ハイブリットセフト攻撃に耐性のある相互認証方式

國貞 勇人† 伊沢 亮一‡ 森井 昌克†

†神戸大学大学院工学研究科
657-8501 兵庫県神戸市灘区六甲台町 1-1
{kunisada@stu., mmorii@}kobe-u.ac.jp

‡独立行政法人情報通信研究機構
184-8795 東京都小金井市貫井北町 4-2-1
isawa@nict.go.jp

あらまし ワンタイムパスワード認証方式とは認証に用いる通信データを認証毎に変化させる方式である．公開鍵暗号系を利用しない方式が多く，計算量や回路規模を小さくする目的で盛んに研究がなされている．しかし従来の方式ではハイブリットセフト攻撃など何らかの攻撃に対して脆弱であり，いまだ安全な方式が提案されていない．ハイブリットセフト攻撃とは，攻撃者が認証サーバや通信路から秘密情報を盗めるという仮定のもとでユーザになりすます攻撃である．サーバの秘密情報が漏洩することを仮定しているため，ハイブリットセフト攻撃を防ぐことは難しい．本稿ではハイブリットセフト攻撃をはじめとする全ての既知の攻撃に耐性をもつ新たなワンタイムパスワード認証方式を提案する．一般的な手法ではサーバとユーザが共通の秘密情報を保持する．一方で，提案方式ではユーザのみが知りうる秘密情報を与えることでハイブリットセフト攻撃への耐性をもたせる．

Mutual Authentication Scheme against Hybrid Theft Attacks

Yuto KUNISADA† Ryoichi ISAWA‡ Masakatu MORII†

†Graduate School of Engineering, Kobe University
1-1 Rokkodai-Cho, Nada-Ku, Kobe-Shi, Hyogo, 657-8501, JAPAN
{kunisada@stu., mmorii@}kobe-u.ac.jp

‡National Institute of Information and Communications Technology
4-2-1 Nukui-Kitamachi, Koganei-Shi, Tokyo 184-8795, JAPAN
isawa@nict.go.jp

Abstract One-Time Password (OTP) schemes are authentication schemes which change the communication data used for authentication in each session. Reserch for it has actively been made in order to reduce computational costs and circuit boards, and most schemes are not based on the public-key encryption system. However, all conventional schemes are vulnerable to various attacks, particularly the hybrid theft attack, and a secure scheme has not been proposed. In the hybrid theft attack, an adversary impersonate a legal user under the assumption she/he can steal secret data from an authentication sever or communication data. Considering the assumption, it is difficult to prevent such an attack. In this paper, we propose a novel OTP scheme against the hybrid theft attack and all existing attacks. In general schemes, the sever and a user share the same secret data. In the proposed scheme, since some of the secret data are only known to the user, the scheme is secure for against hybrid theft attack.

1 Introduction

Recently, we can use many services on the Internet. However, attacks against users and servers such as user spoofings and deny of service attacks have also increased. To prevent such attacks, secure authentication schemes between an authetication server and the users are required. There are three types of authentication scheme. They are a static password authentication like basic and digest access authentications[1], public-key certificates, and OTP authentication. Static password authentication is known as insecure in case users use short and simple passwords. Such passwords can be easily cracked by an adversary,

for example, via a brute force attack. On the other hand, Public-key certificates are highly secure schemes. However, it is difficult to implement them in low-spec mobile devices because they require high computational costs. In contrast, the computational costs of OTP schemes are lower than those of public-key certificates. OTP schemes are typically based on a symmetric encryption funciton or a one-way hash function. Furthermore, OTP schemes generate a different password for every authentication session.

There are three types of OTP schemes: 1) schemes based on time synchronization between the authentication server and the user[2]-[3], 2) schemes that use a mathematical algorithm to generate the new OTP based on a random number called a challenge[4]-[7], and 3) schemes that use a mathematical algorithm to generate a new OTP based on the previous session's OTP. In this study, we focus on the third type because such schemes change not only the communication data included OTP, but also the secret data stored by the server and the user for every authentication session. Therefore, if we use the third type for authentication, we can sefely use our biometric information as the user's initial secret data.

Three common attacks are the stolen verifier attack(SV attack)[8], the theft attack[9], and the hybrid theft attack[9]. In the SV attack, an adversary steals verification data from an authentication server in order to impersonate a legal user. In the theft attack, an adversary steals all the secret data. Furthermore, in the hybrid theft attack, an adversary can obtain communicaton data transmitted between the server and the user. Here, the verification data do not include secret keys used with XOR operation or an encryption funciton. The secret data denote both the verification data and the secret keys. Since the adversary can obtain the sever's data, it is difficult to prevent such attacks, especially the hybrid theft attack. The adversary's impersonation of a legal user, for example, in interest banking, could have serious repercussions. For secure internet services, we require robust authentication schemes that prevent such attacks.

In this paper, we propose a novel OTP scheme against the hybrid theft attack [1]. First, we review conventional schemes and specify the requirements for a secure OTP scheme. The proposed scheme has three advantages: 1) it is secure against all existing attacks, 2) it is based on only a one-way hash function, and 3) it is a mutual authentication scheme. The proposed scheme is more secure because of the first advantage. In addition, if we fabricate a circuit board for our scheme, it would require a smaller area than the schemes combined with an encryption function and a one-way hush function.

The remainder of this paper is organized as follows. In Section 2, we define ten types of attacks on a OTP scheme. In Section 3 and 4, we review existing schemes and specify the requirements for a secure scheme. In Section 5 and 6, we propose a novel scheme, and we verify its security against existing attacks. Finally, in Section 7, we conclude the paper.

2 Attacks on OTP Schemes

There are nine types of attacks on OTP schemes. The replay attack, forgery attack, impersonation attack, and DoS attack have been defined in [12]. The SV attack has been defined in [8]. The theft attack and the server modification attack have been defined in [9]. The SV DoS attack has been defined in [13]. In this paper, we regard the SV DoS attack as the theft DoS attack because they are regarded as a same attack. In addition, we define the hybrid theft attack and the server impersonation attack in this paper.

Replay attack: An adversary obtains communication data transmitted between the server and user in previous authentication sessions. In the current authentication session, she/he replaces all or a specific part of the communication data with previous session's data. If it succeeds, she/he can impersonate a legal user in the current authentication session.

Forgery attack: An adversary modifies the communication data in the current authentication session. If it succeeds, she/he can impersonate a legal user in the next authentication session.

Impersonation attack: An adversary uses the replay attack and the forgery attack to impersonate a legal user.

[1] We have proposed this scheme in FIT2011[10]. In this paper, we clearly distinguish this scheme from conventional schemes, and we elaborate on the security of the proposed scheme using BAN logic[11].

Sever Impersonation attack: An adversary uses the impersonation attack to the server to impersonate a legal sever of a specific user. One-way authentication schemes are vulnerable to this attack.

Denial of Service (DoS) attack: An adversary uses the replay attack or the forgery attack to alter the server's and user's secret data. If it succeeds, the server and the user cannot authenticate each other in the next authentication session.

Stolen Verifier (SV) attack: An adversary steals verification data from the server in the current or previous authentication sessions. Here, the verification data does not include secret keys used with XOR operation or an encryption function. She/He generates communication data by using the stolen data from the server and sends them to the server. If it succeeds, she/he can impersonate a legal user in the next authentication session.

Theft attack: An adversary steals secret data from the server in the current or previous authentication sessions. Here, the secret data mean both the verification data and the secret keys. She/He generates communication data by using the stolen data and sends them to the server. If it succeeds, she/he impersonates a legal user in the next authentication session.

Hybrid Theft attack: An adversary uses the theft attack and the impersonation attack to impersonate a legal user.

Theft DoS attack: An adversary uses the hybrid theft attack to alter the server's or user's secret data. If it succeeds, the server and the user cannot authenticate each other in the next authentication session.

3 Conventional Schemes

In 2002, Tsuji et al. proposed SAS-2 (simple and secure password authentication protocol, ver.2), which reduces the operation time of a one-way hash function at the server and the user[14], [12]. SAS-2 entails lower computational costs than the revised SAS[15], while providing an equivalent level of security. However, Tsuji et al. assumed that adversaries cannot steal the server's secret data.

Chien et al. proposed ROSI (robust and simple authentication protocol), which prevents the SV attack[16]. They assumed that the server possesses the secret key that cannot be stolen by an adversary. Tsuji et al. showed that ROSI is vulnerable to the hybrid theft attack, and they proposed 2GR (two-gene-relation password authentication protocol)[9] in order to prevent the hybrid theft attack. However, Lin et al. found that 2GR is vulnerable to the impersonation attack[17]. In the impersonation attack, an adversary tries to impersonate a legal user by using the intercepted communication data. 2GR is vulnerable to such an attack because it is a one-way authentication scheme (user to server) and the server updates the new verifier without any integrity check. Kuo et al. also indicated that 2GR is vulnerable to the impersonation attack, and they proposed an improved scheme to prevent the hybrid theft attack[18]. Unfortunately, Kim et al. showed that Kuo et al.'s scheme remains vulnerable to the hybrid theft attack[19]. Although Kim et al. proposed a new scheme, they did not verify that their scheme can prevent the hybrid theft attack. In order to prevent the hybrid theft attack, Tsuji et al. proposed SAS-X(2)[20]. However, it is vulnerable to the DoS attack[21]. Thus, all existing schemes have certain vulnerabilities.

4 Requirements for Security of OTP Schemes

In this section, we concretely explain why SAS-2 is vulnerable to the theft attack, and how ROSI prevents the theft attack. Similarly we also concretely explain why ROSI is vulnerable to the hybrid theft attack, and how SAS-X(2) prevents the hybrid theft attack. To explain the above, we briefly summarize the protocol of each scheme. In addition, we discuss the requirements for secure OTP schemes against various attacks.

4.1 Weakness of SAS-2

We explain why SAS-2 is vulnerable to the theft attack, and we summarize the protocol of SAS-2. In the ith (current) authentication session, an authentication server stores the current verifier p_i, which is not protected. A user sends p_i and the $(i+1)$th verifier p_{i+1} to the server. Having received them, the server compares the received p_i with the stored p_i. If they match, the server authenticates the user

and stores p_{i+1} as the next verifier. SAS-2 is vulnerable to the theft attack because the server does not protect the stored p_i, and it does not verify p_{i+1} before storing p_{i+1}. In the theft attack, an adversary steals p_i from the server. She/he sends the stolen p_i and randomly selected p'_{i+1} to the server. The server is convinced that she/he is a valid user on the basis of the received p_i, and it stores p'_{i+1} as the next verifier. Therefore, the adversary impersonates a legal user in SAS-2. ROSI prevents the theft attack. In ROSI, the server stores $h(p_i)$ instead of p_i, where $h(p_i)$ denotes the hash value of p_i. The user sends p_i to the server for authentication. Having received p_i, the server calculates $h(p_i)$ and compares the calculated $h(p_i)$ with the stored $h(p_i)$. An adversary cannot impersonate a legal user because she/he cannot obtain p_i, even if she/he steals $h(p_i)$ from the server.

4.2 Weakness of ROSI

ROSI is vulnerable to the hybrid theft attack. In ROSI, the server stores $h(p_i)$ and q_i, where q_i is used for protecting communication data with XOR operation. Here, the server does not protect the stored q_i. Thus, ROSI is vulnerable to the hybrid theft attack. For authentication, a user sends p_i and the next verifier $h(p_{i+1})$ protected with q_i to the server. Having received them, the server extracts p_i and $h(p_{i+1})$ using the stored q_i. The server calculates $h(p_i)$ and compares it with the stored $h(p_i)$. If they match, the server authenticates the user and stores $h(p_{i+1})$ as the next verifier. Suppose that an adversary steals q_i from the server, and intercepts the communication data p_i and $h(p_{i+1})$ protected with q_i. The adversary extracts p_i using the stolen q_i. She/he randomly selects p'_{i+1} and calculates $h(p'_{i+1})$. Then, she/he sends p_i and $h(p'_{i+1})$ protected with q_i to the server. Having received them, the server extracts p_i and $h(p'_{i+1})$ using the stored q_i. The server calculates $h(p_i)$ and compares it with the stored $h(p_i)$. Since they match, the server is convinced that the adversary is valid user. Then, the server stores $h(p'_{i+1})$ as the next verifier. In the $(i+1)$th authentication session, the adversary can send p'_{i+1} to the server, and impersonate a legal user.

SAS-X(2) prevents the hybrid theft attack. In SAS-X(2), the server stores $h(p_i)$ and $E_{k_i}(p_i)$, where $E_{k_i}(p_i)$ denotes p_i encrypted with a secret key k_i. For authentication, the user sends p_i, k_i, $h(p_{i+1})$, and $E_{k_{i+1}}(p_{i+1})$ to the server. Having received them, the server calculates $h(p_i)$ and compares it with the stored $h(p_i)$. If they match, the server decrypts the stored $E_{k_i}(p_i)$ using the received k_i. The server compares the obtained p_i with the received p_i. If they match, the server authenticates the user and stores $h(p_{i+1})$ and $E_{k_{i+1}}(p_{i+1})$ as the next verifier. The main concept of SAS-X(2) is that the user does not send p_{i+1} and k_{i+1}, and the server does not store them. Because an adversary cannot obtain k_{i+1}, she/he cannot create p'_{i+1}, which equals the decrypted value of $E_{k_{i+1}}(p'_{i+1})$. Even if she/he randomly modifies $h(p_{i+1})'$ and $E_{k_{i+1}}(p_{i+1})'$, the server can detect that $h(p_{i+1})'$ and $E_{k_{i+1}}(p_{i+1})'$ are modified in the $(i+1)$th (next) authentication session. Therefore, SAS-X(2) is secure against the hybrid theft attack. However, SAS-X(2) is vulnerable to the DoS attack. If an adversary randomly modifies $E_{k_{i+1}}(p_{i+1})$ as $E_{k_{i+1}}(p_{i+1})'$, the server directly stores it as the next verifier. Since the server does not verify $E_{k_{i+1}}(p_{i+1})$, the adversary can alter $E_{k_{i+1}}(p_{i+1})$.

4.3 Requirements for Secure OTP Schemes

We can easily prevent the replay attack, forgery attack, impersonation attack, and DoS attack under the assumption that any adversaries cannot steal secret data from the server. Let us specify the requirement for preventing the hybrid theft attack: any adversary cannot create all the user's secret data, even if she/he obtains the server's secret data and the communication data. Naturally, a OTP scheme that is secure against the hybrid theft attack is secure against both the SV attack and the theft attack. Furthermore, considering which data an adversary can obtain and what an adversary do, the hybrid theft attack includes the replay attack, forgery attack and impersonation attack. Similarly, Theft DoS attack includes DoS attack.

In order to prevent the server impersonation attack, we have to design a mutual authentication scheme. Tsuji et al. claimed that an adversary can impersonate the server in mutual authentication schemes under the assumption that the adversary can steal secret data from the user[9]. Nonetheless, one-way

Table 1: List of symbols used in this paper

U	a user who requests the server to authenticate her/himself
S	the authentication server
AD	an adversary
ID	a user's identification
PW	a user's password
$h(x)$	the hash value of the input data x
$\check{x}$	used when U or S compares certain data with $\check{x}$
x', x''	used when A randomly selects instead of a valid x, or calculates certain data y', y'' from x', x''.
X $\Rightarrow$ Y:Z	X sends Z to Y through a secure channel
X $\rightarrow$ Y:Z	X sends Z to Y through a insecure channel
$\|\|$	a concatenation
$\oplus$	XOR operation

authentication schemes are vulnerable to the server impersonation attack under any circumstances. In addition, mutual authentication schemes can adopt the same precaution as one-way authentication schemes. That is, the user receives services doubting the server, even if the user authenticates the server. From the above, we can conclude the OTP scheme against server impersonation attack, hybrid theft attack and theft DoS attack is against all well-known attacks above.

5 Proposed Schemes

Table 1 shows the symbols we use in this paper. The proposed scheme consists of two phases: the registration phase and the authentication phase. In order to prevent the hybrid theft attack, the proposed scheme satisfies the requirement specified in 4.3. In the ith session, U does not send the next verifier A_{i+1} to S, and S does not store it in the ith authentication session. Because U creates A_{i+1} by using a random number Q_i in the $(i-1)$th authentication session, U will never send Q_i to S, and S does not store it.

5.1 Registration Phase

Let us illustrate the protocol as follows.

1. U inputs ID and PW.
2. U $\Rightarrow$ S: ID.
3. S generates three random numbers R_0, R_{-1}, and F_0.
4. S $\Rightarrow$ U: R_0, R_{-1}, F_0.
5. U calculates the following data.
   $A_1 = h(ID||PW||F_0)$
   $F_1 = h(A_1)$
   $Q_1 = h(PW||R_{-1})$
   $A_2 = h(ID||Q_1||F_1)$
   $F_2 = h(A_2)$
   $Q_2 = h(Q_1||R_0)$
   $V_1 = h(A_1||F_2)$
6. U stores ID, Q_2, A_1, F_1, A_2, and F_2.
7. U $\Rightarrow$ S: F_1, V_1.
8. S stores ID, F_1, and V_1.

5.2 Authentication Phase

Let us illustrate the protocol as follows. U stores ID, Q_{i+1}, A_i, F_i, A_{i+1} and F_{i+1}. S stores ID, F_i and V_i.

1. U calculates the following data.
   $A_{i+2} = h(ID||Q_{i+1}||F_{i+1})$
   $F_{i+2} = h(A_{i+2})$
   $F_{i+1} \oplus F_i$
   $A_i \oplus F_{i+1}$
   $V_{i+1} = h(A_{i+1}||F_{i+2})$
   $h(F_i||V_{i+1})$
2. U $\rightarrow$ S: ID, $F_{i+1} \oplus F_i$, $A_i \oplus F_{i+1}$, V_{i+1}, $h(F_i||V_{i+1})$.
3. **S obtains F_{i+1} and A_i, and verifies their validity as follows.**
   S extracts F_{i+1} from the received $F_{i+1} \oplus F_i$ by using the stored F_i, and extracts A_i from the received $A_i \oplus F_{i+1}$ by using the obtained F_{i+1}. S culculates $\check{F}_i = h(A_i)$ by using the obtained A_i, and compares $\check{F}_i$ with the stored F_i. If they match, S is convinced that the received F_{i+1}, F_i, and A_i have not been modified; otherwise, S terminates this authentication session.
4. **S tries to authenticate U using V_i as follows.**
   S calculates $\check{V}_i = h(A_i||F_{i+1})$, and compares $\check{V}_i$ with the stored V_i. If they match, S authenticates U; otherwise, S detects the theft DoS attack and terminates this authentication session.
5. **S verifies the validity of V_{i+1} as follows.**
   S calculates $h(F_i||V_{i+1})$ by using the stored F_i and the received V_{i+1}. Then S compares it with the received $h(F_i||V_{i+1})$. If they match, S is convinced that V_{i+1} has not been modified; otherwise, S terminates this authentication session.

Table 2: List of notation used in BAN logic

P $\mid\equiv X$	P acts as if X is true
P $\triangleleft X$	P receives X
P $\mid\sim X$	P transmitted X
$\sharp(X)$	X has not previously been sent
P $\mid\Rightarrow X$	P can determine X
P $\overset{K}{\rightleftharpoons}$ Q	P and Q communicate with shared key K
P $\overset{X}{\rightleftharpoons}$ Q	X is a secret data known only to P and Q
$\{X\}_K$	X is encrypted with key K

6. S calculates $h(R_i||F_i)$ by using a random number R_i, and stores F_{i+1} and V_{i+1}.
7. S $\rightarrow$ U: R_i, $h(R_i||F_i)$.
8. **U tries to authenticate S as follows.** U calculates $h(R_i||F_i)$ by using the received R_i and the stored F_i. Then U compares it with the received $h(R_i||F_i)$. If they match, U authenticates S; otherwise, U terminates this authentication session.
9. U calculates $Q_{i+2} = h(Q_{i+1}||R_i)$ and stores Q_{i+2}, A_{i+2} and F_{i+2}.

6 Security Analysis of The Proposed Scheme

In this section, we evaluate the security of the proposed scheme against various attacks by the BAN logic[11]. Table 2 shows the notation we use in BAN logic. P and Q are two entities, X and K are data like a key, secret data, or something like that.

Now, we describe the proposed scheme's ith authentication session as follows.

Message 1 U $\rightarrow$ S : $\{A_i, F_{i+1}\}_{F_i}$

1) S $\mid\equiv$ (U $\overset{F_i}{\rightleftharpoons}$ S);
2) S $\triangleleft$ ($\{A_i, F_{i+1}\}_{F_i}$);
3) S $\mid\equiv$ U $\mid\sim$ (A_i, F_{i+1});
4) S $\mid\equiv \sharp(A_i, F_{i+1})$;
5) S $\mid\equiv$ U $\mid\equiv$ (A_i, F_{i+1});
6) S $\mid\equiv$ U $\mid\Rightarrow$ (A_i, F_{i+1});
7) S $\mid\equiv$ (A_i, F_{i+1});

Message 2 S $\rightarrow$ U : $\{R_i\}_{h(),F_i}$

8) U $\mid\equiv$ (U $\overset{h()}{\leftrightarrow}$ S, U $\overset{F_i}{\rightleftharpoons}$ S);
9) U $\triangleleft$ ($\{R_i\}_{h(),F_i}$);
10) U $\mid\equiv$ S $\mid\sim$ (R_i);
11) U $\mid\equiv \sharp(R_i)$;
12) U $\mid\equiv$ S $\mid\equiv$ (R_i);
13) U $\mid\equiv$ S $\mid\Rightarrow$ (R_i);
14) U $\mid\equiv$ (R_i);

In 4.3, we conclude the OTP scheme against server impersonation attack, hybrid theft attack and theft DoS attack is against all well-known attacks. Therefore, we describe the security of the proposed scheme against the three attacks.

Server Impersonation attack

In server impersonation attack, an adversary can obtain communication data transmitted between the server and the user in previous authentication sessions and can modify the communication data in the current authentication session. By using these data, she/he tries to impersonate a legal server.

In the ith authentication session, AD can obtain ID, $F_{i+1} \oplus F_i$, $A_i \oplus F_{i+1}$, V_{i+1}, $h(F_i||V_{i+1})$, R_i and $h(R_i||F_i)$. In the $(i+1)$th session, AD tries to impersonate S to be authenticated by U as follow.

Message 2' AD $\rightarrow$ U : $\{R_{i+1}\}_{h(),F_{i+1}}$

8') U $\mid\equiv$ (U $\overset{h()}{\leftrightarrow}$ AD, U $\overset{F_{i+1}}{\rightleftharpoons}$ AD);
9') U $\triangleleft$ ($\{R_{i+1}\}_{h(),F_{i+1}}$);
10') U $\mid\equiv$ AD $\mid\sim$ (R_{i+1});
11') U $\mid\equiv \sharp(R_{i+1})$;
12') U $\mid\equiv$ AD $\mid\equiv$ (R_{i+1});
13') U $\mid\equiv$ AD $\mid\Rightarrow$ (R_{i+1});
14') U $\mid\equiv$ (R_{i+1});

In the server authentication of the $(i+1)$th session (Message 2'), AD selects and sends random number R'_{i+1} instead of legal random number R_{i+1} used for server authentication. Then AD sends it encrypting with F'_{i+1} instead of legal F_{i+1}. Assumption 8') states that AD has known what hash function U and S use, and secret data F_{i+1} shared by U and S. However, AD cannot obtain F_{i+1} in the previous sessions. Thus assumption 8') does not hold. Accordingly, the proposed scheme is secure against the server impersonation attack.

Theft DoS attack

In theft DoS attack, an adversary can obtain secret data from the server in the current or previous authentication sessions. In addition, she/he can also obtain communication data transmitted between the server and the

user in previous authentication sessions and modify the communication data in the current authentication session. She/He tries to prevent the authentication using these data.

To succeed (theft) DoS attack, AD has to modify the communication data in order to alter S's or U's secret data. Suppose that AD has stolen F_i from S in the ith session. AD randomly modifies V_{i+1} and $h(F_i||V_{i+1})$ as $V'_{i+1} = h(A'_{i+1}||F'_{i+2})$ and $h(F_i||V'_{i+1})$ using A'_{i+1} and F'_{i+2}, respectively. S updates V_i to the received V'_{i+1}. In the $(i+1)$th session, S tries to authenticate U as follow.

Message 1' U $\rightarrow$ S : $\{A'_{i+1}, F'_{i+2}\}_{F_{i+1}}$

1') S $|\equiv$ (U $\overset{F_{i+1}}{\rightleftharpoons}$ S);
2') S $\triangleleft$ $(\{A'_{i+1}, F'_{i+2}\}_{F_{i+1}})$;
3') S $|\equiv$ U $|\sim$ (A'_{i+1}, F'_{i+2});
4') S $|\equiv \sharp(A'_{i+1}, F'_{i+2})$;
5') S $|\equiv$ U $|\equiv$ (A'_{i+1}, F'_{i+2});
6') S $|\equiv$ U $|\Rightarrow$ (A'_{i+1}, F'_{i+2});
7') S $|\equiv$ (A'_{i+1}, F'_{i+2});

In the user authentication of the $(i+1)$th session (Message 1'), U sends A_{i+1}, F_{i+2} instead of illegal secret data A'_{i+1}, F'_{i+2} used for user authentication. Then U sends them encrypting with F_{i+1}. Assumption 1') states that U and S share secret data F_{i+1}. It holds because AD did not modifies F_{i+1}. Assumption 4') states that A'_{i+1} and F'_{i+2} are fresh numbers not used in previous sessions. It also holds because AD selected A'_{i+1}, F'_{i+2} in the ith session, and they are not used for authentication in the ith session. Assumption 6') states that S believes U has jurisdiction over A'_{i+1}, F'_{i+2}. However, they were selected by AD. Thus, assumption 6') does not hold. Accordingly, in the $(i+1)$th authentication session, S can detect the theft DoS attack at step 4 in the authentication phase. However, S cannot authenticate U; hence, S terminates the authentication session. **Here, let us introduce a variation for S to continue the authentication as follows.** Suppose that S and U have the current and previous secret data. When S detects the theft DoS attack, S and U use the previous secret data. Having authenticated each other, they update their own previous secret data to the current secret data. Thus, the proposed scheme is secure against the theft DoS attack.

Hybrid Theft attack

In OTP schemes, AD can certainly impersonate a legal user only once in the ith authentication session as follows. AD intercepts the communication data and directly sends them to S. Because S receives valid data, S is convinced that AD is the legal user. Similarly, in the hybrid theft attack, AD can certainly impersonate a legal user only once. The question we should consider is whether AD can impersonate U in the $(i+1)$th authentication session.

In the proposed scheme, S tries to authenticate U to check $V_{i+1} = h(A_{i+1}||F_{i+2})$ using received A_{i+1} and F_{i+2} in the $(i+1)$th session. In order to impersonate U, AD has to obtain A_{i+1} and F_{i+2}. Suppose that AD steals F_i from S in the ith session. Then, AD intercepts the communication data, and extracts A_i and F_{i+1} from the intercepted data using the stolen F_i. In the $(i+1)$th session, AD tries to impersonate U to be authenticated by S as follow.

Message 1" AD $\rightarrow$ S : $\{A_{i+1}, F_{i+2}\}_{F_{i+1}}$

1") S $|\equiv$ (AD $\overset{F_{i+1}}{\rightleftharpoons}$ S);
2") S $\triangleleft$ $(\{A_{i+1}, F_{i+2}\}_{F_{i+1}})$;
3") S $|\equiv$ AD $|\sim$ (A_{i+1}, F_{i+2});
4") S $|\equiv \sharp(A_{i+1}, F_{i+2})$;
5") S $|\equiv$ AD $|\equiv$ (A_{i+1}, F_{i+2});
6") S $|\equiv$ AD $|\Rightarrow$ (A_{i+1}, F_{i+2});
7") S $|\equiv$ (A_{i+1}, F_{i+2});

In the user authentication of the $(i+1)$th session (Message 1), AD sends A'_{i+1}, F'_{i+2} instead of secret data A_{i+1}, F_{i+2} used for user authentication, which is only known to U. Then AD sends them encrypting with stolen F_{i+1}. Assumption 1") states that AD has secret data F_{i+1} shared by U and S. It holds because AD has stolen the shared secret data F_{i+1} in the ith session. Assumption 4") states that A_{i+1} and F_{i+2} are fresh numbers not used in previous sessions. It also holds because AD can select random number A'_{i+1}, F'_{i+2} instead of legal A_{i+1} and F_{i+2}. Assumption 6") states that S believes AD has jurisdiction over A_{i+1}, F_{i+2}. Now, AD has to obtain $A_{i+1}(= h(ID||Q_i||$ $F_i))$ or Q_i to have jurisdiction over legal A_{i+1}. However, AD cannot obtain them because Q_i and A_{i+1} are not transmitted between U and S, and not stored by S in previous authentication sessions. Thus, assumption 6") does not hold. Accordingly, the proposed scheme is secure against the hybrid theft attack.

7 Conclusion

Existing authentication schemes are vulnerable to various attacks. In this paper, we proposed a novel OTP scheme for security against the hybrid theft attack. The proposed scheme is secure against all existing attacks. In addition, when we fabricate a circuit board for the proposed scheme, the area resources are smaller because the proposed scheme is based on only a one-way hash function. We can apply the proposed scheme to low-spec devices and hence design a secure authentication system.

References

[1] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart, "Http authentication: Basic and digest access authentication," Internet Request For Comments 2617, June 1999.

[2] J. Park, B. Park, J. Park, and J. cheol Ryou, "The improved one-time password algorithm using time," IEICE Trans. Inf. & Syst., vol.E85-D, pp.1962.1966, Dec. 2002.

[3] N. Sklavos and C. Efstathiou, "Securid authenticator: On the hardware implementation efficiency," Proc. 14th IEEE International Conference on Electronics, pp.589.592, Dec. 2007.

[4] M. Peyravian and N. Zunic, "Methods for protecting password transmission," Computers and Security, vol.19, no.5, pp.466.469, July 2000.

[5] T.H. Chen and W.B. Lee, "A new method for using hash functions to solve remote user authentication," Comput. Electr. Eng., vol.34, pp.53.62, Jan. 2008.

[6] F.Y. Yang, T.D.Wu, and M.H. Hsu, "Improvement of h ¨ olbl et al. user authentication protocol and password change protocol," Proc. the 11th International Conference on Information Integration and Webbased Applications & Services, no.5, pp.155.159, Dec. 2009.

[7] J.Y. Kim, H.K. Choi, and J.A. Copeland, "Further improved remote user authentication scheme," IEICE Trans. Fundamentals, vol.E94-A, no.6, pp.1426.1433, June 2011.

[8] C.M. Chen and W.C. Ku, "Stolen-verifier attack on two new strong-password authentication protocols," IEICE Trans. Commun., vol.E85-B, no.11, pp.2519.1521, Nov. 2002.

[9] T. Tsuji and A. Shimizu, "One-time password authentication protocol against theft attacks," IEICE Trans. Commun., vol.E87-B, no.3, pp.523.529, March 2004.

[10] R. Isawa, M. Morii, "One-Time Password Authentication Scheme to Solve Stolen Verifier Problem," Proc. of Forum on Information Technology 2011 (FIT2011), CD-ROM, Sep. 2011.

[11] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," Proc. Royal Soc. London A, vol.426, pp.233-271, 1989.

[12] T. Tsuji and A. Shimizu, "A one-time password authentication method for low spec machines and on internet protocols," IEICE Trans. Commun., vol.E87-B, no.6, pp.1594.1600, June 2004.

[13] Y. Nakayama, T. Tsuji, and A. Shimizu, "A one-time password authentication scheme resistant to dos attacks," IEICE Technical Report, OIS2008-74, vol.108, pp.51.56, Jan. 2009.

[14] T. Tsuji and A. Shimizu, "Simple and secure password authentication protocol, ver.2 (sas-2)," IEICE Technical Report, OIS2002-30, vol.102, pp.7.11, Sept. 2002.

[15] T. Kamioka and A. Shimizu, "The examination of the security of sas one-time password authentication," IEICE Technical Report, OFS2001-48, vol.101, pp.53.58, Nov. 2001.

[16] H.Y. Chien and J.K. Jan, "Robust and simple authentication protocol," Comput. J., vol.46, no.2, pp.193.201, Feb. 2003.

[17] C.L. Lin and C.P. Hung, "Impersonation attack on two-gene-relation password authentication protocol (2gr)," IEICE Trans. Commun., vol.E89-B, no.12, pp.3425.3427, Dec. 2006.

[18] W.C. Kuo and Y.C. Lee, "Attack and improvement on the one-time password authentication protocol against theft attacks," Proc. the Sixth International Conference on Machine Learning and Cybernetics, pp.1918.1922, Aug. 2007.

[19] M. Kim, B. Lee, S. Kim, and D. Won, "Weaknesses and improvements of a one-time password authentication scheme," International Journal of Future Generation Communication and Networking, vol.2, no.4, pp.29.38, Dec. 2009.

[20] T. Tsuji, T. Nakahara, and A. Shimizu, "A one-time password authentication method," IEICE Technical Report, OIS2005-83, vol.105, pp.23.28, Jan. 2006.

[21] K. Uo, Y. Shiraishi, and M. Morii, "Evaluation of a one-time password method to resist stolen-verifier attack," Proc. Computer Security Symposium 2006 (Japanese Edition), CD-ROM, pp. 8A-4, 6pages, Oct. 2006.