ISSN 1392 – 124X, ISSN 2335 – 884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2013, Vol.42, No.2

Cryptanalysis and Improvement of a Password-Based Remote User Authentication Scheme without Smart Cards

Debiao He[1], Ding Wang[2], Shuhua Wu[3]

[1] *School of Mathematics and Statistics, Wuhan University, Wuhan, China*
[2] *School of Electronics Engineering and Computer Science, Peking University, Beijing, China*
[3] *Department of Networks Engineering, Information Engineering University, Zhengzhou, China*
e-mail: hedebiao@163.comAdresas

crossref http://dx.doi.org/10.5755/j01.itc.42.2.2554

Abstract. Recently, Chen et al. [B. Chen, W. Kuo, L. Wuu, A secure password-based remote user authentication scheme without smart cards, Information Technology and Control 41(1) (2012) 53-59] proposed a secure password-based remote user authentication scheme without smart cards and claimed that their scheme could withstand various attacks. Although Chen et al.'s scheme has many benefits, we find that it is vulnerable to the device stolen attack and the privileged insider attack. We also find that their scheme does not support perfect forward secrecy and no key control. Therefore, we propose an improved scheme to overcome weaknesses and maintain the benefits of the original scheme..

Keywords: password-based; smart card; mutual authentication.

1. Introduction

Authentication scheme is an essential security mechanism, through which the user and the server could authenticate each other and generate a session key for future communications. In 1981, Lamport [1] proposed the first authentication scheme by using a one-way hash function. However, the server in his scheme maintains a password table and the system will be crash if the table is stolen by an adversary. To improve security, many password-based authentication schemes with smart cards have been proposed [2-12].

With the development of electronic technology, various common storage devices (e.g., universal serial bus (USB) thumb drives, portable HDDs, mobile phones, Laptop and Desktop PCs) are produced to make human life more convenient. These password-based authentication schemes with smart cards [2-12] cannot be applied in such environment. To satisfy requirement of applications, Rhee et al. [13] proposed a remote user authentication scheme without smart cards. They claimed that their scheme provides mutual authentication with no verification table at the cost of only two messages for login and authentication protocols. They also claimed that their scheme is resistant against impersonation and off-line dictionary attacks. However, Rhee et al.'s scheme is vulnerable to the impersonation attack [14, 15] and the man-in-the-middle attack [14]. To overcome weaknesses in Rhee et al.'s scheme, Chen et al. [16] proposed a password-based remote user authentication and key agreement scheme using common storage devices such as USB drives. They claimed that their scheme not only withstands off-line dictionary and well-known on-line attacks, but also provides mutual authentication. Unfortunately, we found that Chen et al.' scheme is vulnerable to the device stolen attack and the privileged insider attack. We also found that their scheme does not support perfect forward secrecy and no key control. To overcome these weaknesses, we propose an improved password-based remote user authentication scheme without smart cards.

The organization of the paper is sketched as follows. In Sections 2 and 3, we review and analyze Chen et al.'s scheme. In Section 4, we present a new password-based authentication scheme without smart cards. In Section 5, we analyze the security of our proposed scheme. In Section 6, the performance considerations are given. Finally, a conclusion is given in Section 7.

2. Review of Chen et al.'s scheme

Chen et al.'s scheme consists of four phases: registration, login, authentication, and password change. As shown in Fig. 1, the details of these phases are described as follows.

2.1. Registration phase

The server S generates two large prime numbers p and q such that $p = 2q + 1$. S generates a random number $x \in Z_q^*$ as its secret key and selects a one-way hash function H. The following steps will be executed if the user U_i wants to be a legal user.

1) U_i chooses his identity ID_i and password PW_i and sends them to S through a secure channel.

2) Upon receiving ID_i and PW_i, S computes $Y_i = H(ID_i)^{x+PW_i} \bmod p$ and sends the authentication information $\{p, q, H, Y_i\}$ to U_i through a secure channel.

3) Upon receiving $\{p, q, H, Y_i\}$, U_i stores it locally on his memory device, i.e., his USB drive.

2.2. Login phase

When U_i wants to login in S, he will carry out the following steps.

1) U_i generates a random number $\alpha \in Z_q^*$ and computes $Y_i' = Y_i / H(ID_i)^{PW_i} \bmod p$, $C_i = H(ID_i)^{\alpha} \bmod p$, $D_i = Y_i' C_i \bmod p$ and $V_i = H(ID_i, Y_i', C_i, D_i, T_i)$, where T_i is the current time of U_i.

2) U_i sends the login request message $M_1 = \{ID_i, C_i, V_i, T_i\}$ to S.

3) If U_i does not receive S's reply before timeout, U_i must go back to the registration phase and re-obtain his authentication information.

2.3. Authentication phase

In this phase, the sever and the user will authenticate each other and generate a session key for future communications through the following steps.

1) Upon receiving the message $M_1 = \{ID_i, C_i, V_i, T_i\}$, S checks the validity of ID_i and the freshness of T_i. If ID_i is not valid or T_i is not fresh, S stops the session. Otherwise, S computes $Y_i'' = H(ID_i)^x \bmod p$, $D_i' = Y_i'' C_i \bmod p$ and checks whether V_i and $H(ID_i, Y_i'', C_i, D_i', T_1)$ are equal. If they are not equal, S stops the session; otherwise, S computes $V_s = H(ID_i, D_i', T_s)$ and sends the response message $M_2 = \{V_s, T_s\}$ to U_i, where T_s is the current time of S.

2) Upon receiving the message $M_2 = \{V_s, T_s\}$, U_i checks the freshness of T_s. If it is not fresh, U_i stops the session; otherwise, U_i checks whether V_s and $H(ID_i, D_i, T_s)$ are equal. If they are not equal, U_i stops the session; otherwise, S is authenticated.

3) After the mutual authentication finished, U_i and S compute the session key $sk = H(D_i) = H(D_i')$ and use the key to launch a secure communication channel.

2.4. Password Change phase

In the case when the user U_i wants to change his identity ID_i and password PW_i, he can choose his new identity ID_i' and password PW_i', go back to the registration phase, and re-obtain his new authentication information from the server S.

3. Security analysis of Chen et al.'s scheme

In this section, we will analyze the security of Chen et al.'s scheme and point out that their scheme suffers from the device stolen attack and the privileged insider attack. We also point out that their scheme cannot provide perfect forward secrecy and no key control.

3.1. Device stolen attack

The adversary A could read all the authentication information $\{p, q, H, Y_i\}$ from the user's USB device since the device cannot provide tamper-resistant property where $Y_i = H(ID_i)^{x+PW_i} \bmod p$. Besides, we assume A has total control over the communication channel between the user and the server, which means that A can insert, delete, or alter any messages in the channel. Generally speaking, the user often chooses his name as his identity or writes his identity on the device; and moreover the input identity is usually displayed in plain on the screen and thus can be possibly seen when the attacker steals the device [17]. Therefore, A could get the user's identity probably. A could get the password through the following steps.

1) A intercepts all messages $\{ID_i, C_i, V_i, T_i\}$ sent by users. For convenience, A stores these messages in to a database according to the identity ID_i, where $Y_i' = Y_i / H(ID_i)^{PW_i} \bmod p$, $C_i = H(ID_i)^{\alpha} \bmod p$, $D_i = Y_i' C_i \bmod p$ and $V_i = H(ID_i, Y_i', C_i, D_i, T_i)$.

2) A can guess a password PW_i^* and derive the corresponding message $\{ID_i, C_i, V_i, T_i\}$ from his database.

3) A computes $Y_i^* = Y_i / H(ID_i)^{PW_i^*} \bmod p$, $D_i^* = Y_i^* C_i \bmod p$ and checks whether V_i and $H(ID_i^*, Y_i^*, C_i, D_i^*, T_i)$ are equal. If they are equal, the adversary has guessed the correct password PW_i; otherwise, A repeats steps 1), 2) and 3) until the correct password is found.

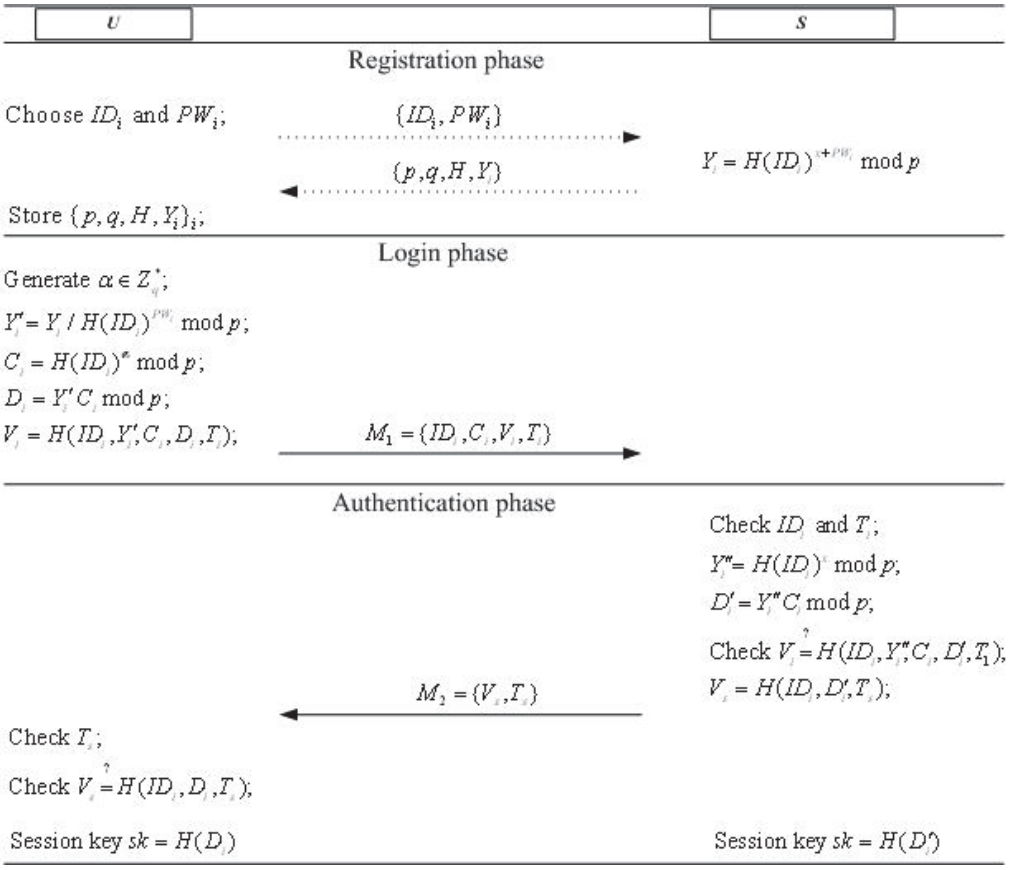

Figure 1. Chen et al.'s scheme

Since A knows the authentication information $Y_i = H(ID_i)^x \bmod p$, he could impersonate the user to login in the server once he gets obtain the correct password PW_i and identity ID_i through the above attack.

3.2. Privileged insider attack

In a real environment, it is a common practice that many users use the same passwords to access different applications or servers for their convenience of remembering long passwords and ease-of-use whenever required [18, 19]. However, if the system manager or a privileged insider A of the server S knows the passwords of user U_i, he may try to impersonate U_i by accessing other servers where U_i could be a registered user. In the user registration phase of Chen et al.'s scheme, A sends his identity ID_A and the password PW_i to S directly. Then, the privileged-insider of S could get the password. Therefore, Chen et al.'s scheme is vulnerable to the privileged insider attack.

3.3. Lack of perfect forward secrecy

We call a key agreement scheme as satisfying the perfect forward secrecy if the adversary still cannot compute the previous session keys when he gets the user's private keys or the server's secret key. In this subsection, we will show Chen et al.'s scheme cannot satisfy the perfect forward secrecy.

- The first attack

  Suppose that the adversary A gets a user U_i's password and the authentication information $\{p,q,H,Y_i\}$, where $Y_i = H(ID_i)^{x+PW_i} \bmod p$. Then, A could get all the previous session keys generated by U_i.

  1) A intercepts a message $\{ID_i, C_i, V_i, T_i\}$ sent by U_i, where $Y_i' = Y_i / H(ID_i)^{PW_i} \bmod p$, $C_i = H(ID_i)^{\alpha} \bmod p$, $D_i = Y_i' C_i \bmod p$ and $V_i = H(ID_i, Y_i', C_i, D_i, T_i)$.

  2) A computes $Y_i' = Y_i / H(ID_i)^{PW_i} \bmod p$ and $D_i = Y_i' C_i \bmod p$.

  3) A computes the session key $sk = H(D_i)$.

- The second attack

  Suppose that the adversary A gets the server's secret key x. Then, A could get all the previous session keys generated by U_i.

  1) A intercepts a message $\{ID_i, C_i, V_i, T_i\}$ sent by U_i, where $Y_i' = Y_i / H(ID_i)^{PW_i} \bmod p$, $C_i = H(ID_i)^{\alpha} \bmod p$, $D_i = Y_i' C_i \bmod p$ and $V_i = H(ID_i, Y_i', C_i, D_i, T_i)$.

2) A computes $Y_i' = H(ID_i)^x \bmod p$ and $D_i = Y_i' C_i \bmod p$.

3) A computes the session key $sk = H(D_i)$.

From the above description, we know that Chen et al.'s scheme cannot satisfy the perfect forward secrecy.

3.4. Lack of no key control

We call a key agreement scheme that satisfies the no key control if neither entity should be able to force the session key to be a pre-selected value.

In Chen et al.'s scheme, the session key is $sk = H(D_i) = H(H(ID_i)^{x+\alpha})$, that is, U_i can compute $H(ID_i)^x \bmod p = Y_i / H(ID_i)^{PW_i} \bmod p$ as α is chosen by U_i. Then, the session key is completely controlled by U_i and does not contain any contribution from S. Therefore, Chen et al.'s scheme cannot satisfy the no key control.

4. Our improved scheme

Like Chen et al.'s scheme does, our scheme also consists of four phases: registration, login, authentication, and password change. The last phase of our scheme is the same as that of Chen et al.'s scheme. To save space, we just give the details of the first three phases. As shown in Figure 2, the details of these phases are described as follows.

4.1. Registration phase

The server S generates two large prime numbers p and q such that $p = 2q + 1$. Then S generates a random number $x \in Z_q^*$ as its secret key and selects a one-way hash function H. The following steps will be executed if the user U_i wants to be a legal user.

1) U_i chooses his identity ID_i and password PW_i, generates a random number $r_i \in Z_q^*$ and sends ID_i and $H(PW_i, r_i)$ to S through a secure channel.

2) Upon receiving ID_i and $H(PW_i, r_i)$, S computes $Y_i = H(ID_i)^{H(ID_i, x)} + H(PW_i, r_i) \bmod p$ and sends the authentication information $\{p, q, H, Y_i\}$ to U_i through a secure channel.

3) Upon receiving $\{p, q, H, Y_i\}$, U_i stores it and r_i locally on his memory device, i.e., his USB drive.

4.2. Login phase

When U_i wants to login in S, he will carry out the following steps.

1) U_i generates a random number $\alpha \in Z_q^*$ and computes $Y_i' = Y_i - H(PW_i, r_i) \bmod p$, $C_i = H(ID_i)^\alpha \bmod p$, $D_i = (Y_i')^\alpha \bmod p$ and $V_i = H(ID_i, C_i, D_i, T_i)$, where T_i is the current time of U_i.

2) U_i sends the login request message $M_1 = \{ID_i, C_i, V_i, T_i\}$ to S.

3) If U_i does not receive S's reply before timeout, U_i must go back to the registration phase and re-obtain his authentication information.

4.3. Authentication phase

In this phase, the sever and the user will authenticate each other and generate a session key for future communications through the following steps.

1) Upon receiving the message $M_1 = \{ID_i, C_i, V_i, T_i\}$, S checks the validity of ID_i and the freshness of T_i. If ID_i is not valid or T_i is not fresh, S stops the session. Otherwise, S computes $D_i' = (C_i)^{H(ID_i, x)} \bmod p$ and checks whether V_i and $H(ID_i, C_i, D_i', T_i)$ are equal. If they are not equal, S stops the session; otherwise, S generates a random number $\beta \in Z_q^*$, computes $E_i = H(ID_i)^\beta \bmod p$, $V_s = H(ID_i, C_i, D_i', T_i, E_i, T_s)$ and sends the response message $M_2 = \{E_i, V_s, T_s\}$ to U_i, where T_s is the current time of S.

2) Upon receiving the message $M_2 = \{E_i, V_s, T_s\}$, U_i checks the freshness of T_s. If it is not fresh, U_i stops the session; otherwise, U_i checks whether V_s and $H(ID_i, C_i, D_i, T_i, E_i, T_s)$ are equal. If they are not equal, U_i stops the session; otherwise, S is authenticated.

3) After the mutual authentication finished, U_i and S compute the session key $sk = H((C_i)^\beta \bmod p) = H((E_i)^\alpha \bmod p)$ and use the key to launch a secure communication channel.

5. Security analysis

5.1. Authentication proof based on BAN-logic

The BAN logic [20] is a well known formal model. It has been widely used to analyze the security of authentication and key distribution protocols. We will demonstrate the validity of our scheme through the BAN logic. For convenience, the notations used in BAN logic analysis are described as follows.

- $P \mid\equiv X$: The principal P believes a statement X, or P is entitled to believe X.
- $\#(X)$: The formula X is fresh.
- $P \Rightarrow X$: The principal P has jurisdiction over the statement X.
- $P \lhd X$: The principal P sees the statement X.

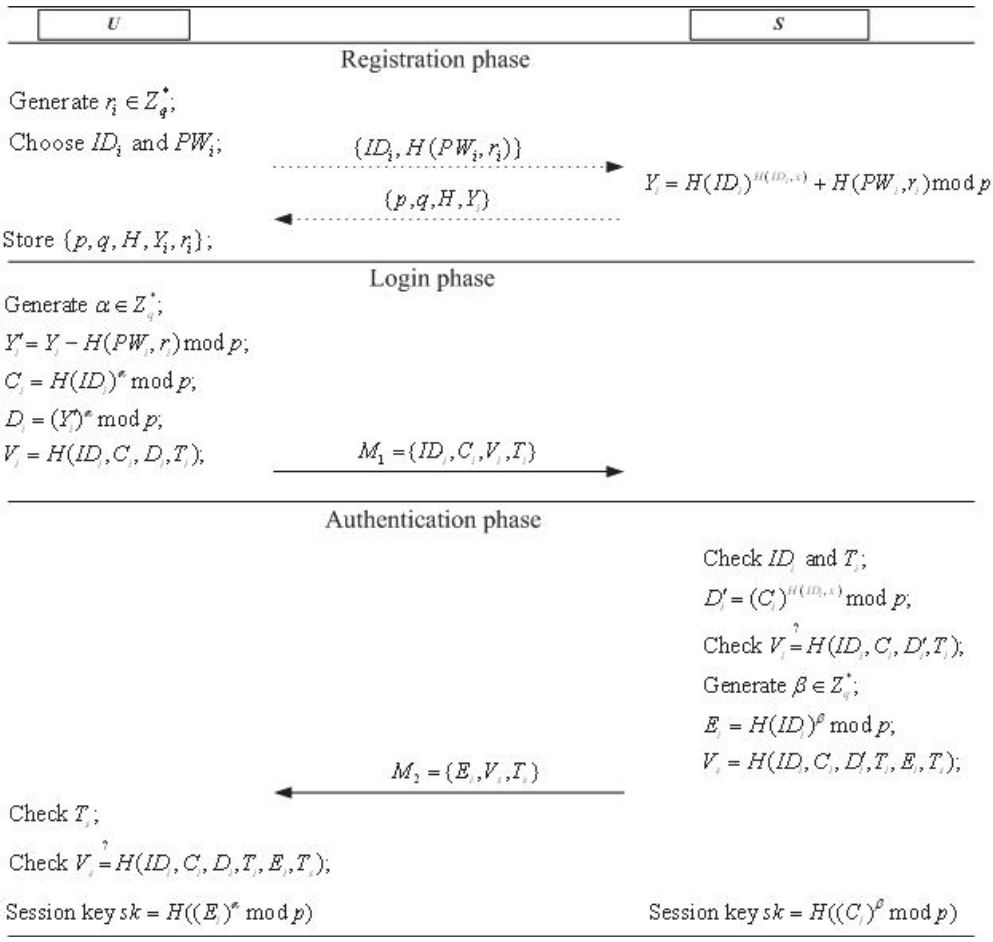

Figure 2. Our scheme

- $P \mid\sim X$: The principal P once said the statement X.
- (X, Y): The formula X or Y is one part of the formula (X, Y).
- $< X >_Y$: The formula X is combined with the formula Y.
- $\{X\}_K$: The formula X is encrypted under the key K.
- $(X)_K$: The formula X is hash with the key K.
- $P \stackrel{K}{\longleftrightarrow} Q$: The principals P and Q use the shared key K to communicate. The key K will never be discovered by any principal except P and Q.
- sk: The session key used in the current session.

We also define some main logical postulates of BAN logic as follows, since they will be used in our proof.

- The message-meaning rule: $\dfrac{P \mid\equiv P \stackrel{K}{\leftrightarrow} Q, P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X}$.
- The freshness-conjuncatenation rule: $\dfrac{P \mid\equiv \#(X)}{P \mid\equiv \#(X, Y)}$.
- The nonce-verification rule: $\dfrac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$.
- The jurisdiction rule: $\dfrac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$.

According to the analytic procedures of BAN logic, the proposed scheme will satisfy the following goals:

Goal 1. $U_i \mid\equiv (U_i \stackrel{sk}{\longleftrightarrow} S)$;

Goal 2. $U_i \mid\equiv S \mid\equiv (U_i \stackrel{sk}{\longleftrightarrow} S)$;

Goal 3. $S \mid\equiv (U_i \stackrel{sk}{\longleftrightarrow} S)$;

Goal 4. $S \mid\equiv U_i \mid\equiv (U_i \stackrel{sk}{\longleftrightarrow} S)$;

First, we transform our proposed scheme to the idealized form as follows:

$U_i \rightarrow S: ID_i, C_i,$

$(ID_i, C_i, U_i \stackrel{sk}{\longleftrightarrow} S, T_i)_{H(ID_i)^{\alpha\, H(ID_i, x)} \bmod p}, T_i$

$S \rightarrow U_i: E_i,$

$(ID_i, C_i, U_i \stackrel{sk}{\longleftrightarrow} S, T_i, E_i, T_s)_{H(ID_i)^{\alpha\, H(ID_i, x)} \bmod p}, T_s$

Second, we make the following assumptions about the initial state of the scheme to analyze the proposed scheme:

$A_1: U_i \mid\equiv \#(T_1)$;

$A_2: S \mid\equiv \#(T_1)$;

$A_3: U_i \mid\equiv \#(T_2)$;

$A_4: S \mid\equiv \#(T_2)$;

$A_5: U_i \mid\equiv U_i \xleftarrow{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p} S$;

$A_6: S \mid\equiv U_i \xleftarrow{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p} S$;

$A_7: S \mid\equiv U_i \Rightarrow (U_i \xleftarrow{sk} S)$;

$A_8: U_i \mid\equiv S \Rightarrow (U_i \xleftarrow{sk} S)$;

Third, we analyze the idealized form of the proposed scheme based on the BAN logic rules and the assumptions. The main proofs are stated as follows:

According to the message M_1, we could get

$$S_1: S \triangleleft (ID_i, C_i, (ID_i, C_i, U_i \xleftarrow{sk} S, T_i)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}, T_i).$$

According to the assumption A_6, we apply the message-meaning rule to get

$$S_2: S \mid\equiv U_i \mid\sim (ID_i, C_i, U_i \xleftarrow{sk} S, T_i)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}.$$

According to the assumption A_2, we apply the freshness-conjuncatenation rule to get

$$S_3: S \mid\equiv \#((ID_i, C_i, U_i \xleftarrow{sk} S, T_i)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}).$$

According to S_2 and S_3, we apply the nonce-verification rule to derive

$$S_4: S \mid\equiv U_i \mid\equiv (ID_i, C_i, U_i \xleftarrow{sk} S, T_i)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}.$$

According to A_6 and S_4, we apply the BAN logic rule to break conjunctions to produce

$$S_5: S \mid\equiv U_i \mid\equiv (U_i \xleftarrow{sk} S). \quad \textbf{(Goal 4)}$$

According to the assumption A_7 and S_5, we apply the jurisdiction rule to get

$$S_6: S \mid\equiv (U_i \xleftarrow{sk} S). \quad \textbf{(Goal 3)}$$

According to the message M_2, we could get

$$S_7: U_i \triangleleft (E_i, (ID_i, C_i, U_i \xleftarrow{sk} S, T_i, E_i, T_s)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}, T_s).$$

According to the assumption A_5, we apply the message-meaning rule to get

$$S_8: U_i \mid\equiv S \mid\sim (ID_i, C_i, U_i \xleftarrow{sk} S, T_i, E_i, T_s)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}.$$

According to the assumption A_3, we apply the freshness-conjuncatenation rule to get

$$S_9: U_i \mid\equiv \#((ID_i, C_i, U_i \xleftarrow{sk} S, T_i, E_i, T_s)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}).$$

According to S_8 and S_9, we apply the nonce-verification rule to derive

$$S_{10}: U_i \mid\equiv S \mid\equiv (ID_i, C_i, U_i \xleftarrow{sk} S, T_i, E_i, T_s)_{H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p}.$$

According to A_5 and S_{10}, we apply the BAN logic rule to break conjunctions to produce

$$S_{11}: U_i \mid\equiv S \mid\equiv (U_i \xleftarrow{sk} S). \quad \textbf{(Goal 2)}$$

According to the assumption A_7 and S_5, we apply the jurisdiction rule to get

$$S_{12}: U_i \mid\equiv (U_i \xleftarrow{sk} S). \quad \textbf{(Goal 1)}$$

According to **(Goal 1)**, **(Goal 2)**, **(Goal 3)** and **(Goal 4)**, we know that both of the user U_i and the server S believe that the session key sk is shared between U_i and S.

5.2. Discussions on the possible attacks

Device stolen attack: An adversary A could read all the authentication information $\{p, q, H, Y_i, r_i\}$ from the user U_i's USB and get a message $M_1 = \{ID_i, C_i, V_i, T_i\}$ sent by U_i, where $Y_i = H(ID_i)^{H(ID_i,x)} + H(PW_i, r_i) \bmod p$, $Y_i' = Y_i - H(PW_i, r_i) \bmod p$, $C_i = H(ID_i)^{\alpha} \bmod p$, $D_i = (Y_i')^{\alpha} \bmod p$ and $V_i = H(ID_i, C_i, D_i, T_i)$. A could guess a password PW_i^* and compute $Y_i^* = Y_i - H(PW_i^*, r_i) \bmod p$. However, A cannot compute D_i from C_i and Y_i^* since he will face with the computational Diffie-Hellman problem. Then, A cannot verify the correctness of PW_i^*. Therefore, our scheme could withstand the device stolen attack.

Privileged insider attack: In the registration of our scheme, the user U_i sends $H(PW_i, r_i)$ to the server S instead of PW_i. The privileged-insider A of the server could get the value $H(PW_i, r_i)$. However, he cannot deduce PW_i since it is protected by the hash function H and the random number r_i. Therefore, our scheme could withstand the privileged insider attack.

Replay attack: An adversary A could intercept the login message $M_1 = \{ID_i, C_i, V_i, T_i\}$ sent by the user U_i. He may re-send it to login in the server. However, the server could find the attack since he will check the freshness of T_i. Therefore, our scheme could withstand the replay attack.

Impersonation attack: Suppose an adversary A wants to impersonate a legal user U_i to login in the server. He could generate a random number $\alpha \in Z_q^*$ and compute $C_i = H(ID_i)^{\alpha} \bmod p$. However, he cannot generate a legal $V_i = H(ID_i, C_i, D_i, T_i)$ to pass the server's verification since he cannot compute $D_i = H(ID_i)^{\alpha\, H(ID_i,x)} \bmod p$ without the knowledge of $H(ID_i)^{H(ID_i,x)} \bmod p$. Therefore, our scheme could withstand the impersonation attack.

Server spoofing attack: Suppose an adversary A wants to impersonate the server to the user U_i. A has

to generate a legal $V_s = H(ID_i, C_i, D_i', T_i, E_i, T_s)$ when he intercepts the login message $M_1 = \{ID_i, C_i, V_i, T_i\}$. However, he cannot compute $D_i' = H(ID_i)^{\alpha\, H(ID_i, x)} \bmod p$ without the knowledge of server's secret key x. Therefore, our scheme could withstand the server spoofing attack.

Perfect forward secrecy: In our scheme, the adversary A may get the server's secret key and the user's authentication information. Since $sk = H(H(ID_i)^{\alpha\,\beta} \bmod p)$, the adversary A has to compute $H(ID_i)^{\alpha\,\beta} \bmod p$ from $C_i = H(ID_i)^{\alpha} \bmod p$ and $E_i = H(ID_i)^{\beta} \bmod p$ if he wants to get the session key generated in previous session. Then, he has to solve the computational Diffie-Hellman problem. Therefore, our scheme could provide perfect forward secrecy.

No key control: In our scheme, the session key is generated by computing $sk = H(E_i^{\alpha} \bmod p) = H(C_i^{\beta} \bmod p) = H(H(ID_i)^{\alpha\,\beta} \bmod p)$, where α and β are two random numbers generated by the user and the server separately. So neither entity can force the session key to a p re-selected value. Therefore, our protocol satisfies the no key control.

6. Performance analysis

In this section, we will compare our scheme with two latest schemes. i.e. Rhee et al.'s scheme [13] and Chen et al.'s scheme [16]. For convenience, some notations are defined as follows.

- t_e : the time complexity of exponentiation operation.
- t_m : the time complexity of multiplication/division operation.
- t_h : the time complexity of hashing operation.

Table 1. Comparison of different schemes

	Rhee et al.'s scheme	Chen et al.'s scheme	Our scheme
Computational cost (user side)	$3t_e+2t_m+3t_h$	$2t_e+2t_m+4t_h$	$3t_e+4t_h$
Computational cost (server side)	$3t_e+t_m+2t_h$	$t_e+t_m+4t_h$	$3t_e+4t_h$
Withstand device stolen attack	No	No	Yes
Withstand privileged insider attack	No	No	Yes
Withstand replay attack	Yes	Yes	Yes
Withstand impersonation attack	Yes	Yes	Yes
Withstand server spoofing attack	Yes	Yes	Yes
Perfect forward secrecy	No	No	Yes
No key control	No	No	Yes

In Table 1, we summarize the comparison results of the related schemes. According to the comparison given in Table 1, Chen et al.'s scheme [16] has better performance than both of Rhee et al.'s scheme [13] and our scheme, whereas our scheme and Rhee et al.'s scheme has similar performance. However, both of Rhee et al.'s scheme [13] and Chen et al.'s scheme [16] are vulnerable to the device stolen attack and the privileged insider attack. Moreover, both of Rhee et al.'s scheme [13] and Chen et al.'s scheme [16] do not support perfect forward secrecy and no key control. Our scheme could overcome these security weaknesses. It is acceptable to enhance security at the cost of increasing the computational complexity slightly. Therefore, our scheme is more suitable for practical applications.

7. Conclusions

In this paper, we analyze the security of Chen et al.'s password-based remote user authentication scheme without smart cards. We point out that four weaknesses exist in their scheme. To improve the security, we also propose a security-enhanced scheme. The analysis shows that our scheme could overcome the weaknesses in Chen et al.'s scheme and is more suitable for practical applications.

Acknowledgments

The authors thank Prof. Vacius Jusas and the anonymous reviewers for their valuable comments. This research was supported by the Specialized Research Fund for the Doctoral Program of Higher Education of China (No. 20110141120003), the National Natural Science Foundation of China (No. 61101112), and the Postdoctoral Science Foundation of China (No. 2011M500775).

References

[1] **L. Lamport.** Password authentication with insecure communication. *Communications of the ACM*, 1981, Vol. 24, No. 11, pp. 770–772.

[2] **M. K. Khan, J. S. Zhang.** Improving the security of 'a flexible biometrics remote user authentication scheme'. *Computer Standards & Interfaces*, 2007, Vol. 29, No. 1, pp. 82–85.

[3] **J. Y. Liu, A. M. Zhou, M. X. Gao.** A new mutual authentication scheme based on nonce and smart cards. *Computer Communications*, 2008, Vol. 31, No. 10, pp. 2205–2209.

[4] **T. Goriparthi, M. L. Das, A. Saxena.** An improved bilinear pairing based remote client authentication protocol. *Computer Standard Interface*, 2009, Vol. 31, No. 1, pp. 181–185.

[5] **C. T. Li, C. C. Lee.** A robust remote user authentication scheme using smart card. *Information Technology and Control*, 2011, Vol. 40, No. 3, pp. 236–245.

[6] **D. He, J. Chen, J. Hu.** Further improvement of Juang et al.'s password-authenticated key agreement scheme using smart cards. *Kuwait Journal of Science & Engineering*, 2011, Vol. 38, No. 2A, pp. 55–68.

[7] **D. He, Y. Chen, J. Chen.** Cryptanalysis and improvement of an extended chaotic maps-based key agreement protocol. *Nonlinear Dynamics*, 2012, Vol. 69, No. 3, pp. 1149–1157.

[8] **D. He, J. Chen, J. Hu.** Improvement on a smart card based password authentication scheme. *Journal of Internet Technology*, 2012, Vol. 13, No. 3, pp. 405–410.

[9] **D. He, J. Chen, J. Hu.** An ID-based client authentication with key agreement protocol for mobile client–server environment on ECC with provable security. *Information Fusion*, 2012, Vol. 13, No. 3, pp. 223–230.

[10] **D. He.** An efficient remote user authentication and key exchange protocol for mobile client-server environment from pairings. *Ad Hoc Networks*, 2012, Vol. 10, No. 6, pp. 1009–1016.

[11] **D. He, J. Chen, R. Zhang.** A more secure authentication scheme for telecare medicine information systems. *Journal of Medical Systems*, 2012, Vol. 36, No. 3, pp. 1989–1995.

[12] **D. He**. Cryptanalysis of an authenticated key agreement protocol for wireless mobile communications. *ETRI Journal*, 2012, Vol. 34, No. 3, pp. 482–484.

[13] **H. S. Rhee, J. O. Kwon, D. H. Lee.** A remote user authentication scheme without using smart cards. *Computer Standards & Interfaces*, 2009, Vol. 31, No. 1, pp. 6–13.

[14] **Z. Tan.** Security analysis of two password authentication schemes. In: *2009 Eighth International Conference on Mobile Business*, 2009, pp. 296–300.

[15] **D. He, J. Chen, J. Hu.** On security of a remote user authentication scheme without using smart cards, https://eprint.iacr.org/2010/306.pdf.

[16] **B. Chen, W. Kuo, L. Wuu.** A secure password-based remote user authentication scheme without smart cards. *Information Technology and Control*, 2012, Vol. 41, No. 1, pp. 53–59.

[17] **S. Wu, Y. Zhu, Q. Pu.** Robust smart-cards-based user authentication scheme with user anonymity. *Security and Communication Networks*, 2012, Vol. 5, No. 2, pp. 236–248.

[18] **H. C. Hsiang, W. K. Shih.** Improvement of the secure dynamic ID based remote user authentication next term scheme for multi-server environment. *Computer Standards & Interfaces*, 2009, Vol. 31, No. 6, pp. 1118–1123.

[19] **D. He, S. Wu, J. Chen.** Note on 'Design of improved password authentication and update scheme based on elliptic curve cryptography'. *Mathematical and Computer Modelling*, 2012, Vol. 55, No. 3-4, pp. 1661–1664.

[20] **M. Burrows, M. Abadi, R. Needham.** A logic of authentication. *ACM Transactions on Computer Systems*, 1990, Vol. 8, No. 1, pp. 18–36.

Received October 2012.