IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received January 10, 2017, accepted February 4, 2017, date of publication February 24, 2017, date of current version March 28, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2675163

A Provable Secure Private Data Delegation Scheme for Mountaineering Events in Emergency System

CHIEN-MING CHEN[1], CHUN-TA LI[2], (Member, IEEE), SHUAI LIU[1], TSU-YANG WU[3], AND JENG-SHYANG PAN[3]

[1]Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, China
[2]Department of Information Management, Tainan University of Technology, Tainan 71002, Taiwan
[3]Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fujian 350118, China

Corresponding author: C.-T. Li (th0040@mail.tut.edu.tw)

The work of C.-M. Chen was supported in part by the Project National Natural Science Foundation of China under Grant 61402135 and in part by the Shenzhen Technical Project under Grant JCYJ20150513151706574. The work of C.-T. Li was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 105-2221-E-165-005 and Grant MOST 105-3114-C-165-001-ES.

ABSTRACT Recently, the sport of mountaineering is a popular leisure activity and many people may injure while mountaineering. In the year of 2014, Chen *et al.* suggested a cloud-based emergency response and SOS system for mountaineering travelers when they encounter dangers. Chen *et al.* claimed that their proposed system is secure against various known attacks and the executive performance of the system is reasonable when the protocol is implemented on the traveler's mobile device. However, in this paper, we discover that Chen *et al.*'s scheme is unable to protect the privacy of mountaineering travelers and the vulnerability allows a malicious attacker to spy on the electronic medical records of all mountaineering travelers by launching eavesdropping attacks. Moreover, Chen *et al.*'s scheme is vulnerable to off-line password guessing attack when the mobile device of the mountaineering traveler is lost or stolen by an attacker. In order to repair these shortcomings existing in Chen *et al.*'s scheme, we suggest an improved version of their scheme, which is provably secure in the random oracle model under the DDH and CDH problems.

INDEX TERMS Cloud computing, emergency system, mobile device, provable security, traveler privacy, user authentication.

I. INTRODUCTION

Recently, outdoor sports, such as mountaineering, river tracing, rafting, etc., have become increasingly popular [1]. However, these kinds of sports often involve considerable dangers. Since these dangers may occur in solitary roads or desert hills, a rapid and safe first aid service is vital for emergency events. Fortunately, with the ever-changing nature of wireless communication technology and the popularity of smart phones, people in danger can easily and rapidly request emergency services.

In 2014, Chen et al. [2] proposed a platform based on cloud computing architecture. In their design, a traveler who is in danger and in need of rescue and send a SOS message to a mountain emergency service center with his smart phone. An investigator or staff of this mountain emergency service center then sends this emergency message to a suitable hospital nearby. Since this traveler may come from other countries and this hospital may not have any useful information about this traveler, this hospital can send an emergency message to CSDH (Cloud Server of Department of Health), which is a cloud server storing EMR (electronic medical record) of all patients, to acquire the EMR of this traveler. With the EMR, this hospital now can arrange a proper doctor for this traveler. To the best of our understanding, this platform is the first one designed for mountaineering events.

Chen et al. [2] also proposed a scheme to protect the secrecy and privacy for their platform. They adopt the Schnorr signature [3], [4], RSA [5], and ElGamal [6]. However, we still found Chen et al.'s scheme has the following drawbacks and weaknesses. First, this scheme fails to protect traveler privacy. In addition, this scheme suffers from unfriendly design in registration phase. This scheme also lacks a random nonce in the delegation phase and in the signing and verification phase. Furthermore, in Chen et al.'s scheme, they claimed that a traveler does not need to worry that his mobile device will be illegally used if his mobile device is lost or stolen by attackers. However, we found that the attacker may derive the password of the traveler by launching off-line password guessing attacks.

We also try to understand the reason why this scheme is rather insecure. It appears this scheme is a common structure problem - not being proven securely in a formal model. In this paper, we first demonstrate that Chen et al.'s scheme [2] still has some drawbacks. In order to fix the drawbacks existing in their scheme, we propose a new scheme that is provable secure in the random oracle model [7], [8] and under the decisional Diffie-Hellman (DDH) and the computational Diffie-Hellman (CDH) problems. According to the performance analysis, our scheme has better efficiency compared with Chen et al's scheme [2].

The reset of the paper is organized as follows. In Section II, we first introduce the entire architecture of a cloud-based emergency system and present Chen et al.'s authentication scheme for mountaineering events in Section III. In Section IV, we demonstrate some security and design drawbacks of Chen et al.'s scheme. The proposed scheme is demonstrated in Section V and we provide a formal security proof in Section VI. In Section VII, we analyze the efficiency of our proposed scheme and compare it with Chen et al.'s scheme. Finally, we conclude this paper in Section VIII.

II. THE ARCHITECTURE OF EMERGENCY SYSTEM FOR MOUNTAINEERING EVENTS

In cloud-based emergency system for mountaineering events, four parties participate in this system: Cloud Server of Department of Health ($CSDH$), Traveler (T), Investigator (I), and Doctor (D). Before accessing the system, every Traveler must register with the Cloud Server of Department of Health ($CSDH$) and $CSDH$ will issue some login parameters for the patient. Note that $CDSH$ is a trusted third party. Moreover, every Investigator must register with $CSDH$. Before mountaineering, T has to delegate the Investigator I and issued a signed warrant for I. When T encounters danger during mountaineering, T sends the SOS message to the designated I and I notifies the appointed Doctor D to prepare to diagnose T. Finally, D can download T's electronic medical record EMR from $CSDH$ and upload freshest EMR to $CSDH$ after diagnosing T. Fig 1 shows the entire architecture of Chen et al.'s emergency system for mountaineering events.

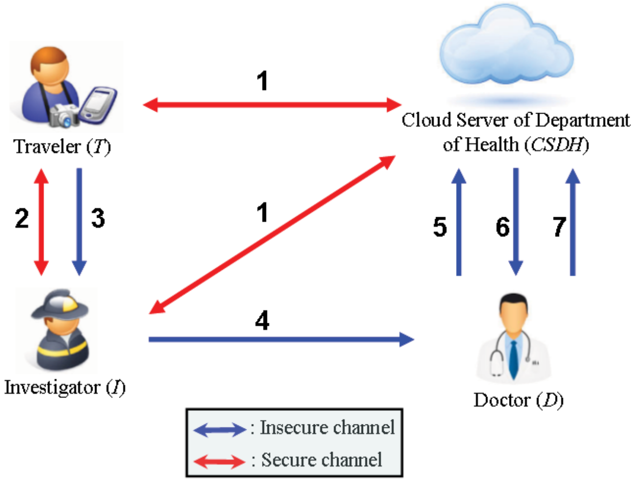

FIGURE 1. The entire architecture of Chen et al.'s cloud-based emergency system for mountaineering events [2].

Step 1. The Travel T goes to the the Cloud Server of the Department of Health $CSDH$ to register to be a legal user. In addition, the Investigator I also registers to be a legal collaborator with $CSDH$.

Step 2. Before mountaineering, the Traveler T has to inform the Investigator I of the mountain emergency service center.

Step 3. In case of danger, the Traveler T sends the SOS message to the designated Investigator I by using his/her mobile device.

Step 4. The Investigator I sends the emergency message to the appointed Doctor D by referring the predefined scheduling table of the hospital.

Step 5. The Doctor D forwards the emergency message to $CSDH$ to request the Traveler's EMR and prepares to diagnose the patient.

Step 6. The $CSDH$ verifies the Investigator's warrant and sends the Traveler's EMR to the Doctor D.

Step 7. After diagnosing the Traveler T, the Doctor D updates T's EMR and uploads it to $CSDH$.

III. REVIEW OF Chen et al.'s SCHEME

This section briefly reviews Chen et al.'s scheme [2] and there are six phases involve in Chen et al.'s scheme: registration phase, login phase, delegation phase, signing and verification phase, password change phase, and revoking the privacy phase. For convenience of description, terminology and notations used in the paper are summarized as follows:

- ID_X: The identity of party X.
- PW_X: The password of party X.
- p, q: Two large prime numbers with $q|(p-1)$.
- g: An element of order q in Z_p^*.
- m: The signed message.
- m_{req}: The SOS request message.
- $Time_i$: The time of the doctor finishes the diagnosis.
- $cert_w$: The warrant issued by Traveler.
- EMR: The electronic medical record of the patients.
- msk: The master secret key of CSDH.
- x_X/y_X: The private/public key pair of party X, where $y_X = g^{x_X} \bmod p$.
- SK_{ab}: The session key shared between a and b.
- $E_k(\cdot)/D_k(\cdot)$: The symmetric encryption/decryption function with key k.

- $sign_x$: A signature generation algorithm by using a private key x to sign a message.
- $verify_y$: A signature verification algorithm by using a public key y to verify the validity of the signature, where $y = g^x \bmod p$.
- $h(\cdot)$: A one-way hash function.
- MAC_i: The ith message authentication code.
- σ_i: The digital signature of m.
- r_X: The random number generated by X.
- N_X: The nonce value generated by X.
- $+$: The addition operation.
- $-$: The subtraction operation.
- $||$: The message concatenation.
- $\oplus$: The XOR operation.

A. REGISTRATION PHASE

In this phase, the Traveler T and Investigator I must register with the Cloud Server of Department of Health $CSDH$ through a secure channel to become legal identities. The details of registration phase are as follows:

Step 1. T submits the identity ID_T to $CSDH$.

Step 2. After receiving T's ID_T, $CSDH$ generates a random number r_{CSDH} and computes $U = ID_T + (EMR||Time_1)$, $G = h(ID_T||h(PW_T))$, $Q = h(r_{CSDH}||msk) \oplus G$, and $V = h(ID_T||h(PW_T) \oplus h(r_{CSDH}||msk))$. Then $CSDH$ stores ID_T and U into the cloud server and sends the message $\{Q, V, h(\cdot)\}$ to T.

Step 3. I submits the identity ID_I to $CSDH$.

Step 4. After obtaining the identity of I, $CSDH$ examines the eligibility of I by a face-to-face manner. $CSDH$ generates a nonce value $N_{CSDH} \in Z_q^*$ and computes $B_{CSDH} = g^{N_{CSDH}} \bmod p$, $S_{CSDH} = (x_{CSDH} \cdot h(ID_i, B_{CSDH}) + N_{CSDH}) \bmod q$ and $MAC_1 = h(ID_I||N_{CSDH})$. Then $CSDH$ stores ID_I and MAC_1 in the cloud server and sends the message $\{B_{CSDH}, S_{CSDH}, MAC_1\}$ to I.

Step 5. After receiving the message from $CSDH$, I verifies the validation of S_{CSDH} by checking whether $g^{S_{CSDH}} = B_{CSDH} \cdot y_{CSDH}^{h(ID_I, B_{CSDH})} \bmod p$.

B. LOGIN PHASE

Before T takes his/her mobile device to mountaineer, T performs the login procedures as follows:

Step 1. In order to verify T is the owner of the mobile device, T enters identity ID'_T and password PW'_T.

Step 2. The mobile device computes $h(r_{CSDH}||msk)' = Q \oplus h(ID'_T||h(PW'_T))$ and $V' = h(ID'_T||h(PW'_T) \oplus h(r_{CSDH}||msk)')$ and checks whether $V' = V$. If it holds, it means T is the legal owner of the mobile device.

C. DELEGATION PHASE

Before mountaineering, the Traveler T has to delegate the Investigator I on a secure channel (e.g., face-to-face). Then T can use his/her mobile device to send the SOS message to I when he/she encounters danger. The details of delegation phase are as follows:

Step 1. T and I establish a session key SK_{T-I}.

Step 2. T generates a random nonce $N_T \in Z_q^*$ and a warrant $cert_w$ and computes $B_T = g^{N_T} \bmod p$, $S_T = (x_T \cdot h(cert_w, B_T, ID_T) + N_T) \bmod q$ and $C_1 = E_{SK_{T-I}}(cert_w||B_T||S_T||ID_T||m_{req})$. Then T sends the message $\{C_1, ID_T\}$ to I.

Step 3. After receiving the message from T, I generates a random nonce N_I and computes $MAC_2 = h(ID_T||C_1||N_I)$. Then I stores MAC_2 into the database and sends MAC_2 to T. In case of danger, T can send the SOS message m_{req} and authorization information through his/her mobile device. The steps are as follows:

Step 4. T sends the SOS message $\{ID_T, C_1, MAC_2\}$ to I via a public channel.

Step 5. After receiving the message, I computes $MAC'_2 = h(ID_T||C_1||N_I)$ and checks whether $MAC'_2 = MAC_2$. If it holds, I uses the session key SK_{T-I} to decrypt the message C_1 by computing $D_{SK_{T-I}}(C_1) = (cert_w||B_T||S_T||ID_T)$. Then I verifies the validity of $cert_w$, S_T and ID_T by checking $g^{S_T} = B_T \cdot y_T^{h(cert_w, B_T, ID_T)} \bmod p$. If it holds, I immediately sends the emergency message to $CSDH$.

D. SIGNING AND VERIFICATION PHASE

In this phase, I sends the emergency message to $CSDH$ via a proxy signature. Then $CSDH$ verifies the validity of I and sends EMR to the Doctor D. The details of this phase are as follows:

Step 1. Before executing a mission, I and $CSDH$ establish a session key SK_{I-CSDH} by RFC 2631 key agreement protocol via a secure channel in off-line model.

Step 2. Before executing a mission, D and $CSDH$ establish a session key SK_{CSDH-D} by RFC 2631 key agreement protocol via a secure channel in off-line model.

Step 3. I computes the proxy signing key $x = (S_T + S_{CSDH}) \bmod q$ and uses x to sign message m by computing $\sigma_1 = Sign_x(m)$, where $m = (B_T, S_T, B_{CSDH}, cert_w, ID_I, ID_T, ID_D, m_{req})$. Then I computes $C_2 = E_{SK_{I-CSDH}}(m||\sigma_1)$ and sends $\{ID_I, C_2, MAC_1, ID_D\}$ to the appointed Doctor by referring the predefined scheduling table of the Hospital.

Step 4. After receiving the message from I, D forwards the message $\{ID_I, C_2, MAC_1, ID_D\}$ to $CSDH$.

Step 5. Upon receiving the message from D, $CSDH$ computes $MAC'_1 = h(ID_I||N_{CSDH})$ and checks whether $MAC'_1 = MAC_1$. If it holds, $CSDH$ uses the session key SK_{I-CSDH} to decrypt the message C_2 by computing $D_{SK_{I-CSDH}}(C_2) = (m||\sigma_1)$. Then

$CSDH$ verifies the validation of warrant $cert_w$ by checking $g^{S_T} = B_T \cdot y_T^{h(cert_w, B_T, ID_T)} \bmod p$, $y = g^{S_T + S_{CSDH}} \bmod p$ and $m = Verify_y(\sigma_1)$. If it holds, according to ID'_T and ID'_D of m, $CSDH$ searches if there are the same identities ID_T and ID_D in the cloud database of $CSDH$ and takes the EMR out by computing $(EMR||Time_1) = U = ID'_T$. $CSDH$ further checks EMR and updates it with the time if it is not the freshest. After confirming EMR is the freshest, $CSDH$ computes $C_3 = E_{SK_{CSDH-D}}(EMR||Time_1)$ and sends it to D.

Step 6. Upon receiving C_3 from $CSDH$, D uses the session key SK_{CSDH-D} to decrypt the message C_3 by computing $D_{SK_{CSDH-D}}(C_3) = (EMR||Time_1)$. After diagnosing, D adds the prescriptions and updates EMR to EMR_{new}. Moreover, D uses his/her private key x_D to sign the message by computing $\sigma_2 = Sign_{x_D}(EMR_{new}||Time_2||ID_D)$ and sends the signed message $\{\sigma_2, EMR_{new}, Time_2, ID_D\}$ to $CSDH$.

Step 7. After receiving the message from D, $CSDH$ uses D's public key y_D to verify the validity of signed message by checking $(EMR_{new}||Time_2||ID_D) = verify_{y_D}(\sigma_2)$. Then $CSDH$ adds T's identity ID_T on the EMR by computing $U_{new} = ID_T + (EMR_{new}||Time_2)$ and stores U_{new} in the cloud server.

E. PASSWORD CHANGE PHASE

When the Traveler T wants to change his/her original password PW_T to a new password $PW_{T_{new}}$, T has to perform the authentication steps which are the same as in the login phase.

Step 1. After confirming that T is a legal owner of the mobile device, T can input a new password and compute $Q_{new} = Q \oplus h(ID_T||h(PW_T) \oplus h(ID_T||PW_{T_{new}})) = h(r_{CSDH}||msk) \oplus h(ID_T|| h(PW_{T_{new}}))$.

Step 2. Finally, T replaces Q with Q_{new} and finishes this phase.

F. REVOKING THE PRIVACY PHASE

If an Investigator I goes against the $CSDH$'s rule, $CSDH$ publishes I's identity ID_I to revoke the I's authority.

IV. DRAWBACKS OF Chen et al.'s SCHEME

In this section, we demonstrate that Chen et al.'s scheme has some security and design drawbacks which are described in the following subsections.

A. FAILS TO PROTECT TRAVELER PRIVACY

In signing and verification phase of Chen et al.'s scheme, after diagnosing, the Doctor D sends the message $\{\sigma_2, EMR_{new}, Time_2, ID_D\}$ to $CSDH$ via an insecure channel. Note that EMR_{new} is transmitted in plaintext format without using any encryption methods. Because of this design, the attacker can know the freshest electronic medical record EMR_{new} of the Traveler T by eavesdropping attack. As a result, we show that Chen et al.'s scheme cannot achieve the requirement of Traveler privacy.

B. UNFRIENDLY DESIGN IN REGISTRATION PHASE

In registration phase of Chen et al.'s scheme, the Traveler T chooses the identity ID_T and sends it to $CSDH$ via a secure channel. Then $CSDH$ computes some parameters (U, G, Q, V) and sends the message $\{Q, V, h(\cdot)\}$ to T, where $G = h(ID_T||h(PW_T))$ and $Q = h(r_{CSDH}||msk) \oplus G$. Note that the password PW_T is randomly generated by $CSDH$ and the Traveler T cannot freely choose the password he/she wants. Moreover, the Traveler T cannot derive password PW_T from the received message $\{Q, V, h(\cdot)\}$ and this design flaw will affect the authentication of mobile device holder during login phase. Therefore, Chen et al.'s scheme suffers from unfriendly design in registration phase.

C. LACK OF RANDOM NONCE IN DELEGATION PHASE

In the design of delegation phase of Chen et al.'s scheme, we observe that the Investigator I only stores MAC_2 in its database without storing the random nonce N_I. Consider that the Traveler T sends the SOS message $\{ID_T, C_1, MAC_2\}$ to I in Step 4 of delegation phase, where $MAC_2 = h(ID_T||C_1||N_I)$. However, in fact, I cannot compute $MAC'_2 = h(ID_T||C_1||N_I)$ and compare it with received MAC_2 without having the random nonce N_I.

D. LACK OF RANDOM NONCE IN SIGNING AND VERIFICATION PHASE

In the registration phase of Chen et al.'s scheme, we observe that $CSDH$ only stores ID_I and MAC_1 in the cloud server without storing the random nonce N_{CSDH}. Consider that the Investigator I sends $\{ID_I, C_2, MAC_1, ID_D\}$ to the appointed Doctor and D forwards the message to $CSDH$ in the signing and verification phase, where $MAC_1 = h(ID_I||N_{CSDH})$. However, in fact, $CSDH$ cannot compute $MAC'_1 = h(ID_I||N_{CSDH})$ and compare it with received MAC_1 without knowing the random nonce N_{CSDH}.

E. STOLEN MOBILE DEVICE ATTACKS

In Chen et al.'s scheme, they claimed that the Traveler does not need to worry his/her mobile device will be illegally used if the Traveler's mobile device is lost or stolen by attackers. However, we found that the attacker may derive the password of the Traveler by launching off-line password guessing attacks. We further provide the detailed explanation of this attack through the following steps:

Step 1. The attacker eavesdrops the Traveler's identity ID_T from public channels.

Step 2. The attacker collects the parameters $(Q, V, h(\cdot))$ stored in Traveler's mobile device, where $Q = h(r_{CSDH}||msk) \oplus h(ID_T||h(PW_T))$ and $V = h(ID_T||h(PW_T) \oplus h(r_{CSDH}||msk))$.

Step 3. The attacker guesses a candidate password PW_T^* and computes $G^* = h(ID_T||h(PW_T^*))$, $R^* = Q \oplus G^*$ and $V^* = h(ID_T||h(PW_T^*) \oplus R^*)$ in off-line manner.

Step 4. The attacker checks whether the computed V^* is equal to the stored V or not. If it is equal, the Traveler's password is successfully guessed. Otherwise, the attacker repeats Step 3 and Step 4 until the correct password is found.

From the above descriptions, any password stored in Traveler's mobile device will not safe because there is unable to limit attacker's off-line computation.

V. THE PROPOSED SCHEME

In this section, an improved scheme is proposed to repair the drawbacks existing in [2] and the presented scheme is composed of seven phases: registration phase, login phase, delegation phase, verification phase, recovery phase, password change phase, and revoking the privacy phase. The detailed descriptions of the proposed scheme are as follows.

A. REGISTRATION PHASE

In this phase, the Traveler T, Investigator I and Doctor D must register with the Cloud Server of Department of Health $CSDH$ through a secure channel to become legal identities. The details of registration phase are as follows:

Step 1. T chooses the identity ID_T and password PW_T and submits them to $CSDH$.

Step 2. After receiving T's ID_T and PW_T, $CSDH$ generates a random number $r_{CSDH-T} \in_R [1, p-1]$ and computes $Y_{CSDH-T} = g^{r_{CSDH-T}} \bmod p$ and $R_{CSDH-T} = (Y_{CSDH-T} + PW_T) \bmod p$. Then $CSDH$ stores ID_T, r_{CSDH-T} and R_{CSDH-T} into its private cloud server and sends the message $\{R_{CSDH-T}\}$ to T. Then T stores R_{CSDH-T} into its mobile device.

Step 3. I submits the identity ID_I to $CSDH$.

Step 4. After receiving the identity of I, $CSDH$ examines the eligibility of I by a face-to-face manner. Then $CSDH$ generates a nonce value $N_{CSDH-I} \in Z_q^*$ and computes $K_{CSDH-I} = y_I^{x_{CSDH}} \bmod p$, where $y_I = g^{x_I}$ is I's public key. Then $CSDH$ stores ID_I, K_{CSDH-I} and N_{CSDH-I} into its private cloud server and sends the message $\{N_{CSDH-I}\}$ to I.

Step 5. After receiving the message from $CSDH$, I computes $K_{I-CSDH} = y_{CSDH}^{x_I} \bmod p$ and stores K_{I-CSDH} with N_{CSDH-I}, where $y_{CSDH} = g^{x_{CSDH}}$ is $CSDH$'s public key.

Step 6. Before accessing the system, the Doctor D and $CSDH$ construct a common secret key $K_{CSDH-D} = (y_D)^{x_{CSDH}} \bmod p = g^{x_D x_{CSDH}} \bmod p = (y^{CSDH})^{x_D} \bmod p = K_{D-CSDH}$ and $CSDH$ stores ID_D and K_{CSDH-D} into its private cloud server, where $y_D = g^{x_D}$ is D's public key.

B. LOGIN PHASE

Before T takes his/her mobile device to mountaineer, T needs to login $CSDH$ and performs the login procedures as follows:

Step 1. In order to verify T is the owner of the mobile device, T enters identity ID_T and password PW_T. Then T's mobile device generates a random number $r_T \in_R [1, p-1]$ and computes $Y_T = g^{r_T} \bmod p$, $Y_{T-CSDH} = R_{CSDH-T} - PW_T$, $K_{T-CSDH} = (Y_{T-CSDH})^{r_T} \bmod p$ and $V_T = h(ID_T||Y_T||K_{T-CSDH}||Time_T)$, where $Time_T$ is the current timestamp of T. After computed the above parameters, T's mobile device sends the message $\{ID_T, V_T, Y_T, Time_T\}$ to $CSDH$ via a public channel.

Step 2. Upon receiving the message from T, $CSDH$ first checks if $Time_{CSDH} - Time_T \leq \Delta T$ holds, where $Time_{CSDH}$ is the current timestamp of $CSDH$ and $\leq \Delta T$ is a preset transmission delay. If the time interval is not valid, the session is terminated by $CSDH$; otherwise, $CSDH$ computes $K_{CSDH-T} = Y_T^{r_{CSDH-T}} \bmod p$ and $V'_T = h(ID_T||Y_T||K_{CSDH-T}||Time_T)$ and checks whether $V'_T = V_T$. If it holds, $CSDH$ further computes $SK_{CSDH-T} = (Y_T)^{x_{CSDH}} \bmod p$ and $V_{CSDH} = h(ID_T||K_{CSDH-T}||SK_{CSDH-T}||Time_{CSDH})$ and sends the message $\{success, V_{CSDH}, Time_{CSDH}\}$ to the mobile device of T via a public channel. $CSDH$ stores SK_{CSDH-T} with ID_T.

Step 3. Upon receiving the message from $CSDH$, T's mobile device checks if $Time_T - Time_{CSDH} \leq \Delta T$ holds. If it is valid, the mobile device computes $SK_{T-CSDH} = (y_{CSDH})^{r_T} \bmod p$ and $V'_{CSDH} = h(ID_T||K_{T-CSDH}||SK_{T-CSDH}|| \; Time_{CSDH})$ and checks whether $V'_{CSDH} = V_{CSDH}$. If it holds, it means T is successful to log into $CSDH$ and stores r_T and SK_{T-CSDH} into his/her mobile device.

C. DELEGATION PHASE

Before mountaineering, the Traveler T has to delegate the Investigator I on a secure channel (e.g., face-to-face). Then T and I communicate with $CSDH$ to identify each other and construct a session key for securing later communications. When T encounters danger, T can use his/her mobile device to send the SOS message to I and the details of delegation phase are as follows:

Step 1. T utilizes his/her mobile device to compute $Q_T = h(ID_I||ID_T||SK_{T-CSDH}||Time_T)$ and sends $\{ID_T, ID_I, Q_T, Time_T\}$ to $CSDH$ via a public channel.

Step 2. As T sending the message to $CSDH$, I does the similar steps and sends the message $\{ID_I, ID_T, Q_I, Time_I\}$ to $CSDH$ via a public channel, where $Q_I = h(ID_I||ID_T||K_{I-CSDH}||Time_I)$.

Step 3. After receiving two messages from T and I, $CSDH$ checks if $Time_{CSDH} - Time_T \leq \Delta T$ and $Time_{CSDH} - Time_I \leq \Delta T$. If both of them are valid, $CSDH$ computes $Q'_T = h(ID_I||ID_T||SK_{CSDH-T}||Time_T)$ and $Q'_I = h(ID_T|| ID_I||K_{CSDH-I}||Time_I)$ and checks whether $Q'_T = Q_T$ and $Q'_I = Q_i$. If both of them are valid, $CSDH$ generates a random number $N_{CSDH-T,I} \in Z_q^*$ with the same bit lengths of K and SK and computes the following parameters:

$$\begin{aligned} M_{CSDH-T} &= N_{CSDH-T,I} \oplus SK_{CSDH-T} \oplus \\ &\quad \times R_{CSDH-T} \\ M_{CSDH-I} &= N_{CSDH-T,I} \oplus K_{CSDH-I} \oplus \\ &\quad \times N_{CSDH-I} \\ Q_{CSDH-T} &= h(ID_I||SK_{CSDH-T}||N_{CSDH-T,I}|| \\ &\quad \times Time_{CSDH}) \\ Q_{CSDH-I} &= h(ID_T||K_{CSDH-I}||N_{CSDH-T,I}|| \\ &\quad \times Time_{CSDH}) \end{aligned}$$

Then $CSDH$ sends $\{ID_{CSDH}, ID_I, Q_{CSDH-T}, Time_{CSDH}, M_{CSDH–T}\}$ and $\{ID_{CSDH}, ID_T, Q_{CSDH–I}, Time_{CSDH}, M_{CSDH-I}\}$ to T and I, respectively.

Step 4. After receiving the message from $CSDH$, T's mobile device checks if $Time_T - Time_{CSDH} \leq \Delta T$. If it holds, the mobile device computes $N'_{CSDH-T,I} = M_{CSDH-T} \oplus SK_{T-CSDH} \oplus R_{CSDH-T}$ and $Q'_{CSDH-T} = h(ID_I||SK_{T-CSDH}||N'_{CSDH-T,I}||Time_{CSDH})$ and checks whether $Q'_{CSDH-T} = Q_{CSDH-T}$. If it holds, the mobile device stores $N_{CSDH-T,I}$ with ID_I into its memory space.

Step 5. After receiving the message from $CSDH$, I checks if $Time_I - Time_{CSDH} \leq \Delta T$. If it holds, I computes $N'_{CSDH-T,I} = M_{CSDH-I} \oplus K_{I-CSDH} \oplus N_{CSDH-I}$ and $Q'_{CSDH-I} = h(ID_T||K_{I-CSDH}||N'_{CSDH-T,I}||Time_{CSDH})$ and checks whether $Q'_{CSDH-I} = Q_{CSDH-I}$. If it holds, I stores $N_{CSDH-T,I}$ with ID_T.

Step 6. In case of danger, T can use his/her mobile device to compute Q_{T-CSDH}=$h(ID_T||SK_{T-CSDH}||m_{req})$ and $Q_{T-I} = h(ID_T||N_{CSDH-T,I}||Time_T||m_{req})$. Then T sends $\{ID_T, m_{req}, Q_{T-CSDH}, Q_{T-I}, Time_T\}$ with his/her location to I via a public channel.

Step 7. After receiving the message from T, I checks if $Time_T - Time_I \leq \Delta T$ holds. If it holds, I computes $Q'_{T-I} = h(ID_T||N_{CSDH-T,I}||Time_T||m_{req})$ and checks whether $Q'_{T-I} = Q_{T_i}$. If it is valid, I will transmit SOS message to $CSDH$ in next phase.

D. VERIFICATION PHASE

In this phase, I sends the emergency message to $CSDH$ via a proxy signature. Then $CSDH$ verifies the validity of I and sends EMR to the Doctor D. The details of this phase are as follows:

Step 1. After checking the SOS request message, the Investigator I computes Q_{I-CSDH}=$h(ID_I \,||\, K_{I-CSDH} ||Time_I||m_{req}||ID_D)$ and sends the message $\{ID_T, ID_I, Q_{T-CSDH}, Q_{I-CSDH}, Time_I\}$ to $CSDH$ via a public channel.

Step 2. After receiving the message from I, $CSDH$ checks if $Time_{CSDH}$-$Time_I \leq \Delta T$. If it holds, $CSDH$ computes Q'_{I-CSDH}=$h(ID_I||K_{CSDH–I}||Time_I||m_{req}|| ID_D)$ and Q'_{T-CSDH}=$h(ID_T||SK_{CSDH-T}||m_{req})$ and checks whether Q'_{I-CSDH}=Q_{I-CSDH} and Q'_{T-CSDH}=Q_{T-CSDH}. If they are valid, $CSDH$ computes Q_{CSDH-D}=$h(ID_{CSDH} \,||\, ID_T \,||\, K_{CSDH–D}|| Time_{CSDH})$ and C_1=$E_{K_{CSDH-D}}(EMR||time_1)$ and sends the message $\{ID_{CSDH}, ID_T, Q_{CSDH-D}, C_1, Time_{CSDH}\}$ to D via a public channel, where $(EMR||time_1)$ is the most recently EMR of T.

Step 3. After receiving the message from $CSDH$, D checks if $Time_D - Time_{CSDH} \leq \Delta T$. If it holds, D computes $Q'_{CSDH-D} = h(ID_{CSDH}||ID_T||K_{D–CSDH}||Time_{CSDH})$ and checks whether $Q'_{CSDH-D} = Q_{CSDH-D}$. If it is valid, D reveals $(EMR||time_1)$ by computing $D_{K_{D-CSDH}}(C_1)$.

Step 4. After diagnosing, D adds the prescriptions and updates EMR to EMR_{new}. Then D computes $C_2 = E_{K_{D-CSDH}}(EMR_{new}||Time_D)$ and $Q_{D-CSDH} = h(ID_D||ID_T||K_{D-CSDH}||Time_D)$ and sends the message $\{ID_D, ID_T, Q_{D-CSDH}, C_2, Time_D\}$ to $CSDH$ via a public channel.

Step 5. After receiving the message from D, $CSDH$ checks if $Time_{CSDH} - Time_D \leq \Delta T$. If it holds, $CSDH$ computes $Q'_{D-CSDH} = h(ID_D||ID_T||K_{CSDH-D}||Time_D)$ and checks whether $Q'_{D-CSDH} = Q_{D-CSDH}$. If it is valid, $CSDH$ reveals $(EMR_{new}||Time_D)$ by computing $D_{K_{CSDH-D}}(C_2)$ and sets $time_2$ as $Time_D$. Finally, $CSDH$ replaces T's $(EMR||time_1)$ with $(EMR_{new}||time_2)$.

E. PASSWORD CHANGE PHASE

When the Traveler T wants to change his/her original password PW_T to a new password $PW_{T_{new}}$, T must input his/her identity ID_T and old password PW_T to start the mobile device and input the new password $PW_{T_{new}}$.

Step 1. T's mobile device computes $PW_{T_{new}} \oplus h(SK_{T-CSDH}||Time_T)$ and sends the message $\{ID_T, PW_{T_{new}} \oplus h(SK_{T-CSDH}||Time_T), Time_T\}$ to $CSDH$ via a public channel.

Step 2. After receiving the message from T's mobile device, $CSDH$ checks if $Time_{CSDH} - Time_T \leq \Delta T$. If it holds, $CSDH$ reveals T's new password $PW_{T_{new}}$ by computing $PW_{T_{new}} \oplus h(SK_{T-CSDH}||Time_T) \oplus h(SK_{CSDH-T}||Time_T)$ and further computes $Y_{new} = g^{r_{new}} \bmod p$, $R_{new} = (Y_{new} + PW_{T_{new}}) \bmod p$ and $Q_{new} = h(ID_T||SK_{CSDH-T}||Time_{CSDH})$, where $r_{new} \in_R [1, p - 1]$ and it is randomly chosen by $CSDH$. Then $CSDH$ replaces $(ID_T, r_{CSDH-T}, R_{CSDH-T})$ with (ID_T, r_{new}, R_{new}) sends the message $\{Q_{new}, R_{new}, Time_{CSDH}\}$ to T via a public channel.

Step 3. After receiving the message from $CSDH$, T's mobile device checks if $Time_T - Time_{CSDH} \leq \Delta T$. If it holds, T's mobile devic computes $Q'_{new} = h(ID_T||SK_{T-CSDH}||Time_{CSDH})$ and checks whether $Q'_{new} = Q_{new}$. If it is valid, T replaces R_{CSDH-T} with R_{new} and finishes this phase.

F. RECOVERY PHASE

In case of Traveler's mobile device is lost or stolen by attackers, T can smoothly shift the emergency service to his/her new mobile device without changing his/her identity with $CSDH$ and the detailed steps of recovery phase are shown as follows:

Step 1. T inputs the identity ID_T and the password PW'_T into his/her new mobile device . Then T's new mobile device computes $Y_T = g^{r_T} \bmod p$, $SK_{T-CSDH} = (y_{CSDH})^{x_T} \bmod p$ and $U_{T-CSDH} = PW'_T \oplus h(Y_T||SK_{T_CSDH}||Time_T)$ and sends $\{ID_T, U_{T-CSDH}, Y_T, Time_T\}$ to $CSDH$ via a public channel, where $r_T \in_R [1, p-1]$ and it is randomly chosen by T.

Step 2. After receiving the message from T's new mobile device, $CSDH$ checks if $Time_{CSDH} - Time_T \leq \Delta T$. If it holds, $CSDH$ computes $SK_{CSDH-T} = (y_T)^{x_{CSDH}} \bmod p$, $PW_T" = U_{T-CSDH} \oplus h(Y_T||SK_{CSDH-T}||Time_T)$, $Y_{CSDH-T} = g^{r_{CSDH-T}} \bmod p$ and $R'_{CSDH-T} = (Y_{CSDH-T} + PW_T") \bmod p$ and checks whether $R'_{CSDH-T} = R_{CSDH-T}$. If it holds, it means $PW'_T = PW_T"$ and $CDSH$ convinces that T is a valid Traveler.

Step 3. $CSDH$ computes $H_{CSDH-T} = h(ID_{CSDH}|| SK_{CSDH-T}||Time_{CSDH})$ and $U_{CSDH-T} = {}_{CSDH-T} \oplus h(ID_T||SK_{CSDH-T}||Time_{CSDH})$ and sends $\{ID_{CSDH}, H_{CSDH-T}, U_{CSDH-T}, Time_{CSDH}\}$ to T via a public channel.

Step 4. After receiving the message from $CSDH$, T checks if $Time_T - Time_{CSDH} \leq \Delta T$. If it holds, T computes $H'_{CSDH-T} = h(ID_{CSDH}||SK_{T-CSDH}||Time_{CSDH})$ and checks whether $H'_{CSDH-T} = H_{CSDH-T}$. If it holds, T further computes $R_{CSDH-T} = U_{CSDH-T} \oplus h(ID_T||SK_{T-CSDH}||Time_{CSDH})$ and stores R_{CSDH-T} into his/her new mobile device.

G. REVOKING THE PRIVACY PHASE

In this phase, the executed steps are the same as Chen et al.'s scheme.

VI. SECURITY PROOFS OF THE PROPOSED SCHEME

Various security protocols for different environments or applications have been proposed in the recent years. Some of them identified a vulnerability of their precedences and presented their improved protocols [9]–[14]. These papers formed a paper cracking loop and seemed never end. The root of the problem is that these protocols only work on a heuristic security analysis but not a formal security proof. For this reason, in this section we present a proof of security of the Random Oracle [7], [8] and a logic proof based on BAN logic [15].

A. SECURITY OF LOGIN PHASE

In this subsection, we show that the security of login phase in our scheme. Note that in the login phase, we use a two-party authenticated key exchange protocol. After executing the login phase, the traveler T and $CSDH$ can establish a common key to communication. Here, we adopt a modified security model based on [7], [16], [17] to analyze the security of login phase in our scheme. In this model, we define the following assumptions:

(1) There are two entities: Traveler T and $CSDH$;
(2) T can execute an authenticated key exchange (AKE) protocol with $CSDH$ repeatedly;
(3) Each entity involved in a session can be view as oracle. We denote a oracle Π^i_T as the i-th instance of T and a oracle Π^j_{CSDH} as the j-th instance of $CSDH$ in a session for $i, j \in \mathbb{N}$;
(4) There is an adversary $\mathcal{A}$ can access the oracle by issue some queries.

Note that according to the oracle queries between $\mathcal{A}$ and oracles, it can be used to simulate some attacks made by $\mathcal{A}$ in a real AKE protocol.

1) ADVERSARIAL MODEL

Here, we define the capability of adversary $\mathcal{A}$. An adversary $\mathcal{A}$ can be viewed as a probabilistic polynomial time (PPT) algorithm. We assume $\mathcal{A}$ can potentially control all communications in the networks and $\mathcal{A}$ is allowed to make the following queries. Let $E \in \{T, CSDH\}$.

- $Execute(T, CSDH)$: This query can be used to get a complete transcript of honest execution between T and $CSDH$. Note that this query models passive attack that $\mathcal{A}$ can eavesdrop a real execution in an AKE protocol.
- $Send(\Pi^i_E, m)$: This query can be used to send a message m to oracle Π^i_E. Upon receiving m, Π^i_E executes an AKE protocol and responds the result to $\mathcal{A}$. Note that $\mathcal{A}$ may make this query to launch some active attacks such as inserting, deleting, or modifying messages of AKE protocol. Thus, this query models impersonation attack and man-in-the-middle attack in an AKE protocol.
- $Reveal(\Pi^i_E)$: This query can be used to get a session key for oracle Π^i_E. Note that this query models known session key attack in an AKE protocol.
- $Corrupt(T)$: This query can be used to get password for T. Note that this query models forward secrecy in an AKE protocol.
- $Test(\Pi^i_E)$: Once $\mathcal{A}$ makes this query to fresh oracle Π^i_E, it randomly selects a coin $b \in \{0, 1\}$ and responds the session key, if $b = 1$. Otherwise, it returns a random string. Note that this query models the semantic security of session key.

Definition 1 (Fresh Oracle): We say that an oracle Π^i_E is called fresh, if (1) Π^i_E has accepted a session key; (2) Π^i_E and its partner haven't been made for a Reveal query; (3) Π^i_E hasn't been made for a Corrupt query.

2) SECURITY OF AKE PROTOCOL

The security of AKE protocol is defined by the following game between the adversary $\mathcal{A}$ and an infinite set of oracles Π^i_E for $E \in \{T, CSDH\}$ and $i \in \mathbb{N}$.

- Initialization: In this phase, public parameters are setting. Traveler's password and *CSDH*'s publc/private key pair are assigned through out the registration phase related to the parameters.
- Queries: $\mathcal{A}$ may make some queries to oracles and gets back the results corresponding to the AKE protocol.
- Test: Finally, $\mathcal{A}$ outputs its guess b' for the coin b in Test query and terminates.

In the above game, we define the advantage of $\mathcal{A}$ as the measurement of ability to distinguish a session key from a random string, ie. guessing b. Let *Succ* be the event that $\mathcal{A}$ correctly guesses the coin b in Test query. Then, the advantage (probability) of $\mathcal{A}$ in attacking an AKE protocol P is defined by $Adv_{\mathcal{A},P}(k) = |2 \cdot \Pr[Succ] - 1|$.

Definition 2 (Secure AKE): We say that an AKE protocol is called secure, if it satisfies the following two properties:

- Correctness: a oracle and its partner accepts the same key.
- Indistinguishability: for any adversary $\mathcal{A}$, the advantage $Adv_{\mathcal{A},P}(k)$ is negligible.

We first show that given Traveler's identity ID_T, forging Traveler's and CSDH's transcripts are intractable in the random oracle model [8], [18].

Lemma 1: In the random oracle model, assume that there is an attacker $\mathcal{A}$ can make at most q_H and q_S times to the Hash and Send queries to forge Traveler's transcript with a non-negligible advantage ϵ_1. Then, there exists a challenger $\mathcal{C}$ can solve the computational Diffie-Hellman (CDH) problem with an advantage $\epsilon_1' \geq \epsilon_1 - \frac{q_H \cdot q_S}{p \cdot 2^k}$, where k denotes the length of hash value.

Proof: $\mathcal{C}$ is given an instance (g, g^a, g^b) of the CDH problem, the goal of $\mathcal{C}$ is to compute g^{ab} for some $a, b \in \mathbb{Z}_p^*$. Then, $\mathcal{C}$ runs $\mathcal{A}$ as a subroutine and simulates the attack environment. Firstly, $\mathcal{C}$ set public parameters $\{G_p, g, p\}$ and sends it to $\mathcal{A}$. Then, $\mathcal{A}$ can make following queries to $\mathcal{C}$. Note that in order to avoid consistently and collusion of the results, $\mathcal{C}$ maintains a list L_H which initially empty.

- Hash query. When $\mathcal{A}$ makes a $Hash(m)$ query to $\mathcal{C}$, $\mathcal{C}$ returns a random number h and adds (m, h) into L_H.
- Send query. When $\mathcal{A}$ makes a $Send(\Pi_T^i, (ID_T, Time_{T_i}))$ query to $\mathcal{C}$, $\mathcal{C}$ selects $r_{T_i} \in_R \mathbb{Z}_p^*$, $Y_{T-CSDH_i} \in \mathbb{G}_p$ and computes $Y_{T_i} = g^{r_{T_i}}$, $K_{T-CSDH_i} = (Y_{T-CSDH_i})^{r_{T_i}}$, and $V_{T_i} \in L_H$. Finally, $\mathcal{C}$ returns (Y_{T_i}, V_{T_i}) to $\mathcal{A}$.

Eventually, $\mathcal{A}$ outputs a new valid message tuple $(ID_T, V_T, Y_T, Time_T)$. It means that *CSDH* accepts $(ID_T, V_T, Y_T, Time_T)$ but it has not been produced by Traveler T. Hence, it could be the following two situations:

1) $\mathcal{A}$ guesses that value (V_T, Y_T) with the probability less than $\frac{q_H}{2^k} \times \frac{q_S}{p}$.
2) $\mathcal{A}$ had asked for $(ID_T, Y_T, K_{T-CSDH}, Time_T)$ to Hash query.

We use symbol $Forge_T$ to denote the event that $\mathcal{A}$ forges Traveler's transcript. Thus, we can obtain

$$\Pr[Forge_T] \leq \Pr[V_T = h(ID_T||Y_T||K_{T-CSDH}||Time_T)| \\ K_{T-CSDH} \leftarrow \mathbb{G}_p] + \frac{q_H \cdot q_S}{p \cdot 2^k}.$$

Set $Y_T = g^a$ and $Y_{T-CSDH} = g^b$. The challenger $\mathcal{C}$ can use $\mathcal{A}$ to compute $K_{T-CSDH} = g^{ab}$. In other words, $\mathcal{C}$ can solve the CDH problem with advantage $\epsilon_1' \geq \epsilon_1 - \frac{q_H \cdot q_S}{p \cdot 2^k}$.

Lemma 2: In the random oracle model, assume that there is an attacker $\mathcal{A}$ can make at most q_H and q_S times to the Hash and Send queries to forge *CSDH*'s transcript with a non-negligible advantage ϵ_2. Then, there exists a challenger $\mathcal{C}$ can solve the CDH problem with an advantage $\epsilon_2' \geq \epsilon_2 - \frac{q_H}{2^k} - \frac{q_H \cdot q_S^2}{p \cdot 2^k}$, where k denotes the length of hash value.

Proof: $\mathcal{C}$ is given an instance (g, g^a, g^b) of the CDH problem, the goal of $\mathcal{C}$ is to compute g^{ab} for some $a, b \in \mathbb{Z}_p^*$. Then, $\mathcal{C}$ runs $\mathcal{A}$ as a subroutine and simulates the attack environment. Firstly, $\mathcal{C}$ set public parameters $\{G_p, g, p, y_{CSDH} = g^b\}$ and sends it to $\mathcal{A}$. Then, $\mathcal{A}$ can make following queries to $\mathcal{C}$. Note that in order to avoid consistently and collusion of the results, $\mathcal{C}$ maintains a list L_H which initially empty.

- Hash query. When $\mathcal{A}$ makes a $Hash(m)$ query to $\mathcal{C}$, $\mathcal{C}$ returns a random number h and adds (m, h) into L_H.
- Send query. When $\mathcal{A}$ makes a query $Send(\Pi_{CSDH}^i, (ID_T, Time_{CSDH_i}))$ to $\mathcal{C}$, $\mathcal{C}$ selects $r_{CSDH_i}, x_{CSDH_i} \in_R \mathbb{Z}_p^*$ and computes $K_{CSDH-T_i} = (Y_T)^{r_{CSDH_i}}$, $SK_{CSDH-T_i} = (Y_T)^{x_{CSDH-T_i}}$, and $V_{CSDH} \in L_H$. Finally, $\mathcal{C}$ returns V_{CSDH_i} to $\mathcal{A}$.

Eventually, $\mathcal{A}$ outputs a new valid message tuple $(V_{CSDH}, Time_{CSDH})$. It means that after having sent $(ID_T, V_T, Y_T, Time_T)$ Traveler accepts $(V_{CSDH}, Time_{CSDH})$ but it has not been produced by *CSDH*. Hence, it could be the following three situations:

1) $\mathcal{A}$ guesses that value V_{CSDH} with the probability less than $\frac{q_H}{2^k}$.
2) The values V_T and Y_T were obtained in other session. The probability is $\frac{q_H}{2^k} \times \frac{q_S}{p} \times (q_S - 2)$ which is less than $\frac{q_H \cdot q_S^2}{p \cdot 2^k}$.
3) $\mathcal{A}$ had asked for $(ID_T, K_{CSDH-T}, SK_{CSDH-T}, Time_{CSDH})$ to Hash query.

We use symbol $Forge_{CSDH}$ to denote the event that $\mathcal{A}$ forges *CSDH*'s transcript. Thus, we can obtain

$$\Pr[Forge_{CSDH}] \\ \leq \Pr[V_{CSDH} = h(ID_T||K_{CSDH-T}||SK_{CSDH-T}||Time_{CSDH}) \\ |K_{CSDH-T}, SK_{CSDH-T} \leftarrow \mathbb{G}_p] + \frac{q_H}{2^k} + \frac{q_H \cdot q_S^2}{p \cdot 2^k}.$$

Set $Y_T = g^a$. The challenger $\mathcal{C}$ can use $\mathcal{A}$ to compute $SK_{CSDH-T} = g^{ab}$ using Y_T and y_{CSDH}. In other words, $\mathcal{C}$ can solve the CDH problem with advantage $\epsilon_2' \geq \epsilon_2 - \frac{q_H}{2^k} - \frac{q_H \cdot q_S^2}{p \cdot 2^k}$.

Theorem 1: In the random oracle model, the proposed two-party authenticated key agreement protocol P in the login phase is a secure AKE providing forward secrecy under the hardness of the decisional Diffie-Hellman (DDH) and the CDH problems. Precisely,

$$Adv_P^{AKE-fs}(t, q_{ex}) \leq 2q_{ex} \cdot Adv_{\mathbb{G}_p}^{DDH}(t) + Adv_P^{Forge}(t),$$

where q_{ex} is the maximum times of making the Execute query and $Adv_P^{Forge}(t)$ denotes the advantage of any attacker forges the proposed AKE protocol P.

proof: Assume that $\mathcal{A}$ is an active adversary in attacking P with a non-negligible advantage. Note that $\mathcal{A}$ is called active, if it can make all queries mentioned in the adversarial model. Then, $\mathcal{A}$ obtain the advantage in the following two cases:

1) $\mathcal{A}$ forges Traveler's and $CSDH$'s transcripts.
2) $\mathcal{A}$ breaks P without forging any transcripts.

For the case 1, we use $\mathcal{A}$ to construct a forger $\mathcal{F}$ which returns two valid messages $(ID_T, V_T, Y_T, Time_T)$ and $(V_{CSDH}, Time_{CSDH})$ as follows: $\mathcal{F}$ generates all parameters and keys for the system and simulates the oracle queries of $\mathcal{A}$. This simulation is perfect indistinguishable from $\mathcal{A}$'s queries except making Corrupt queries for Traveler or $CSDH$. It it occurs, $\mathcal{F}$ terminates. Eventually, $\mathcal{F}$ returns two valid messages $(ID_T, V_T, Y_T, Time_T)$ and $(V_{CSDH}, Time_{CSDH})$, while $\mathcal{A}$ outputs the two messages. Let *Forge* be the event that $\mathcal{A}$ generates the two valid messages. Then,

$$\Pr[Forge] \leq Adv_{\mathcal{F},P}^{forge}(t) \leq Adv_P^{forge}(t).$$

By Lemmas 1 and 2, Pr[*Forge*] is negligible.

For the case 2, we compute the upper bound of the advantage that $\mathcal{A}$ breaks P without forging any transcripts. When $\mathcal{A}$ makes an $Execute(T, CSDH)$ query for Traveler T and $CSDH$ chosen by $\mathcal{A}$, the real execution is returned by the equations, shown at the bottom of this page.

where T denotes the transcript and SK_{CSDH-T} denotes the establishing session key. Since $\mathcal{A}$ can obtain T's password PW_T and hash values V_T, V_{CSDH} by making Corrupt and Hash queries. However, there values offer no information about x_{CSDH} and r_T under the discrete logarithm algorithm.

Then, we can define the distribution

We want to show that the problem to distinguish *Real* from *Fake* can be reduced to solve the decisional Diffie-Hellman (DDH) problem. Let $\epsilon(t) = Adv_{\mathbb{G}_p}^{DDH}(t)$.

Claim: For any algorithm $\mathcal{A}$ running in time t,

$$\Pr[(T, SK_{CSDH-T}) \leftarrow Real] : \mathcal{A}(T, SK_{CSDH-T}) = 1| \\ -|| \Pr[(T, SK_{CSDH-T}) \leftarrow Fake] : \mathcal{A}(T, SK_{CSDH-T}) = 1| \\ \leq \epsilon(t).$$

Proof: By the contradiction proof, suppose that $\mathcal{A}$ can distinguish *Real* from *Fake*. Then, we can construct an algorithm $\mathcal{D}$ which can solve the decisional Diffie-Hellman (DDH) problem, i.e. to distinguish (g^a, g^b, g^{ab}) from (g^a, g^b, R_2) for $a, b \in \mathbb{Z}_q^*$ and $R_2 \in \mathbb{G}_p$.

We set $y_{CSDH} = g^a$ and $Y_T = g^b$ as the input of $\mathcal{D}$. Then, D returns $T = (ID_T, V_T, Y_T, Time_T, V_{CSDH}, Time_{CSDH})$ and computes $R_1 = R_2$. Finally, $\mathcal{D}$ sends (T, R_1) to $\mathcal{A}$. Upon receiving (T, R_1), $\mathcal{A}$ can determine whether $SK = R_1$. If it is true, $g^{ab} = R_2$. In other words, $\mathcal{D}$ can run $\mathcal{A}$ as a subroutine to distinguish (g^a, g^b, g^{ab}) from (g^a, g^b, R_2), a contradiction.

Since$| \Pr[(T, SK_{CSDH-T_0}) \leftarrow Fake; SK_{CSDH-T_1} \leftarrow \mathbb{G}_p; b \leftarrow \{0, 1\} | \mathcal{A}(T, SK_{CSDH-T_b}) = 1] = \frac{1}{2}$, we can obtain the resulted advantage on the event $\neg$*Forge* which is bounded by $2 \cdot Adv_{\mathbb{G}_p}^{DDH}(t)$. Thus, it implies

$$Adv_P^{AKE-fs}(t, 1) \leq 2 \cdot Adv_{\mathbb{G}_p}^{DDH}(t) + Adv_P^{Forge}(t)$$

for the case $q_{ex} = 1$. Finally, for the case $q_{ex} > 1$ we have

$$Adv_P^{AKE-fs}(t, q_{ex}) \leq 2q_{ex} \cdot Adv_{\mathbb{G}_p}^{DDH}(t) + Adv_P^{Forge}(t).$$

B. SECURITY OF DELEGATION PHASE

In this subsection, we want to prove the security of our scheme in the delegation phase by the BAN logic [15], [19]–[22]. We will show that: Traveler T and Investigator I share a secret $N_{CSDH-T,I}$ which is chosen by $CSDH$ so that T can send the SOS message to I using this secret while T encountering danger. Firstly, we define some notations and rules about the BAN logic as follows:

1) NOTATIONS

1) $P \models X$: P believes X or called P would be entitled to believe X. In particular, P may act as though X is true.
2) $P \lhd X$: P sees X. Someone has sent a message containing X to P and P can read and repeat X.

$$Param = \left\{ \begin{array}{l} \mathbb{G}_p; g \leftarrow \mathbb{G}_p; ID_T, PW_T \leftarrow \{0,1\}^*; x_{CSDH}, r_{CSDH-T} \leftarrow \mathbb{Z}_p^*; \\ y_{CSDH} = g^{x_{CSDH}}, R_{CSDH-T} = g^{r_{CSDH-T}} + PW_T : \\ (\mathbb{G}_p, g, y_{CSDH}, ID_T) \end{array} \right\}$$

and

$$Real = \left\{ \begin{array}{l} r_T \leftarrow \mathbb{Z}_p^*; V_T, V_{CSDH} \leftarrow \{0,1\}^k; Y_T = g^{r_T}; \\ Y_{T-CSDH} = R_{CSDH-T} - PW_T, K_{T-CSDH} = (Y_{T-CSDH})^{r_T}; \\ SK_{CSDH-T} = (y_{CSDH})^{r_T}; \\ T = (ID_T, V_T, Y_T, Time_T, V_{CSDH}, Time_{CSDH}) : (T, SK_{CSDH-T}) \end{array} \right\}$$

$$Fake = \left\{ \begin{array}{l} r_T \leftarrow \mathbb{Z}_p^*; V_T, V_{CSDH} \leftarrow \{0,1\}^k; R_1 \leftarrow \mathbb{G}_p; Y_T = g^{r_T}; \\ Y_{T-CSDH} = R_{CSDH-T} - PW_T, K_{T-CSDH} = (Y_{T-CSDH})^{r_T}; \\ SK_{CSDH-T} = R_1; \\ T = (ID_T, V_T, Y_T, Time_T, V_{CSDH}, Time_{CSDH}) : (T, SK_{CSDH-T}) \end{array} \right\}$$

3) $P \mid\sim X$: P once said X. P sent a message including X at some time. Note that it does not know whether the message was sent long ago or during the current run of the protocol, but it knows that $P \models X$ when the message was sent.
4) $P \mid\Rightarrow X$: P has jurisdiction over X. P controls X which is subject to jurisdiction of P and P is trusted for X.
5) $\sharp(X)$: X is fresh. X has not been sent in a message at any time before the execution of current round of the protocol.
6) $P \overset{K}{\longleftrightarrow} Q$: P and Q may use the shared key K to communicate securely. We say that K is good, if K will never be discovered by any principal except P or Q, or a principal trusted by either P or Q.
7) $P \overset{X}{\rightleftharpoons} Q$: The formula X is a secret known only to P and Q, and possibly to principals trusted by P and Q.
8) $\langle X\rangle_Y$: The formula X is combined with a secret Y.

2) RULES

1) Message meaning rule for shared secrets: $\dfrac{P \models Q \overset{Y}{\rightleftharpoons} P,\ P \triangleleft \langle X\rangle_Y}{P \models Q \mid\sim X}$. It means that if P believes that Y is a secret known only to P and Q and P sees X under Y, then P believes that Q once said X.
2) Nonce verification rule: $\dfrac{P \models \sharp(X),\ P \models Q \mid\sim X}{P \models Q \models X}$. It means that if P believes that X is fresh and Q once said X, then P believes Q believes X.
3) Jurisdiction rule: $\dfrac{P \models Q \mid\Rightarrow X,\ P \models Q \models X}{P \models X}$. It means that if P believes that Q has jurisdiction over X and believes Q believes X, then P believes X.
4) Belief rule: $\dfrac{P \models Q \models (X, Y)}{P \models Q \models X}$. It means that if P believes Q believes (X, Y) then P believes Q believes X.

3) GOALS

We want to show that our scheme should achieve the following goals:

$G_1 : T \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

$G_2 : I \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

$G_3 : CSDH \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

$G_4 : I \models T \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

$G_5 : CSDH \models T \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

$G_6 : CSDH \models I \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

4) INITIAL ASSUMPTIONS

We define some initial assumptions of our scheme as follows:

$A_1 : CSDH \models \sharp(R_{CSDH-T})$.

$A_2 : CSDH \models \sharp(N_{CSDH-I})$.

$A_3 : CSDH \models \sharp(N_{CSDH-T,I})$.

$A_4 : T \models T \overset{R_{CSDH-T}}{\rightleftharpoons} CSDH$.

$A_5 : CSDH \models T \overset{R_{CSDH-T}}{\rightleftharpoons} CSDH$.

$A_6 : I \models I \overset{N_{CSDH-I}}{\rightleftharpoons} CSDH$.

$A_7 : CSDH \models I \overset{N_{CSDH-I}}{\rightleftharpoons} CSDH$.

$A_8 : T \models T \overset{SK_{T-CSDH}}{\longleftrightarrow} CSDH$.

$A_9 : CSDH \models T \overset{SK_{T-CSDH}}{\longleftrightarrow} CSDH$.

$A_{10} : I \models I \overset{K_{CSDH-I}}{\longleftrightarrow} CSDH$.

$A_{11} : CSDH \models I \overset{K_{CSDH-I}}{\longleftrightarrow} CSDH$.

A_1, A_2, and A_3 mean that $CSDH$ generates fresh random values R_{CSDH-T}, N_{CSDH-I}, and $N_{CSDH-T,I}$, respectively. Hence, we assume that they are freshness. A_4 and A_5 are valid because R_{CSDH-T} is computed by $CSDH$ and shares with T. Similarly, A_6 and A_7 are valid because N_{CSDH-I} is computed by $CSDH$ and shares with I. After T logs $CSDH$ successfully, they can establish a common key SK_{T-CSDH} to communicate. Thus, A_8 and A_9 are valid. In the registration phase, I and $CSDH$ can compute a common key K_{CSDH-I} to communicate. Thus, A_{10} and A_{11} are valid.

5) IDEALIZE THE COMMUNICATION MESSAGES

Here, we idealize the communication messages of our scheme listed as below:

$M_1 : T \rightarrow CSDH : \{ID_T, ID_I, Q_T, Time_T\}$.

$M_2 : I \rightarrow CSDH : \{ID_I, ID_T, Q_I, Time_I\}$.

$M_3 : CSDH \rightarrow T : \{ID_{CSDH}, ID_I, Q_{CSDH-T}, Time_{CSDH}, M_{CSDH-T}\}$.

$M_4 : CSDH \rightarrow I : \{ID_{CSDH}, ID_T, Q_{CSDH-I}, Time_{CSDH}, M_{CSDH-I}\}$.

$M_5 : T \rightarrow I : \{ID_T, m_{req}, Q_{T-CSDH}, Q_{T-I}, Time_T\}$.

$M_6 : I \rightarrow CSDH : \{ID_T, ID_I, Q_{T-CSDH}, Q_{I-CSDH}, Time_I\}$.

6) DETAILED DESCRIPTION

Based on the rules of the BAN logic, we prove that our scheme can achieve the defined goals using the initial assumptions.

a: FOR THE GOAL 1

By message M_3, we can obtain $T \models CSDH \models M_{CSDH-T}$. Since $M_{CSDH-T} = N_{CSDH-T,I} \oplus SK_{CSDH-T} \oplus R_{CSDH-T}$ is computed by $CSDH$ and by A_5, A_9, it implies T can obtain $N_{CSDH-T,I}$ and $T \models CSDH \mid\Rightarrow N_{CSDH-T,I}$. By the similar approach, using message M_4 and assumptions A_7, A_{11}, it implies I can obtain $N_{CSDH-T,I}$ and $I \models CSDH \mid \Rightarrow N_{CSDH-T,I}$. Thus, we have $S_1 : T \models CSDH \mid\Rightarrow (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$. In other aspect, $T \models CSDH \models N_{CSDH-T,I}$ is also true. In other words, we have $S_2 : T \models CSDH \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$. Finally, according to S_1 and S_2 we can obtain $T \models (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the jurisdiction rule.

TABLE 1. Performance comparisons among the proposed scheme and Chen et al.'s scheme.

Phase→ Scheme↓	Login	Delegation	Signing and verification	Total
Chen et al.'s scheme				
Traveler T	$4T_H+T_{xor}$	$T_{exp}+T_{mul}+$ $T_H+T_{add}+$ $T_{sym}+T_{RFC2631}$	N/A	$T_{exp}+5T_H+$ $T_{sym}+T_{mul}+$ $T_{xor}+T_{add}+T_{RFC2631}$
Investigator I	N/A	$T_{mul}+3T_H+$ $2T_{exp}+T_{sym}+$ $T_{RFC2631}$	$T_{add}+T_{sym}+$ $T_{sign}+T_{RFC2631}$	$2T_{exp}+3T_H+$ $2T_{sym}+T_{sign}+$ $T_{mul}+T_{add}+2T_{RFC2631}$
Doctor D	N/A	N/A	$T_{sym}+T_{sign}+$ $T_{RFC2631}$	$T_{sym}+T_{sign}+T_{RFC2631}$
Cloud server $CSDH$	N/A	N/A	$T_{mul}+2T_H+$ $4T_{add}+3T_{exp}+$ $2T_{sym}+2T_{verify}+$ $2T_{RFC2631}$	$3T_{exp}+2T_H+2T_{sym}+$ $2T_{verify}+T_{mul}+4T_{add}+$ $2T_{RFC2631}$
Proposed scheme				
Traveler T	$3T_{exp}+2T_{add}+$ $2T_H$	$2T_{xor}+3T_{add}+$ $5T_H$	N/A	$3T_{exp}+7T_H+$ $5T_{add}+2T_{xor}$
Investigator I	N/A	$2T_{xor}+3T_{add}+$ $5T_H$	T_H	$2T_{xor}+3T_{add}+6T_H$
Doctor D	N/A	N/A	$2T_{sym}+T_{add}+$ $2T_H$	$2T_{sym}+T_{add}+2T_H$
Cloud server $CSDH$	$2T_{exp}+T_{add}+$ $2T_H$	$4T_{xor}+2T_{add}+$ $4T_H$	$2T_{sym}+2T_{add}+$ $4T_H$	$4T_{xor}+2T_{exp}+10T_H+$ $2T_{sym}+5T_{add}$

b: FOR THE GOAL 2

By the similar approach in the goal 1, we can obtain S_3 : $I \mid\equiv CSDH \mid\Rightarrow (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ and S_4 : $I \mid\equiv CSDH \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$. Hence, according to S_3 and S_4 it implies $I \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the jurisdiction rule.

c: FOR THE GOAL 3

According to the proofs in the goals 1 and 2, it is easy to see that $CSDH \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$.

d: FOR THE GOAL 4

By M_5, we can obtain S_5 : $I \triangleleft \langle Q_{T-I} \rangle_{N_{CSDH-T,I}}$ because $Q_{T-I} = h(ID_T || N_{CSDH-T,I} || Time_T || m_{req})$. From the goal 2, we have S_6 : $I \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$. According S_6 and S_5, it implies S_7 : $I \mid\equiv T \mid\sim Q_{T-I}$ by the message meaning rule. Then, we can apply the nonce verification rule to obtain S_8 : $I \mid\equiv T \mid\equiv Q_{T-I}$ because Q_{T-I} is a hash value, $I \mid\equiv \sharp(Q_{T-I})$. Since Q_{T-I} contains $N_{CSDH-T,I}$ and $T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I$, we can obtain $I \mid\equiv T \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ from S_8 by the belief rule.

e: FOR THE GOAL 5

By M_3, it is easy to see that S_9 : $CSDH \triangleleft \langle Q_{CSDH-T} \rangle_{SK_{T-CSDH}}$ because $Q_{CSDH-T} = h(ID_I || SK_{T-CSDH} || N_{CSDH-T,I} || Time_{CSDH})$. According to A_9 and S_9, we can obtain S_{10} : $CSDH \mid\equiv T \mid\sim Q_{CSDH-T}$ by the message meaning rule. Since Q_{CSDH-T} is a hash value and computed by the $CSDH$, it implies that S_{11} : $CSDH \mid\equiv \sharp(Q_{CSDH-T})$. According to S_{11} and S_{10}, we can obtain $CSDH \mid\equiv T \mid\equiv Q_{CSDH-T}$ by the nonce verification rule. Because Q_{CSDH-T} contains $N_{CSDH-T,I}$ and $T \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the goal 1, it implies that $CSDH \mid\equiv T \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the belief rule.

f: FOR THE GOAL 6

By M_4, it is easy to see that S_{12} : $CSDH \triangleleft \langle Q_{CSDH-I} \rangle_{K_{CSDH-I}}$ because $Q_{CSDH-I} = h(ID_T || K_{CSDH-I} || N_{CSDH-T,I} || Time_{CSDH})$. According to A_{11} and S_{12}, we can obtain S_{13} : $CSDH \mid\equiv I \mid\sim Q_{CSDH-I}$ by the message meaning rule. Since Q_{CSDH-I} is a hash value and computed by the $CSDH$, it implies that S_{14} : $CSDH \mid\equiv \sharp(Q_{CSDH-I})$. According to S_{14} and S_{13}, we can obtain $CSDH \mid\equiv I \mid\equiv Q_{CSDH-I}$ by the nonce verification rule. Because Q_{CSDH-I} contains $N_{CSDH-T,I}$ and $I \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the goal 2, it implies that $CSDH \mid\equiv I \mid\equiv (T \overset{N_{CSDH-T,I}}{\rightleftharpoons} I)$ by the belief rule.

C. SECURITY OF FINAL PHASE

1) *CSDH* AUTHENTICATES INVESTIGATOR *I*

Theorem 2: The message $(ID_T, ID_I, Q_{T-CSDH}, Q_{I-CSDH}, Time_I)$ sent by the authorized Investigator I with delegated Traveler T can not be forged under the security of SK_{T-CSDH} and K_{I-CSDH}.

Proof: The value $Q_{T-CSDH} = h(ID_T||SK_{T-CSDH}||m_{req})$, where SK_{T-CSDH} is a shared key establishing by T and $CSDH$ in the login phase. The security of SK_{T-CSDH} can be referred to Theorem 1. The value $Q_{I-CSDH} = h(ID_I||K_{I-CSDH}||Time_I||m_{req}||ID_D)$, where K_{I-CSDH} is a shared key establishing by I and $CSDH$ in the registration phase. Note that the procedures in the registration phase are over a secure channel. In other words, no one can forge Q_{T-CSDH} and Q_{I-CSDH} except T and I, respectively.

2) DOCTOR *D* AUTHENTICATES *CSDH*

Theorem 3: The message $(ID_{CSDH}, ID_T, Q_{CSDH-D}, C_1)$ sent by $CSDH$ can not be forged under the security of K_{CSDH-D} and the security of adopted encryption algorithm E.

Proof: The value $Q_{CSDH-D} = h(ID_{CSDH}||ID_T||K_{CSDH-D}||Time_{CSDH})$, where K_{CSDH-D} is a shared key establishing by Doctor D and $CSDH$ in the registration phase. Note that the procedures in the registration phase are over a secure channel. The ciphertext $C_1 = E_{K_{CSDH-D}}(EMR||time_1)$. Only D with K_{CSDH-D} can decrypt C_1 to obtain EMR unless the adopted encryption algorithm E is insecure.

VII. PERFORMANCE ANALYSIS OF THE PROPOSED SCHEME

In this section, we show the performance analysis of our proposed scheme with Chen et al.'s scheme. Let T_H, T_{xor}, T_{exp}, T_{add}, T_{mul}, T_{sym}, T_{sign}, T_{verify} and $T_{RFC2631}$ denote the time complexity of hash function, XOR operation, exponential operation, addition/subtraction operation, multiplication operation, symmetric encryption/decryption, signature signing operation, signature verification operation and constructing a session key by RFC 2631 protocol, respectively.

Table 1 lists the computation cost of all phases of our proposed scheme and Chen et al.'s scheme. Due to the hardware restrictions of Traveler T's mobile device, in the login and delegation phases, we do not take T_{sign}, T_{verify} and $T_{RFC2631}$ into account and it is well-known that the time complexity of T_H, T_{add} and T_{xor} are negligible as compared to other operations. In addition, the total computational cost of the Investigator I, the Doctor D and Cloud server $CSDH$ in our scheme are $2T_{xor}+3T_{add}+6T_H$, $2T_{sym}+T_{add}+2T_H$ and $4T_{xor}+2T_{exp}+10T_H+2T_{sym}+5T_{add}$ separately and it is obvious that the performance of our proposed scheme is superior than Chen et al.'s scheme. Finally, the proposed scheme is more appropriate for cloud-assisted emergency system due to it ensures desirable security and is comparable in terms of computational cost with the previous work.

VIII. CONCLUSIONS

In this paper, we briefly reviewed Chen et al.'s cloud-based emergency system and shown that the process of data upload in the signing and verification phase is insecure. Although the identities of system participants are strictly verified, the attacker can still spy on the traveler's electronic medical record transmitted via public channels. In addition, Chen et al.'s scheme is also vulnerable to off-line password guessing attack in the case that the mobile device of traveler is lost or stolen. To resist these shortcomings, we put forward an improved scheme preserving traveler privacy by employing the concept of authenticated key exchange and message authentication. We have proved that our improved scheme achieves the goals of mutual authentication and key agreement in the random oracle model and the BAN logic. The analysis shows that our proposed scheme improves the security flaws of Chen et al.'s scheme while maintains the computation efficiency in cloud-based emergency system for mountaineering events.

REFERENCES

[1] *Sport England: Primary Offer Data Information Pack for Mountaineering*, accessed on Jul. 03, 2016. [Online]. Available: http://www.sportengland.org/

[2] C.-L. Chen, Y.-Y. Chen, C.-C. Lee, and C.-H. Wu, "Design and analysis of a secure and effective emergency system for mountaineering events," *J. Supercomput.*, vol. 70, no. 1, pp. 54–74, 2014.

[3] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, 1991.

[4] B. Lee, H. Kim, and K. Kim, "Strong proxy signature and its applications," in *Proc SCIS*, vol. 1. 2001, pp. 603–608.

[5] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[6] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Proc. Workshop Theory Appl. Cryptograph. Techn.*, 1984, pp. 10–18.

[7] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *Proc. Annu. Int. Cryptol. Conf.*, 1993, pp. 232–249.

[8] R. Canetti, O. Goldreich, and S. Halevi, "The random oracle methodology, revisited," *J. ACM*, vol. 51, no. 4, pp. 557–594,

[9] C.-M. Chen, W. Fang, K.-H. Wang, and T.-Y. Wu, "Comments on 'An improved secure and efficient password and chaos-based two-party key agreement protocol,'" *Nonlinear Dyn.*, vol. 87, no. 3, pp. 2073–2075, 2017.

[10] S. A. Chaudhry, M. S. Farash, H. Naqvi, S. Kumari, and M. K. Khan, "An enhanced privacy preserving remote user authentication scheme with provable security," *Secur. Commun. Netw.*, vol. 8, no. 18, pp. 3782–3795, 2015.

[11] C.-M. Chen, L. Xu, T.-Y. Wu, and C.-R. Li, "On the security of a chaotic maps-based three-party authenticated key agreement protocol," *J. Netw. Intell.*, vol. 1, no. 2, pp. 61–65, 2016.

[12] D. He, N. Kumar, and N. Chilamkurti, "A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks," *Inf. Sci.*, vol. 321, pp. 263–277, Nov. 2015.

[13] C.-C. Lee, D.-C. Lou, C.-T. Li, and C.-W. Hsu, "An extended chaotic-maps-based protocol with key agreement for multiserver environments," *Nonlinear Dyn.*, vol. 76, no. 1, pp. 853–866, 2014.

[14] S. Kumari, S. A. Chaudhry, F. Wu, X. Li, M. S. Farash, and M. K. Khan, "An improved smart card based authentication scheme for session initiation protocol," *Peer-Peer Netw. Appl.*, vol. 10, no. 1, pp. 92–105, 2017.

[15] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. Roy. Soc. London A, Math., Phys. Eng. Sci.*, vol. 426, no. 1871, pp. 233–271, 1989.

[16] K. Y. Choi, J. Y. Hwang, D. H. Lee, and I. S. Seo, "ID-based authenticated key agreement for low-power mobile devices," in *Proc. Austral. Conf. Inf. Secur. Privacy*, 2005, pp. 494–505.

[17] T.-Y. Wu, Y.-M. Tseng, and T.-T. Tsai, "A revocable id-based authenticated group key exchange protocol with resistant to malicious participants," *Comput. Netw.*, vol. 56, no. 12, pp. 2994–3006, 2012.

[18] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proc. 1st ACM Conf. Comput. Commun. Secur.*, 1993, pp. 62–73.

[19] C.-C. Lee, C.-T. Li, S.-T. Chiu, and Y.-M. Lai, "A new three-party-authenticated key agreement scheme based on chaotic maps without password table," *Nonlinear Dyn.*, vol. 79, no. 4, pp. 2485–2495, 2015.

[20] L. Zhang, S. Tang, and S. Zhu, "A lightweight privacy preserving authenticated key agreement protocol for SIP-based VoIP," *Peer-Peer Netw. Appl.*, vol. 9, no. 1, pp. 108–126, 2016.

[21] Q. Jiang, M. K. Khan, X. Lu, J. Ma, and D. He, "A privacy preserving three-factor authentication protocol for e-health clouds," *J. Supercomput.*, vol. 72, no. 10, pp. 3826–3849, 2016.

[22] M. Nikooghadam, R. Jahantigh, and H. Arshad, "A lightweight authentication and key agreement protocol preserving user anonymity," *Multimedia Tools Appl.*, pp. 1–23, Jul. 2016.

CHIEN-MING CHEN is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China. He had published over 70 international journal and international conference papers in his research fields. His current research interests include network security, mobile Internet, wireless sensor network, and cryptography. He serves as an Associate Editor of three international journals: *Journal of Information Hiding and Multimedia Signal Processing*, *Data Science and Recognition*, and *Journal of Network Intelligence*.

CHUN-TA LI (M'10) is currently an Associate Professor with the Department of Information Management, Tainan University of Technology, Tainan, Taiwan. He has authored or co-authored over 100 international journal and international conference papers in his research fields. His research interests include information and network security, cloud computing/RFID/IoTs security, and security protocols for telemedicine systems. He is an Editorial Board Member of the *International Journal of Network Security*.

SHUAI LIU is currently pursuing the degree with the School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, China. His current research interests include information and network security.

TSU-YANG WU received the Ph.D. degree in mathematics from the National Changhua University of Education, Taiwan, in 2010. He was an Assistant Professor with the Harbin Institute of Technology, Shenzhen, China. He is an Associate Professor with the College of Information Science and Engineering, Fujian University of Technology, China. His research interests include applied cryptography, pairing-based cryptography, and network security. He is a member of the China Computer Federation and the Chinese Cryptology and Information Security Association. He serves as an Editor of two international journals: *Data Science and Recognition* and *Journal of Network Intelligence*.

JENG-SHYANG PAN received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology in 1986, the M.S. degree in communication engineering from National Chiao Tung University, Taiwan, in 1988, and the Ph.D. degree in electrical engineering from the University of Edinburgh, U.K., in 1996. He is currently the Director of the Fujian Provincial Key Lab of Big Data Mining and Applications, and an Assistant President with the Fujian University of Technology. He is also the Professor with the Harbin Institute of Technology. He is the IET Fellow, U.K., and has been the Vice Chair of the IEEE Tainan Section. He was offered Thousand Talent Program in China in 2010.