taichung, Taiwan, R.O.C.*
[2]*Department of Computer Science and Information Engineering, National Cheng Kung University, No 1. Ta-Hsueh Road, Tainan City, Taiwan 701, R.O.C.*

SUMMARY

Mutual authentication and key exchange protocols (MAKEP) provide two parties in communication with guarantee of true identity. And then the two parties end up sharing a common session key for privacy and data integrity during the session. In MAKEP, public-key-based schemes and symmetric-key-based schemes are often used. However, the former requires high computation complexity and hence, it is not suitable for applications in wireless settings. The latter has to maintain many distinct keys for different parties. Wong *et al.* proposed the Linear MAKEP to solve these problems. But in term of storage space, it is not optimal. In this paper, we propose a scheme that uses the geometric properties of line to achieve mutual authentication and key exchange. Compared with Wong *et al.*'s scheme, our scheme is efficient and requires less storage space. It can withstand the replay attack and the unknown key-share attack, and the server does not bear much more computation cost than the client in each session, hence we call it a protocal with balanced computational power. Copyright © 2004 AEI.

1. INTRODUCTION

Mutual authentication and key exchange protocols (MAKEP) [14, 15, 19] provide two parties in communication with guarantee of true identity. And then the two parties end up sharing a common session key known only to them. This session key can be used to provide privacy and data integrity during the session.

In MAKEP, public-key-based schemes [2, 7, 10, 12] and symmetric-key-based schemes [4, 5, 18, 21, 22] are the two kinds of schemes most commonly used. The former requires complex en/decryption computation, it is hard to satisfy for wireless settings. Since the wireless settings are characterized by the low-power mobile device (mobile host, MH) with limited memory, low computational power and limited bandwidth. If an MH depends on a battery, then its power will be depleted very quickly. The latter scheme is more suitable for the wireless settings but two parties in communication need to share a long-life key, which means each party must maintain many distinct keys for different parties.

To relieve of such high computation complexity as the public-key-based scheme has and such restriction as the symmetric-key-based scheme has, several schemes [16, 25, 26] have been proposed for the wireless settings with unbalanced computation complexity power which means that the base station BS must bear much more computation cost than the client in each session. They use a pre-computation technique to reduce the computation complexity of the MH and to store pre-computation results in its memory to relieve itself from complex computations. But in [16, 25], the pre-computation results are on the side of the BS, hence if the MH moves into the realm of another BS without the pre-computation results based on the public

Correspondence to: Professor Chou-Chen Yang, Department of Management Information Systems, National Chung Hsing University, 250 Kuo Kuang Road, 402 Taichung, Taiwan, R.O.C. E-mail: ccyang@cyut.edu.tw
Contract/grant sponsor: National Science Council, Taiwan, R.O.C.; contract/grant number: NSC 91-2622-E-324-010-CC3.

Received 18 September 2002
Revised 16 June 2003
Accepted 3 September 2003

key of this new BS, then the MH is not able to perform these protocols. In addition, the protocol in Reference [16] is susceptible to a variety of interleaving attacks brought up by Wong *et al.* [25].

In some unbalanced computation schemes [16, 25, 26] reduced the computation complexity of the MH but the BS still has to do complex operations, such as public-key encryption/decryption or modular exponentiation computation. As the number of MH increases, the BS may have to deal with many complex computations simultaneously, and these communication transactions will be delayed. Therefore, the computation complexity on the BSs side also needs to be reduced in order to have more efficient performance no matter where it is in the wired or wireless settings.

Wong *et al.* [26] proposed the linear MAKEP. This scheme is free from the above restrictions and reduces the computation complexity of the MH, so it is very suitable for the wireless settings. But in term of storage space, the scheme is not optimized, because the MH must have more memory to store n pairs of private keys and n certificates in its memory, where n is the total amount of that the MH wants to run the protocol.

In this paper, we propose a scheme that uses the geometric properties of line to achieve authentication and key exchange. The computation complexity of the BS is lower than the protocols in Reference [16, 25, 26]. Compared with Wong *et al.*'s scheme, our scheme is efficient and requires less storage space. Furthermore, our scheme can withstand the replay attack and the unknown key-share attack [3, 6]. And the MH and BS does not bear unbalanced computation cost, we call it a protocol with balanced computation power.

The remainder of this paper is organized as follows. In the next section, we will briefly review Wong *et al.*'s scheme. Then, in Section 3, we shall illustrate how our proposed scheme will work in detail. After that, in Sections 4 and 5, the security analysis and performance analysis will be presented. Finally, in the last section, we shall offer our conclusion.

Notations

Some notations used throughout this paper are the following:

- A $\longrightarrow$ B: m: denotes that A sends the message m to B;
- E_K/D_K: an encryption/decryption transformation of symmetric cryptosystem with secret key K;
- PK_A: a public key of entity A;
- SK_A: a private key of entity A;
- E_{PK_A}: the encryption transformation of a public-key cryptosystem with PK_A;
- $Sig_A(m)$: a signature of the message m signed by entity A;
- $H(\cdot)$: a one way hash function;
- ID_A: an identification information of entity A;
- $Cert_A$: a certificate of entity A;
- $r \leftarrow \{0,1\}^k$: a nonce which is an k-bit strong random number, where k is a secure parameter.

2. REVIEW OF WONG *ET AL.*'S SCHEME

In the linear MAKEP [26], an BS has a private key (SK_{BS}) and the corresponding public key (PK_{BS}). The PK_{BS} is publicly known. The system chooses a prime p in Z_p and then chooses a $g \in Z_p^*$, where p and g are public parameters. For MH, it must perform a pre-computation technique in advance and store the pre-computation results in its memory. Then, to ask for service or resource from the BS, the MH can run the protocol by means of the pre-computation results. This technique and protocol run as follows (shown in Fig. 1):

Step 0. MH: Pre-computation
First, the MH randomly chooses a sequence of integers (a_1, a_2), (a_3, a_4), ..., (a_{2i-1}, a_{2i}) in Z_{p-1} as its private keys, $1 \le i \le n$, where i is the i-th run of the protocol and n are the total amount of that the MH wants to run the protocol. The corresponding sequence of public key pairs are (g^{a_1}, g^{a_2}), (g^{a_3}, g^{a_4}), ..., $(g^{a_{2i-1}}, g^{a_{2i}})$ in Z_p, where $1 \le i$. Second, the signatures $\mathrm{Sig}_{TA}(ID_{MH}, g^{a_{2i-1}}, g^{a_{2i}})_{1 \le i}$ are obtained from the trusty authority (TA).

Step 1. MH $\rightarrow$ BS: Cert^i_{MH}
At the i-th run of the protocol, the MH constructs a certificate denoted by

$$\mathrm{Cert}^i_{MH} = \langle ID_{MH}, g^{a_{2i-1}}, g^{a_{2i}}, \mathrm{Sig}_{TA}(ID_{MH}, g^{a_{2i-1}}, g^{a_2 i}) \rangle$$

and sends it to the BS.

Step 2. BS $\rightarrow$ *MH*: r_{BS}
Upon receiving messages of Step 1, the BS confirms the validity of the certificate and sends back a nonce r_{BS}.

Step 3. MH: Upon receipt of messages of Step 2.
The MH chooses another nonce r_{MH} and computes $x = E_{PK_{BS}}(r_{MH})$. Then it computes y as

$$y = a_{2i-1}(x \oplus r_{BS}) + a_{2i} \bmod (p-1) \qquad (1)$$

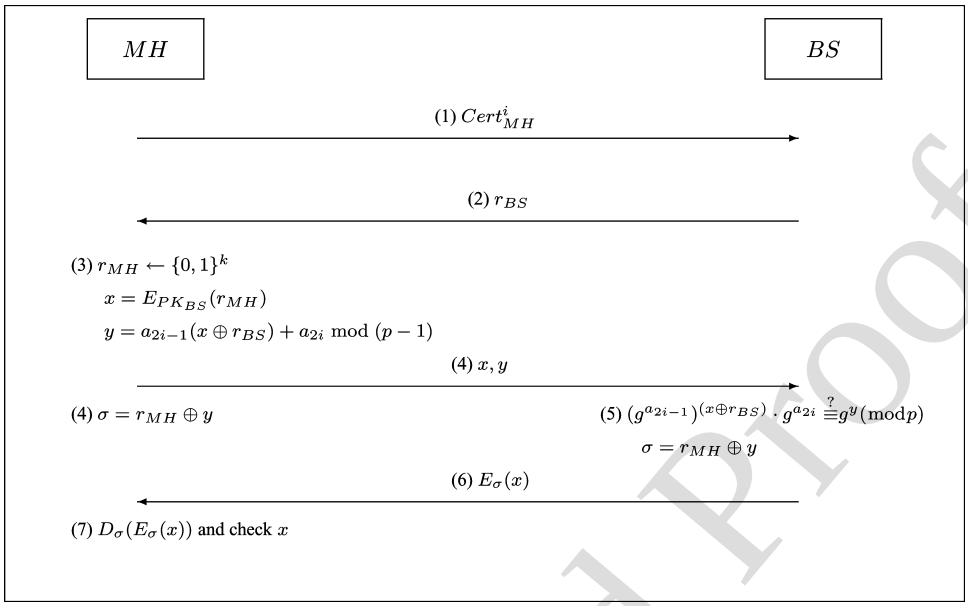

Figure 1. The linear MAKEP.

Step 4. MH $\rightarrow$ BS: x, y
The MH sends x, y to the BS and computes a new session key σ as $r_{\mathrm{MH}} \oplus y$.

Step 5. BS: Upon receipt of messages of Step 4.
The BS checks the equation

$$\left(g^{a_{2i-1}}\right)^{(x \oplus r_{\mathrm{BS}})} \cdot g^{a_{2i}} \stackrel{?}{\equiv} g^y \bmod p.$$

If the equation holds, the BS derives r_{MH} by decrypting x and then computes a new session key σ as $r_{\mathrm{MH}} \oplus y$; otherwise, this communication is rejected and the protocol halts.

Step 6. BS $\rightarrow$ MH: $E_\sigma(x)$

Step 7. MH: Upon receipt of messages of Step 6.
The MH decrypts the message and then checks whether the decrypted message is x.

3. THE PROPOSED SCHEME

In this section, we propose a scheme which uses the geometric properties of lines to achieve mutual authentication and key exchange between a low-power MH and an BS. For the BS, it has a private key $\mathrm{SK_{BS}}$ and the corresponding public key $\mathrm{PK_{BS}}$. We assume that the public key of the BS is publicly known. For the MH, it must perform a pre-computation technique in advance. Then, to ask for service or resource from the BS, the MH can run the protocol by means of the pre-computation results. This technique and the proposed protocol are described in detail as follows (shown in Fig. 2).

Step 0. MH: pre-computation
The MH selects linear equations $L_i(x) = a_i x + b_i$, $1 \leq i \leq n$, where a_i and b_i are nonzero real numbers, x is a variable, i is the i-th run of the protocol, and n is the total amount of that the MH wants to run the protocol. The corresponding certificates are given from the TA by

$$\mathrm{Cert}^i_{\mathrm{MH}} = \langle \mathrm{ID_{MH}}, B_i, \mathrm{H}(b_i, B_i), \mathrm{Sig_{TA}}(\mathrm{ID_{MH}}, B_i, H(b_i, B_i)) \rangle,$$

where B_i is a point on the linear equation $L_i(x)$.

Step 1. MH: Compute a point ($A_i = (r_{\mathrm{MH}}||T, L_i(r_{\mathrm{MH}}||T))$ on $L_i(x)$
At the i-th run of our protocol, the MH computes a point

$$A_i = (x, L_i(x)) = (r_{\mathrm{MH}}||T, L_i(r_{\mathrm{MH}}||T))$$

on the linear equation $L_i(x)$, where r_{MH} is randomly chosen by the MH and x is derived from concatenating r_{MH} with the timestamp T.

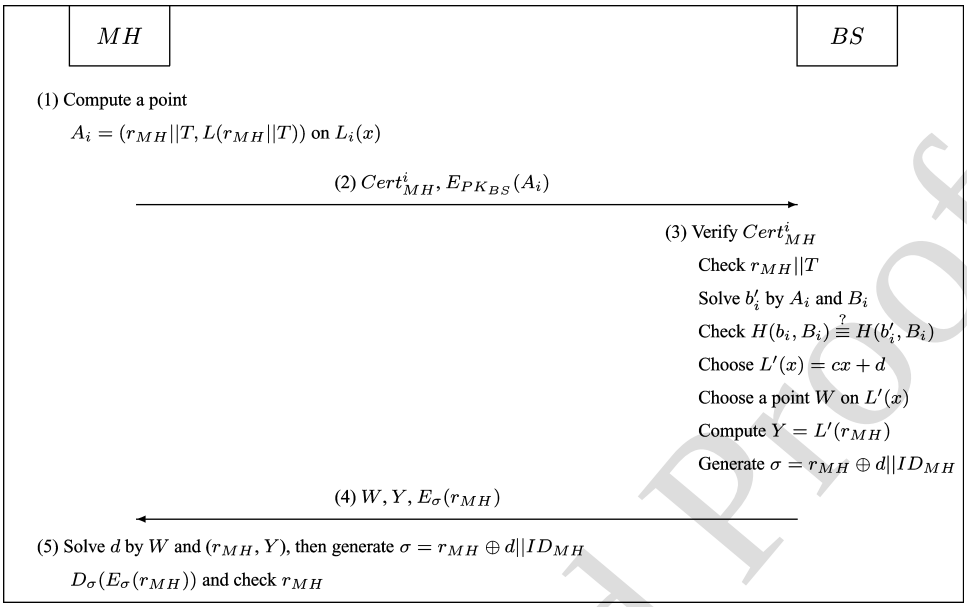

Figure 2. The proposed scheme.

Step 2. MH $\longrightarrow$ BS: Cert^{i}MH, $\mathrm{E}_{\mathrm{PK_{BS}}}(A_i)$

Step 3. BS: receipt of messages of Step 2.
First, the BS verifies the correctness of the certificate $\mathrm{Cert}^i_{\mathrm{MH}}$ and then decrypts $\mathrm{E}_{\mathrm{PK_{BS}}}(A_i)$ by using its private key and checks the timestamp T in A_i.

Second, the BS uses the two points A_i and B_i to reconstruct $L_i(x)$ and then derives the constant term b'_i. Then, the BS computes $H(b'_i, B_i)$ and checks the equation $H(b_i, B_i) \stackrel{?}{=} H(b'_i, B_i)$. If the equation holds, that means the MH has been authenticated successfully. Otherwise, this communication is rejected and the protocol halts.

Last, the BS chooses a linear equation $L'(x) = cx + d$ which is different from $L_i(x)$, where c and d are nonzero real numbers and x is a variable. Then, the BS randomly chooses a point W on $L'(x)$ and computes $Y = L'(r_{\mathrm{MH}})$. In the meantime, the session key $\sigma = r_{\mathrm{MH}} \oplus d||\mathrm{ID}_{\mathrm{MH}}$ can be generated.

Step 4. BS $\longrightarrow$ MH: $W, Y, \mathrm{E}_\sigma(r_{\mathrm{MH}})$

Step 5. MH: receipt of messages of Step 4.
The MH takes the two points W and (r_{MH}, Y) on the linear equation $L'(x) = cx + d$ to reconstruct $L'(x)$ and then derives the constant term d and generates the session key $\sigma = r_{\mathrm{MH}} \oplus d||ID_{\mathrm{MH}}$. After that, the MH takes the session key σ to decrypt $\mathrm{E}_\sigma(r_{\mathrm{MH}})$ and checks whether the decrypted message is r_{MH}. If so, that means the BS has been authenticated successfully; otherwise, this communication is rejected and the protocol halts.

The mutual authentication is achieved by the pairs of challenge and response messages ($\mathrm{Cert}^i_{\mathrm{MH}}$, A_i) and (r_{MH}, $\mathrm{E}_\sigma(r_{\mathrm{MH}})$). For ($\mathrm{Cert}^i_{\mathrm{MH}}$, A_i), the BS can authenticate the MH by the means of point A_i which is provided by the MH and the trusted certificate. Furthermore, to avoid the replay attack, we use a timestamp T at the point $A_i = (r_{\mathrm{MH}}||T, L_i(r_{\mathrm{MH}}||T))$ on $L_i(x)$. For (r_{MH}, $\mathrm{E}_\sigma(r_{\mathrm{MH}})$), since r_{MH} is encrypted by the public key of the BS, the MH can have confidence in the BS by checking r_{MH} in $\mathrm{E}_\sigma(r_{\mathrm{MH}})$.

The BS generates a Linear equation $L'(x) = cx + d$ different from $L_i(x)$ and hides the factor d of the session key in the constant term. If an adversary or illegal receiver wants to reconstruct $L'(x)$, he/she must find two points on $L'(x)$. The adversary knows only one public point W. To find another point on $L'(x)$ would be extremely difficult. Therefore, only the MH owns simultaneously the two points (r_{MH}, Y) and W to reconstruct $L'(x)$ and derive the session key σ. Besides, r_{MH} and d are bound together to provide the MH with the ability to confirm the freshness of the session key.

4. SECURITY ANALYSIS

In this section, some security properties and some possible attacks will be raised and fought against to demonstrate the security of our scheme. First of all, we will review the properties of one way hash function and public key cryptosystem.

Definition 4.1 A one way hash function, $H : x \rightarrow y$, has the following properties [8, 20, 23]:

1. The function H can take a message of arbitrary-length input and produce a message digest of a fixed-length output.
2. The function H is one-way, given x, it is easy to compute $H(x) = y$. However, given y, it is hard to compute $H^{-1}(y) = x$.
3. The function H, given x, it is computationally infeasible to find $x' \neq x$ such that $H(x') = H(x)$.
4. The function H, it is computationally infeasible to find any two pair x and x' such that $x' \neq x$ and $H(x') = H(x)$.

Definition 4.2. In a public key cryptosystem, a user owns a private key SK and the corresponding public key PK, where the private key is kept secretly by the user and the public key is publicly known. Besides, there are encryption and decryption function ($\mathrm{E_{PK}}(\cdot)$ and $\mathrm{D_{SK}}(\cdot)$), such as RSA [24] and ElGamal [11]. When a sender wants to send a secret message M to a receiver which has a private key ($\mathrm{SK_R}$) and public key ($\mathrm{PK_R}$), he can encrypt M with the public key of receiver $\mathrm{PK_R}$, as

$$\mathrm{C} = \mathrm{E_{PK_R}}(M),$$

where C denotes cipher text . Only the valid receiver which has the correct $\mathrm{SK_R}$ can derive M by decrypting with $\mathrm{SK_R}$ as follows:

$$M = \mathrm{D_{SK}}(\mathrm{C}).$$

Since the $\mathrm{SK_R}$ is only kept by the valid receiver, no one can derive the message M without knowing $\mathrm{SK_R}$.

In the proposed MAKEP scheme, two parties in communication (MH and BS) must mutually authenticate with each other. Then they agree a session key to provide privacy and data integrity during a session. The following shows that the proposed scheme satisfies the above properties, and some possible attacks will be raised and fought against to demonstrate the security of our scheme.

Attack 1: An adversary tries to break the mutual authentication between the MH and BS.

Analysis of attack 1: The mutual authentication between MH and HA is achieved by the challenge-response method. The pairs of challenge and response messages are ($\mathrm{Cert}^i_{\mathrm{MH}}$, $\mathrm{E_{PK_{BS}}}(A_i)$) and ($r_{\mathrm{MH}}$, $\mathrm{E}_\sigma(\mathrm{r_{MH}})$), respectively. An adversary tries to impersonate the MH, he/she must fake $\mathrm{Cert}^i_{\mathrm{MH}}$ and A_i. However, since the A_i and b_i are protected by $\mathrm{PK_{BS}}$ and a one way hash function $\mathrm{H}(b_i, B_i)$ respectively, the adversary can not derive $L_i(x)$. Besides, he/she can not compute a correct certificate without knowing the private key of TA. Similarly, the adversary tries to fake $\mathrm{E}_\sigma(r_{\mathrm{MH}})$ to impersonate the BS. However, the adversary can not compute the correct session key $\sigma = r_{\mathrm{MH}} \oplus d||\mathrm{ID_{MH}}$ without knowing the private key of BS.

Attack 2: An adversary tries to break the privacy and integrity of communication messages between the MH and BS.

Analysis of Attack 2: Since the session key σ is computed as $r_{\mathrm{MH}} \oplus d||\mathrm{ID_{MH}}$, the adversary tries to break the privacy and integrity of communication messages between the MH and BS, he/she must derive A_i and d. However, the adversary has no knowledge of the private key of BS and d. Note that, r_{MH} and d are bound together to provide the MH with the ability to confirm the freshness of the session key.

Attack 3: An adversary tries to mount the replay attack.

Analysis of Attack 3: An adversary tries to raise the replay attack. The random numbers and timestamps are employed in our scheme, such as $A_i = (x, L_i(x)) = (r_{\mathrm{MH}}||T, L_i(r_{\mathrm{MH}}||T))$, it can successfully withstand the replay attack.

Attack 4: An adversary tries to mount the unknown key-share attack [3,6].

Analysis of Attack 4: If an adversary E tries to mount the unknown key-share attack on an authenticated key agreement protocol, an entity A ends up believing that she shares a session key with an entity B. However, B mistakenly believes that the session key is instead shared with the adversary E. In this situation, B has been led to false beliefs. To against the unknown key-share attack, we employ an identification of MH $\mathrm{ID_{MH}}$ to a session key ($\sigma = r_{\mathrm{MH}} \oplus d||\mathrm{ID_{MH}}$) in our scheme. For example, if the unknown key-share attack happens, the session key $\sigma = r_{\mathrm{MH}} \oplus d||\mathrm{ID_E}$ (where $\mathrm{ID_E}$ is the identifier of an adversary) which the BS computes will not be equal to the $\sigma = r_{\mathrm{MH}} \oplus d||\mathrm{ID_{MH}}$ which the MH computes. Therefore, the unknown key-share attack can be withstood.

5. PERFORMANCE AND STORAGE ANALYSIS

In this section, we analyze the computation complexity, the number of communication times, the total size of communication messages and memory demand of our proposed scheme, compared with the Wong *et al.*'s scheme [26].

To analyze the computation complexity of the Wong *et al.*'s scheme and our scheme, we first define related notations as follows:

- $H(\cdot)$ is an operation of hash function.
- O_{PK} is an operation of encrypting a message, in length 512 bits, using public-key cryptosystems (i.e. RSA).
- O_{SK} is an operation of encrypting a message, in length 64 bits, using secret-key cryptosystems (i.e. DES).
- MOD_M and MOD_A are a modular multiplication and a modular addition respectively.
- $R_{L(x)}$ is an operation of reconstructing a linear equation $L(x)$. In general, to construct $R_{L(x)}$ it requires one multiplication, one division and three subtract.
- $Mod_M(n)$ and $Mod_A(n)$ denote an operation of modular multiplications and modular additions of length n bits respectively.
- D(n), M(n), A(n) and MOD(n) denote operation, of divisions, multiplications, additions and modulus of length n bits respectively.
- S denotes an operation of shift.
- l(n) denotes lengths of n.
- $Total_{O–MH}$ and $Total_{O–BS}$ denote the total number of operations of MH and BS in our scheme respectively.
- $Total_{W–MH}$ and $Total_{W–BS}$ denote the total number of operations of MH and BS in Wong *et al.*'s scheme, respectively.

5.1. Storage analysis

Comparing the storage space, we adopt the assumption by Wong *et al.* in Reference [26]. In our scheme, the secret values (a_i and b_i) consist of $L_i(x)$, both values are like the private keys as the Wong *et al.*'s scheme. Similarly, the corresponding public key is one point of $L_i(x)$, B_i. If an adversary tries to guess the $L_i(x)$, the probability of guessing successfully is $\frac{1}{2^l} \cdot \frac{1}{2^l}$, where l is the length of bit of a_i and b_i. However, many guessing failures will be detected by the BS, the adversary can not guess $L_i(x)$ all the time. Here, we assume that l is 160 bits, the same as the private key. The storage comparison is shown in Table 1.

Table 1. A comparison to the storage space of the MH.

	Wong *et al.*'s scheme	Our scheme
ID_{MH}	128 bits	128 bits
Private key/a_i and b_i	2×160 bits	2×160 bits
Public key/B_i	2×512 bits	320 bits
$H(\cdot)$	0 bit	128 bits
Signature	512 bits	512 bits
$Total_{one\text{-}round}$	1984 bits	1408 bits
$Total_{n\text{-}rounds}$	$128 + 1856n$ bits	$128 + 1280n$ bits

n is total rounds of that MH wants to perform the protocol.

5.2. Computation complexity

The analysis of computation cost is shown in the Appendix. The total number of operations of the MH in the Wong *et al.*'s scheme and our scheme are as following:

$$\begin{aligned} Total_{W-MH} &= 2MOD(32) + 131M(32) + 635A(32) \\ &\quad + 124S + O_{PK} + O_{SK} + 2XOR \end{aligned}$$

and

$$\begin{aligned} Total_{O-MH} &= 135M(32) + 830A(32) \\ &\quad + 132S + O_{PK} + O_{SK} + XOR. \end{aligned}$$

Although the total number of operations of the MH of our scheme is greater $4M(32)$, $185A(32)$ and $8S$ than that of Wong *et al.*'s scheme, but our scheme is less $2MOD(32)$ and XOR than that of Wong *et al.*'s scheme. In 32-bits microprocessor [13], the computing time is almost the same between Wongs' and our scheme.

On the other hand, in most protocols for wireless settings, although the computation complexity of the MH is reduced, the BS still has to do very complex computations. Unfortunately, as the number of MH increases, the BS may have to deal with many high complex computations simultaneously and the communication transactions will be delayed. Therefore, we analyze our scheme and Wong *et al*'s scheme in term of the computation complexity of the BS further. The total number of operations of the BS in Wong *et al.*'s scheme and our scheme are as following:

$$\begin{aligned} Total_{W-BS} &= 961MOD(32) + 192721M(32) \\ &\quad + 1316895A(32) + 188876S + O_{PK} \\ &\quad + O_{SK} + 2XOR \end{aligned}$$

and

$$\begin{aligned} Total_{O-BS} &= 162M(32) + 1000A(32) + 158S \\ &\quad + H(\cdot) + O_{PK} + O_{SK} + XOR. \end{aligned}$$

Although our scheme requires a hash function $H(\cdot)$, but is less 961MOD(32), 191736M(32), 1315895A(32), 188718S and one XOR than Wongs'.

5.3. Total size of communication messages

Last, we analyze the total size of communication messages as follows. In Wong *et al.*'s scheme, the total of communication messages include 1664 bits of $\text{Cert}^i_{\text{MH}}$, 64 bits of r_{BS}, 512 bits of x, 160 bits of y and 512 bits of $E_\sigma(x)$. The total size of communication messages of Wong *et al.*'s scheme is 2912 bits. In our scheme, the total of communication messages include 1038 bit of $\text{Cert}^i_{\text{MH}}$, 512 bits of $\text{E}_{\text{PK}_{\text{BS}}}(A_i)$, 320 bits of W, 160 bits of Y and 64 bits of $\text{E}_\sigma(r_{\text{MH}})$. The total size of communication messages of our scheme is 2094 bits.

6. CONCLUSION

In this paper, we have proposed a new scheme which uses the geometric properties of lines to achieve authentication and key exchange. After the security and performance analysis, our proposed scheme is proven to be efficient and able to withstand the replay attack and the unknown key-share attack [3,6]. Furthermore, our proposed scheme requires low computation power on both the MHs side and the BSs side. Therefore, our proposed scheme is suitable for the application in wireless settings.

APPENDIX

In terms of computation complexity, to analyze more precisely, we use the addition chain method [17] to analyze the complexity of computing the $(x^y \bmod z)$. The modular exponentiation can be denoted as $\text{MOD}_{\text{E}}(y,z)$ and shown as follows:

$$\text{MOD}_{\text{E}}(y,z) = 1.5l(y)[M(l(z)) + 2\text{MOD}(l(z)) + 1]. \quad (2)$$

And we analyze the number of operations for $M(512)$, $M(160)$, MOD(512) and MOD(160) by using divide and conquer [1], together with the computational analysis of divisions, multiplications and additions as outlined by Davida and Wells [9] is as follows:

$$D(n) = 3 \times M(n) + 2S,$$
$$M(n) = \begin{cases} 1, & \text{if } n = 32 \\ 3M(n/2) + 5A(n) + 2S & \text{if } n > 32 \end{cases}$$

and

$$A(n) = \begin{cases} 1, & \text{addition if } n = 32 \\ k, & \text{additions if } n = 32k \end{cases}$$

If one uses the recursion down to the 32-bit level then

$$M(160) = 27M(32) + 160A(32) + 26S,$$
$$M(512) = 81M(32) + 650A(32) + 80S$$

Using divide and conquer, the number of modulo is thus as follows:

$$\text{MOD}(n) = \begin{cases} 1, & \text{if } n = 32 \\ \text{MOD}(n/2) + 4M(n/2) & \text{if } n > 32 \\ \quad + 3/2A(n) + 3S & \end{cases}$$

If one uses the recursion down to the 32-bit level then

$$\begin{aligned} \text{MOD}(160) &= \text{MOD}(32) + 52M(32) \\ &\quad + 235A(32) + 49S, \text{MOD}(512) \\ &= \text{MOD}(32) + 160M(32) \\ &\quad + 1045A(32) + 156S \end{aligned}$$

In term of the computation complexity of the MH, the total number of operations of the MH in the Wong *et al.*'s scheme is equal to

$$\begin{aligned} \text{Total}_{\text{W-MH}} &= \text{Mod}_{\text{M}}(160) + \text{Mod}_{\text{A}}(160) + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} \\ &\quad + \text{XOR}, \\ &= M(160) + \text{MOD}(160) + A(160) \\ &\quad + \text{MOD}(160) + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} + 2\text{XOR}, \\ &= 2\text{MOD}(32) + 131M(32) + 635A(32) \\ &\quad + 124S + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} + 2\text{XOR} \end{aligned} \quad (3)$$

In our scheme, the total number of operations of the MH is equal to

$$\begin{aligned} \text{Total}_{\text{O-MH}} &= M(160) + A(320) + R_{L(x)} + \text{O}_{\text{PK}} \\ &\quad + \text{O}_{\text{SK}} + \text{XOR}, \\ &= 2M(160) + 2A(320) \\ &\quad + 2A(160) + D(160) + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} + \text{XOR}, \\ &= 135M(32) + 830A(32) \\ &\quad + 132S + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} + \text{XOR} \end{aligned} \quad (4)$$

The total number of operations of the BS in Wong *et al.*'s scheme is equal to

$$\begin{aligned} \text{Total}_{\text{W-BS}} &= 2\text{MOD}_E(160,512) + \text{MOD}_M(512) \\ &\quad + \text{O}_{\text{PK}} + \text{O}_{SK} + 2X\text{OR}, \\ &= 961\text{MOD}(32) \\ &\quad + 192721M(32) + 1316895A(32) \\ &\quad + 188876S + \text{O}_{\text{PK}} + \text{O}_{\text{SK}} + 2\text{XOR} \end{aligned} \quad (5)$$

where

$$\begin{aligned}\mathrm{Mod_E}(160,512) &= 1.5\times 160[M(512)+2\mathrm{MOD}(512)+1]\\ &= 480\mathrm{MOD}(32)+96240M(32)\\ &\quad +657600A(32)+94320S\end{aligned}$$

In our scheme, the total number of operations of the BS is equal to

$$\begin{aligned}\mathrm{Total_{O-BS}} &= 2M(160)+2A(320)+R_{L(x)}+H(\cdot)+\mathrm{O_{PK}}\\ &\quad +\mathrm{O_{SK}}+\mathrm{XOR},\\ &= 3M(160)+D(160)\\ &\quad +3A(320)+2A(160)+\mathrm{H}(\cdot)+\mathrm{O_{PK}}+\mathrm{O_{SK}}\\ &\quad +\mathrm{XOR},\\ &= 162M(32)+1000A(32)+158S\\ &\quad +H(\cdot)+\mathrm{O_{PK}}+\mathrm{O_{SK}}+\mathrm{XOR} \qquad (6)\end{aligned}$$

ACKNOWLEDGEMENTS

We are grateful to the anonymous reviewer and Ting-Yi Chang whose comments have helped us to improve the protocol. Part of this research was supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC 91-2622-E-324-010-CC3.

REFERENCES

1. Aho AV, Hopcroft JE, Ullman JD. *The Design and Analysis of Computer Algorithms*. Addison-Wesley: Massachusetts, 1974.
2. Aziz A, Diffie W. Privacy and authentication for wireless local area networks. *IEEE Personal Communications* 1994; **1**(1):24–31.
3. Baek J, Kim K. Remarks on the Unknown Key-Share Attacks. *IEICE Transactions on Fundamentals*, 2000; **E83-A**(12): 2766–2769.
4. Bellare M, Rogaway P. Provably Secure Session Key Distribution the Three Party Case. In *Proceedings of the 27th ACM Symposium on the Theory of Computing*, 1995.
5. Q1 Bellare M, Rogaway P. Entity authentication and key distribution. *Advances in Cryptology—CRYPTO'93* 1993; 232–249.
6. Blake-Wilson S, Menezes A. Unknown key-share attacks on the station-to-station (STS) protocol. In *Public Key Cryptography (PKC'99) Proceedings*, LNCS 1560, 1999; 154–170.
7. Blake-Wilson S, Menezes A. Authenticated Diffie-Hellman key agreement protocols. In *Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC'98)*, Lecture Notes in Computer Science, Vol. 1556, 1998; 339–361.
8. Q1 Damgard IB. A design principle for hash functions. In *Advances in Cryptology, CRYPTO'89*, 1989; 416–427. Q1
9. Davida GI, Wells DL, Kam JB. A database encryption system with subkeys. In *ACM Transactions on Database Systems* 1981; **6**:312–328.
10. Diffie W, Hellman ME. New directions in cryptography. In *IEEE Transactions on Information Theory* 1976; **IT-22**:644–654.
11. ElGamal T. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory* 1985; **31**(4):469–472.
12. Freier A, Karlton P, Kocher P. The SSL Protocol Version 3.0. *Internet-Draft*, Nov 1996.
13. Gupta A, Toong HMD. An architectural comparsion of 32-bits microprocessors. *IEEE Micro* 1983; **3**:9–22.
14. Hwang M-S, Lee C-H. Authenticated key-exchange in a mobile radio network. *European Transactions on Telecommunications* 1997; **8**(3):265–269.
15. Hwang M-S, Yang W-P. Conference key distribution protocols for digital mobile communication systems. *IEEE Journal on Selected Areas in Communications* 1995; **13**(2):416–420.
16. Q2 Jakobsson M, Pontcheval D. Mutual Authentication for Low-Power Mobile Devices. In *Proceeding of Financial Cryptography* 2001; Springer-Verlag. Q2
17. Knuth DE. *The Art of Computer Programming, Vol. 2 (Seminumerical Algorithms)*, (2nd edn). Addison-Wesley: Massachusetts, 1980.
18. Kohl J, Neuman C. The Kerberos Network Authentication Service (V5). In *IETF RFC1510*, September 1993.
19. Lee C-H, Hwang M-S, Yang W-P. Enhanced privacy and authentication for the global system of mobile communications. *Wireless Networks* 1999; **5**:231–243.
20. Merkle RC. One-way hash functions and DES. In *Advances in Cryptology, CRYPTO'89*, 428–446, Lecture Notes in Computer Science, Vol. **435**, 1989.
21. Q1 Needham RM, Schroeder MD. Using encryption for authentication in large networks of computers. *Communication of the ACM* 1978; 993–999. Q1
22. Q3 Otway D, Rees O. Efficient and timely mutual authentication. *Operating Systems Review*, 1987. Q3
23. Rivest R. The MD5 message digest algorithm. *Technical Report RFC 1321*, April 1992.
24. Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM*, 1978; **21**:120–126.
25. Q3 Wong DS, Chan AH. Efficient and mutually authenticated key exchage for Low Power Computing Device. In *Advances in cryptology, Asiacypt'01, LNCS 2248*, 2001; Springer-Verlag, 272–289. Q3 Q1
26. Wong DS, Chan AH. Mutual Authentication and Key Exchange for Low Power Wireless Communications. *Military Communications for Network-Centric Operations Conference Proceedings* 2001; **1**:39–43.

AUTHORS' BIOGRAPHIES

Chou-Chen Yang received his B.S. in Industrial Education from the National Kaohsiung Normal University, in 1980, his M.S. in Electronic Technology from the Pittsburg State University, in 1986 and his Ph.D. in computer science from the University of North Texas, in 1994. He has been an associate professor in Department of Computer Science and Information Engineering, since 1994. His current research interests include network security, mobile computing, and distributed system.

Jian-Wei Li received the B.S. in Information Engineering and Computer Science from Feng Chia University, Taichung, Taiwan, Republic of China (R.O.C.), from 1997 to 2001; the M.S. in Computer Science and Information Engineering from Chaoyang University of Technology, Taichung, Taiwan, in 2001 and in 2003. He is currently pursuing his Ph.D. in Computer Science and Information Engineering from National Cheng Kung University, Taiwan. His current research interests include information security, network security, cryptography, computer network and mobile computing.

Min-Shiang Hwang was born on 27 August 1960 in Tainan, Taiwan, Republic of China (R.O.C.). He received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, R.O.C., in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988 and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984 to 1986. Dr. Hwang passed the National Higher Examination in field 'Electronic Engineer' in 1988. He also passed the National Telecommunication Special Examination in field 'Information Engineering', qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999–2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002–2003. He obtained the 1997, 1998, 1999, 2000 and 2001 distinguished research awards of the National Science Council of the Republic of China. He is currently a professor of the department of Management Information Systems, National Chung Hsing University, Taiwan, ROC. He is a member of IEEE, ACM and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression and mobile computing. Dr Hwang had published 80 articles on the above research fields in international journals.

Author Query Form (ETT/957)

Special Instructions: Author please write responses to queries directly on Galley proofs and then fax back. Alternatively please list responses in an e-mail.

Q1: Author: Please provide the volume number for feferences [5],[8],[21]

Q2: Author: Please provide the pulisher location for reference [16], [25].

Q3: Author: Please provide complete publication details.