The Weakness of Moon et al.’s Password Authentication Scheme

Bam-bang Irawan\textsuperscript{1,2} and Min-shiang Hwang\textsuperscript{1,3,*}

\textsuperscript{1}Department of Computer Science and Information Engineering, Asia University
No. 500, Lioufeng Rd., Wufeng, Taichung, Taiwan 41354
\textsuperscript{2}Department of Computer Science, Esa Unggul University
Arjuna Utara 9 Rd, Kebon Jeruk 11510, West Jakarta, Indonesia
\textsuperscript{3}Department of Medical Research, China Medical University Hospital, China Medical University, No.91, Hsueh-Shih Road, Taichung, Taiwan 40402

*Email: mshwang@asia.edu.tw

Abstract. Using smart cards make remote transactions easier for users in Internet. It’s important to identity the legal users to have the access right to obtain the resources. In 2017, Liu et al. proposed an efficient and secure smart card based password authentication scheme. Recently, Moon et al. pointed some weaknesses of Liu et al.’s scheme. They also proposed a password authentication scheme to overcome Liu et al.’s weaknesses. They claim that their scheme is more secure and practical as a remote user authentication scheme. However, we find that some weaknesses of Moon et al.’s scheme. In this article, we will show that Moon et al.’s scheme is vulnerable to the guessing identity and impersonation attacks.

1. Introduction
Security is the need for everyone at home, at the office, on the street, and in every place, because it makes a person safely use security systems and prevent things that should not happen. The security system should be flexible, inexpensive and work continuously without being limited by working hours [1-3]. Smartcard-RFID is an advanced information technology embedded into a card as an information storage medium [4-8]. Implementation of smartcards has currently spread almost in all areas, whether it is used in hotels, homes, attendance at offices and educational institutions, as tough data security [9-12]. Many schemes were applied a smart card to authenticate the legal users in multi-server environment [13-18]. Other schemes are list in [19-34].

In this paper, we propose modifications to the schemes provided by Moon et al.’s [35]. In their papers, we find a weakness during the phase registration, login and authentication, which attacks the security of data transmitted. We have made improvements by modifying the mathematical equations in the 3rd phase. From the given scheme, it can handle the problem of weaknesses during anonymous attacks and impersonation attacks.

This paper, we find that the security weaknesses of the two-factor authentication scheme by Moon et al. After careful analysis, we demonstrate that their scheme does not actually resist anonymous intercepts and user impersonation attacks. To overcome these security vulnerabilities, we propose a new biometrics-based authentication and key agreement scheme using a smart card. In addition, we demonstrate that the proposed authentication scheme is highly more resistant to various attacks, compared with other related schemes.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
For more details we divide this paper into 4 Sections as follows: We briefly introduce some cryptographic definitions in Section 2, where we briefly review Moon et al.’s smart card-based password authentication scheme. In Section 3 its weaknesses is analyzed. Finally, we make a conclusion of the paper in Section 4.

2. Review of Moon et al.’s Scheme
In this section, we show that Moon et al.’s scheme, Secure Smart Card Based Password Authentication [35], is insecure. Their scheme is an improvement of Liu et al.’s scheme [36]. In the scheme, there are two participants, the user U_i and the server S. The scheme consists of four phases: registration, login, authentication, and password changing phase. Some notations used in the scheme are described in Table 1.

| Notation | Meaning |
|----------|---------|
| U_i | The i-th user |
| ID_i, PW_i | The identity and password of the user i |
| S | The server |
| x | The master secret key stored in the S |
| P | The base point of the elliptic curve E |
| rP | The point multiplication defined as rP = P + P + · · · + P. |
| T_i | The timestamp of the user U_i |
| T'_i | The time of receiving the login request message |
| T_s | The timestamp of the S |
| T'_s | The time of receiving the mutual authentication message |
| R_i, P_i | The U_i’s nearly random binary string and auxiliary binary string |
| h(·) | A collision-resistant hash function |
| ⊕ | Exclusive-or operation |
| || | Concatenation operation |
| sk | The shared session key |

2.1. Registration Phase
The server S selects the master secret key x, the base point P of the elliptic curve E and a collision-resistant one-way hash function h(·). Then, the user U_i registers to the server S by the way below:

**Step 1.** The U_i imprints the personal biometric information BIO_i at the device sensor. The device sensor then scans the BIO_i, extracts (R_i, P_i) from Gen (BIO_i) → (R_i, P_i), and stores P_i in the memory. Next, U_i selects the identity ID_i and password PW_i, and calculate RPW_i = h(PW_i||R_i). Lastly, the U_i sends the registration request message {ID_i, RPW_i} to the S over a secure channel.

**Step 2.** After receiving the registration request message from the U_i, the server S verifies whether ID_i is valid, and computes the following parameters:

\[A_i = h(ID_i \oplus x),\]
\[B_i = h(A_i) \oplus RPW_i,\]
\[C_i = h(ID_i||RPW_i),\]
\[D_i = x \oplus A_i \oplus h(x).\]

**Step 3.** The server S stores the data {B_i, C_i, D_i, h(·), P_i} on a new smart card and issues the smart card to the user U_i over a secure channel.

**Step 4.** The user U_i stores the random string P_i into the smart card.

2.2. Login Phase
After performing the registration phase, then the user proceeds on the login phase invoke U_i user to log into server S. The steps of this phase are done as follows.
Step 1. The $U_i$ inserts his/her smart card into the card reader and enters the identity ID$_i$ and password PW$_i$, and imprints the biometrics BIO$_i^*$ at the sensor. The sensor then sketches BIO$_i^*$ and recovers R$_i$ from $\text{Rep}(\text{BIO}_i^*, \text{Pi}) \rightarrow \text{R}_i \text{BIO}_i^*$. 

Step 2. The smart card first computes two parameters: RPW$_i = h(\text{PW}_i \| \text{R}_i)$ and C$_i' = h(\text{ID}_i \| \text{RPW}_i)$. The smart card then examines whether $C'_i$ is equal to the stored $C_i$. If this holds, the smart card continues to perform Step 3; otherwise, the smart card terminates this session.

Step 3. The smart card randomly generates a number $\alpha$ and $n_i$, and computes the following parameters: $h(A_i) = B_i \oplus \text{RPW}_i$, AID$_i = \text{ID}_i \oplus h(A_i)$, $E_i = aP$, $F_i = h(\text{ID}_i \| h(A_i) \| \text{E}_i \| T_i)$, where $T_i$ is the current timestamp of the user $U_i$.

Step 4. The smart card sends the login request message $\{\text{AID}_i, D_i, E_i, F_i, T_i\}$ to the server $S$.

2.3. Authentication Phase

Completing this phase, the user $U_i$ and the server $S$ could mutually authenticate each other and establish a shared session key for the subsequent secret communication. These steps of the authentication phase are shown as follows:

Step 1. The server $S$ verifies whether $T'_i - T_i \leq \Delta T$, where $T'_i$ is the time of receiving the login request message and $\Delta T$ is a valid time threshold. If both conditions are true, the server $S$ continues to execute Step 2; otherwise, the server $S$ rejects the login request.

Step 2. The server $S$ computes the following parameters:

$$A'_i = D_i \oplus x \oplus h(x),$$
$$\text{ID}'_i = \text{AID}_i \oplus h(A'_i),$$
$$F'_i = h(\text{ID}'_i \| h(A'_i) \| E_i \| T_i).$$

The server $S$ then compares whether $F'_i$ is equals $F_i$. If this holds, the server $S$ confirms that the user $U_i$ is valid and the login request is accepted; otherwise, the server $S$ rejects the login request.

Step 3. Next, the server $S$ randomly generates a number $\beta$ and computes the following parameters:

$$F_i = \beta P,$$
$$G_i = h(\text{ID}'_i \| h(A'_i) \| F_i \| T_i).$$

where $T_s$ is the current timestamp of the server $S$.

Step 4. The server $S$ sends the mutual authentication message $\{F_i, G_i, T_s\}$ to the user $U_i$.

Step 5. Upon receiving the message $\{F_i, G_i, T_s\}$ from the $S$, the user $U_i$ checks the validity of the $T_s$. If $T'_s - T_s \leq \Delta T$, where $T'_s$ is the time of receiving the mutual authentication message, the user $U_i$ continues to perform Step 6; otherwise, the user $U_i$ terminates this connection.

Step 6. The user $U_i$ computes $G'_i = h(\text{ID}_i \| h(A_i) \| F_i \| T_s)$, then checks whether $G'_i$ is equal to the received $G_i$. If this holds, the validity of the server $S$ is authenticated; otherwise, the session is terminated.

Step 7. Finally, the user $U_i$ and the server $S$ construct a shared session key:

$$sk = a \beta P$$

to ensure the secret communication.

3. Cryptanalysis of Moon et al.’s Scheme

Moon et al.’s scheme is based on the elliptic curve cryptosystem (ECC). There are two weaknesses: Gussing identity and user impersonation attacks.

- Gussing Identity Attack:
  Moon et al.’s scheme [35] did not hide the ID user $U_i$ in the login phase and authentication phase. The attacker could intercept AID$_i$, ID$_i$, F$_i$, T$_s$, and G$_i$ from the login and authentication phases:

  User → Server: $\{\text{AID}_i, D_i, E_i, F_i, T_i\}$
  Server → User: $\{F_i, G_i, T_s\}$.
The attacker can guess or steal it easily from an unsecure public channel. Then the attacker could check with guessing identity ID', to hold the following equation:
\[ h(\text{ID}') \| (\text{AID} \oplus \text{ID}') \| F_s \| T_s) = G_s. \]

In general, the identity was named by the user and the length of the identity is between 6 - 12 characters (26 alphabets and 10 digits). Therefore, the probability of guessing the identity is \( 1/(36^{12}) \) in the worse cases.

- **User Impersonation**

  After knowing the user identity U_i, AID_i, and D_i by the guessing identity attack, the attacker could impersonate the user U_i as follows:

  **Steps 1 & 2:** The attacker by passes Steps 1 and 2 of the login phase.

  **Step 3:** The attacker randomly generates a number \( \alpha' \) and \( n_i \), and computes the following parameters:
  \[
  h(A_i) = \text{AID}_i \oplus \text{ID}_i, \\
  E'_s = \alpha' P, \\
  F'_s = h(\text{ID}_i \| h(A_i) \| E'_s \| T_s),
  \]
  where \( T_s \) is the current timestamp of the attacker.

  **Step 4:** The attacker sends the login request message \{AID_i, D_i, E'_s, F'_s, T_s\} to the server S.

  Next, the server authenticates the identity of the attacker (an impersonated user) and establishes a shared session key for the subsequent secret communication. These steps of the authentication phase are shown as follows:

  **Step 1.** The server S verifies whether \( T_s - T_a \leq \Delta T \), where \( T_s \) is the time of receiving the login request message and \( \Delta T \) is a valid time threshold. If both conditions are true, the server S continues to execute **Step 2**; otherwise, the server S rejects the login request.

  **Step 2.** The server S computes the following parameters:
  \[
  A'_i = D_i \oplus x \oplus h(x), \\
  \text{ID}'_i = \text{AID}_i \oplus h(A'_i), \\
  F'_s = h(\text{ID}'_i \| h(A'_i) \| E'_s \| T_s).
  \]

  The server S then compares whether \( F'_s \) is equals \( F_s \). If this holds, the server S confirms that the attack is a legal user U_i and the login request is accepted; otherwise, the server S rejects the login request.

  **Step 3.** Next, the server S randomly generates a number \( \beta \) and computes the following parameters:
  \[
  F_s = \beta P, \\
  G_i = h(\text{ID}'_i \| h(A'_i) \| F_s \| T_s),
  \]
  where \( T_s \) is the current timestamp of the server S.

  **Step 4.** The server S sends the mutual authentication message \{F_s, G_i, T_s\} to the user U_i.

  **Step 5.** Upon receiving the message \{F_s, G_i, T_s\} from the S, the attacker checks the validity of \( T_s \). If \( T_s - T_a \leq \Delta T \), where \( T_s \) is the time of receiving the mutual authentication message, the attacker continues to perform **Step 6**.

  **Step 6.** The attacker computes \( G'_s = h(\text{ID}'_i \| h(A'_i) \| F_s \| T_s) \), then checks whether \( G'_s \) is equal to the received \( G_s \). If this holds, the validity of the server S is authenticated.

  **Step 7.** Finally, the attacker and the server S construct a shared session key:
  \[
  sk = \alpha' \beta P
  \]
  to ensure the secret communication.

4. **Conclusion**

In this paper, we have shown that the weaknesses of Moon et al.’s Scheme. Their scheme could not against the guessing identity attack and the user impersonation attack. In general, the probability of guessing the identity is \( 1/(36^{12}) \) in the worse cases, if the user selects his/her identity with 12 characters (26 alphabets and 10 digits).
5. Acknowledgments

This work was partially supported by the Ministry of Science and Technology, Taiwan, under grant MOST 106-3114-E-005-001, MOST 106-2221-E-468-002, and MOST 106-2221-E-845-001.

6. References

[1] AbdElElmaen A D 2018 *Int. J. Electron. Inf. Engineering* **8** 40
[2] Al-Shaikhly M H R, El-Bakry H M and Saleh A A 2018 *Int. J. Electron. Inf. Engineering* **8** 96
[3] Nabi F and Nabi M M 2017 *Int. J. Electron. Inf. Engineering* **6** 40
[4] Chen T Y, Lee C C, Hwang M S and Jan J K 2013 *J. Supercomputing* **66** 1008
[5] Chen T Y, Ling C H and Hwang M S 2014 *Proc. IEEE Workshop on Electronic. Computer and Applications* (IEEE) p 771
[6] Wei J, Liu W and Hu X. 2016 *Int. J. Netw. Secur.* **18** 782
[7] Hwang M S, Lo J W, Liu C Y and Lin S C 2005 *Pakistan J. Appl. Sci.* **5** 99
[8] Wijayanto H and Hwang M S 2015 *Int. J. Netw. Secur.* **17** 160
[9] Anwar N, Riadi I and Luthfi A 2016 *Int. J. Electron. Inf. Engineering* **4** 71
[10] Tsai C Y, Pan C S and Hwang M S 2017 *Advances in Intelligent Systems and Computing, Recent Developments in Intelligent Systems and Interactive Applications* (Springer) **541** 194
[11] Yang L, Ma J F and Jiang Q 2012 *Int. J. Netw. Secur.* **14** 156
[12] Li C T and Hwang M S 2010 *Int. J. Innovative Computing Inf. Control* **6**, 2181
[13] Feng T H, Ling C H and Hwang M S 2014 *Proc. 2nd Congress on Computer Science and Application* p 111
[14] Feng T H, Ling C H and Hwang M S 2014 *Int. J. Netw. Secur.* **16** 318
[15] Pan H T, Pan C S, Tsaur S C and Hwang M S 2017 *Proc. Int. Conf. on Computational Intelligence and Security* p 590
[16] Ling C H, Chao W Y, Chen S M and Hwang M S 2015 *Advances in Engineering Research* (Springer) **15** 981
[17] He D, Zhao W and Wu S 2013 *Int. J. Netw. Secur.* **15** 282
[18] Amin R 2016 *Int. J. Netw. Secur.* **18** 172
[19] Mohan N B M, Chakravarty A S N and Ravindranath C 2018 *Int. J. Netw. Secur.* **20** 217
[20] Ghosh D, Li C and Yang C 2018 *Int. J. Netw. Secur.* **20** 414
[21] Hwang M S, Lee C C and Tang Y L 2001 *Informatica* **12** 297
[22] Chang T Y, Yang W P and Hwang M S 2005 *Computers & Math. with Applications* **49** 703
[23] Lee C C, Liu C H and Hwang M S 2013 *Int. J. Netw. Secur.* **15** 64
[24] Yang C C, Chang T Y, Li J W and Hwang M S 2003 *IEICE Trans. Commun.* **E86-B** 2178
[25] Yang C C, Chang T Y and Hwang M S 2003 *Informatica* **14** 551
[26] Zhuang X, Chang C C, Wang Z H and Zhu Y 2014 *Int. J. Netw. Secur.* **16** 271
[27] Liu C W, Tsai C Y and Hwang M S 2017 *Advances in Intelligent Systems and Computing, Recent Developments in Intelligent Systems and Interactive Applications* (Springer) **541** 188
[28] Thandra P K, Rajan J and Murty S A V 2016 *Int. J. Netw. Secur.* **18** 362
[29] Pan C S, Tsai C Y, Tsaur S C and Hwang M S 2016 *Proc. Int. Conf. on Systems and Informatics* p 732
[30] Prakash A 2014 *Int. J. Netw. Secur.* **16** 65
[31] Zhu H and Zhang Y 2017 *Int. J. Netw. Secur.* **19** 487
[32] Wu M, Chen J and Wang R 2017 *Int. J. Netw. Secur.* **19** 785
[33] Feng T H, Chao W Y and Hwang M S 2014 *Proc. Int. Conf. on Future Commun. Technology and Engineering* p 103
[34] Hou G and Wang Z 2017 *Int. J. Netw. Secur.* **19** 904
[35] Moon J, Lee D, Jung J and Won D 2017 *Int. J. Netw. Secur.* **19** 1053
[36] Liu Y, Chang C C and Chang S C 2017 *Int. J. Netw. Secur.* **19** 1