ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2015, T. 44, Nr. 4

A Three-Party Password-based Authenticated Key Exchange Protocol for Wireless Communications

Yanrong Lu[1,2], Lixiang Li[1,2], Haipeng Peng[1,2], Yixian Yang[1,2]

[1] Information Security Center, State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, China
[2] National Engineering Laboratory for Disaster Backup and Recovery,
Beijing University of Posts and Telecommunications,China
e-mail: li_lixiang2006@163.com

crossref http://dx.doi.org/10.5755/j01.itc.44.4.9729

Abstract. A three-party password-based authenticated key exchange (3PAKE) protocol is an important cryptographic primitive which allows two entities to establish a session key with the help of a trusted server through an insecure channel. Recently, Farash and Attari (Information Technology and Control 43(2), 143-150, 2014) presented an improved 3PAKE protocol to erase the security flaws found in Tallapally's 3PAKE protocol (Information Technology and Control 41(1), 15-22, 2012). They claimed that their improved protocol could withstand many security attacks. However, we identified that Farash and Attari's protocol was still sensitive to the off-line password guessing attack which directly resulted in defencelessness to the impersonation attack. In order to cope with the loopholes of Farash and Attari's protocol, we proposed a modified 3PAKE protocol without using smart cards for wireless communications. We demonstrate that the proposed protocol can mitigate all the problems of the protocol of Farash and Attari and possess more security properties. In addition, we make a comparison among the proposed protocol and the other related protocols regarding the performance and security properties.

Keywords: three-party; password-based; authenticated key exchange; wireless communications.

1. Introduction

With the rapid development of wireless communication, various portable devices (mobile phone, Laptop, USB thumb drives and PDAs.) have dramatically increased to provide a more convenient life to people. People can roam freely and use mobile services almost everywhere. However, open access to wireless services for wireless environment has raised a number of security concerns. Authenticated key exchange (AKE) protocols were proposed to authenticate the identities of entities involved in the open communication. Password-based authenticated key exchange (PAKE) protocol is a type of AKE protocols. It is where two or more parties, based only on their knowledge of a password, establish a session key by messages exchange, thus preventing an unauthorized entity from participating in the protocol. Hitherto, PAKE is widely applied because of its simplicity, convenience, adaptability, mobility, and less hardware requirement [1].

Bellovin and Merritt [2] proposed the first two-party authenticated key exchange (2PAKE) protocol which is employed to establish a session key between two communication parties. After that, numerous 2PAKE protocols were presented for different communication environments [3-12]. However, 2PAKE protocols cannot be applied in large-scale peer-to-peer architecture since each user must store a different password for each partner it communicates with which may strain the storage capacity of the users. Subsequently, researchers provided three-party authenticated key exchange (3PAKE) to conquer this problem effectively [13-19]. In 2009, Huang [20] proposed a 3PAKE protocol without using smart cards in the hope that the proposed scheme was secure against various attacks. Unfortunately, Tallapaly [21] pointed out that the protocol proposed by Huang was prone to suffer from undetectable online password guessing attack. In order to thwart security attack found in Huang, Tallapaly developed an enhanced 3PAKE protocol which requires only four message transmission rounds. Nevertheless, Farash and Attari [22] observed that Tallapally's protocol was still vulnerable to undetectable online password guessing and off-line password guessing attacks. Farash and Attari then proposed their modified protocol based on Tallapally's protocol and claimed that their protocol was immune to many kinds of security attacks.

In this paper, we focus on the protocol provided by Farash and Attari. We found that the protocol was still insecure against off-line password guessing attack which directly resulted in defencelessness to the impersonation attack. In order to cope with the loopholes of Farash and Attari's protocol, we proposed a modified 3PAKE protocol without using smart cards for wireless communications. We demonstrate that the proposed protocol can mitigate all the problems of the scheme of Farash and Attari's protocol and possess more security properties. In addition, we make a comparison between the proposed protocol and the other related protocols regarding the performance and security properties.

The remainder of this paper is organized as follows. Section 2 and Section 3 review and analyze Farash and Attari's protocol. Section 4 presents our proposed protocol. Section 5 analyzes the proposed protocol. The performance and security properties comparison among the proposed protocol and other related schemes are shown in Section 6. Section 7 is a brief conclusion.

2. Review of Farash and Attari's protocol

In this section, we briefly review Farash and Attari's 3PAKE protocol. First, in Table 1, we introduce some notations used in this paper.

Table 1. Notations

Notation	Description
A, B	Users
S	Server
pw_n	The password of n
p, q	Large prime numbers, where $q = 2p - 1$
g	A generator of G
Z_q	The ring of integers modulo$q$
Z_q^*	The multiplicative group of Z_q
$F_S(\cdot)$	A trapdoor function
$h(\cdot)$	A one-way hash function
$\oplus$	Exclusive-or operation
$\|$	Concatenation operation

1) A randomly selects x , $r_a \in Z_p^*$, and computes $\mathrm{R}_a = g^x + h(pw_a, A, B) \bmod q$ .Then, he sends the message $\{A, B, R_a\}$ to S . Similarly, B also generates two random numbers y , $r_b \in Z_p^*$, and computes $\mathrm{R}_b = g^y + h(pw_b, A, B) \bmod q$. B sends $\{A, B, R_b\}$ to S.
2) Upon receiving the messages, S first computes $R_a^{'} = R_a \oplus h(A \| pw_a \| B) \bmod q$, and $R_b^{'} = R_b \oplus h(A \| pw_b \| B) \bmod q$. After that, S selects a random number $z \in Z_p^*$ and computes, $K_{sa} = (R_a^{'})^z \bmod q$, $K_{sb} = (R_b^{'})^z \bmod q$, $Z_a = h(pw_a \| A \| K_{sa} \| B \| 0 \| S)$, $T_s = g^z \bmod q$, $Z_b = h(pw_b \| 0 \| K_{sb} \| B \| A \| S)$. Finally, S sends back the message $\{Z_a, T_s\}$and $\{Z_b, T_s\}$ to A and$B$, respectively.
3) After receiving the message, A computes, $K_{as} = (T_s)^x \bmod q$ and checks whether $Z_a^{'} \overset{?}{=} Z_a$. If holds, A computes $V_a = h(A \| B \| K_{as} \| pw_a \| 0 \| T_s \| S)$ and sends V_a to S . Simultaneously, once receiving the message, B also computes $K_{bs} = (N_s)^y \bmod q$ and verifies whether $Z_b \overset{?}{=} h(pw_b \| A \| K_{sb} \| B \| 0)$. If it is correct, S is authenticated. Then, B computes $V_b = h(A \| B \| K_{bs} \| pw_b \| 0 \| T_s \| S)$ and sends V_b to S.
4) When receiving the messages, S checks whether $V_a \overset{?}{=} h(A \| B \| K_{as} \| pw_a \| 0 \| T_s \| S)$ and $V_b \overset{?}{=} h(A \| B \| K_{bs} \| pw_b \| 0 \| T_s \| S)$. If hold, S computes $X_a = K_{sb} + h(1 \| A \| B \| S \| K_{sa} \| pw_a)$ and $X_b = K_{sa} + h(1 \| A \| B \| S \| K_{sb} \| pw_b)$. Then, S sends X_a and X_b to A and B, respectively.
5) When receiving the message, A computes $K_{sb} = \mathrm{X}_a - h(A \| B \| K_{as} \| pw_a \| S \| 1)$, $K_{ab} = (K_{sb})^x \bmod q$, $S_a = h(K_{ab} \| A)$ and sends S_a to B . Simultaneously, B computes $K_{sa} = \mathrm{X}_b - h(A \| B \| K_{bs} \| pw_b \| S \| 1)$, $K_{ba} = (K_{sa})^y \bmod q$, $S_b = h(K_{ba} \| B)$ and sends S_b to A .
6) Finally, A and B check the correctness of S_b and S_a and respectively compute the session key $sk = h(A \| B \| S \| K_{as} \| K_{ab} \| K_{bs})$.

3. Security analysis of Farash and Attari's protocol

In this section, we analyze the protocol proposed by Farash and Attari. The following attacks are based on the assumption that a malicious adversary has totally supervised the communication channel and has the capacity to intercept, insert, delete, refresh or update any information delivered in the public channel [23].

3.1. Off-line password guessing attack

Assume$A$ is a malicious user. He can obtain the password of the real user A by performing the following process.

1) A guesses a password $pw_a^{'}$ and computes $R_a = g^x + h(pw_a^{'}, A, B)$, where x is a random

number of A. Finally, A submits $\{A,B,R_a\}$ and $\{B,A,R_b\}$ to S.

2) When receiving the messages, S computes $R_a' = R_a - h(pw_a, A, B)$, $R_b' = R_b - h(pw_b, A, B)$. After that, S generates a random number $z \in Z_q^*$ and computes the values $T_s = g^z$, $K_{sa} = (R_a')^z$, $K_{sb} = (R_b')^z$, $Z_a = h(0, A, B, S, pw_a, K_{sa})$ and $Z_b = h(0, A, B, S, pw_b, K_{sb})$. Then, S sends back $\{T_s, Z_a\}$and $\{T_s, Z_b\}$ to A and B, respectively.
3) When A receives the message, he computes $K_{as} = (T_s)^x$ at first. A then checks whether $h(0, A, B, S, pw_a', K_{as}) \overset{?}{=} Z_a$. If the equation is true, A has guessed the correct password; Otherwise, he continues to guess a candidate password and repeat 1)-3) until he succeeds.

3.2. Impersonation attack

As described in the previous subsection, when an adversary got the password of one of the entities, he could easily impersonate as the legal user to cheat the server. The details will follow.

1) Assume A's password is leaked. The adversary A generates a random number x' , computes $R_a = g^{x'} + h(pw_a, A, B)$, and sends $\{A, B, R_a\}$ and $\{B, A, R_b\}$ to S.
2) After receiving the message, S performs the original protocol in their scheme without detecting A who is actually a disguiser. Finally, S respectively delivers $\{T_s, Z_a\}$ and $\{T_s, Z_b\}$ to A and B.
3) When receiving the message, the adversary first checks the correctness of Z_a . If it holds, he computes $V_a = h(A, B, S, pw_a, K_{as}, T_s)$, and sends V_a and V_b to S.
4) After receiving the message, S surely verifies the correctness of V_a since all the values indeed come from A . Then, S computes $X_a = K_{sb} + h(1, A, B, K_{sa}, S, pw_a)$, $X_b = K_{sa} + h(1, A, B, K_{sb}, S, pw_b)$ and sends X_a and X_b to A and B, respectively.
5) When receiving the message, A computes $K_{sb}' = h(1, A, B, S, pw_a, K_{as})$, $S_a = h(K_{ab}, A)$, $K_{ab} = (K_{sb}')^x = g^{xyz}$, $S_b = h(K_{ab}, B)$ and the session key $sk = h(A, B, S, K_{as}, K_{sb}, K_{ab})$. That is, A successfully has been authenticated by the server.

4. The proposed protocol

In this section, we introduce our enhanced protocol which inherits the merits and remedies the weaknesses of Farash and Attari's protocol. The proposed protocol is shown in Fig. 1.

1) A chooses two random numbers x, $r_a \in Z_p^*$ and computes $H_a = h(pw_a) \oplus r_s$, $V_a = h(pw_a \| r_a \| A)$, and $R_a = g^x \oplus V_a \bmod q$ .A sends $\{H_a, V_a, R_a, A\}$ to S. Similarly, B also generates two random numbers y , $r_b \in Z_p^*$ and calculates the values $H_b = h(pw_b) \oplus r_b$, $V_b = h(pw_b \| r_b \| B)$ and $R_b = g^y \oplus V_b \bmod q$. B sends $\{H_b, V_b, R_b, B\}$ to S.
2) When receiving the messages, S first derives the two random numbers r_a and r_b using the known passwords pw_a and pw_b . Then, S verifies whether $h(A \| pw_a \| r_a)$ and $h(B \| pw_b \| r_b)$ are equal to the received V_a and V_b, respectively. If hold, S computes $R_a' = R_a \oplus h(A \| pw_a \| r_a)$, and $R_b' = R_b \oplus h(A \| pw_b \| r_b)$. After that, S selects a random number $z \in Z_p^*$ and computes $N_s = g^z \bmod q$, $K_{sa} = (R_a')^z \bmod q$, $K_{sb} = (R_b')^z \bmod q$, $T_a = h(pw_a \| r_a \| K_{sa} \| A)$, $T_b = h(pw_b \| r_b \| K_{sb} \| B)$. Finally, S sends back the messages $\{T_a, B, N_s\}$and $\{T_b, N_s, A\}$ to A and B, respectively.
3) Upon receiving the message, A computes $K_{as} = (N_s)^x \bmod q$ and verifies whether $T_a \overset{?}{=} h(pw_a \| r_a \| K_{as} \| A)$. If the equation is true, A computes $W_a = h(A \| B \| K_{as} \| pw_a \| r_a)$ and sends $\{A, W_a\}$ to S. Simultaneously, when receiving the message, B also computes $K_{bs} = (N_s)^y \bmod q$ and checks whether $T_b \overset{?}{=} h(pw_b \| r_b \| K_{bs} \| B)$. If the equation is true, B computes $W_b = h(A \| B \| K_{bs} \| pw_b \| r_b)$ and sends $\{B, W_b\}$ to S.
4) Once receiving the messages, S checks whether $W_a \overset{?}{=} h(A \| B \| K_{as} \| pw_a \| r_a)$ and $W_b \overset{?}{=} h(A \| B \| K_{bs} \| pw_b \| r_b)$. If hold, S computes $U_b = K_{sa} \oplus h(A \| B \| K_{sb} \| pw_b \| r_b)$. Finally, S sends U_a and respectively U_b to A and B.
5) After receiving the message, A computes $K_{sb} = U_a \oplus h(A \| B \| K_{sa} \| pw_a \| r_a)$, $sk = (K_{sb})^x \bmod q$, and $A_a = h(sk \| A \| B)$.

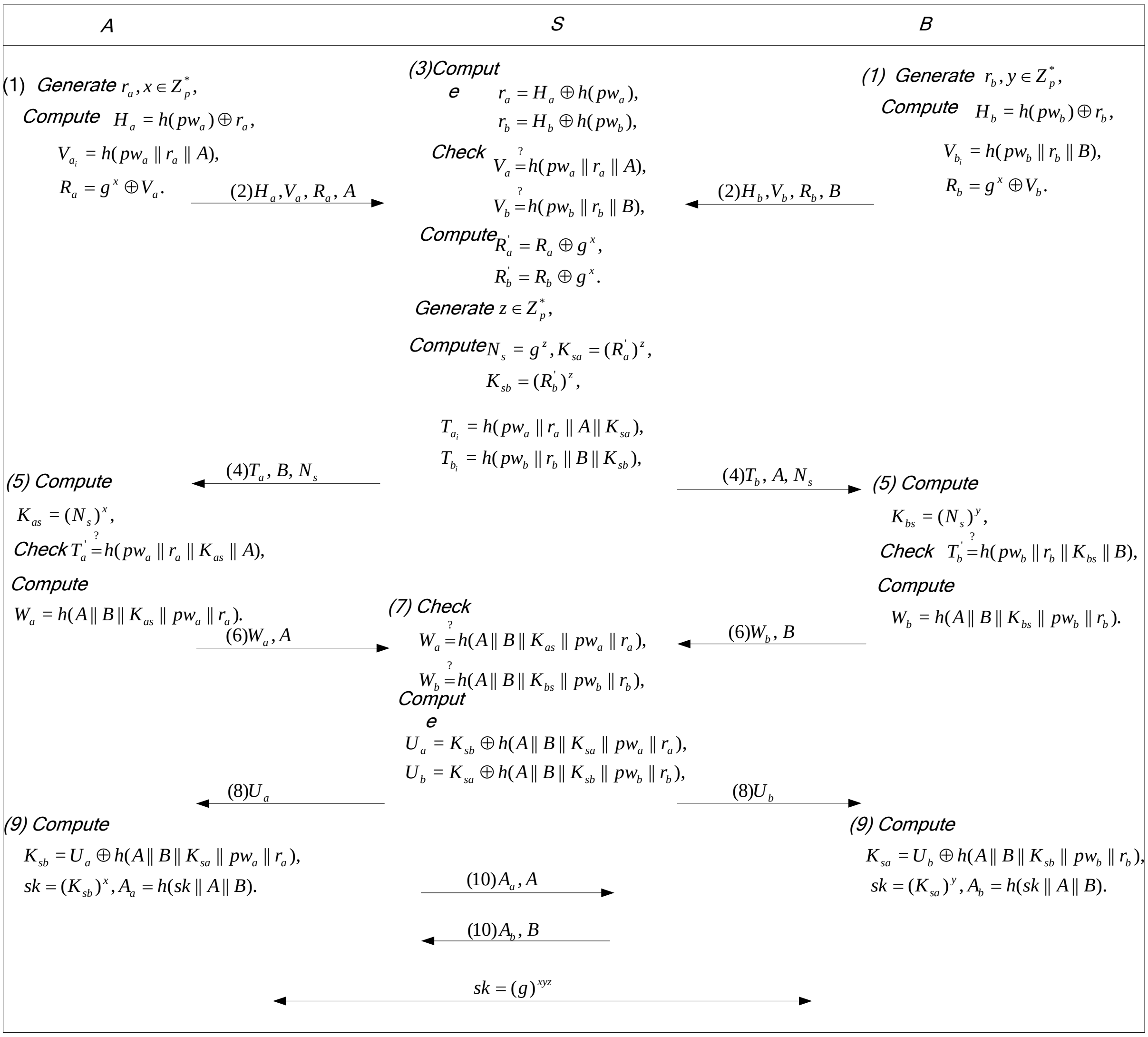
A
S
B
(1) Generate $r_a, x \in Z_p^*$,
Compute $H_a = h(pw_a) \oplus r_a$,
$V_{a_i} = h(pw_a \| r_a \| A)$,
$R_a = g^x \oplus V_a$.
$(2) H_a, V_a, R_a, A$
(3)Compute $r_a = H_a \oplus h(pw_a)$,
$r_b = H_b \oplus h(pw_b)$,
Check $V_a \overset{?}{=} h(pw_a \| r_a \| A)$,
$V_b \overset{?}{=} h(pw_b \| r_b \| B)$,
Compute $R_a^{'} = R_a \oplus g^x$,
$R_b^{'} = R_b \oplus g^x$.
Generate $z \in Z_p^*$,
Compute $N_s = g^z, K_{sa} = (R_a^{'})^z$,
$K_{sb} = (R_b^{'})^z$,
$T_{a_i} = h(pw_a \| r_a \| A \| K_{sa})$,
$T_{b_i} = h(pw_b \| r_b \| B \| K_{sb})$,
(1) Generate $r_b, y \in Z_p^*$,
Compute $H_b = h(pw_b) \oplus r_b$,
$V_{b_i} = h(pw_b \| r_b \| B)$,
$R_b = g^x \oplus V_b$.
$(2) H_b, V_b, R_b, B$
$(4) T_a, B, N_s$
$(4) T_b, A, N_s$
(5) Compute
$K_{as} = (N_s)^x$,
Check $T_a^{'} \overset{?}{=} h(pw_a \| r_a \| K_{as} \| A)$,
Compute
$W_a = h(A \| B \| K_{as} \| pw_a \| r_a)$.
(5) Compute
$K_{bs} = (N_s)^y$,
Check $T_b^{'} \overset{?}{=} h(pw_b \| r_b \| K_{bs} \| B)$,
Compute
$W_b = h(A \| B \| K_{bs} \| pw_b \| r_b)$.
$(6) W_a, A$
$(6) W_b, B$
(7) Check
$W_a \overset{?}{=} h(A \| B \| K_{as} \| pw_a \| r_a)$,
$W_b \overset{?}{=} h(A \| B \| K_{bs} \| pw_b \| r_b)$,
Compute
$U_a = K_{sb} \oplus h(A \| B \| K_{sa} \| pw_a \| r_a)$,
$U_b = K_{sa} \oplus h(A \| B \| K_{sb} \| pw_b \| r_b)$,
$(8) U_a$
$(8) U_b$
(9) Compute
$K_{sb} = U_a \oplus h(A \| B \| K_{sa} \| pw_a \| r_a)$,
$sk = (K_{sb})^x, A_a = h(sk \| A \| B)$.
(9) Compute
$K_{sa} = U_b \oplus h(A \| B \| K_{sb} \| pw_b \| r_b)$,
$sk = (K_{sa})^y, A_b = h(sk \| A \| B)$.
$(10) A_a, A$
$(10) A_b, B$
$sk = (g)^{xyz}$

Figure 1. The proposed protocol

Then, A sends A_a to B. Upon receiving the message, B retrieves K_{sa} by computing $U_b \oplus h(A \| B \| K_{sb} \| pw_b \| r_b)$ and then computes $sk = (K_{sa})^y, A_b = h(sk \| A \| B)$.

Finally, B sends A_a to A.

6) After receiving the message, A and B verify the correctness of A_b and respectively A_a. If they respectively hold, A and B successfully agree on the session key $sk = (g)^{xyz}$ with the help of the server.

5. Security analysis

This section will present a cryptanalysis of the proposed protocol. The following attacks are based on the assumption that a malicious adversary has totally supervised the communication channel and has the capacity to intercept, insert, delete, refresh or update any information delivered in the public channel [23].

5.1. Off-line password guessing attack

Without loss of generality, we assume the adversary has intercepted the message $\{H_a, V_a, R_a, A, T_a, B, N_s, W_a, U_a\}$ transmitted from A to S. However, the adversary cannot guess correctly of the password even if he knows the transmitted message. In the proposed protocol, A's password pw_a is mingled with the random number r_a, which is needed if the adversary intends to verify a guessed password. This random number cannot be obtained without knowledge of the password. Therefore, the proposed protocol is secure against the off-line password guessing attack.

5.2. On-line password guessing attack

Without loss of generality, an adversary may eavesdrop $\{H_a, V_a, R_a, A, T_a, B, N_s, W_a, U_a\}$ and plan to pretend to be a legal user A. But the adversary cannot send a new valid message $\{H_a, V_a, R_a, A, T_a, B, N_s, W_a, U_a\}$ to the trusted server unless he has guessed the correct password. And all the values $\{H_a, V_a, R_a, A, T_a, B, N_s, W_a, U_a\}$ are hidden from the one-way hash functions, the adversary cannot get them because of the security of hash function. Therefore, our protocol can resist the un- detectable online password guessing attack.

5.3. Mutual authentication

In the proposed protocol, S authenticates A by computing V_a and W_a. Furthermore, S authenticates B by computing V_b and W_b. S is verified through the correctness of T_a and T_b. Therefore, mutual authentication is achieved in the proposed protocol.

5.4. Impersonation attack

Assume that the adversary wants to impersonate A to cheat S via intercepting the messages transmitted in the public channel. However, he cannot succeed until he can get the password of A. As described in the previous subsection, the adversary is not possible to launch an off-line password guessing attack. That is, the proposed protocol is immune to the impersonation attack.

5.5. The session key perfect forward secrecy

Suppose A has compromised all the passwords of the communication entities. In order to get the session key $sk = (g)^{xyz}$, he needs to know the random numbers $\{x, y, z\}$. Nevertheless, he will face the DLP if he tries to get $\{x, y, z\}$ from $\{K_{as}, K_{bs}, N_s\}$. Therefore, the proposed protocol can ensure the perfect forward secrecy.

5.6. Replay attack

In our protocol, we use numerous random numbers for each session to resist replay attack. If the adversary intends to resend the old messages, the corresponding receiver will immediately detect the attack from the sender. Therefore, it seems to be impossible to perform the replay attack to our protocol.

6. Performance and security properties comparison

In this section, we compare the performance and security properties of our protocol with other related protocols [20-22] in Fig. 2 and Table 2.

In Fig. 2, our proposed protocol has a similar efficiency with Farash and Attari's protocol even if the costs of our protocol are slightly higher than Tallapaly and Huang's protocol. However, Tallapaly and Huang's protocol is vulnerable to the on-line password guessing and off-line password guessing attacks and Huang's protocol cannot provide mutual authentication. As shown in Table 2, none of the other protocols can resist the off-line password guessing attack. Therefore, our protocol is efficient and secure compared with other related protocols.

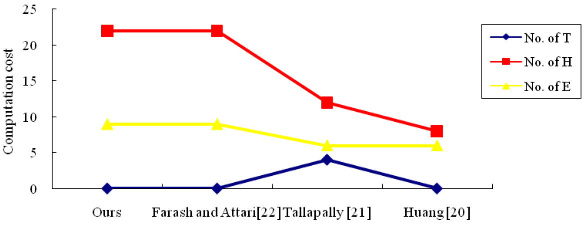

Figure 2. Performance comparison

No.of T: the number of trapdoor functions;
No.of H: the number of hashing operations;
No.of E: the number of exponentiation operations.

Table 2. Comparison of security properties

	The proposed protocol	Farash and Attari [22]	Tallapally [21]	Huang [20]
N1	Yes	Yes	-	No
N2	Yes	Yes	Yes	Yes
N3	Yes	No	No	No
N4	Yes	Yes	No	No
N5	Yes	No	-	-
N6	Yes	Yes	Yes	Yes

N1: Provide mutual authentication;
N2: Provide the session key perfect forward secrecy;
N3: Withstanding off-line password guessing attack;
N4: Withstanding on-line password guessing attack;
N5: Withstanding impersonation attack;
N6: Withstanding replay attack.

7. Conclusion

This paper mainly discussed the protocol proposed by Farash and Attari. We found that their protocol was insecure against the off-line password guessing attack, thus suffering from an impersonation attack. In order to eliminate the weaknesses of Farash and Attari's protocol, we presented an enhanced three-party password-based authenticated key exchange protocol for wireless communications. We demonstrated that the proposed protocol could withstand various kinds of attacks including attacks found in Farash and Attari's protocol. In addition, we compared the proposed protocol with other related protocols regarding the performance and security features. The results showed that the proposed protocol was more secure than Farash and Attari's protocol without increasing the computation cost. All in all, the proposed protocol is suitable for the wireless environments.

Acknowledgments

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper. This paper is supported by the National Natural Science Foundation of China (Grant Nos. 61472045, 61573067), the BUPT Excellent Ph.D. Students Foundation (Grant No. CX2015310), the Asia Foresight Program under NSFC Grant (Grant No. 61411146001) and the Beijing Natural Science Foundation (Grant No. 4142016).

References

[1] **J. Yang, T. 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.

[2] **S. M. Bellovin, M. Merritt.** Encrypted key exchange: password-based protocols secure against dictionary attacks. *In: Proceedings of the 1992 IEEE symposium on research in security and privacy,* 1992, pp. 72-84.

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

[4] **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, 82-85.

[5] **H. Sun, Q. Wen, H. Zhang, Z. Jin.** A strongly secure pairing-free certificate less authenticated key agreement protocol for low-power devices. *Information Technology and Control,* 2013, Vol. 42, No. 2, 113-123.

[6] **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.

[7] **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, pp. 53-59.

[8] **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, pp. 150-158.

[9] **Y. Lu, L. Li, Y. Yang**. Robust and efficient authentication scheme for session initiation protocol. *Mathematical Problems in Engineering,* 2015, Article ID 894549.

[10] **Y. Lu, L. Li, H. Peng, X. Yang, Y. Yang.** A light-weight ID based authentication and key agreement protocol for multi-server architecture. *International Journal of Distributed Sensor Networks,* 2015, Article ID 635890.

[11] **Y. Lu, L. Li, H. Peng, Y. Yang.** An enhanced biometric-based authentication scheme for telecare medicine information systems using elliptic curve cryptosystem. *Journal of Medical Systems*. 2015, Vol.39, No.3, 1-8.

[12] **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.

[13] **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**.**

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

[15] **D. Zhao, H. Peng, L. Li, Y. Yang.** A secure and effective anonymous authentication scheme for roaming service in global mobility networks. *Wireless Personal Communication*s, 2014, Vol. 78, No. 1, 247-269.

[16] **T.-F. Lee.** Verifier-based three-party authentication schemes using extended chaotic maps for data exchange in telecare medicine information systems. *Computer Methods and Programs in Biomedicine,* 2014, Vol. 117, No. 3, 464-472.

[17] **J. Zhao, D. Gu.** Provably secure three-party password-based authenticated key exchange protocol. *Information Sciences*, 2012, Vol. 184, 310-323.

[18] **H. Yang, Y. Zhang, Y. Zhou, X. Fu, H. Liu, A. Vasilakos.** Provably secure three-party authenticated key agreement protocol using smart cards. *Computer Networks*, 2014, Vol. 58, 29-38.

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

[20] **S. Tallapally.** Security enhancement on simple three-party PAKE protocol. *Information Technology and Control,* 2012, Vol. 41, No. 1, 15-22.

[21] **M. S. Farash, M. A. Attari.** An enhanced and secure thee-party password-based authenticated key exchange protocol without using server's public-keys and symmetric crypto-systems. *Information Technology and Control,* 2014, Vol. 43, No. 2, 143-150.

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

Received January 2015.