Attacks and Solutions of an Authenticated Key Agreement Protocol Based on NFC for Mobile Payment

Chienming Chen[1], Weicheng Fang[1], Kinghang Wang[2*], Tsuyang Wu[3, 4]

[1] Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China.
[2] Hong Kong University of Science and Technology, Hong Kong, China.
[3] Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou, China
[4] National Demonstration Center for Experimental Electronic Information and Electrical Technology Education, Fujian University of Technology, Fuzhou, China

* Corresponding author. Tel.: 852-2358 8834; email: kevinw@cse.ust.hk
Manuscript submitted May 24, 2017; accepted July 20, 2017.
doi: 10.177606/ijcce.6.3.173-180

Abstract: The popularization of the word "Fin-tech" thanks to many non-technical individuals being amazed by the unconventional way of payments, such as mobile payment over NFC. Undoubtedly speaking security/privacy is considered as the most important factor when a new Fin-tech is introduced; at least psychologically, it is. Recently Seo *et al.* presented an authenticated key agreement protocol for mobile payment over NFC. The protocol intended to provide secure pairing over untrusted devices with client's anonymity and forward secrecy. Unfortunately, in this paper we found that their protocol is indeed very insecure when an attacker has different levels of network controls. We presented the man-in-the-middle attacks and the replay attacks against this protocol. Under these attacks the attackers can successfully impersonate an anonymous client or can tap the communication between two legitimate clients without being detected by anyone. Then we suggested some improvements, with adequate analysis, to avoid these problems.

Key words: Authenticated key agreement, near field communication, security.

1. Introduction

Mobile-commerce (m-commerce) is defined as any electronic transaction with a mobile device and is considered the next generation e-commerce. This allows a more user friendly, more convenient, any-time-any-where model of B2C or even C2C model. However, various security problems regarding m-commerce appeared [1], [2] that discourage the use of m-commerce. The success of m-commerce is relying on how these security issues be handled carefully and not to lose any confidence from the public.

In this paper we focus at the protocol level of m-commerce security. When a transaction happens, the system performs an authenticated key agreement protocol to authenticate the mobile user and the merchant that ensures the confidentiality and integrity of the transaction. Recently, many researchers have been studying such protocols. In 2009, Yang *et al.* [3] proposed an efficient three-party authenticated key agreement protocol using elliptic curve cryptography (ECC) for m-commerce environment. However, Pu *et al.* [4] first discovered that Yang *et al.*'s protocol suffers from unknown key-share attacks. Then, Tan *et al.* [5] further pointed out that their protocol is insecure against impersonation attacks and parallel attacks, and

proposed an enhanced protocol. Later, Nose [6] pointed out the enhanced protocol is susceptible to man-in-the-middle attacks and impersonation attack. To fix the problem, He *et al.* [7] employed timestamps to design an ID-based protocol, while Islam *et al.* [8] suggested an improved protocol using hash function and ECC only with nonces.

A very recent article published on the journal by Seo *et al.* [9] discusses the issue about mobile payment over the near field communication (NFC). As highlighted in their paper, mobile payments have taken up a significant role nowadays. While NFC is gaining more and more trusts from end-users and is supported by more and more devices, it is a good channel to bootstrap a secure communication between mobile devices in mobile payments.

The mobile payment environment assumes the existence of an authentication server (AuC), representing a credit card issuer or similar authority that all mobile payment users have registered under it. Any payer and payee in a mobile payment transactions are modelled as a mobile user. The payee would be a card reader installed in a retail store (in B2C setting) or a mobile phone (in P2P setting), as long as these mobile users have both network connectivity to the AuC and NFC connectivities to the payer. A payer is modelled as another mobile user which can be a credit card with sufficient computation power or a mobile phone. Yet, the payer only requires to have NFC connectivities but not internet connectivity to the AuC.

According to various previous research [10]-[13], the security requirements of an authentication protocol needed in this setting are listed below.

1) **Mutual Authentication.** This requires participants can be convinced that the communicating partner over the network is not being impersonated or unauthorized.
2) **Forward Secrecy.** This requirement states the session key established in a session will not be known to an attack if later the long term secret is exploited to an attacker.
3) **Resistant to Password Guessing Attack.** This requires an attacker cannot guess a user's password without being detected.
4) **Resistant to Man-in-the-middle Attack.** This requires an attacker cannot eavesdrop or manipulate a session content by standing in the middle of communicating parties.
5) **Resistant to Replay Attack.** This requires an attacker cannot interfere the protocol by replaying messages.

The remaining sections are organized as follows. Section 2 reviews Seo *et al.*'s protocol. In Section 3, we demonstrate that this protocol is vulnerable to several attacks. In Section 4, we propose an improved protocol. Section 5 provides a detailed security analysis of the improved protocol. Finally, we draw a conclusion in Section 6.

2. Review of Seo *et al.*'s Protocol

In this section, we review Seo *et al.*'s Protocol. This protocol contains two phases, the registration phase and the authenticated key agreement phase.

The registration phase is involved when a payer desires to register to the server. For example, if a payer A wants to register to an AuC, A generates a random number r_A and computes APW_A where $APW_A = h(PW_A||r_A)$. Note that PW_A is A's password and $h()$ means a one-way hash function. Then, A sends $\{ID_A, APW_A\}$ to the AuC. After receiving $\{ID_A, APW_A\}$, the AuC computes $V_A = h(ID_A||x) \oplus APW_A$, where x is the secret key of the AuC and $\oplus$ denotes an exclusive-or operation. The AuC then stores $\{V_A, PU_{AuC}\}$ in a smart device (or a smart card) and issues this device to A. PU_{AuC} is the public key of the AuC. Now A also stores r_A to this smart device.

The authenticated key agreement phase is involved if two users desire to establish a common session key. This phase is summarized in the Fig. 1. The detailed steps are listed as follows.

1) A payer A starts the protocol by selecting a nonce a. He retrieves a stored secret V_A and PU_{AuC} from

his smart device (or smart card), enters the password PW_A and computes Y_A, R_A, CID_A, MAC_A as shown in the Fig. 1. He then sends $\{R_A, CID_A, MAC_A\}$ to the payee B.

2) B computes similar values as A does, then appends the message and sends $\{R_B, CID_B, MAC_B, R_A, CID_A, MAC_A\}$ to the AuC.
3) The AuC retrieves the ID of A and B from CID_A and CID_B by operating $\oplus$ operation with $(R_A)^x$ and $(R_B)^x$ respectively. The AuC then computes Y_A' and Y_B'. Now the AuC can validate MAC_A and MAC_B. If both hold, it chooses a nonce c, computes R_C, SK_{CB}, SK_{CA}, MAC_{CB}, MAC_{CA} as described in the Fig. 1, and returns the message $\{R_C, MAC_{CB}, MAC_{CA}\}$ to B.
4) After B receives the message from the AuC, he computes the session key SK_{BC}, and then utilizes SK_{BC}, Y_B, R_B and R_C to validate MAC_{CB}. After that, B computes the session key SK_{BA} used between A and B with R_A and b and sends the authentication message $\{R_B, R_C, MAC_{CA}, MAC_{BA}\}$ to A.
5) In this last step, A computes SK_{AC}, validates MAC_{CA}, computes the session key SK_{AB}, and verifies MAC_{BA}. Now A and B have established a common session key.

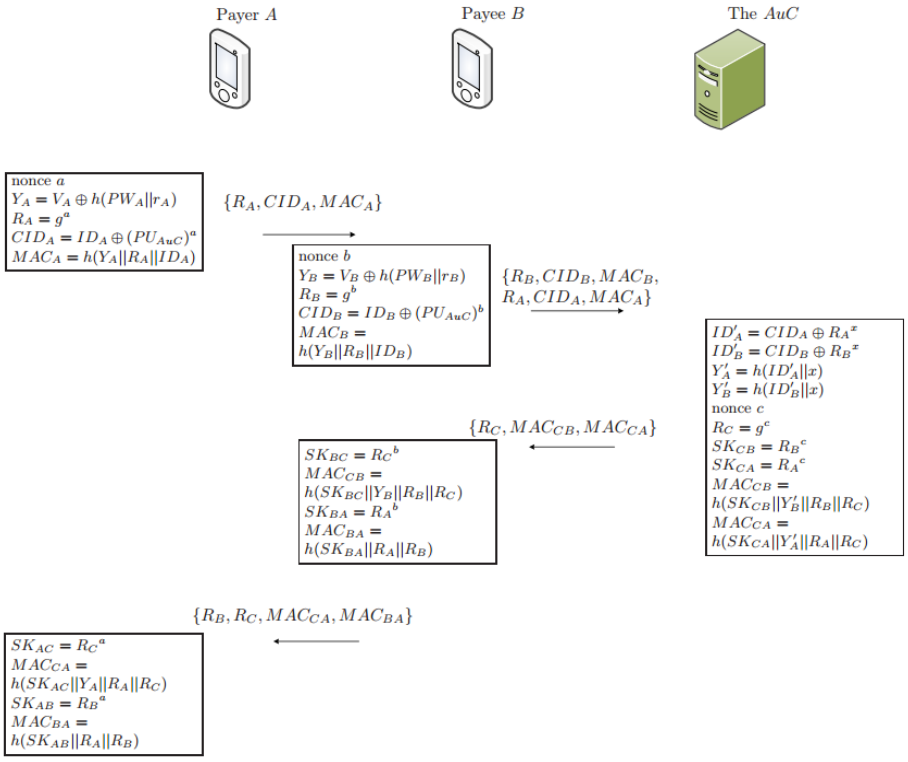

Fig. 1. Illustration of SEO *et al.*'s scheme.

3. Security Analysis

In this section, we present several attacks on the protocol.

3.1. A Man-in-the-middle Attack #1

The protocol is vulnerable to a man-in-the-middle attack. In this attack a middleman attacker E has a control over the NFC channel and has registered an account with the AuC. When the payer initiate the protocol by sending out $\{R_A, CID_A, MAC_A\}$, E replaces the whole set of message by $\{R_E, CID_E, MAC_E\}$ using E's secret and sends this to the payee B. We call this communication session be Session 1. Concurrently E initiates another session and we name this as Session 2. In Session 2 E takes B's message and sends $\{R_E, CID_E, MAC_E, R_A, CID_A, MAC_A\}$ to the AuC. Both sessions will be authenticated by the AuC while A will accept the protocol in Session 2, B will accept the protocol in Session 1. A and B will think they have established a

secure connection with each other but in fact they are connected to the middle man E in two separated sessions. This is because in the authentication message sent from the AuC does not contain the identity of A or B. Therefore A or B has no way to ensure they are communicate with each other.

3.2. A Man-in-the-middle Attack #2

In this man-in-the-middle attack, the middleman attacker E has a control over the NFC channel and the public network. When A sends the first message $\{R_A, CID_A, MAC_A\}$ to B, E replaces it by $\{R_E, CID_A, MAC_A\}$ where $R_E = g^e$ which is computed by E. B will continue the protocol since B cannot validate the message sent from A. B will output $\{R_B, CID_B, MAC_B, R_E, CID_A, MAC_A\}$ and send it to the AuC. E replaces this message with $\{R_B, CID_B, MAC_B, R_A, CID_A, MAC_A\}$. This message will be accepted by the AuC. In the next step B will also accept the message sent from the AuC and output $\{R_B, R_C, MAC_{CA}, MAC_{\mathrm{BA}}\}$ to A. E again replaces this message by $\{R_E, R_C, MAC_{CA}, MAC_{EA}\}$ such that $MAC_{EA} = h(SK_{EA}||R_A||R_E)$ and $SK_{EA} = (R_A)^e$. A will also accept this message as MAC_{CA} which does not contain any information about R_B and MAC_{EA} is indistinguishable to a normal MAC sent from B. But, the session key SK_{BA} computed by B and SK_{AB} computed by A are different and both known to E. In this attack, even the AuC would have no idea that E is ever involved in attack.

3.3. Replay Attacks and Other Attacks

This attack does not require the attacker to have control over the NFC channel. An attacker E records the messages sent by B over the public network in Step 2 and later replays it to the AuC. Since there are no further challenge and response involved, the AuC will simply accept this protocol as the message are valid and consider payer A and payee B are trying to conduct a mobile payment. This process can be further extended by mixing two separated sessions together. Let's assume a session by a payer A and a payee B were logged as Session 1 and another session by a payer C and a payee D was also logged as Session 2. The attacker E can replay part of the message from Session 1 and Session 2 to convince the AuC that payer \$A\$ is indeed proceeding a transaction with payee D, which is never happened.

This attack could lead to other attack patterns like impersonation attacks if the attacker has control over the public network. Say for example E is not a legitimate user. He initiates the protocol with a payee B with $\{R_E, random, random\}$. E replaces the message from B by $\{R_B, CID_B, MAC_B, R_A, CID_A, MAC_A\}$ where R_A, CID_A, MAC_A were recorded previously. B and the AuC will accept the protocol and B will agree a session key with E.

4. The Improved Protocol

Mutual authentication normally requires a two-way challenge and response. Without adding additional rounds to the protocol, it is not easy to proof the one with the knowledge of the discrete log of R_A and R_B has also the knowledge of password and long term key. For instance, one may consider using non-interactive zero knowledge proof (NIZK) [14], hash chains [15], timestamps [16], [17], or human-verifiable hash [18] to avoid replay attacks. However, even if the protocol is secure against replay attack, it cannot be shown the protocol is secure. As a result, we propose an improved protocol in this section. Our improvement focuses on the protection against the attacks mentioned in the Section 3.

4.1. The Proposed Protocol

In our design, there is no need to modify the registration phase. The modified authentication phase is summarized in the Fig. 2. The detailed steps are described as follows.

1) A payer A first chooses a nonce a, computes Y_A, R_A, and CID_A as usual, and then sends $\{R_A, CID_A, MAC_A, t_A\}$ to the payee B, where t_A is a timestamp, but the MAC is computed as $MAC_A = h(Y_A||R_A||ID_A||ID_B||t_A)$.
2) B follows in a way like A, where $MAC_B = h(Y_B||R_B||ID_B||t_B)$. After appending A's messages, B send them

to the AuC.

3) Upon receiving the messages, the AuC first checks the timestamps of A and B. If either do not lie in a valid time interval, the authentication session is immediately rejected. Otherwise, with the retrieved IDs, the computed Y_A' and Y_B' and the received plain text R_A, R_B, t_A, and t_B, the AuC can validate MAC_A and MAC_B. If both hold, it computes R_C, SK_{CB}, and SK_{CA} as usual. But the MACs are computed as $MAC_{CB} = h(SK_{CB}||Y_B'||R_B||R_C||R_A)$ and $MAC_{CA} = h(SK_{CA}||Y_A'||R_A||R_C||R_B)$. Next, the AuC sends $\{R_C, MAC_{CB}, MAC_{CA}\}$ to B.
4) After B receives the message from the AuC, he verifies MAC_{CB} using the message sent from A and the AuC. If the verification passes, B further computes the session key SK_{BA} and the MAC_{BA} as usual, and then sends $\{R_B, R_C, MAC_{CA}, MAC_{BA}\}$ to A.
5) After A receives the message from B, he verifies MAC_{CA} and MAC_{BA} with the message sent from B as well as the computed session key SK_{AB}. Once they are verified, the session key SK_{AB} is shared between A and B.

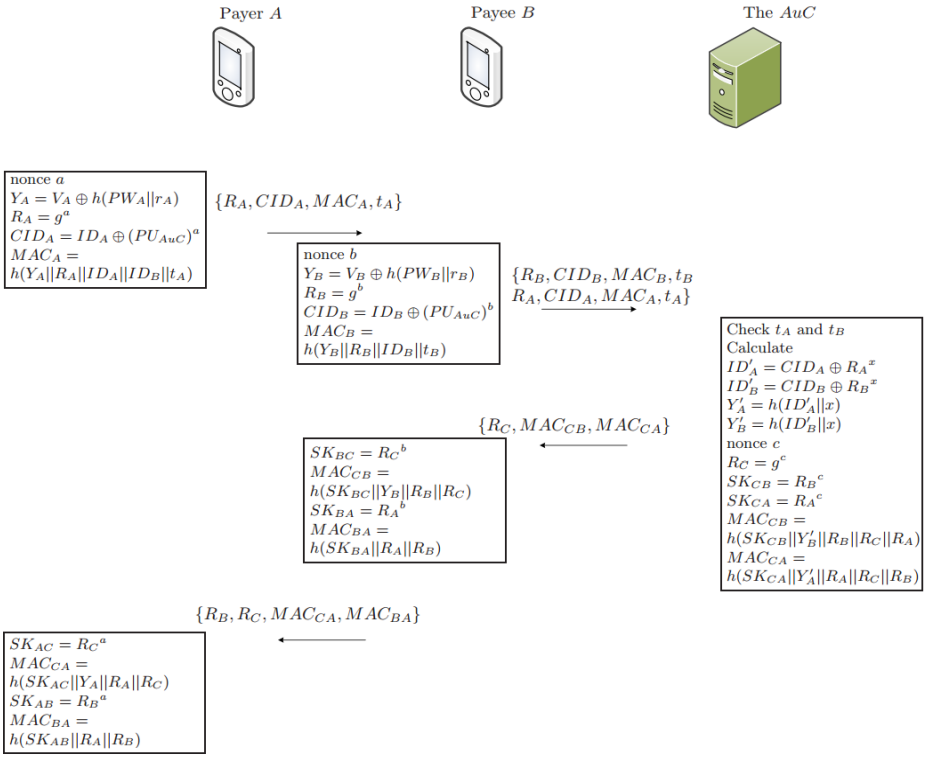

Fig. 2. Illustration of the proposed scheme.

4.2. Discussion

There are three patches applied to Seo *et al.*'s scheme.

1) In Seo *et al.*'s protocol, the authentication between payer A and payee B is direction independent. The AuC may be confused about the identity of the users since the MAC (MAC_A and MAC_B) has the same form. We add B's ID in MAC_A so that the authentication sessions and identities involved become distinguishable.
2) Both payer A and payee B should have authenticated each other through the AuC. However, under our man-in-the-middle attacks they cannot, because the authentication messages sent by the AuC are independent of identities of both A and B. For example, the message $MAC_{CB} = h(SK_{CB}||Y_B'||R_B||R_C)$ sent to B is irrelevant with A's messages. Therefore, if they simply verifies the authentication messages, neither of them can ensure that the message R_A that is sent by A (or R_B sent by B) has not been

modified. Note R_A and R_B are used to generate the session key. Once they are modified according to the man-in-the-middle attacks, A and B will agree on the same session key. Thus, it is indispensable to make sure that B should receive the true R_A while A the true R_B. Our quick fix is to include R_A in the authentication message of MAC_{CB}. Similarly, we include R_B in MAC_{CA}.

3) Finally, we include timestamps to make sure that the messages are fresh.

5. Security Analysis of Our Proposed Protocol

To demonstrate the effect of our improvement, we analyze the improved protocol in terms of resistance to several attacks.

5.1. Resistance to the Man-in-the-middle Attack #1

In this attack, when the payer \$A\$ initiates the protocol by sending out $\{R_A, CID_A, MAC_A, t_A\}$, the message will include the payee's ID, ID_B. Then an adversary E replaces the whole message by $\{R_E, CID_E, MAC_E, t_E\}$, and sends it to B. When the payee B receives E's message, he will follow the protocol honestly. This is Session 1. Next, E starts another session by sending $\{R_A, CID_A, MAC_A, t_A, R_E, CID_E, MAC_E, t_E\}$. Session 1 will proceed as usual since as a registered user, E initiates a session with B. However, the attack fails when the AuC verifies the second session. In Session 2, A wants to communicate with B, but the AuC finds that the payee's ID in MAC_A does not corresponds to the received one. Thus the man-in-the-middle attack #1 will not applied to our improved scheme.

5.2. Resistance to the Man-in-the-middle Attack #2

In this attack, an adversary E replaces A's initial message by $\{R_E, CID_A, MAC_A, t_A\}$. After that, B will simply forward the message, and send the message $\{R_E, CID_A, MAC_A, t_A, R_B, CID_B, MAC_B, t_B\}$ to the AuC. Then E changes R_E to R_A in this message. The AuC accepts this message and sends feedback to B. However, B will reject the session since $MAC_{CB} = h(SK_{CB}||Y_B'||R_B||R_C||R_A)$ does not match with $MAC_{CB} = h(SK_{CB}||Y_B'||R_B||R_C||R_E)$. Thus the adversary cannot launch such attacks.

5.3. Resistance to Replay Attacks and Other Attacks

In these attacks, adversary records messages sent by payers or payees, and then replays them to attack the network. However, the initiation messages contain timestamps and identities in their MACs. When the AuC receives the replayed message, they will be detected immediately.

5.4. Mutual Authentication

In our protocol, the AuC authenticates the payer and the payee by checking if $MAC_A = h(Y_A'||R_A||ID_A'||ID_B'||t_A)$ and $MAC_B = h(Y_B'||R_B||ID_B'||t_B)$ hold. If both hold, the AuC concludes that they own their passwords, PW_A and PW_B, and have managed to compute the issued secrets, $h(ID_A||x)$ and $h(ID_B||x)$. Then, both A and B is successfully authenticated by the AuC. As to the authentication of the AuC by the users, A checks if $MAC_{CA} = h(SK_{AC}||Y_A||R_A||R_C||R_B)$ holds, while B checks if $MAC_{CB} = h(SK_{BC}||Y_B||R_B||R_C||R_A)$ holds. If both hold, the user concludes that the AuC holds the server's secret x. Then, the AuC is authenticated by both A and B.

5.5. Forward Secrecy

In our protocol, both payer A and payee B compute the session key as $SK = (R_A)^b = (R_B)^a = g^{ab}$, where the nonces a and b are chosen by A and B respectively. As the scheme assumes that the computational Diffie-Hellman problem is intractable, it is infeasible to compute SK as long as the nonces are unknown to an adversary. Even if the server's secret key x, the users' identities ID_A and ID_B, and their passwords PW_A and PW_B are compromised, since they are not involved in the computation, it does not help the adversary to

crack the session keys.

6. Conclusion

In this paper, through our cryptanalysis of Seo *et al.*'s authenticated key agreement protocol based on NFC for mobile payment, we demonstrate that their protocol is not secure against man-in-the-middle attack and replay attack. In order to enhance the security, we propose an improved protocol that can resist these attacks. Also, we provide a security analysis of our improved protocol.

Acknowledgment

The work was supported in part by the Project NSFC (National Natural Science Foundation of China) under Grant number 61402135 and in part by Shenzhen Technical Project under Grant number JCYJ20150513151706574.

References

[1] Goth, G. (2012). Mobile security issues come to the forefront. *IEEE Internet Computing, 16(3)*, 7–9.

[2] Goyal, V., Pandey, U. S., & Batra, S. (2012). Mobile banking in India: Practices, challenges and security issues. *International Journal of Advanced Trends in Computer Science and Engineering, 1(2)*.

[3] Yang, J. H., & Chang, C. C. (2009). An efficient three-party authenticated key exchange protocol using elliptic curve cryptography for mobile-commerce environments. *Journal of Systems and Software, 82(9)*, 1497–1502.

[4] Pu, Q., Zhao, X., & Ding, J. (2009). Cryptanalysis of a three-party authenticated key exchange protocol using elliptic curve cryptography. *Proceedings of the 2009 International Conference on Research Challenges in Computer Science* (pp. 7–10). IEEE Computer Society.

[5] Tan, Z. (2010). An enhanced three-party authentication key exchange protocol using elliptic curve cryptography for mobile commerce environments. *Journal of Communications, 5(5)*, 436–443.

[6] Nose, P. (2011). Security weaknesses of authenticated key agreement protocols. *Information Processing* Letters, *111(14),* 687–696.

[7] He, D., Chen, Y., & Chen, J. (2013). An ID-based three-party authenticated key exchange protocol using elliptic curve cryptography for mobile-commerce environments. *Arabian Journal for Science and Engineering, 38(8)*, 2055–2061.

[8] Islam, S. H., Amin, R., Biswas, G. P., Farash, M. S., Li, X., & Kumari, S. (2015). An improved three party authenticated key exchange protocol using hash function and elliptic curve cryptography for mobile-commerce environments. *Journal of King Saud University - Computer and Information Sciences*.

[9] Seo, B., Lee, S. W., & Kim, H. (2016). Authenticated key agreement based on NFC for mobile payment. *International Journal of Computer and Communication Engineering, 5(1)*, 71–78.

[10] Chen, C. M., Li, C. T., Liu, S., Wu, T. Y., & Pan, J. S. (2017). A provable secure private data delegation scheme for mountaineering events in emergency system. *IEEE Access*, *5,* 3410-3422.

[11] Chen, C. M., Xu, L., Wu, T. Y., Li, C. R. (2016). On the security of a chaotic maps-based three-party authenticated key agreement protocol. *Journal of Network Intelligence*, *1(2),* 61–66.

[12] Sun, H. M., He, B. Z., Chen, C. M., Wu, T. Y., Lin, C. H., & Wang, H. (2015). A provable authenticated group key agreement protocol for mobile environment. *Information Sciences*, *321,* 224-237.

[13] Chen, Y., Chen, C. M., Pan, J. S., Wu, T. Y., & Liu, S. (2016). Security Improvement on a three party password based authenticated key exchange scheme suing chaotic maps. *Journal of Information Hiding and Multimedia Signal Processing*, *7(6),* 1365-1372.

[14] Blum, M., Feldman, P., & Micali, S. (1988). Non-interactive zero-knowledge and its applications.

Proceedings of the 20th Annual ACM Symposium on Theory of Computing (pp. 103–112). ACM.

[15] Yeh, L. Y., Tsaur, W. J., & Juang, T. Y. (2016). Cryptanalysis and efficient improvement of a robust and scalable one-way hash chain authentication protocol in vehicular communication. *Proceedings of the International Conference on Wireless Networks* (pp. 17–20). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing.

[16] Hu, X., & Zhang, Z. (2014). Cryptanalysis and enhancement of a chaotic maps-based three-party password authenticated key exchange protocol. *Nonlinear Dynamics, 78(2)*, 1293–1300.

[17] Lee, C. C., Li, C. T., & Hsu, C. W. (2013). A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps. *Nonlinear Dynamics, 73(1-2)*, 125–132.

[18] Lin, Y. H., Studer, A., Hsiao, H. C., McCune, J. M., Wang, K. H., Krohn, M., *et al*. (2009). Spate: Small-group PKI-less authenticated trust establishment. *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services* (pp. 1–14). ACM.

Chienming Chen received his Ph.D from the National Tsing Hua University, Taiwan in 2010. He is currently an associate professor of the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen, China. Dr. Chen serves as the associate editor of three international journals: Journal of Information Hiding and Multimedia Signal Processing, Data Science and Recognition, Journal of Network Intelligence. His current research interests include network security, mobile internet, wireless sensor network and cryptography. Dr. Chen had published more than 70 international journal and international conference papers on the above research fields.

Weicheng Fang is currently pursuing a master's degree in School of Computer Science and Technology, Shenzhen Graduate School, Harbin Institute of Technology. His research interests include authenticated key exchange protocols and anonymous authentication.

Kinghang Wang received his Ph.D from the National Tsing Hua University and BEng from the Chinese University of Hong Kong. He worked in the Hong Kong Institute of Technology in 2010 as a lecturer. He joined the Department of Computer Science and Engineering of the Hong Kong University of Science and Technology as a teaching staff since 2015. His research focus is cryptography, mobile security, and provable authentication.

Tsuyang Wu received the Ph.D degree in Department of Mathematics, National Changhua University of Education, Taiwan in 2010. Currently, he is an associate professor in College of Information Science and Engineering at Fujian University of Technology, China. In the past, he is an assistant professor in Innovative Information Industry Research Center at Shenzhen Graduate School, Harbin Institute of Technology, He is a member of Chinese Cryptology and Information Security Association (CCISA) and China Computer Federation (CCF). He serves as associate editor of international journals: Data Science and Recognition, Journal of Network Intelligence. His research interests include cryptography and network security.