Cryptanalysis of Hu-Niu-Yang's Multi-server Password Authenticated Key Agreement Schemes Using Smart Card

Sang-Gon Lee, Meng-Hui Lim, Hoon-Jae Lee, *Member, KIMICS*

Abstract—Multi-server password authentication schemes enable remote users to obtain service from multiple servers with single password without separately registering to each server. In 2007, Hu-Niu-Yang proposed an improved efficient password authenticated key agreement scheme for multi-server architecture based on Chang-Lee's scheme proposed in 2004. This scheme is claimed to be more efficient and is able to overcome a few existing deficiencies in Chang-Lee's scheme. However, we find false claim of forward secrecy property and some potential threats such as offline dictionary attack, key-compromise attack, and poor reparability in their scheme. In this paper, we will discuss these issues in depth.

Index Terms — Multi-server password authentication, forward Secrecy, cryptography, key agreement.

I. INTRODUCTION

PASSWORD authentication is one of the simplest and the most convenient authentication mechanisms over public networks. Conventional password authentication schemes are suitable for solving the privacy and security problems of a single user under a client-server architecture. As the network becomes larger, the password authentication schemes that only support a single server are clearly not sufficient to address the users' growing needs. Therefore, many dynamic password authentication schemes have been proposed for the multiserver architecture [1-5]. In addition, most of them provide an added functionality to derive a session key, such as an encryption key or MAC key, for each session, particularly for subsequent secure communication use, to protect the messages transmitted during each session. These schemes are therefore referred to as *password authenticated key agreement protocols*.

In 2004, Juang proposed a password authenticated key agreement scheme for a multiserver architecture [3]. However, Juang's scheme lacks efficiency, and each server needs to additionally protect and securely maintain an encrypted key table. In the same year, Chang and Lee proposed a similar but more efficient and secure scheme [1]. Their scheme is claimed to be able to achieve following six vital requirements of the multiserver password authentication scheme: choose and change password at will; low computation; security; mutual authentication; single registration; session key agreement. In 2007, Hu-Niu-Yang identified some deficiencies in Chang-Lee's scheme and subsequently proposed an improved password authenticated key agreement scheme for multi-server architecture [2] based on Chang-Lee's scheme. This scheme is claimed to be more efficient, capable of overcoming such deficiencies and able to satisfy the six vital requirements which have been defined by Chang and Lee. However, we argue that this scheme is not as ideal as described by the authors. We discover that the fulfillment of forward secrecy property is falsely claimed and some potential threats to the scheme, such as offline dictionary attack, key-compromise attack, and poor reparability, have been negligently unexplored in the corresponding security analysis. In this paper, we will analyze these issues in detail.

The structure of this paper is organized as follows. In the next section, we will revisit Hu et al.'s multi-server password authenticated key agreement scheme using smart cards. In section 3, we will discuss a few cryptographic flaws of Hu et al.'s scheme and provide countermeasures to satisfy the forward secrecy property and offline dictionary attack. In the last section, we will conclude this paper.

II. REVIEW OF HU ET AL.'S SCHEME

The main notations used throughout this paper are listed as follows.

Manuscript received August 9, 2009; revised August 31, 2009

Sang-Gon Lee is with the Division of Computer & Information Engineering, Dongseo University, Busan, 617-716, Korea (Tel: +82-51-320-1760, Fax: +82-51-320-8255, Email: nok60@dongseo.ac.kr)

RC – Registration center
U_i – User U_i
S_j – Server S_j
ID_{U_i} – Identity of user U_i
ID_{S_j} – Identity of server S_j
PW_{U_i} – Password of user U_i
K_x – Pre-shared secret key between RC and every S_j
SK_{U_i,S_j} – Session key derived by user U_i and server S_j
N_{U_i} – Nonce generated by user U_i
N_{S_j} – Nonce generated by server S_j
$\{\cdot\}_k$ – Symmetric encryption with key k
$D_k\{\cdot\}$ – Symmetric decryption with key k
$H(\cdot)$ – One-way collision-resistant hash function
$\oplus$ – XOR operation
$\|$ – Concatenation
$\Rightarrow$ – Secure channel transfer
$\rightarrow$ – Common channel transfer

This subsection reviews Hu et al.'s multi-server password authenticated key agreement scheme [2]. Hu et al.'s scheme basically consists of 3 phases: *registration* phase, *login* phase and *authentication and key agreement* phase. The details of these phases along with a password change method can be described as follows:

II.1 Registration phase:

Step R1. $U_i \Rightarrow RC: \{ID_{U_i}, PW_{U_i}\}$

Initially, a user U_i, with identity ID_{U_i}, chooses a password PW_{U_i} and submits $\{ID_{U_i}, PW_{U_i}\}$ to the RC through a secure channel.

Step R2. $RC \Rightarrow U_i$: Smart card with in-formation $\{ID_{U_i}, H(\cdot), G_{U_i}, V_{U_i}\}$.

On reception of the same, RC computes

$$G_{Ui} = H(K_x, ID_{U_i}) \quad (1)$$

and

$$V_{U_i} = G_{Ui} \oplus H(PW_{U_i}). \quad (2)$$

Then, the RC successfully issues a smart card containing information $\{ID_{U_i}, H(\cdot), G_{U_i}, V_{U_i}\}$ to U_i through a secure channel.

II.2 Login phase:

Step L1. U_i inserts the smart card into the input device and enters ID_{U_i} and PW_{U_i} to request access to the server S_j.

Step L2. $U_i \rightarrow S_j: \{ID_{U_i}, \{N_{U_i}, M_{U_i}\}_{G_{U_i}}\}$.

The smart card checks the validity of ID_{U_i}, computes $G'_{U_i} = H(PW_{U_i}) \oplus V_{U_i}$, and verifies

$$G'_{U_i} \stackrel{?}{=} G_{U_i}. \quad (3)$$

If they are not equivalent, the access request is rejected. Otherwise, the smart card generates a nonce N_{U_i}, computes

$$M_{U_i} = H(N_{U_i} \| ID_{U_i}), \quad (4)$$

encrypts $\{N_{U_i}, M_{U_i}\}$ under key G_{U_i}, and sends to S_j.

II.3 Authentication and key agreement phase:

Step A1. $S_j \rightarrow U_i: \{ID_{S_j}, L, R_{S_j}, M_{S_j}\}$.

S_j verifies the validity of ID_{U_i}, computes G_{U_i} as in Eq. (1), and performs

$$D_{G_{U_i}}(\{N_{U_i}, M_{U_i}\}_{G_{U_i}}) \quad (5)$$

by using key G_{U_i} to obtain N_{U_i} and M_{U_i}. Then, S_j computes $M'_{U_i} = H(N_{U_i} \| ID_{U_i})$ and verifies

$$M'_{U_i} \stackrel{?}{=} M_{U_i}. \quad (6)$$

If the result is negative, S_j terminates the session. Otherwise, S_j generates a nonce N_{S_j} and computes

$$M_{S_j} = H(N_{S_j} \| ID_{S_j}) \quad (7)$$

$$L = M_{U_i} \oplus N_{S_j} \quad (8)$$

and

$$R_{S_j} = H(N_{U_i} + 1) \quad (9)$$

and sends $\{ID_{S_j}, L, R_{S_j}, M_{S_j}\}$ to U_i.

Step A2. $U_i \rightarrow S_j: \{ID_{U_i}, R_{U_i}\}$.

The smart card computes $R'_{S_j} = H(N_{U_i} + 1)$ and verifies whether

$$R'_{S_j} \stackrel{?}{=} R_{S_j}. \quad (10)$$

If Eq. (10) does not hold, the smart card terminates the session. Otherwise, the smart card retrieves

$$N_{S_j} = M_{U_i} \oplus L \quad (11)$$

computes $M'_{S_j} = H(N_{S_j} \| ID_{S_j})$ and verifies

$$M'_{S_j} \stackrel{?}{=} M_{S_j}. \quad (12)$$

If Eq. (12) does not hold, the smart card aborts. Otherwise, the smart card computes the session key

$$SK_{U_i,S_j} = H(N_{U_i}, N_{S_j}, G_{U_i}), \quad (13)$$

$$R_2 = H(N_{S_j} + 1), \quad (14)$$

and sends $\{ID_{U_i}, R_{U_i}\}$.

Step A3. S_j computes $R'_{U_i} = H(N_{S_j} + 1)$ and verifies

$$R'_{U_i} \stackrel{?}{=} \mathrm{R}_{U_i}. \quad (15)$$

If it does not hold, S_j aborts. Otherwise, S_j computes the session key SK_{U_i,S_j} as in Eq. (13).

II.4 Password change method:

Step P1. U_i inserts the smart card into the input device and enters ID_{U_i} and PW_{U_i} to request for a password change.

Step P2. The smart card checks the validity of ID_{U_i}, computes $G'_{U_i} = H(PW_{U_i}) \oplus V_{U_i}$, and verifies

$$G'_{U_i} \stackrel{?}{=} \mathrm{G}_{U_i}. \quad (16)$$

If they are not equivalent, the access request is rejected. Otherwise, the smart card prompts for a new password $PW^*_{U_i}$. Then, the smart card computes

$$\begin{aligned} V^*_{U_i} &= V_{U_i} \oplus H(PW_{U_i}) \oplus H(PW^*_{U_i}) \\ &= H(PW^*_{U_i}) \oplus G_{U_i} \end{aligned} \quad (17)$$

and replaces V_{U_i} with $V^*_{U_i}$.

III. WEAKNESSES OF HU ET AL.'S SCHEME

III.1 False Claim of Forward Secrecy Property

Provision of the forward secrecy property by a password based authenticated key agreement protocol is of the utmost importance for avoiding past session keys from being recovered by the compromise of any participating entity's long-term secret key. According to Hu et al., their scheme offers this vital property as the session key computed in Eq. (13) is claimed to be unrecoverable without the nonce values N_{U_i} and N_{S_j} even when G_{U_i} or K_x happens to be compromised. However, we observe that the forward secrecy property has been falsely claimed by the authors.

Assume that an adversary, eavesdropping on the channel, has obtained all the messages exchanged in a previous session and has compromised K_x at a later time. We find that the adversary is able to derive G_{U_i} using Eq. (1), decrypts $\{N_{U_i}, M_{U_i}\}_{G_{U_i}}$ in step L2 to retrieve N_{U_i}, compute Eq. (11) to obtain N_{S_j} and recover SK_{U_i,S_j} in Eq. (13) with the knowledge of derived information $\{G_{U_i}, M_{U_i}, N_{S_j}\}$. In this case, the security of the scheme is significantly violated, resulted from the absence of forward secrecy property.

Countermeasure. From our experience we can say that forward secrecy in a key agreement protocol can not be achieved using only the hash function and symmetric encryption schemes. We need to use public key techniques (e.g., exponentiations in a multiplicative group). An efficient approach to the preservation of the forward secrecy property is to slightly tweak the protocol specification. Instead of choosing N_{U_i} and N_{S_j} as random nonces, we redefine them as the user's and the server's ephemeral public key computed respectively as follows:

$$N_{U_i} = g^{r_{U_i}}, \text{ for } r_{U_i} \in_R Z^*_q \quad (18)$$

and

$$N_{S_j} = g^{r_{S_j}}, \text{ for } r_{S_j} \in_R Z^*_q \quad (19)$$

Moreover, we include additional component $N^{r_{U_i}}_{S_j} = N^{r_{S_j}}_{S_i} = g^{r_{U_i} r_{S_j}}$ in the session key derivation function such that

$$U_i : SK_{U_i S_j} = H(N^{r_{U_i}}_{S_j}, N_{U_i}, N_{S_j}, G_{U_i}); \textit{and}$$

$$S_j : SK_{U_i S_j} = H(N^{r_{S_j}}_{U_i}, N_{U_i}, N_{S_j}, G_{U_i}).$$

With these modifications, if an adversary happens to learn K_x after completion of a key establishment session, he/she will only be able to derive G_{U_i}, N_{U_i}, and N_{S_j} but not the shared secret $g^{r_{U_i} r_{S_j}}$ (bound by the intractability of Computational Diffie-Hellman Problem) because he/she does not possess the knowledge of any private ephemeral key r_{U_i} or r_{S_j}. Intuitively, the secrecy of the session key and the security of the scheme can be significantly preserved.

III.2 Offline Dictionary Attack

Tamper resistance of smart cards is widely assumed in most smart card based schemes. However, such an assumption may be unrealistic in practice. Several research works have pointed out that the secret information stored in current existing smart cards could be extracted by either monitoring the power consumption [7] or analyzing the information leakage

[8]. According to Hu et al.'s protocol specification, secret information stored in the user's smart card include $\{ID_{U_i}, H(\cdot), G_{U_i}, V_{U_i}\}$. If any adversary happens to reveal such information from a user's smart card successfully, she could then stand a chance to mount an offline guessing attack to recover the respective password.

Although ensuring password selected from a large password space can resist such an exhaustive attempt, most users would select passwords only from a small subset of the full password space for the sake of easy remembrance, resulting in some weak passwords with low entropy chosen. The adversary upon obtaining the secret information in the smart card could simply try for any password PW_A from the smaller password space and subsequently verify whether

$$G_{U_i} \stackrel{?}{=} (PW_A) \oplus V_{U_i} \quad (20)$$

If this equation holds, the adversary is deemed to have guessed the respective password PW_{U_i} correctly.

Countermeasure. It is a proven fact that *public key techniques (e.g., exponentiations in a multiplicative group)* are absolutely necessary to make password systems secure against offline dictionary attacks, whereas the involvement of public key cryptosystems under a PKI(e.g., public key encryption and digital signature schemes) is not essential [9, 10].

III.3 Insider Impersonation Attack

Hu et al. assumed all servers $S_j (1 \le j \le n)$ to be honest in their protocol specification. We find that this assumption is rather too strong and it is in fact impractical for a multi-server environment in practice. We observe that most multi-server password authenticated key agreement schemes in the literature would at most assume the trustworthiness of RC but not on the trusted servers [3, 5, 6].

Servers are always the attacking targets of the adversaries and they can be compromised at anytime. Since all the servers pre-share a single secret K_x with RC in Hu et al.'s scheme, a mere compromise of a server which results in a revelation of K_x would render the whole system insecure. The adversary who has acquired this secret could impersonate any legitimate user or any other legitimate server to request for access or to authenticate other users since G_{U_i} can be easily derived using K_x.

III.4 Poor Reparability

A secure key agreement protocol is *reparable* from the compromised secret keys if the security breaches due to these compromised keys are removed once secure keys replace the compromised keys [12, 13].

As described in III.2, once an adversary successfully extracts a user's secret information he can act as the user or a server of which the user is a legal member. The adversary can use G_{U_i} to impersonate U_i to login S_j, or impersonate S_j to full U_i. Moreover the adversary can use G_{U_i} to launch man-in-the middle attack by sitting between U_i and S_j and establishing parallel session with them respectively.

Until G_{U_i} is updated among RC, U_i, and all servers, the above attacks can not be stopped even when U_i has detected that the G_{U_i} has been compromised. Because the user authentication key G_{U_i} is not related to U_i's password but only to U_i's identifier ID_{U_i} and pre-shared secret key K_x between RC and every S_j, RC can not change G_{U_i} for U_i unless ID_{U_i} or K_x can be changed. However, for K_x is commonly shared with all servers, it is unreasonable and inefficient if K_x should be changed to recover the security of U_i only. In addition, it is also impractical to change ID_{U_i} which should be tied to U_i in most application systems. Therefore, Hu et al.'s scheme is not easily reparable [13].

IV. CONCLUSIONS

In this paper, we have revisited the security of Hu et al.'s password authenticated key agreement scheme. Particularly, we have pointed out the absence of forward secrecy property which has been falsely alleged to have fulfilled by Hu et al. We also have further highlighted some possible threats due to offline dictionary attack, key-compromised impersonate attack, and poor reparability. Besides, we provided countermeasures to re-satisfy the forward secrecy property and offline dictionary attack. We hope that our discussion will provide useful awareness to the protocol designers in taking appropriate security considerations while designing any password authenticated key agreement schemes so as to resist the discussed vulnerabilities.

ACKNOWLEDGMENT

For the first author, this work was supported by 2008 Research Fund of Dongseo University.

REFERENCES

[1] C.C. Chang and J.S. Lee, "An Efficient and Secure Multi-server Password Authentication Scheme using Smart Cards," *International Conference on Cyberworlds(CW '04)*, pp. 417-422, 2004.

[2] L. Hu, X. Niu, and Y. Yang, "An Efficient Multi-server Password Authenticated Key Agreement Scheme using Smart Cards," *International Conference on Multimedia and Ubiquitous Engineering (MUE '07)*, IEEE, pp. 903-907, 2007.

[3] W.S. Juang, "Efficient Multi-server Password Authenticated Key Agreement using Smart Cards," *IEEE Trans. on Consumer Electronics*, vol. 50, no. 1, pp. 251-255, 2004.

[4] C.-L. Lin and T. Hwang, "A Password Authentication Scheme with Secure Password Updating," *Computer and Security*, vol. 22, no. 1, pp. 68-72 , 2003.

[5] I.-C. Lin, M.-S. Hwang, and L.-H. Li, "A New Remote User Authentication Scheme for Multi-server Architecture," *Future Generation Computer Systems*, vol. 19, pp. 13-22, 2003.

[6] L.-H. Li, I.-C. Lin, and M.-S. Hwang, "A Remote Password Authentication Scheme for Multi-server Architecture using Neural Networks," *IEEE Trans. on Neural Networks*, vol. 12, no. 6, pp. 1498-1504, 2001.

[7] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis," *Advances in Cryptology (CRYPTO '99)*, LNCS 1666, pp. 388-397, 1999.

[8] T.S. Messerges, E.A. Dabbish, and R.H. Sloan, "Examining Smart-Card Security Under the Threat of Power Analysis Attacks," *IEEE Trans. on Computers*, vol. 51, no. 5, pp. 541-552, 2002.

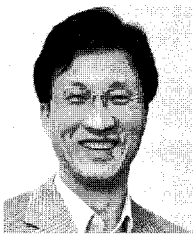
[9] S. Halevi and H. Krawczyk, "Public-Key Cryptography and Password Protocols," *Proc. ACM Conf. Computer and Comm. Security*, pp. 122-131, 1998.

[10] Y. Yang, R. H. Deng, and F. Bao, "A Practical password-based two-server authentication and key exchange system," *IEEE Trans. On Dependable and Secure Computing*, vol.3, no.3 pp. 105-114, 2006.

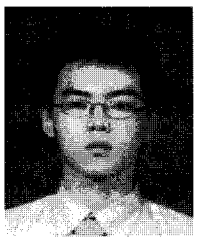
[11] S. B. Wilson, and A. Menezes, "Authenticated Diffie-Hellman key agreement protocols," *Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC 98)*, LNCS, vol. 1556, pp. 339-361, 1998.

[12] T. Hwang and W.-C. Ku, "Reparable Key distribution protocols for Internet environments," *IEEE Trans. Commun.*, vol.43, no.5 pp.1947-1949, May 1995.

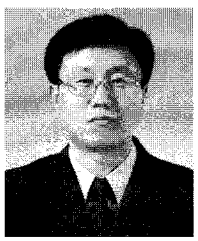
[13] W.-C. Ku, H.-M Chuang, and M.-H Chiang, "Cryptanalysis of a Multi-Server Authenticated Key Agreement Scheme Using Smart Cards," *IEICE Trans. Fundamentals*, vol. E88-A, no.11 Nov. 2005.



Sang-Gon Lee
received the BEng, MEng, and PhD degree in electronics engineering from Kyungpook National University, Korea, in 1986, 1988, and 1993, respectively. He is a professor at the Division of Computer & Information Engineering, Dongseo University. He was an assistant/associate professor at Chang-shin College from 1991 to 1993 and a visiting scholar at QUT, Australia from 1993 to 1994. His research areas include information security, network security, and digital right managements.



Meng-Hui Lim
received his B.Eng (Hons) Electronics (Telecommunications) from Multimedia University, Malaysia in 2006 and his M.Eng (Ubiquitous IT) from Dongseo University, Korea in 2009. He is currently a PhD candidate in the School of Electrical and Electronics Engineering in Yonsei University. His research interests include cryptography, key exchange protocol, information security and biometrics security.



Hoon-Jae Lee
received BS, MS, and PhD Degrees in electronic engineering from Kyungpook National University, Daegu, Korea in 1985, 1987, and 1998, respectively. He is currently an associate professor in the School of Computer and Information Engineering at Dongseo University. From 1987 to 2000, he was a research associate at the Agency for Defense Development (ADD). His current research interests include developing secure communication system, side-channel attack and USN/RFID security.