Reviews and analyses the privacy-protection system for multi-server

Min-Shiang Hwang[1,2], Eko Fajar Cahyadi[1,3], Shu-Fen Chiou[4], and Cheng-Ying Yang[5,*]

[1]Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan 41354 (Email: mshwang@asia.edu.tw)
[2]Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan 40402
[3]Department of Telecommunication Engineering, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia
[4]Department of Information Management, National Taichung University of Science and Technology, Taiwan
[5]Department of Computer Science, University of Taipei, Taipei, Taiwan
(*Email: cyang@utaipei.edu.tw)

Abstract. Zhu proposed a provable privacy-protection system which could achieve two kinds of privacy protection and switch between them optionally by users. He claimed his scheme could achieve anonymous, privacy-protection, and practical multi-server architecture. His scheme is based on chaotic maps. He thus claimed his scheme can not only own high efficiency and unique functionality but is also robust to various attacks and achieves perfect forward secrecy. However, in this article, we will review Zhu's scheme, applied in a multi-server environment and adopted in the provable privacy protection system. In addition to the review, we analyzed its protocol and elaborated its weakness of resulting insecurity.

1. Introduction

The key agreement scheme is a method for establishing a common secret key to be shared between two parties over the internet. Then, they could use the secret key to be the secret key cryptosystem to establish a secure communication [1-7]. Many schemes had been proposed for key agreement scheme. One of these schemes was developed for multi-servers [8-13]. One of these schemes was developed for anonymous [14-16]. One of these schemes was developed for Chaotic maps [17-19]. One of these schemes was developed for authenticated [20-22]. One of these schemes was developed for group key [23-26].

In 2015, Zhu [9] proposed a provable privacy protection system for a multi-server environment, it constructed in a multi-server architecture. In addition, it combined a provable privacy protection system which can achieve users' identities in protection. It gave up putting a user ID in a scheme to replace with an anonymous ID. The server and the RC knew that it was legal or paying members. However, we found it had a serious problem to make it unsafe. Because of not preventing Bergamo et al.'s attack, it would explore a user's long-term key in the transmission process. Furthermore, Zhu's protocol was inefficient. When users and servers wanted to authenticate the identity in the system, they should send a verified message to the registration center (RC). If the number of users increases

linearly, it would bring that RC must waste a lot of computation resource to deal with authentication. It is not a good way to solve this problem.

2. Review of Zhu's Privacy-Protection System for Multi-Server

There are six phases in Zhu's system [9]: System initialization phase, server registration phase, user registration phase, anonymous authenticated key agreement phase, hiding identity authenticated key agreement phase and password changing phase.

2.1. System initialization phase

In Zhu's Multi-server, RC firstly generates a private key *k* to compute $T_k(x)$ and release it as public information. Public information stores $\{ID_{RC}, H, E_k(.)/D_k(.), (x, T_k(x))\}$ which is offered to entities for conducting the below phases.

2.2. Server registration phase

Step 1. Server $S_i \rightarrow$ RC: $\{ID_{Si}\}$

The server S_i chooses his identity ID_{Si} which is then submitted to RC via a secure channel.

Step 2. RC $\rightarrow$ Server S_i: $\{R\}$

When RC receives ID_{Si} from the server S_i, RC publishes ID_{Si} as public information and computes $R=H(ID_{Si}||k)$, where *k* is the private key for RC. Then, RC sends back R via a secure channel.

2.3. User registration phase

Step 1. User $U_j \rightarrow$ RC: $\{ID_U, H(r_a||PW)\}$

The user U chooses his identity ID_U, and produces PW, a random number r_a to take them into computation $H(r_a \| PW)$, which submits ID_U, $H(r_a \| PW)$ to RC via a secure channel.

Step 2. RC $\rightarrow$ User U_j : $\{B, B_A\}$

When RC receives ID_U, $H(r_U \| PW)$ from the user U, RC computes $B=H(ID_U \| k) \oplus H(r_U \| PW)$ and $B_A = H(\text{Anonymous} \| k) \oplus H(r_U \| PW)$, where k is the private key for RC. Then, RC sends back B, B_A via a secure channel. Upon receiving up, he stores $\{ID_U, r_U, B, B_A\}$.

2.4. Anonymous authenticated key agreement phase

Step 1. User $U_j \rightarrow$ Server S_i:$\{m_1\}$

The user U inputs password PW to compute $B_A^* = B_A \oplus H(r_U \| PW)$, and chooses a random integer a to take them into computation of getting $K_{U\text{-}RC}=T_aT_k(x)$ as secret key, $H_A=H(B_A^* \| ID_{Si} \| T_a(x))$ as verified message, $C_1 = E_{K_{U\text{-}RC}}(Ano_{Si} \| ID_{Si}) \| H_A)$,where Ano_{Si} acts temporary ID of the user U. After that, the user sends message $m_1 = \{Ano_{Si}, T_a(x), C_1\}$ to the server S_i.

Step 2. Server $S_i \rightarrow$RC:$\{m_2\}$

Upon receiving the message $m_1 = \{Ano_{Si}, T_a(x), C_1\}$ from the user U. The server S_i selects random r_i to compute $T_{ri}(x)$ and $C_2 = H(ID_{Si})||m_1||R||T_{ri}(x))$. Then, the user U sends message $m_2 = \{ID_{Si}, T_{ri}(x), C_2, m_1\}$ to the RC.

Step 3. RC $\rightarrow$ Server S_i:$\{ID_{RC}, C_3\}$ and RC $\rightarrow$ User U_j: $\{ID_{RC}, C_4\}$

RC gets the message $m_2 = \{ID_{Si}, T_{ri}(x), C_2, m_1\}$ to conduct the following activity:

(1) Authentication of the server S_i:

RC computes $R'=H(ID_{Si} \| k)$ and $C'_2 = H(ID_{Si}||m_1||R'||T_{ri}(x))$ for verifying $C'_2 \stackrel{?}{=} C_2$.

(2) Judgment of Anonymous authentication:

RC computes $K_{RC\text{-}U} = T_kT_a(x)$ to decrypt $D_{K_{RC\text{-}U}}(C_1)$ and computes. By getting the parameters, RC finds Ano_{Si} to judge its anonymous authentication so that it makes RC conduct to compute $B_A^* = H(\text{Anonymous} \| k)$ with using parameter Anonymous.

(3) Challenge of verified message:

RC gets H_A by decrypting to compute $H'_A = H(B_A^* \| ID_{Si} \| T_a(x))$ to verify $H'_A \ ?= H_A$.

(4) The computation of the session key:

RC computes $H_{RC-U} = H(B_A^* \| ID_{RC} \| ID_{Si} \| T_{ri}(x))$ to get $C_3 = H(ID_{RC} \| ID_{Si} \| m_1 \| R \| T_{ri}(x))$, $C_4 = E_{RC-U}(ID_{RC} \| ID_{Si} \| m_1 \| T_{ri}(x) \| H_{RC-U})$. Next, RC respectively sends the messages to the user U and the server S_i. ID_{RC}, C_3 belongs to the server S_i. ID_{RC}, and C_4 belongs to the user U.

Step 4. Session Key

The user U and the server S_i, respectively, receive the message. It can conduct to authenticate and produce the session key for building the secure link, by the following activity:

(1) For Servers:

The server S_i computes $C'_3 = H(ID_{RC} \| ID_{Si} \| m_1 \| R \| T_{ri}(x))$ to check $C'_3 \ ?= C_3$. If it is successful, the server S_i produces session key $SK= T_{ri}T_a(x)$.

(2) For Users:

The user U uses K_{U-RC} to decrypt C_4. After it computes $H'_{RC-U} = H(B_A^* \| ID_{RC} \| ID_{Si} \| T_{ri}(x))$, it can verify $H'_{RC-U} \ ?= H_{RC-U}$. If it is successful, the user U produces session key $SK= T_aT_{ri}(x)$.

2.5. Hiding identity authenticated key agreement phase

Step 1. User $U_j \rightarrow$ Server S_i: $\{m_1\}$

The user U inputs password PW to compute $B^*=B \oplus H(r_U \| PW)$, and chooses a random integer a to take them into computation getting $K_{U-RC} = T_aT_k(x)$ as a secret key, $H_A = H(B^* \| ID_{Si} \| T_a(x))$ as verified message, $C_1 = E_{K_{U-RC}}(ID_U \| ID_{Si} \| H_A)$. After that, the user sends message $m_1 = \{T_a(x), C_1\}$ to the server S_i.

Step 2. Server $S_i \rightarrow$ RC: $\{m_2\}$

Upon receiving the message $m_1 = \{T_a(x), C_1\}$ from the user U. The server S_i selects random r_i to compute $T_{ri}(x)$ and $C_2 = H(ID_{Si} \| m_1 \| R \| T_{ri}(x))$. Then, the user U sends message $m_2 = \{ID_{Si}, T_{ri}(x), C_2, m_1\}$ to the RC.

Step 3. RC $\rightarrow$ Server S_i: $\{ID_{RC}, C_3\}$ and RC $\rightarrow$ User U_j: $\{ID_{RC}, C_4\}$

RC gets the message $m_2 = \{ID_{Si}, T_{ri}(x), C_2, m_1\}$ to conduct the following activity:

(1) Authentication of the server S_i:

RC computes $R'=H(ID_{Si} \| k)$ and $C'_2 = H(ID_{Si}) \| m_1 \| R' \| T_{ri}(x))$ for verifying $C'_2 \ ?= C_2$.

(2) Judgment of hiding identity authentication:

RC computes $K_{RC-U} = T_kT_a(x)$ to decrypt $D_{K_{RC-U}}(C_1)$ to compute. By getting the parameters, RC finds ID_U to judge its using hiding identity authentication so that it makes RC conduct to compute $B^*=H(ID_U \| k)$.

(3) Challenge of verified message:

RC gets H_A by decrypting to compute $H'_A = H(B^* \| ID_{Si} \| T_a(x))$ to verify $H'_A \ ?= H_A$.

(4) The computation of the session key:

RC computes a secret key $K_{S-RC} = T_kT_{ri}(x)$, $H_{RC-S} = H(ID_U \| ID_{RC} \| ID_{Si} \| T_{ri}(x) \| R \| m_1)$, $H_{RC-U} = H(B\text{^}* \| ID_{RC} \| ID_{Si} \| T_{ri}(x))$ to get $C_3 = E\_(RC\text{-}S) (ID_{RC} \| ID_{Si} \| m_1 \| R \| T_{ri}(x) \| H_{RC-S})$, $C_4 = E_{RC-U}(ID_{RC} \| ID_{Si} \| m_1 \| T_{ri}(x) \| H_{RC-U})$. Next, RC respectively sends the messages to the user U and the server S_i. ID_{RC}, C_3 belongs to the server S_i. ID_{RC}, C_4 belong to the user U.

Step 4. Session Key

The user U and the server S_i respectively receive the message. It can conduct to authenticate and produce the session key for building the secure link, by the following activity:

(1) For Servers:

The server S_i uses $K_{S\text{-}RC}$ to decrypt C_3.The server S_i computes $H'_{RC\text{-}S} = H(ID_U \| ID_{RC} \| ID_{Si} \| T_{ri}(x) \| R \| m_1)$ to check $H'_{RC\text{-}S} ?= H_{RC\text{-}S}$. If it is successful, the server S_i produces a session key $SK= T_{ri}T_a(x)$.

(2) For Users:

The user U uses $K_{U\text{-}RC}$ to decrypt C_4. After it computes $H'_{RC\text{-}U} =H(B \| ID_{RC} \| ID_{Si} \| T_{ri}(x))$, it can verify $H'_{RC\text{-}U} ?= H_{RC\text{-}U}$. If it is successful, the user U produces a session key $SK= T_aT_{ri}(x)$.

2.6. Password changing phase

Step 1. User $U_j \rightarrow$RC:$\{m_1\}$

The user U chooses a new password PW', two random integers: r'_a, a and computes $B^*=B \oplus H(r_U \| PW)$, $T_a(x)$, $K_{U\text{-}RC} = T_aT_k(x)$, $H_A =H(B^* \| ID_{RC} \| T_a(x))$ and $C_1 = E_{K_{U\text{-}RC}} (ID_U \| H(r'_a \| PW') \| H_A)$. Then, he sends $m_1 = \{T_a(x), C_1\}$ to RC.

Step 2. RC$\rightarrow$User U_j: $\{ID_{RC}, C_2\}$

Upon receiving the message $m_1 = \{T_a(x), C_1\}$ from the user U. RC decrypts C_1 and computes $B^*=H(ID_U \|k)$ and $H'_A =H(B^* \| ID_{RC} \| T_a(x))$ to check $H'_A ?= H_A$. If it is successful, RC computes $'=H(ID_U \|k) \oplus H(r'_U \|PW')$, $B'_A =H(\text{Anonymous} \|k) \oplus H(r'_U \|PW')$, $H_{RC} = H(ID_{RC}\|ID_U\|B'\|B'_A)$ and $C_2 = E_{K_{U\text{-}RC}}(ID_{RC} \| ID_U \|B' \|B'_A \| H_{RC})$. After that, RC sends ID_{RC}, C_2 to the user U.

Step 3. User U_j: $\{ID_U, r'_U, B', B'_A\}$

The user U gets the message ID_{RC}, C_2 to decrypt C_2, which can conduct to compute $H'_{RC} =H(ID_{RC} \| ID_U \|B' \|B'_A)$ in local. Then, he can verify $H'_{RC} ?= H_{RC}$. If it is successful to confirm, the user U stores $\{ID_U, r'_U, B', B'_A\}$ to replace original out.

3. Security and Performance Analysis of Zhu's Protocol

In this protocol, the main contribution proposed by Zhu is using provable privacy-protection system (PPPS) which can offer users whether to select anonymous services or not. It can help users protect their privacy. Certainly, the protocol can resist all common attacks so that it can achieve a secure structure.

However, we found it had serious problems, unsafe and inefficient. If the number of users increases linearly, it would bring about that RC must waste a lot of computation resources to deal with authentication.

3.1. The weakness of the inefficient performance

When users and servers want to authenticate the identity in the system, the user will send a verified message to the server, and the server will make computation and produce the message including their messages to send to registration center (RC), which helps them check both identities.

However, in his protocol, all of its authentications are performed in RC. If the number of users continually increases up to exceed hardware computation, it will be overloaded and results in a serious burden.

3.2. The weakness of the Denial-of-Register attack

We assume another situation that a lot of legal users have been controlled locally by a hacker. They submit the authenticated request to the servers simultaneously. Meanwhile, when the servers receive numerous requests from the users, each server sends all of the requests to RC.

However, the mechanism does not exist in this protocol to resist this probable from happening so that it will be easy to achieve an illegal action by attackers. There are not the timestamp or random number to protect and prevent the combination with denial-of-Register attack and reply attack. Therefore, it totally has a potential threat. As long as attackers intercept lots of messages and transmit heavily to the server. Meanwhile, the server also does the same thing as users gather their authenticated challenge to send to RC. The server and RC will collapse simultaneously, and the system will stop to operate.

4. Conclusion

In summary, we have shown the weaknesses of Zhu's privacy-protection system for multi-server. Zhu claimed his scheme can not only own high efficiency and unique functionality but is also robust to various attacks and achieves perfect forward secrecy. However, in this article, we have shown that Zhu's scheme is inefficient and insecure. If the number of users increases linearly, it would bring about that RC must waste a lot of computation resources to deal with authentication. Furthermore, Zhu's scheme could not against the denial-of-register attack.

Acknowledgments

This work was partially supported by the Ministry of Science and Technology, Taiwan, under grant MOST 108-2622-8-468-001-TM1, MOST 107-2221-E-845-002-MY3, and MOST 107-2221-E-845-001-MY3.

References

[1] M. Rajaram and T. D. Suresh. An interval-based contributory key agreement. International Journal of Network Security, 2011, 13(2): 92-97.

[2] S. F. Chiou, H. T. Pan, E. F. Cahyadi, and M. S. Hwang. Cryptanalysis of the mutual authentication and key agreement protocol with smart cards for wireless communications. International Journal of Network Security, 2019, 21(1): 100-104.

[3] L. C. Hwang and M. S. Hwang. An efficient MQV key agreement scheme. International Journal of Network Security, 2014, 16(2): 157-160.

[4] C. Guo, C. C. Chang. A novel threshold conference-key agreement protocol based on generalized chinese remainder theorem [J]. International Journal of Network Security, 2015, 17(2): 165-173.

[5] C. Guo, C. C. Chang, S. C. Chang. A secure and efficient mutual authentication and key agreement protocol with smart cards for wireless communications [J]. International Journal of Network Security, 2018, 20(2): 323-331.

[6] M. Ramadan, F. Li, C. X. Xu, A. Mohamed, H. Abdalla, A. Abdalla. User-to-user mutual authentication and key agreement scheme for lte cellular system [J]. International Journal of Network Security, 2016, 18(4): 769-781.

[7] Y. K. Peker. A new key agreement scheme based on the triple decomposition problem [J]. International Journal of Network Security, 2014, 16(6): 426-436.

[8] T. H. Feng, C. H. Ling, M. S. Hwang. Cryptanalysis of Tan's improvement on a password authentication scheme for multi-server environments [J]. International Journal of Network Security, 2014, 16: 318-321.

[9] D. He, W. Zhao, and S. Wu. Security analysis of a dynamic ID-based authentication scheme for multi-server environment using smart cards [J]. International Journal of Network Security, 2013, 15: 282-292.

[10] L. H. Li, I. C. Lin, M. S. Hwang. A remote password authentication scheme for multi-server architecture using neural networks [J]. IEEE Transactions on Neural Networks, 2001, 12: 1498-1504.

[11] I. C. Lin, M. S. Hwang, L. H. Li. A new remote user authentication scheme for multi-server architecture [J]. Future Generation Computer Systems, 2003, 19: 13-22.

[12] R. Amin. 2016. Cryptanalysis and efficient dynamic id based remote user authentication scheme in multi-server environment using smart card [J]. International Journal of Network Security, 2016, 18(1): 172-181.
[13] H. Zhu. A provable privacy-protection system for multi-server [J]. Nonlinear Dynamics, 2015, 82(1): 835–849.
[14] A. Kumar and S. Tripathi. Anonymous ID-based group key agreement protocol without pairing [J]. International Journal of Network Security, 2016, 18(2): 263-273.
[15] Min Wu, Jianhua Chen, Ruibing Wang. An enhanced anonymous password-based authenticated key agreement scheme with formal proof [J]. International Journal of Network Security, 2017, 19(5): 785-793.
[16] Yanjun Liu, Chin-Chen Chang, Chin-Yu Sun. Notes on "An anonymous multi-server authenticated key agreement scheme based on trust computing using smart card and biometrics"[J]. International Journal of Network Security, 2016, 18(5): 997-1000.
[17] Hongfeng Zhu, Yifeng Zhang. An improved two-party password-authenticated key agreement protocol with privacy protection based on Chaotic maps [J]. International Journal of Network Security, 2017, 19(4): 487-497.
[18] Hongfeng Zhu. Secure Chaotic maps-based group key agreement scheme with privacy preserving [J]. International Journal of Network Security, 2016, 18(6): 1001-1009.
[19] Hongfeng Zhu, Yifeng Zhang, Yan Zhang and Haiyang Li. A novel and provable authenticated key agreement protocol with privacy protection based on chaotic maps towards mobile network [J]. International Journal of Network Security, 2016, 18(1): 116-123.
[20] Hongfeng Zhu, Yan Zhang, Haiyang Li, and Lin Lin. A novel biometrics-based one-time commitment authenticated key agreement scheme with privacy protection for mobile network [J]. International Journal of Network Security, 2016, 18(2): 209-216.
[21] Hai-Duong Le, Ngoc-Tu Nguyen, and Chin-Chen Chang. Provably secure and efficient three-factor authenticated key agreement scheme with untraceability [J]. International Journal of Network Security, 2016, 18(2): 335-344.
[22] P. Hiranvanichakorn. Provably authenticated group key agreement based on braid groups - the dynamic case [J]. International Journal of Network Security, 2017, 19(4): 517-527.
[23] Q. Cheng. Security analysis of a pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile networks [J]. International Journal of Network Security, 2015, 17(4): 494-496.
[24] R. S. Ranjani, D. L. Bhaskari, P. S. Avadhani. An extended identity based authenticated asymmetric group key agreement protocol [J]. International Journal of Network Security, 2015, 17(5): 510-516.
[25] V. S. Naresh and N. V.E.S. Murthy. Elliptic curve based dynamic contributory group key agreement protocol for secure group communication over ad-hoc networks [J]. International Journal of Network Security, 2015, 17(5): 588-596.
[26] Q. Cheng and C. Tang. Cryptanalysis of an ID-based authenticated dynamic group key agreement with optimal round [J]. International Journal of Network Security, 2015, 17(6): 678-682.