ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2014, T. 43, Nr. 2

An Enhanced and Secure Three-Party Password-based Authenticated Key Exchange Protocol without Using Server's Public-Keys and Symmetric Cryptosystems

Mohammad Sabzinejad Farash

Department of Mathematics and Computer Sciences,
Kharazmi University, Tehran, Iran,
e-mail: sabzinejad@khu.ac.ir

Mahmoud Ahmadian Attari

Faculty of Electrical and Computer Engineering,
K.N. Toosi University of Technology, Tehran, Iran,
e-mail: mahmoud@eetd.kntu.ac.ir

crossref http://dx.doi.org/10.5755/j01.itc.43.2.3790

Abstract. Password-based authenticated key exchange protocol is a type of authenticated key exchange protocols which enables two or more communication entities, who only share weak, low-entropy and easily memorable passwords, to authenticate each other and establish a high-entropy secret session key. In 2012, Tallapally proposed an enhanced three-party password-based authenticated key exchange protocol to overcome the weaknesses of Huang's scheme. However, in this paper, we indicate that the Tallapally's scheme not only is still vulnerable to undetectable online password guessing attack, but also is insecure against off-line password guessing attack. Therefore, we propose a more secure and efficient scheme to overcome the security flaws.

Keywords: Password-based key exchange protocol; Password guessing attack; Client-server authentication.

1. Introduction

Authenticated key exchange (AKE) protocols [1–8] help communicating entities, who are communicating over an insecure network, to establish a secret session key to be used for protecting their subsequent communication. Password-based authenticated key exchange (PAKE) protocol is a type of AKE protocols which enables two or more communication entities, who only share weak, low-entropy and easily memorable passwords, to authenticate each other and establish a high-entropy secret session key.

PAKE protocols were first proposed in the twoparty setting (2PAKE) which are quite suitable for the client-server architecture [9–13]. However, these protocols are very inconvenient for large scale clientclient communication environments. Since each client needs to remember different password for each partner who communicates with, for a large network, it may strain the storage capacity of the clients. To avoid this problem, PAKE protocols in the three-party setting (3PAKE) are developed. In a 3PAKE protocol, a trusted server mediates between two communication clients and each client only needs to share a password with the server.

In order to design a secure and practicable 3PAKE, many protocols have been proposed. The main security threats for the 3PAKE protocols are dictionary attacks. To protect these protocols against dictionary attacks, there are three main approaches: using the server's public key [14–18], using symmetric cryptosystems [19–21], and without using server's public keys and symmetric cryptosystems [22–36]. In this paper, we restrict our attention to the 3PAKE protocols that require neither the server public keys nor symmetric cryptosystems.

In 2009, Huang [26] proposed a 3PAKE protocol without any server's public key and symmetric cryptosystems. However, Yoon and Yoo [27] demonstrated that Huang's 3PAKE protocol is vulnerable to undetectable online password guessing attacks and off-line password guessing attacks by any other user. Based on the Yoon and Yoo's attacks, Wu et al. [28] showed that Huang's protocol is also vulnerable to key compromise impersonation attack. Wu et al. [28] then proposed an enhanced 3PAKE protocol which uses server's public key.

In 2011, Lee et al. [29] proposed a 3PAKE protocol without server public keys in order to reduce the number of steps in communication. After that, Chang et al. [30] proposed an efficient 3PAKE protocol which requires neither the server public key nor symmetric cryptosystems based upon Lee et al.'s protocol. However,Wu et al. [31] pointed out that Chang et al.'s 3PAKE is insecure by the password guessing attacks. Very recently, Xiong et al. [18] demonstrated that the Wu et al.'s 3PAKE protocol is vulnerable to key compromise impersonation attack and proposed an improved 3PAKE protocol as well as an enhanced scheme which uses the server's public key. The attack on these protocols demonstrates the necessity of considering KCI resilience for 3PAKE protocols. To overcome the security problems of the Chang et al.'s scheme, Tso [32] also proposed an improved scheme without using the server public key or symmetric cryptosystems.

In 2011, Chien [33] proposed an efficient 3PAKE protocol using verifiers which requires neither the server public key nor symmetric cryptosystems. He claimed that his protocol could resist against various dictionary attacks. However, Pu et al. [34] pointed out that Chien's scheme still has a critical security weakness, which opens door to a partition attack (off-line dictionary attack). To overcome the security weaknesses of the Chien's scheme, Pu et al. [34] and Liu et al. [15] independently proposed enhanced schemes. The Pu et al.'s scheme [34] is a verifier-based 3PAKE protocol which requires neither the server public key nor symmetric cryptosystems, while the Liu et al.'s scheme [15] uses the server public key.

In 2012, an unknown key share attack was demonstrated on the Huang's 3PAKE protocol [26] by Tallapally [35]. She then proposed an enhanced 3PAKE protocol without using server's public keys and symmetric cryptosystems to overcome the security flaws of the Huang's protocol. However, this paper indicates that the Tallapally's scheme [35] not only is still vulnerable to undetectable online password guessing attack, but also is insecure against offline password guessing attack. Therefore, we propose a more secure and efficient scheme to overcome the security flaws of the previous schemes.

The rest of this paper is organized as follows. In Section 2, we review the Tallapally's 3PAKE protocol. In Section 3, we propose undetectable online password guessing attack and off-line password guessing attack on the Tallapally's protocol. An enhanced 3PAKE protocol is proposed in Section 4. In Section 5, we analyze the security of the proposed scheme. In Section 6, we make a comparison between our scheme and some related schemes. Finally, we conclude our paper in Section 7.

2. A brief review of the Tallapally's 3PAKE protocol

This section briefly reviews the Tallapally's threeparty password-based key exchange (3PAKE) protocol [35].

Table 1. The Notations

Notation	Description
A, B	legitimate users
pw	the password of a legitimate user
S	a remote server
p	a large prime number
$\mathbb{Z}_p^*$	the non-zero residues mod p
q	a large prime with $q\|(p-\ 1)$
G	a multiplicative group of order q
g	a generator of G
$F_S(.)$	a trapdoor function
$h(.)$	a conventional hash function
$\oplus$	the exclusive-or operation

2.1. Notations

The notations used throughout this paper are summarized in Table 1.

2.2. Protocol description

For a detailed analysis, we review the Tallapally's three-party password-based key exchange (3PAKE) protocol [35].We show this protocol in Fig. 1 and more details are provided as follows:

Step 1. A randomly chooses $x,\ r_A \in \mathbb{Z}_q^*$ and computes $F_S(r_A)$ and $R_A = (g^x) \oplus h(r_A, pw_A, A, B)$. Then, she sends $(A, R_A, F_S(r_A))$ to S. Similarly, B also selects $y, r_B \in \mathbb{Z}_q^*$, computes $F_S(r_B)$ and $R_B = (g^y \bmod p) \oplus\ h(r_B, pw_B, A, B)$, and sends $(B, R_B, F_S(r_B))$ to S.

Step 2. Upon receiving the messages $(A, R_A, F_S(r_A))$ and $(B, R_B, F_S(r_B))$, S first extracts R_A and R_B from $F_S(r_A)$ and $F_S(r_B)$, and obtains $g^x = R_A \oplus h(r_A,\ pw_A, A, B)$ and $g^y = R_B \oplus h\ (r_B,\ pw_B, A, B)$. Then, S chooses a random number $z \in_R \mathbb{Z}_q^*$ and computes $a = g^{xz}$ and $b = g^{yz}$. Finally, S computes $Z_A = b \oplus h(r_A,\ pw_A, g^x)$ and $Z_B = a \oplus h(r_B,\ pw_B,\ g^y)$, and sends (Z_A) and (Z_B) to A and B, respectively.

Step 3. After receiving the message (Z_A), A computes $b = Z_A \oplus h(r_A, pw_A, g^x)$ and the session key $K = b^x = g^{xyz}$. Then she computes $S_A = h(K, A)$ and sends to B. At the same time, upon receiving the message (Z_B), B computes $a = Z_B \oplus h(r_B,\ pw_B, g^y)$ and the session

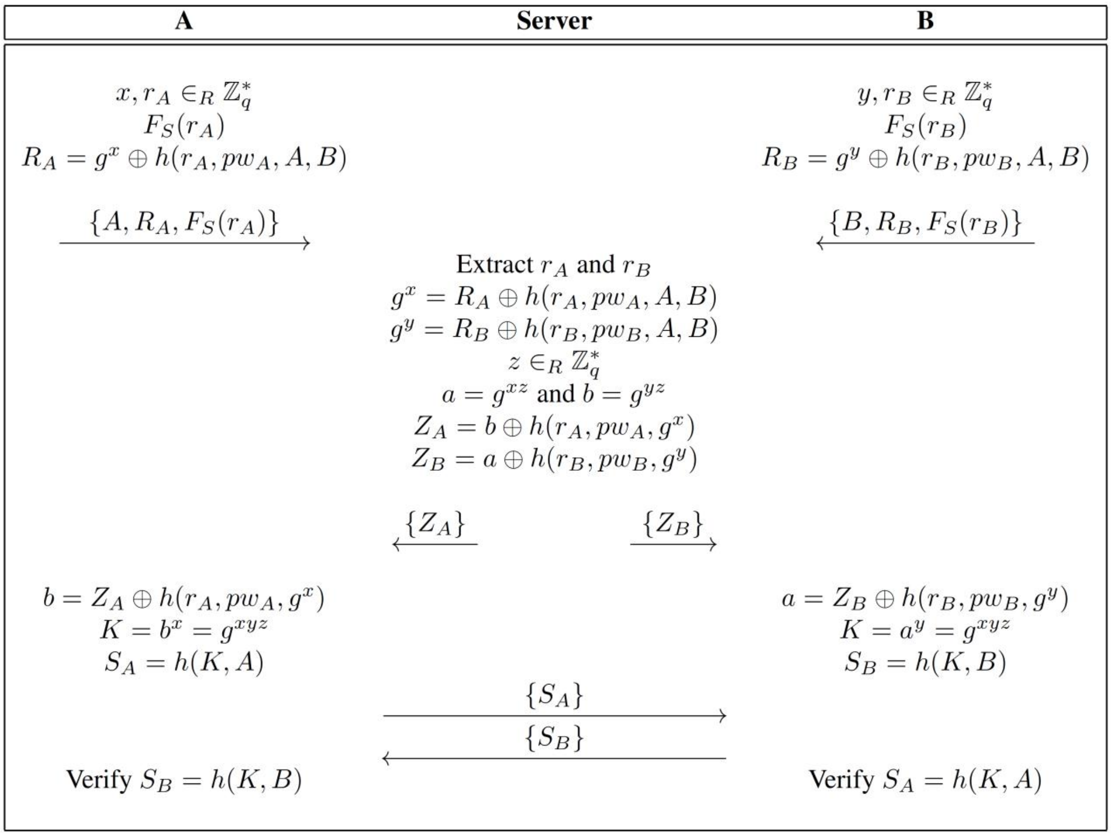
A
Server
B
$x, r_A \in_R \mathbb{Z}_q^*$
$F_S(r_A)$
$R_A = g^x \oplus h(r_A, pw_A, A, B)$
$y, r_B \in_R \mathbb{Z}_q^*$
$F_S(r_B)$
$R_B = g^y \oplus h(r_B, pw_B, A, B)$
$\{A, R_A, F_S(r_A)\}$
$\{B, R_B, F_S(r_B)\}$
Extract r_A and r_B
$g^x = R_A \oplus h(r_A, pw_A, A, B)$
$g^y = R_B \oplus h(r_B, pw_B, A, B)$
$z \in_R \mathbb{Z}_q^*$
$a = g^{xz}$ and $b = g^{yz}$
$Z_A = b \oplus h(r_A, pw_A, g^x)$
$Z_B = a \oplus h(r_B, pw_B, g^y)$
$\{Z_A\}$
$\{Z_B\}$
$b = Z_A \oplus h(r_A, pw_A, g^x)$
$K = b^x = g^{xyz}$
$S_A = h(K, A)$
$a = Z_B \oplus h(r_B, pw_B, g^y)$
$K = a^y = g^{xyz}$
$S_B = h(K, B)$
$\{S_A\}$
$\{S_B\}$
Verify $S_B = h(K, B)$
Verify $S_A = h(K, A)$

Figure 1. The Tallapally's 3PAKE protocol

key $K = a^x = g^{xyz}$. Then he computes $S_B = h(K, B)$ and sends to A.

Step 4: Finally, A and B verify S_A and S_B, respectively.

3. Cryptanalysis of the Tallapally's protocol

3.1. Undetectable online password guessing attack

The scenario of the proposed undetectable online password guessing attack is outlined in Fig. 2. In this attack, a malicious user B performs as follows:

Step 1. $B \rightarrow S : (A,\ R'_A,\ F_S(r'_A)),\ (B,\ R_B, F_S(r_B))$

Suppose that, B is a malicious user mediating between S and A .Without any contribution from A, B randomly chooses $r'_A,\ r_B \in \mathbb{Z}_q^*$ and $X \in \mathbb{Z}_p^*$, guesses a password pw'_A, and computes $F_S(r'_A), F_S(r_B), R'_A = X \oplus h(r'_A, pw'_A, A,\ B)$ and $R_B = X \oplus h(r_B,\ pw'_B, A, B)$. Finally, B sends $(A, R'_A, F_S(r'_A))$ and $(B, R_B, F_S(r_B))$ to S.

Step 2. $S \rightarrow A : Z_A$ and $S \rightarrow B : Z_B$

Upon receiving the messages $(A, R'_A, F_S(r'_A))$ and $(B, R_B, F_S(r_B))$, S first extracts r'_A and r_B from $F_S(r'_A)$ and $F_S(r_B)$ and obtains $X' = R'_A \oplus h(r'_A, pw_A, A, B)$ and $X = R_B \oplus h(r_B, pw_B, A, B)$. Then, S chooses a random number $z \in \mathbb{Z}_q^*$ and computes $a = X'^z$ and $b = X^z$. Finally, S computes $Z_A = b \oplus h(r'_A, pw_A, X')$ and $Z_B = a \oplus h(r_B, pw_B, X)$, and sends Z_A and Z_B to A and B, respectively.

Step 3. Upon receiving Z_B and intercepting Z_A, B obtains $a = Z_B \oplus h(r_B,\ pw_B, X)$ and $b = Z_A \oplus\ h(r'_A, pw'_A, X)$, and checks if $a = b$. If it holds, then B confirms that the guessed password pw'_A is correct. It is clear that if the guessed password is correct, then $a = b = X^z$.

Step 4. Otherwise, B repeatedly performs the above steps (1–3) to find the correct password.

3.2. Off-line password guessing attack

The scenario of the proposed off-line password guessing attack is outlined in Fig. 3. In this attack, a malicious user B performs as follows:

Step 1. $B \rightarrow S : (A, R'_A, F_S(r'_A)), (B, R_B, F_S(r_B))$

Suppose that, B is a malicious user mediating between S and A .Without any contribution from A, B randomly chooses $r'_A, r_B \in \mathbb{Z}_q^*$ and $X \in \mathbb{Z}_p^*$, and computes $F_S(r'_A)$, $F_S(r_B)$, $R'_A = X$ and $R_B = 1 \oplus h(r_B, pw_B, A, B)$. Finally, B sends $(A, R'_A, F_S(r'_A))$ and $(B, R_B, F_S(r_B))$ to S.

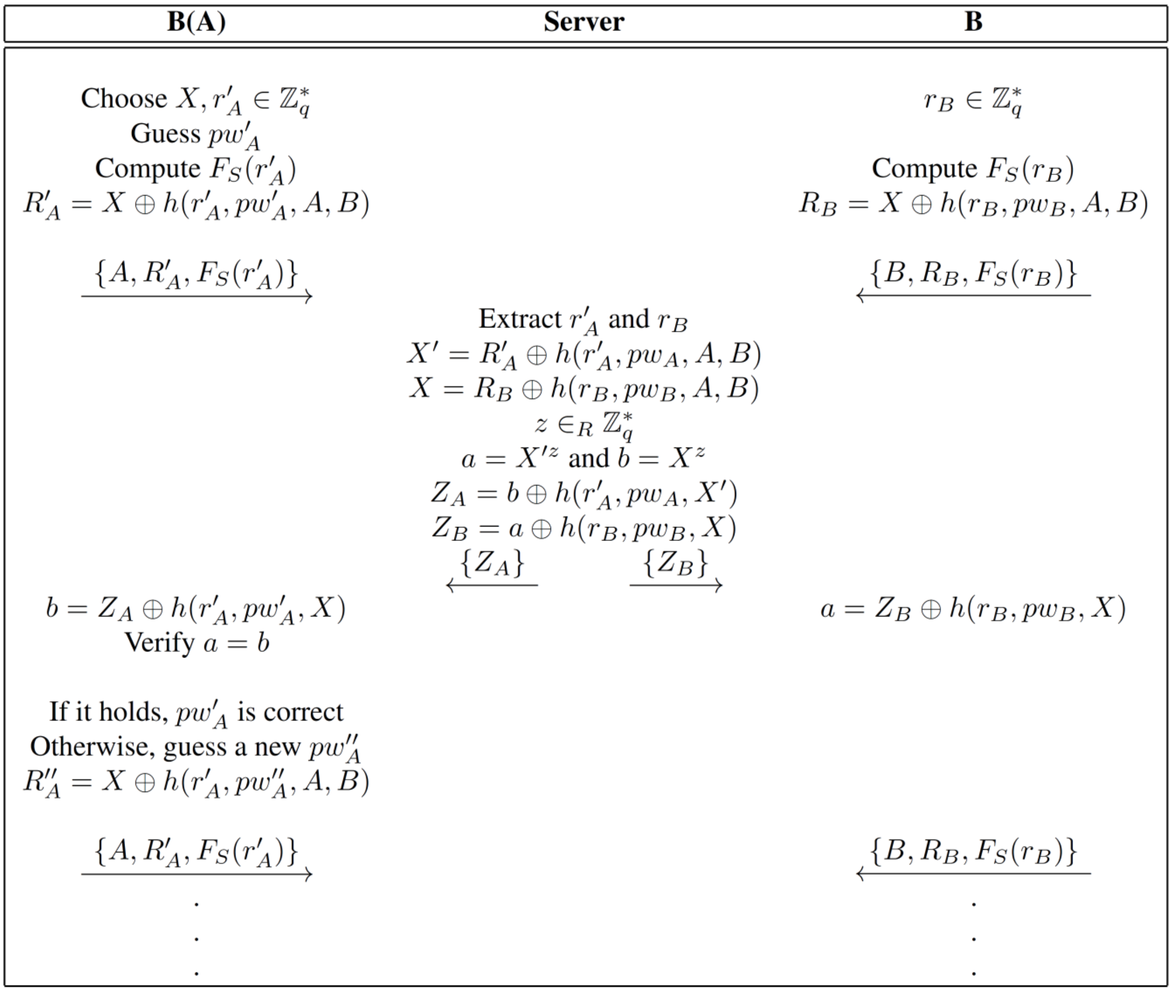
B(A)
Server
B
Choose $X, r'_A \in \mathbb{Z}_q^*$
Guess pw'_A
Compute $F_S(r'_A)$
$R'_A = X \oplus h(r'_A, pw'_A, A, B)$
$r_B \in \mathbb{Z}_q^*$
Compute $F_S(r_B)$
$R_B = X \oplus h(r_B, pw_B, A, B)$
$\{A, R'_A, F_S(r'_A)\}$
$\{B, R_B, F_S(r_B)\}$
Extract r'_A and r_B
$X' = R'_A \oplus h(r'_A, pw_A, A, B)$
$X = R_B \oplus h(r_B, pw_B, A, B)$
$z \in_R \mathbb{Z}_q^*$
$a = X'^z$ and $b = X^z$
$Z_A = b \oplus h(r'_A, pw_A, X')$
$Z_B = a \oplus h(r_B, pw_B, X)$
$\{Z_A\}$
$\{Z_B\}$
$b = Z_A \oplus h(r'_A, pw'_A, X)$
Verify $a = b$
$a = Z_B \oplus h(r_B, pw_B, X)$
If it holds, pw'_A is correct
Otherwise, guess a new pw''_A
$R''_A = X \oplus h(r'_A, pw''_A, A, B)$
$\{A, R'_A, F_S(r'_A)\}$
$\{B, R_B, F_S(r_B)\}$

Figure 2. Undetectable online password guessing attack

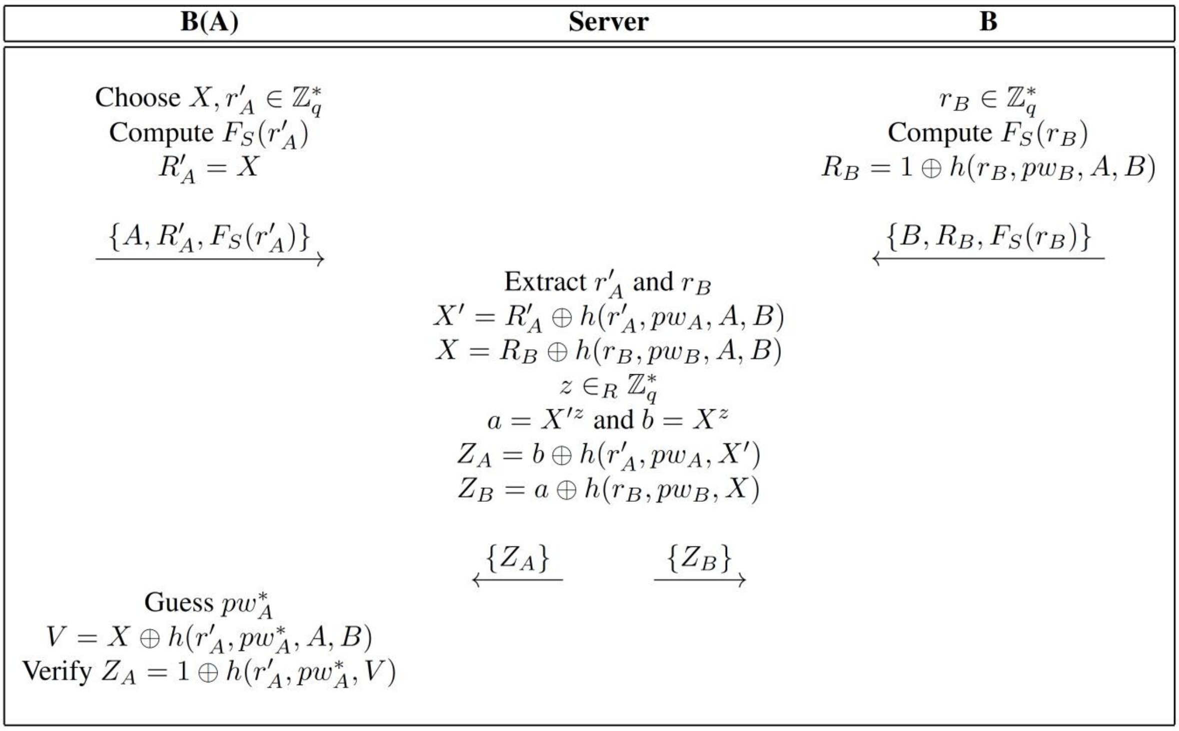
B(A)
Server
B
Choose $X, r'_A \in \mathbb{Z}_q^*$
Compute $F_S(r'_A)$
$R'_A = X$
$r_B \in \mathbb{Z}_q^*$
Compute $F_S(r_B)$
$R_B = 1 \oplus h(r_B, pw_B, A, B)$
$\{A, R'_A, F_S(r'_A)\}$
$\{B, R_B, F_S(r_B)\}$
Extract r'_A and r_B
$X' = R'_A \oplus h(r'_A, pw_A, A, B)$
$X = R_B \oplus h(r_B, pw_B, A, B)$
$z \in_R \mathbb{Z}_q^*$
$a = X'^z$ and $b = X^z$
$Z_A = b \oplus h(r'_A, pw_A, X')$
$Z_B = a \oplus h(r_B, pw_B, X)$
$\{Z_A\}$
$\{Z_B\}$
Guess pw^*_A
$V = X \oplus h(r'_A, pw^*_A, A, B)$
Verify $Z_A = 1 \oplus h(r'_A, pw^*_A, V)$

Figure 3. Off-line password guessing attack

Step 2. $S \rightarrow A : Z_A$ and $S \rightarrow B : Z_B$

Upon receiving the messages $(A, R'_A, F_S(r'_A))$ and $(B, R_B, F_S(r_B))$, S first extracts r'_A and r_B from $F_S(r'_A)$ and $F_s(r_B)$ and obtains $X' = R'_A \oplus h(r'_A, pw_A, A, B)$ and $Y = R_B \oplus h(r_B, pw_B, A, B)$. We can see Y is equal to one because of $R_B \oplus h(pw_B, A, B) = (1 \oplus h(pw_B, A, B)) \oplus h(pw_B, A, B) = 1$. Then, S chooses a random number $z \in \mathbb{Z}_q^*$ and computes $a = X^z$ and $b = Y^z = 1$. Finally, S computes $Z_A = b \oplus h(r'_A, pw_A, X')$ and $Z_B = a \oplus h(r_B, pw_B, Y)$, and sends Z_A and Z_B to A and B, respectively.

Step 3. After receiving Z_B and intercepting Z_A, B guesses a password pw'_A and checks if $Z_A = 1 \oplus h(r'_A, pw'_A, X \oplus h(r'_A, pw'_A, A, B))$. If it holds, then B confirms that the guessed password pw'_A is the correct one. It is obvious that, if the guessed password is correct, then

$$\begin{aligned} Z_A &= b \oplus h(r'_A, pw'_A, X') \\ &= 1 \oplus h(r'_A, pw'_A, X' \oplus h(r'_A, pw'_A, A, B)), \end{aligned}$$

because of b = 1 and

$$\begin{aligned} X' &= R'_A \oplus (r'_A, pw'_A, A, B) \\ &= X \oplus h(r'_A, pw'_A, A, B). \end{aligned}$$

Step 4. Otherwise, B chooses another password pw'_A and repeatedly performs above Step 3 to obtain the correct password.

4. The proposed 3PAKE protocol

This section aims to propose an enhanced 3PAKE protocol to overcome the above mentioned problems with the Tallapally's protocol. Detailed steps of the proposed protocol are in Fig. 4 and are described as follows:

Step 1. A randomly chooses $x \in \mathbb{Z}_q^*$ and computes $R_A = g^x + h(pw'_A, A, B) \bmod p$. Then, she sends (A, B, R_A) to S. Similarly, B also selects $y \in_R \mathbb{Z}_q^*$, computes $R_B = g^y + h(pw_A, A, B) \bmod p$, and sends (B, A, R_B) to S.

Step 2. Upon receiving the messages (A, B, R_A) and (B, A, R_B), S obtains $g^x = R_A \oplus h(pw_A, A, B) \bmod p$ and $g^y = R_B \oplus h(pw_B, A, B) \bmod p$, then chooses a random number $z \in \mathbb{Z}_q^*$ and computes $T_S = g^z \bmod p$, $K_{SA} = g^{xz} \bmod p$ and $K_{SB} = g^{yz} \bmod p$. Finally, S computes $Z_A = h(0, A, B, S, pw_A, K_{SA})$ and $Z_B = (0, A, B, S, pw_B, K_{SB})$, and sends (Z_A, T_S) and (Z_B, T_S) to A and B, respectively.

Step 3. After receiving the message (Z_A, T_S), A computes $K_{AS} = (T_S)^x \bmod p$ and verifies $h(A, B, S, pw_A, K_{AS})? = Z_A$. If it holds, she computes $V_A = h(0, A, B, S, pw_A, K_{AS}, T_S)$ and sends to S. At the same time, upon receiving the message (Z_B, T_S), B computes $K_{BS} = (T_S)^y \bmod p$ and verifies $h(0, A, B, S, pw_B, K_{BS})\, ? = Z_B$. If it holds, she computes $V_B = h(A, B, S, pw_B, K_{BS}, T_S)$ and sends to S.

Step 4. Upon receiving the messages V_A and V_B, S verifies $h(A, B, S, pw_A, K_{SA}, T_S)? = V_A$ and $h(A, B, S, pw_B, K_{SB}, T_S)? = V_B$. If they hold, S computes $X_A = K_{SB} + h(1, A, B, S, pw_A, K_{SA}) \bmod p$ and $X_B = K_{SA} + h(1, A, B, S, pw_B, K_{SB}) \bmod p$, and send$X_A$s and X_B to A and B, respectively.

Step 5. After receiving X_A, A computes $K'_{SB} = X_A - h(1, A, B, S, pw_A, K_{SA}) \bmod p$ and the secret shared $K_{AB} = (K'_{SB})^x = g^{xyz} \bmod p$. Then she computes $S_A = h(K_{AB}, A)$ and sends to B. Simultaneously, upon receiving the message X_B, B computes $K'_{SA} = X_B - h(1, A, B, S, pw_B, K_{SB}) \bmod p$ and the secret shared $KBA = (K'_{SA})^y = g^{xyz} \bmod p$. Then he computes $S_B = h(K_{BA}, B)$ and sends to A.

Step 6. Finally, A and B verify S_A and S_B, and compute the session keys $SK = h(A, B, S, K_{AS}, K'_{SB}, K_{AB})$ and $SK = h(A, B, S, K'_{SA}, K_{BS}, K_{BA})$, respectively.

5. Security analysis

5.1. Online password guessing attack

This attack is divided into detectable and undetectable attacks. Detectable online password guessing attack can be preserved by limiting the login times. However, this solution is not useful to prevent the undetectable one. The general solution to prevent undetectable online password guessing attack is that each entity verifies the correctness of the received messages.As can be clearly seen in the proposed protocol, the users and the server verify each other in Step 3 and Step 4, respectively. Therefore, an adversary cannot execute any undetectable online guessing attack on the proposed scheme.

5.2. Off-line password guessing attack

As mentioned in Section 3.2, the vulnerability of Tallapally's protocol to the off-line password guessing attack is dealt with using g^x and g^y in Z_A and Z_B, respectively. Therefore, to overcome this problem, it is sufficient to make use of gxz and gyz in Z_A and Z_B, respectively. As can be seen, this approach is considered in the proposed protocol to compute Z_A and Z_B in Step 2, V_A and V_B in Step 3, and X_A and X_B in Step 4. Therefore, the proposed scheme is secure against the off-line password guessing attack.

..

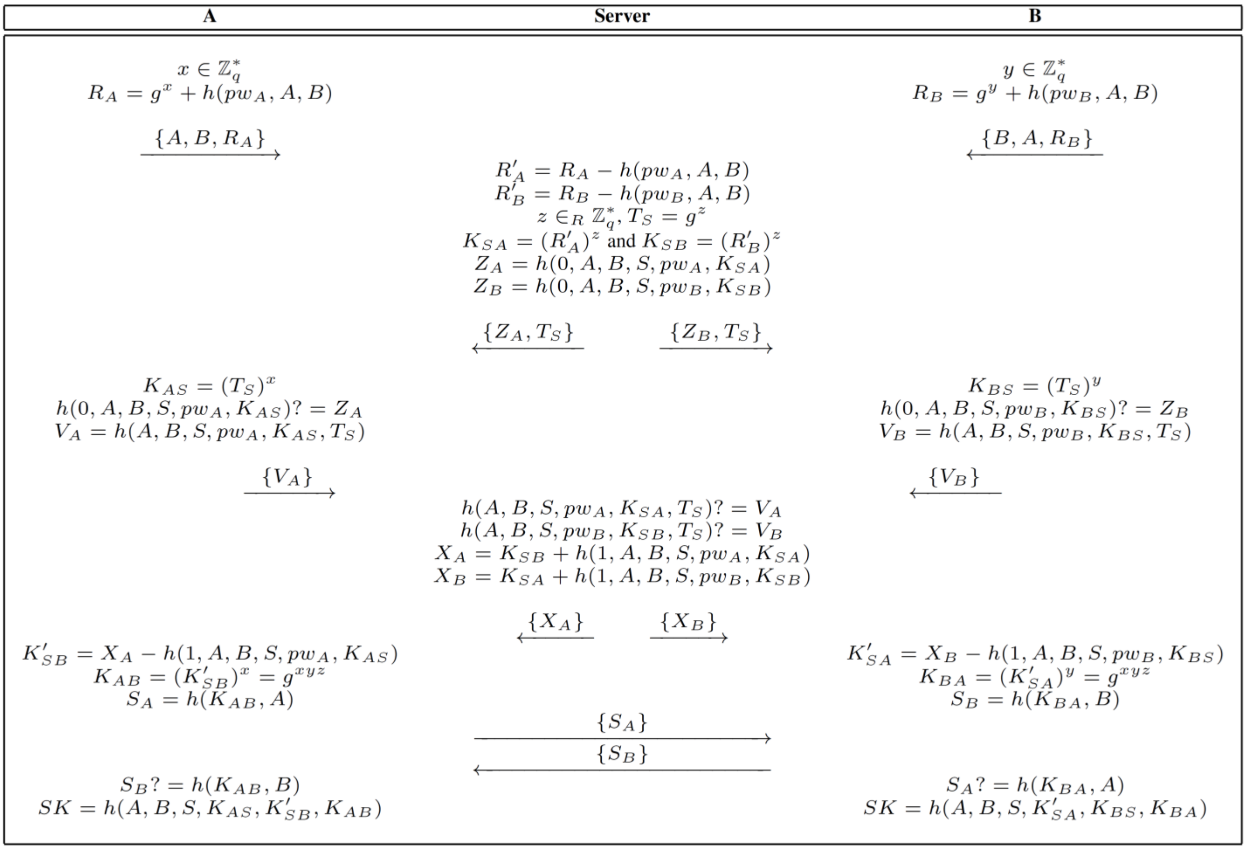
A
Server
B
$x \in \mathbb{Z}_q^*$
$R_A = g^x + h(pw_A, A, B)$
$y \in \mathbb{Z}_q^*$
$R_B = g^y + h(pw_B, A, B)$
$\{A, B, R_A\}$
$\{B, A, R_B\}$
$R'_A = R_A - h(pw_A, A, B)$
$R'_B = R_B - h(pw_B, A, B)$
$z \in_R \mathbb{Z}_q^*, T_S = g^z$
$K_{SA} = (R'_A)^z$ and $K_{SB} = (R'_B)^z$
$Z_A = h(0, A, B, S, pw_A, K_{SA})$
$Z_B = h(0, A, B, S, pw_B, K_{SB})$
$\{Z_A, T_S\}$
$\{Z_B, T_S\}$
$K_{AS} = (T_S)^x$
$h(0, A, B, S, pw_A, K_{AS})? = Z_A$
$V_A = h(A, B, S, pw_A, K_{AS}, T_S)$
$K_{BS} = (T_S)^y$
$h(0, A, B, S, pw_B, K_{BS})? = Z_B$
$V_B = h(A, B, S, pw_B, K_{BS}, T_S)$
$\{V_A\}$
$\{V_B\}$
$h(A, B, S, pw_A, K_{SA}, T_S)? = V_A$
$h(A, B, S, pw_B, K_{SB}, T_S)? = V_B$
$X_A = K_{SB} + h(1, A, B, S, pw_A, K_{SA})$
$X_B = K_{SA} + h(1, A, B, S, pw_B, K_{SB})$
$\{X_A\}$
$\{X_B\}$
$K'_{SB} = X_A - h(1, A, B, S, pw_A, K_{AS})$
$K_{AB} = (K'_{SB})^x = g^{xyz}$
$S_A = h(K_{AB}, A)$
$K'_{SA} = X_B - h(1, A, B, S, pw_B, K_{BS})$
$K_{BA} = (K'_{SA})^y = g^{xyz}$
$S_B = h(K_{BA}, B)$
$\{S_A\}$
$\{S_B\}$
$S_B? = h(K_{AB}, B)$
$SK = h(A, B, S, K_{AS}, K'_{SB}, K_{AB})$
$S_A? = h(K_{BA}, A)$
$SK = h(A, B, S, K'_{SA}, K_{BS}, K_{BA})$

Figure 4. The proposed 3PAKE protocol

5.3. Replay attack

Suppose an active attacker $\mathcal{A}$ intercepts the message of A or B in Step 1 and replays it to impersonate each of them. However, $\mathcal{A}$ cannot compute a correct $K_{AS} = (T_S)^x$ or $K_{BS} = (T_S)^y$ and deliver it to S in Step 3 unless he/she can correctly guess the password pw_A or pw_B and guess z from T_S. When $\mathcal{A}$ tries to guess z from T_S, he/she will face the Discrete Logarithm Problem (DLP) which is untractable. Therefore, the proposed scheme is secure against the replay attack.

5.4. Denning-Sacco attack

Although attacker $\mathcal{A}$ may obtain the session key $SK = h(A, B, S, K_{AS}, K'_{SB}, K_{AB}) = h(A, B, S,\ K'_{SA},\ K_{BS},\ K_{BA})$, for some reasons, he/she cannot obtain the secret passwords pw_A and pw_B, because he/she will face to obtain K_{AS}, K'_{SB} and K_{AB} which are protected by a hash function.

5.5. Impersonation attack

An adversary $\mathcal{A}$ cannot masquerade as server, because he/she cannot compute $Z_A = h(A, B, S, pw_A, K_{SA})$ and $Z_B = h(A, B, S, pw_B, K_{SB})$ without knowing the users' secret passwords. $\mathcal{A}$ also cannot impersonate the users to authenticate with the server, because he/she cannot construct $V_A = h(A, B,\ S, pw_A, K_{AS}, T_S)$ or $V_B = h(A, B, S, pw_B, K_{BS}, T_S)$ without the knowledge of pw_A and pw_B. Therefore, the proposed scheme resists the impersonation attack.

5.6. Modification attack

An adversary $\mathcal{A}$ cannot modify the communicated messages between users and the server, because the users and the server can detect the modification by verifying the received messages in Steps 3-6.

5.7. Known-key security

In this attack, the adversary who has some previous session keys is willing to compute the next session keys. Assume that some previous session keys are known for the adversary $\mathcal{A}$. This does not give him/her any useful information for computing the next session keys because the random numbers x, y and z are changed in each session. Therefore, the proposed protocol satisfies the known-key security.

5.8. Perfect forward secrecy

Perfect forward secrecy means that if long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by the trusted entities is not affected. In the proposed protocol, the adversary who knows pw_A and pw_B cannot determine the previous session keys, because

Table 2. Performance comparison of 3PAKE which have neither the server public keys nor symmetric cryptosystems

	Huang's [26]		Chang's [30]		Wu's [31]		Tallapally's [35]		Ours	
	User	Server	User	Server	User	Server	User	Server	User	Server
No. of exponentiation	2	2	3	4	0	0	2	2	3	2
No. of hash func.	4	4	5	4	4	4	4	4	6	8
No. of trapdoor func.	0	0	0	0	1	2	0	0	0	0
No. of scaler mult.	0	0	0	0	3	4	0	0	0	0
No. of exclusive or	2	4	1	2	0	0	2	4	0	0
Security	DLP		DLP		ECDLP		DLP		DLP	

he/she does not know the shared secret $K_{AB} = K_{BA}$. Therefore, the proposed protocol satisfies the perfect forward secrecy.

6. Performance comparison

In this section, we evaluate the performance and functionality of our proposed protocol and make comparisons with some related 3PAKE protocols. Table 2 shows the performance comparisons of our protocol and some other related protocols. From Table 2, it is obvious that our improved protocol has worth several additional hash operations and an exponentiation more than the other protocols to achieve the security and functionality attributes.

7. Conclusions

In this paper, we briefly reviewed the Tallapally's 3PAKE protocol. We demonstrated that like Huang's scheme, the Tallapally's scheme is still vulnerable to undetectable online password guessing attack. Additionally, we pointed out that the Tallapally's scheme also suffers from the off-line password guessing attack. The main flaw of the Tallapally's protocol is that each entity does not verify the correctness of the received messages. Therefore, we proposed an improved scheme to overcome the security weaknesses of the related schemes and showed that it satisfies the common security attributes.

References

[1] **M. S. Farash, M. Bayat, M. A. Attari.** Vulnerability of two multiple-key agreement protocols. *Computers & Electrical Engineering*, 2011, Vol. 37, No. 2, 199-204.

[2] **M. S. Farash, M. A. Attari, M. Bayat.** A Certificateless Multiple-key Agreement Protocol without Oneway Hash Functions Based on Bilinear Pairings. *IACSIT International Journal of Engineering and Technology*, 2012 Vol. 4, No. 3, 321-325.

[3] **M. S. Farash, M. A. Attari, R. E. Atani, M. Jami.** A new efficient authenticated multiple-key exchange protocol from bilinear pairings. *Computers & Electrical Engineering*, 2013, Vol. 39, No. 2, 530-541.

[4] **M. S. Farash, M. A. Attari.** Provably Secure and Efficient Identity-Based Key Agreement Protocol for Independent PKGs Using ECC. *The ISC International Journal of Information Security*, 2013, Vol. 5, No. 1, 1-15.

[5] **M. S. Farash, M. A. Attari.** A Pairing-free ID-based Key Agreement Protocol with Different PKGs. *International journal of Network Security*, 2014, Vol. 16, No. 2, 143-148.

[6] **S. Haiyan, W. Qiaoyan, Z. Hua, J. Zhengping.** A Strongly Secure Pairing-free Certificateless Authenticated Key Agreement Protocol for Low-Power Devices. *Information Technology and Control*, 2013, Vol. 42, No. 2, 105-112.

[7] **Y. M. Tseng, C. H. Yu, T. Y. Wu.** Towards Scalable Key Management for Secure Multicast Communication. *Information Technology and Control*, 2012, Vol. 41, No. 2, 173-182.

[8] **M. S. Farash, M. A. Attari.** A secure and efficient identity-based authenticated key exchange protocol for mobile client-server networks. *The Journal of Supercomputing*, 2014, doi: 10.1007/s11227-014-1170-5.

[9] **B. L. Chen, W. C. Kuo, L. C. Wuu.** A Secure Password-Based Remote User Authentication Scheme without Smart Cards. *Information Technology and Control*, 2012, Vol. 41, No. 1, 53-59.

[10] **Q. Jiang, J. Ma, G. Li, Z. Ma.** An Improved Password-Based Remote User Authentication Protocol without Smart Cards. *Information Technology and Control*, 2013, Vol. 42, No. 2, 150-158.

[11] **M. Bayat, M. S. Farash, A. Movahed.** A Novel Secure Bilinear Pairing Based Remote User Authentication Scheme with Smart Card. In: *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC)*, 2010, pp. 578-582.

[12] **M. S. Farash, M. A. Attari.** An Enhanced Authenticated Key Agreement for Session Initiation Protocol. *Information Technology and Control*, 2013, Vol. 42, No. 4, 333-342.

[13] **M. S. Farash, M. A. Attari.** Cryptanalysis and improvement of a chaotic maps-based key agreement protocol using Chebyshev sequence membership testing. *Nonlinear Dynamics*, 2013, doi:10.1007/s11071-013-1204-1

[14] **Q. Xie, N. Dong, X. Tan, D. S.Wong, G.Wang.** Improvement of a Three-Party Password-Based Key Exchange Protocol with Formal Verification. *Information Technology and Control*, 2013, Vol. 42, No. 3, 231-237.

[15] **T. Liu, Q. Pu, Y. Zhao, S. Wu.** ECC-Based Password-Authenticated Key Exchange in the Three-Party Setting. *Arabian Journal for Science And Engineerin*, 2013, Vol. 38, No. 8, 2069-2077.

[16] **H. Y. Chien, T. C. Wu.** Provably secure password-based three-party key exchange with optimal message steps. *Computer Journal*, 2009, Vol. 52, No. 6, 646-655.

[17] **T. F. Lee, J. L. Liu, M. J. Sung, S. B. Yang, C. M. Chen.** Communication-efficient three-party protocols for authentication and key agreement. *Computers & Mathematics with Applications*, 2009, Vol. 58, No. 4, 641-648.

[18] **H. Xiong, Y. Chen, Z. Guan, Z. Chen.** Finding and fixing vulnerabilities in several three-party password authenticated key exchange protocols without server public keys. *Information Sciences,* 2013, Vol. 235, No. 1, 329-340.

[19] **H. B. Chen, T. H. Chen, W. B. Lee, C.C. Chang.** Security enhancement for a three-party encrypted key exchange protocol against undectectable on-line password guessing attacks. *Computer Standards & Interfaces*, 2008, Vol. 30, No. 1-2, 95-99.

[20] **J. Zhao, D. Gu.** Provably secure three-party password-based authenticated key. *Information Sciences*, 2012, Vol. 184, No. 1, 310-323.

[21] **J. H. Yang, T. J. Cao.** Provably secure three-party password authenticated key exchange protocol in the standard model. *Journal of Systems and Software*, 2012, Vol. 85, No. 2, 340-350.

[22] **H. S. Kim, J. Y. Choi.** Enhanced password-based simple three-party key exchange protocol. *Computers & Electrical Engineering*, 2009, Vol. 35, No. 1, 107-114.

[23] **J. Nam, J. Paik, H. K. Kang, U. M. Kim, D. Won.** An off-line dictionary attack on a simple three-party key exchange protocol, *IEEE Communications Letters*, 2009, Vol. 13, No. 3, 205-207.

[24] **J. Nam, J. Paik, D. Won.** A security weakness in Abdalla et al.'s generic construction of a group key exchange protocol. *Information Sciences*, 2011, Vol. 181, No. 1, 234-238.

[25] **D. C. Lou, H. F. Huang.** Efficient three-party password-based key exchange scheme. *International Journal of Communication Systems*, 2010, Vol 24, No. 4, 504-512.

[26] **H. F. Huang.** A simple three-party password-based key exchange protocol. *International Journal of Communication Systems*, 2009, Vol. 22, No. 7, 857-862.

[27] **E. J. Yoon, K. Y. Yoo.** Cryptanalysis of a simple three-party password-based key exchange protocol. *International Journal of Communication Systems*, 2011, Vol. 24, No. 4, 532-542.

[28] **S. Wu, K. Chen, and Y. Zhu.** Enhancements of A Three-Party Password-Based Authenticated Key Exchange Protocol. *International Arab Journal of Information Technology*, 2013, Vol. 10, No. 3, 215.

[29] **T. F. Lee, T. Hwang.** Simple password-based threeparty authenticated key exchange without server public keys. *Information Sciences,* 2010, Vol. 180, No. 9, 1702-1714.

[30] **T. Y. Chang, M. S. Hwang, W. P. Yang.** A communication-efficient three-party password authenticated key exchange protocol. *Information Sciences*, 2011, Vol. 181, No. 1, 217-226.

[31] **S. Wu, Q. Pu, S. Wang, D. He.** Cryptanalysis of a communication-efficient three-party password authenticated key exchange protocol. *Information Sciences*, 2012, Vol. 215, No. 1, 83-96.

[32] **R. Tso.** Security analysis and improvements of a communication-efficient three-party password authenticated key exchange protocol. *The Journal of Supercomputing*, 2013, doi:10.1007/s11227-013-0917-8.

[33] **H. Chien.** Secure verifier-based three-party key exchange in the random oracle model. Journal of *Information Science and Engineering*, 2011, Vol. 27 No. 4, 1487-1501.

[34] **Q. Pu, J. Wang, S. Wu, J. Fu.** Secure verifierbased three-party password-authenticated key exchange. *Peer-to-Peer Networking and Applications*, 2013, Vol. 6, No. 1, 15-25.

[35] **S. Tallapally.** Security Enhancement on Simple Three-Party PAKE Protocol. *Information Technology and Control*, 2012, Vol. 41, No. 1, 15-22.

[36] **M. S. Farash, M. A. Attari.** An efficient and provably secure three-party password-based authenticated key exchange protocol based on Chebyshev chaotic maps. *Nonlinear Dynamics*. 2014, doi:10.1007/s11071-014-1304-6.

Received March 2013.