

## **Cryptanalysis and Improvements of a Timestamp-Based User Authentication Scheme for Wireless Communications**

<sup>1</sup>Jaewook Jung, <sup>2</sup>Yoonsung Choi, <sup>2</sup>Youngsook Lee and <sup>3</sup>Dongho Won

<sup>1</sup>LG Electronics Seocho R&D Campus, Yangjae-daero 11-gil 19, Seocho-gu, Seoul,  
Republic of Korea

<sup>2</sup>Department of Computer Science, Major of Cyber Security, Howon University, Gunsan-si,  
54058 Jeollabuk-do, Republic of Korea

<sup>3</sup>Department of Computer Engineering, Sungkyunkwan University, 2066 Seoburo, Suwon,  
16419 Gyeonggi-do, Republic of Korea

---

**Abstract:** Remote user authentication schemes universally utilize to communicate between authorized users and remote servers through an unsafe network. By the benefit of its simplicity and convenience, this method is commonly employed in many conditions such as sensor networks or remote host login systems. In recent years, a few schemes taking advantages of smart cards for remote user authentication have been proposed. Lately, proposed a timestamp-based user authentication scheme by smart cards. They insisted that their scheme could withstand password guessing attack on off-line. However, there are some weaknesses in scheme. First of all, their scheme could not prevent off-line guessing password attack, impersonation attack, stolen verifier attack and privileged insider attack. Also, their scheme could not notice the wrong passwords in login phase and make it unshielded to change the user's passwords in changing password phase. Moreover, their scheme could not able to guarantee the user's anonymity. In this study, we propose a modified timestamp-based user authentication scheme to resolve the aforementioned vulnerabilities. Our proposed scheme is based on the RSA cryptosystem and our analysis demonstrates that this scheme ensures the safety and is more efficient than the previous schemes.

**Key words:** Authorized, remote, login systems, timestamp, scheme, authentication

---

### **INTRODUCTION**

With the rapidly increasing use of internet services and telecommunications technologies, many users have utilized various electronic transactions via internet networks by the medium of an authentication protocol. Remote user authentication schemes are vital to ensure secure and efficient telecommunication. In the existing remote user authentication schemes, not only does a remote server verify a user's legitimacy over an unsafe communication channel but also a legal login user verifies the remote server regardless of the server's reliability for the mutual authentication purpose. In spite of these weaknesses, this method is widely used because of its efficiency and simplicity in many areas such as sensor network environments or remote host login systems. Since, Lamport (1981) has proposed a remote password authentication protocol for insecure channels in 1981 at first, many remote user authentication schemes (Yang and Shieh, 1999; Chan and Cheng, 2001; Fan *et al.*,

2002; Shen *et al.*, 2003; Liu *et al.*, 2008; Sun *et al.*, 2009; Awasthi *et al.*, 2011; Huang *et al.*, 2014; Ku *et al.*, 2003; Hwang and Ku, 1995; Hsu, 2004) have been proposed to address password authentication problems. Essentially, a secure and efficient password-based user authentication scheme should fulfill some security requirements and defend some different types of attacks. Based on previous studies (Lamport, 1981; Yang and Shieh, 1999; Chan and Cheng, 2001; Fan *et al.*, 2002; Shen *et al.*, 2003; Liu *et al.*, 2008; Sun *et al.*, 2009; Awasthi *et al.*, 2011; Huang *et al.*, 2014), we analyze them and classify some requirements in order to design an ideal timestamp-based user authentication scheme. The security requirements can be divided into two parts: Functionality [F1-F6] and attack resistance [A1-A8]. These requirements will be used to scrutinize the security of our proposed scheme in the section 6. Detailed security requirements are as follow:

**F1; User anonymity:** Anonymity is of increasing importance and is achieved when the user's identity is not disclosed to an unauthorized group.

**F2; Single registration:** This method means that if a user registers only once which allows the user to access the registration center.

**F3; Freely change password:** This method allows the users to change or update their passwords without communicating to the server.

**F4; Securely change password:** This method allows the users to securely change or update their password without communicate to the server.

**F5; Provide mutual authentication:** This method makes the login users and the remote servers to authenticate each other. The server verifies the user's legitimacy and the user checks the communicating server's validity.

**F6; No verification table:** This requirement means that it is no need to retain the password's table in the server. Also, timestamp-based user authentication schemes should withstand different types of attacks. The most typical attacks include:

**A1; Replay attacks:** An attacker intercepts data packets for the purpose of making use of that data in some manner. Typically, this type of attack connotes copying and possibly modifying the data in various ways before releasing it for the delivery to the intended receiver.

**A2; User impersonation attack:** An attacker pretends to be the registered user with the forged login message by using the secret or public information that is collected from the smart cards and the data packets.

**A3; Server impersonation attack:** An attacker pretends to be the legitimate server with the forged authentication message by using the secret or public information that is collected from the smart cards and the data packets.

**A4; Man-in-the-middle attack:** This attack means that an attacker intercepts the messages which were sent between the server and the user. Then, the attacker utilizes these intercepted messages in the valid time frame window.

**A5; Off-line password guessing attack:** An adversary tries to guess a password and eventually find out the exact password in an off-line environment by using the information stored in the smart card.

**A6; Stolen smart card attack:** An attacker robs a user's smartcard to take out the contents by the power

consumption attack and reverses the engineering techniques. Then, the attacker guesses the users password in an off-line manner form the extracted.

**A7; Privileged-insider attack:** A privileged-insider attack literally means the attack mounted by a malicious insider. The malicious insiders have a noticeable advantage over external adversaries because they have an authorized system admission and also may be familiar with the network design and system actions. Commonly, the malicious insiders want to obtain the users private information such as their passwords.

**A8; Stolen-verifier attack:** A stolen-verifier attack means that an attacker steals verifier tables stored in the data base of the server and the attacker directly uses this value in the guise of a legitimate user.

Yang and Shieh (1999) proposed a timestamp-based password remote authentication scheme using smart cards to remove the need for password tables or verification table at the server end. However, this scheme was found to be vulnerable to forged login attack by Chan and Cheng, (2001), Fan *et al.* (2002) and Shen *et al.* (2003) who all independently performed successful attacks. Shen *et al.* (2003) proposed an improved timestamp-based password authentication scheme to remove the weaknesses of (Yang and Shieh, 1999).

They claimed that their scheme resists forged login attack and provides mutual authentication. However, in Liu *et al.* (2008), pointed out that Shen *et al.* (2003) scheme is still vulnerable to forged login attack. To overcome this problem, they proposed a new improved time stamp-based authentication scheme based on nonce. Sun *et al.* (2009) showed that Liu *et al.* (2008) scheme did not resist the forged login attack as in previous version. Awashi *et al.* (2011) pointed out that (Shen *et al.*, 2003) suffered from lost smart card and the forged login attack and proposed an improved timestamp-based authentication scheme. Recently, Huang *et al.* (2014) pointed out that (Awasthi *et al.*, 2011) was vulnerable to impersonation attack and proposed an enhanced timestamp-based user authentication scheme with smart cards. Huang *et al.* (2014) claimed that their scheme can resist off-line password guessing attack and provide mutual authentication. However, after careful analysis, we find that Huang *et al.* (2014) scheme cannot resist off-line password guessing attack, user impersonation attack, stolen verifier attack and privileged insider attack. Besides, their scheme cannot detect the wrong password in login phase and make it insecure to changing the user's password in password change phase. Furthermore, their scheme fails to preserve user anonymity. In order to eliminate all the above

problems by Huang *et al.* (2014) scheme, we propose a modified timestamp-based user authentication scheme for wireless communications.

## MATERIALS AND METHODS

In this study, we give a brief introduction to the RSA cryptosystem (Rivest *et al.*, 1978) and one-way hash functions (Stallings, 2003), these are the security basis of Huang *et al.* (2014) scheme and our proposed scheme. Detailed information about the RSA cryptosystem and one-way hash functions can be found by Rivest *et al.* (1978) and Stallings (2003), respectively.

**RSA cryptosystem:** Rivest *et al.* (1978) proposed a public-key cryptosystem, namely the RSA cryptosystem. The RSA cryptosystem is the most commonly (frequently) used system to guarantee privacy and ensure digital data's confidentiality. The security of the RSA algorithm is based on the intractability of integer factorization. The principle of the RSA cryptosystem is described as follows:

**Key generation:** Choose two large prime numbers  $p$  and  $q$ . Compute their product  $n = p \cdot q$  and calculate  $\phi(n) = (p-1)(q-1)$  where,  $\phi$  is Euler function. After that choose an integer  $e$  where,  $1 < e < \phi$  and  $\gcd(e, \phi) = 1$ . Calculate integer  $d$  to where,  $1 < d < \phi$  and  $e \cdot d = 1 \pmod{\phi(n)}$ . Thus, we have public key pair and private key pair as  $(e, n)$  and  $(d, n)$ , respectively.

**Encryption:** To encrypt a plaintext  $M$ , the sender uses a public key  $(e, n)$ . Thus, the cipher text denoted by  $C$  is computed as  $C = M^e \pmod{n}$ . the sender can transfer his cipher text  $C$  to the receiver.

**Decryption:** The receiver receives the cipher text  $C$ . He decrypts it by using the private key  $(d, n)$ . For this, he performs the operation  $M = C^d \pmod{n}$ . Thus, the receiver can get the original plain text  $M$ .

**One-way hash function:** A hash function Stallings (2003)  $f: \{0, 1\}^* \rightarrow \{0, 1\}^n$  is a one-way function which deals with hash functions taking an arbitrary-length input  $x \in \{0, 1\}^*$  and outputs a fixed-length bit string, called the message digest or hash value  $f(x) \in \{0, 1\}^n$ . When considering cryptographic hash functions there are commonly three levels of security considered.

**One-way property:** From a given hash value  $y = f(x)$  and the given hash function  $f(\cdot)$  it is computationally infeasible to derive the input  $x$ .

Table 1: Notations

Notations	Descriptions
$U_i$	Remote user
$S$	Authentication server
KIC	Key Information Center
$ID_i, PW_i$	Identity and Password of $U_i$
$SID_i$	Smart card's identity corresponding to $ID_i$
$p, q$	Large prime numbers
$e, d$	Systems public key and private key
$n$	The modulus of RSA cryptosystem
$f(\cdot)$	One-way hash function
$r$	Random number
$T_e, T_s'$	Current timestamp of $U_i$
$T_s, T_s'$	Current timestamp of $S$
$\Delta T$	The maximum of transmission delay time

**Weak-collision resistance property:** For any given input  $x$ , finding any other input  $x'$  where,  $x' \neq x$  such that  $f(x') = f(x)$  is computationally infeasible.

**Strong-collision resistance property:** Finding a pair of inputs  $(x, x')$ , where,  $x \neq x'$  such that  $f(x) = f(x')$  is also computationally infeasible.

**Review of Huang *et al.* (2014) scheme:** In this study, we briefly review (Huang *et al.*, 2014). We describe each phase of Huang *et al.* (2014) scheme in sections 3.1-3.5 and Table 1 shows the notations used in the remainder of the study.

### Initialization phase:

- KIC generates two large primes  $p$  and  $q$  and computes  $n = p \cdot q$
- Choose two integers  $e$  and  $d$  such that  $e \cdot d = 1 \pmod{\phi(n)}$  where,  $\phi(n) = (p-1)(q-1)$  and  $e$  is the system's public key and  $d$  is the corresponding private key which should be kept secret by the server
- Determine an integer  $g$  which is a primitive element in both  $GF(p)$  and  $GF(q)$  and is the public information of the system

### Registration phase:

- $U_i$  selects  $ID_i$  and  $PW_i$  and then  $U_i$  sends a  $\{ID_i, PW_i\}$  to KIC in secure channels
- After receiving the  $\{ID_i, PW_i\}$ , the KIC computes  $CID_i = f(ID_i \oplus d)$  and  $S_i = (CID_i^d \pmod{n}) \oplus f(PW_i)$
- KIC stores  $\{n, e, S_i, ID_i\}$  into a smart card and issues the smart card to user  $U_i$  through a secure channel (Fig. 1)

### Login phase:

- $U_i$  inserts  $U_i$ 's smart card into a card reader and inputs the  $ID_i$  and  $PW_i$
- The smart card computes  $X_i = S_i \oplus f(PW_i)$  and  $Y_i = X_i^{f(ID_i, T_e)} \pmod{n}$
- $U_i$  sends the login request message  $M = \{ID_i, n, e, T_e, Y_i\}$  to the server  $S$  through a communication channel

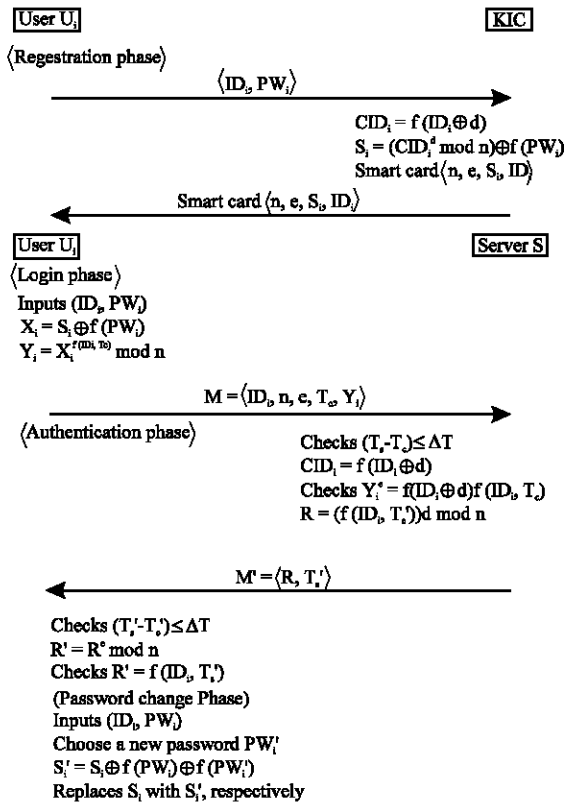


Fig. 1: Huang et al. (2014) scheme

**Authentication phase:** After receiving the login request message  $M$ , the S verifies whether the  $ID_i$  is a legitimate identifier of user  $U_i$ . Then it checks, if  $(T_s - T_c) \leq \Delta T$ . If  $(T_s - T_c) \leq \Delta T$ , then, the next step proceeds otherwise, this phase is terminated. The S computes  $CID_i = f(ID_i \oplus d)$  and checks  $Y_i^e = f(ID_i \oplus d)^{f(ID_i, T_c)} \text{ mod } n$ . If this is satisfied, the S accepts the login request otherwise, the login request is rejected and this phase is terminated. The S computes  $R = (f(ID_i, T_s))^d \text{ mod } n$ . Then, S sends the  $M' = \{R, T_s'\}$  to user  $U_i$ . After receiving the  $M'$  at time  $T_c'$ , the  $U_i$  checks the timestamp  $T_s'$  in the received message with the condition  $(T_s' - T_c') \leq \Delta T$ . If  $(T_s' - T_c') \leq \Delta T$ ,  $U_i$  accepts  $M'$ ; otherwise, the  $M'$  is rejected and this phase is terminated. The user  $U_i$  computes  $R' = R^e \text{ mod } n$  and then checks the equation  $R' = f(ID_i, T_s')$ . If there are satisfied,  $U_i$  accepts the S, otherwise, it rejects the S.

**Password change phase:**

- $U_i$  inserts  $U_i$ 's smart card into a card reader and inputs the  $ID_i$  and  $PW_i$
- $U_i$  chooses a new Password  $PW_i'$ . Then the smart card computes  $S_i' = S_i \oplus f(PW_i) \oplus f(PW_i')$  where,  $PW_i$  is the old password of  $U_i$
- The smart card replaces the existing value  $S_i$  with the new value  $S_i'$

**RESULTS AND DISCUSSION**

**Security analysis of Huang et al. (2014) scheme:** In this study, we are going to explain the possible attacks on Huang et al. (2014) scheme. We postulate that an attacker can intercept or eavesdrop on all the messages that were sent or received among the insecure channels. It has more details on the related studies (Kocher et al., 1999; Kim et al., 2014; Choi et al., 2014).

**Off-line password guessing attack:** The authentication schemes for all password-based users should be designed to prevent a guessing attack. Huang et al. (2014) has maintained that their scheme is able to resist an off-line password guessing attack even in the situation where an attacker could steal the secret values stored in a user's smart card. Despite of their claims, we found that their scheme has vulnerability in this situation, so, we present a scenario for the off-line password guessing attack instead of Huang et al. (2014) scheme. The detailed description follows.

**Step 1:** After an attacker has stolen the smart card, attacker can extract  $\{n, e, S_i, ID_i\}$  in  $U_i$ 's smart card.

**Step 2:** Attacker can use the eavesdropped login request message  $M = \{ID_i, n, e, T_s, Y_i\}$  from the communication network.

**Step 3:** Attacker selects a password candidate  $PW_i^*$ .

**Step 4:** Attacker computes  $X_i^* = S_i \oplus f(PW_i^*)$ .

**Step 5:** Attacker computes  $Y_i^* = (X_i^*)^{f(ID_i, T_c)} \text{ mod } n$  and  $Y_i^* = (S_i \oplus f(PW_i^*))^{f(ID_i, T_c)} \text{ mod } n$ .

**Step 6:** The attacker repeats above steps from 3-5 until the computed result  $Y_i^*$  equals the breached secret  $Y_i$ .

**Step 7:** If they correspond with each other,  $PW_i^*$  would be the accurate password. If not, the attacker repeats the above steps until finding the correct password. Through the aforementioned descriptions, we can realize that Huang et al. (2014) scheme is under the off-line password guessing attacks.

**User impersonation attack:** As is well known, general password-based user authentication is based on intelligence of the password. Thus, in general, if an attacker gains a password, the attacker can pretend to be a legal user. Huang et al. (2014) scheme allows an attacker to arrogate a legal user if the attacker obtains the user's password. After obtaining the password  $PW_i$  through the

guessing attack described in section 4.1 an attacker can impersonate a legal user  $U_i$  by performing the following steps.

**Step 1:** When a user  $U_i$  sends the login request message  $M = \{ID_i, n, e, T_c, Y_i\}$  to the server  $S$  in the login phase, the attacker intercepts the login request message  $M$  and computes  $X_i = S_i \oplus f(PW_i)$  and  $Y_i = X_i^{f(ID_i, T_c)} \bmod n$ .

**Step 2:** Attacker sends login request message  $M = \{ID_i, n, e, T_c, Y_i\}$  to  $S$ .

**Step 3:** After receiving the  $M$  from the attacker, the  $S$  then executes the verification procedure.

**Step 4:**  $S$  successfully verifies the  $M$  because the  $M$ , made by an attacker, correctly equals a legitimate user's login request message. Through the aforementioned descriptions, we can realize that the attacker can successfully impersonate the legal user  $U_i$ .

**Privileged insider attack:** Privileged insider attack is an attack where the user's password can be derived by a privileged insider of the server in the registration phase. Unfortunately, in the registration phase of Huang *et al.* (2014) scheme, the user  $U_i$  sends his/her Identity  $ID_i$  and Password  $PW_i$  to the server directly without any security measures. So, the Password  $PW_i$  of the  $U_i$  will be revealed and the privileged insider can easily obtain the user's Password  $PW_i$ . If the privileged insider of the server obtains the user's Password  $PW_i$ , an insider can impersonate the user's login by abusing the obtained legitimate user's Password  $PW_i$  and can access other remote systems. Therefore, Huang *et al.* (2014) scheme is vulnerable to privileged insider attack.

**Wrong password cannot be quickly detected:** In the login phase of Huang *et al.* (2014) scheme if the  $U_i$  inputs his/her  $ID_i$  and  $PW_i$ , the smart card does not verify the validity of user's password in itself. Therefore, even if the  $U_i$  inputs his/her password incorrectly by mistake, the login and authentication phases are still performed until it checked by the Server  $S$ . This leads to unnecessary waste a lot of communication and computation costs during the login and authentication phases. The detailed description is as follows:

Assume that the  $U_i$  inputs a wrong Password  $PW_i^*$  in the login phase then the smart card computes  $X_i^* = S_i \oplus f(PW_i^*)$  and  $Y_i^* = X_i^{*f(ID_i, T_c)} \bmod n$ . Then,  $U_i$  sends the login request messages  $M = \{ID_i, n, e, T_c, Y_i\}$  to  $S$ . After receiving the login request message  $M$ , the  $S$  checks whether the  $ID_i$  is a legitimate user or not and checks the timestamp  $T_s$ . Then, the  $S$  calculates  $CID_i = f(ID_i \oplus d)$  and

checks the equation  $Y_i^{*e} = f(ID_i \oplus d)^{f(ID_i, T_c)} \bmod n$ . If there are satisfied, the  $S$  accepts the login request. If not, the login request is rejected.

It is obvious that  $Y_i^* \neq Y_i$  since,  $X_i^* \neq X_i$  and  $Y_i^* \neq Y_i$ , therefore, the  $S$  will reject  $U_i$ 's login request. From above demonstration, if the  $U_i$  inputs a wrong password in the login phase it cannot be quickly detected. To prevent this weakness, verifying user's password should be examined in the beginning of the login phase as soon as possible. It can be helpful to prohibit the mentioned situation and lead to save the unnecessary waste of countless communication and computation costs.

**Weakness in password change phase:** In the phase of changing password an unauthorized user inserts his/her smart card into a smart card reader. Then, the  $U_i$  inputs the  $ID_i$  and  $PW^*$  which is created arbitrarily by the unauthorized user to ask for changing password. After requesting a password change, unauthorized user inputs the  $PW^*$  and the smart card calculates  $S_i^* = S_i \oplus f(PW_i) \oplus f(PW_i^*)$  which yields  $CID_i \oplus f(PW^*)$ . Eventually, the unauthorized user successfully replaces  $S_i$  with  $S_i^*$  without any confirmation process. As mentioned above, if an illegal attacker robs a  $U_i$ 's smart card and tries to change a password, the legal  $U_i$  would be rejected to log in except re-registering the remote server.

**Failure to preserve user anonymity:** Huang *et al.* (2014) scheme states a user's Identity  $ID_i$  is in the static condition and in plaintext form in the login phase. Using such an eavesdropping attack, the attacker can monitor maliciously the communication channels between user  $U_i$  and server  $S$ . The attacker also can identify some valued information on the transmitted messages over the public communication channel. In this way, once the login messages are eavesdropped, an attacker collects the communicating user's plaintext identities without difficulties. By analyzing all of the eavesdropped messages, the attacker can track down the connections between the user  $U_i$  and the server  $S$ . For this reason, a user's anonymity cannot be preserved by Huang *et al.* (2014) scheme.

**The proposed scheme:** In this study, we propose an improved authentication scheme to overcome the security weaknesses of Huang *et al.* (2014) scheme. The notations in the proposed scheme are summarized in Table 1. We describe each phase in detail in sections 5.1-5.5.

**Initialization phase:** The initialization phase is the first step of our proposed scheme and there is the same phase by Huang *et al.* (2014) scheme. In this initialization phase, KIC not only plays a role in producing some global parameters but also computes user's classified data.

**Registration phase:**  $U_i$  selects  $ID_i$  and  $PW_i$  and then  $U_i$  sends a  $\{ID_i, f(PW_i)\}$  to KIC in secure channels. After receiving the  $\{ID_i, PW_i\}$ , the KIC computes  $CID_i = f(ID_i \oplus d)$ ,  $S_i = (CID_i^d \text{ mod } n) \oplus f(PW_i)$  and  $H_i = g^d \text{ mod } n$ . KIC stores  $(ID_i, SID_i)$  about the user  $U_i$  and writes  $\{n, e, g, S_i, H_i, SID_i\}$  into the smartcard's memory. KIC sends the smartcard to  $U_i$  through a secure channel. Upon receiving the smartcard,  $U_i$  inserts his/her password into the smart card and then the smart card computes  $B_i = f(ID_i) \oplus f(PW_i)$ . Finally, the smart card contains the information  $\{B_i, n, e, g, S_i, H_i, SID_i\}$ .

**Login phase:**  $U_i$  inserts  $U_i$ 's smart card into a card reader and inputs the  $ID_i$  and  $PW_i$ . The smart card computes  $B_i^* = f(ID_i) \oplus f(PW_i)$ . The smart card then verifies  $B_i^* = B_i$ . If they are not equal, the smart card rejects  $U_i$ 's login request, otherwise goes to the next step. The smart card generates random number  $r$  and computes  $X_i = g^r \text{ mod } n$  and  $Y_i = (S_i \oplus f(PW_i)) \cdot H_i^{r \cdot f(ID_i, T_e)} \text{ mod } n$ .  $U_i$  sends the login request message  $M = \{SID_i, n, e, g, T_e, X_i, Y_i\}$  to the server  $S$  through a communication channel (Fig. 2).

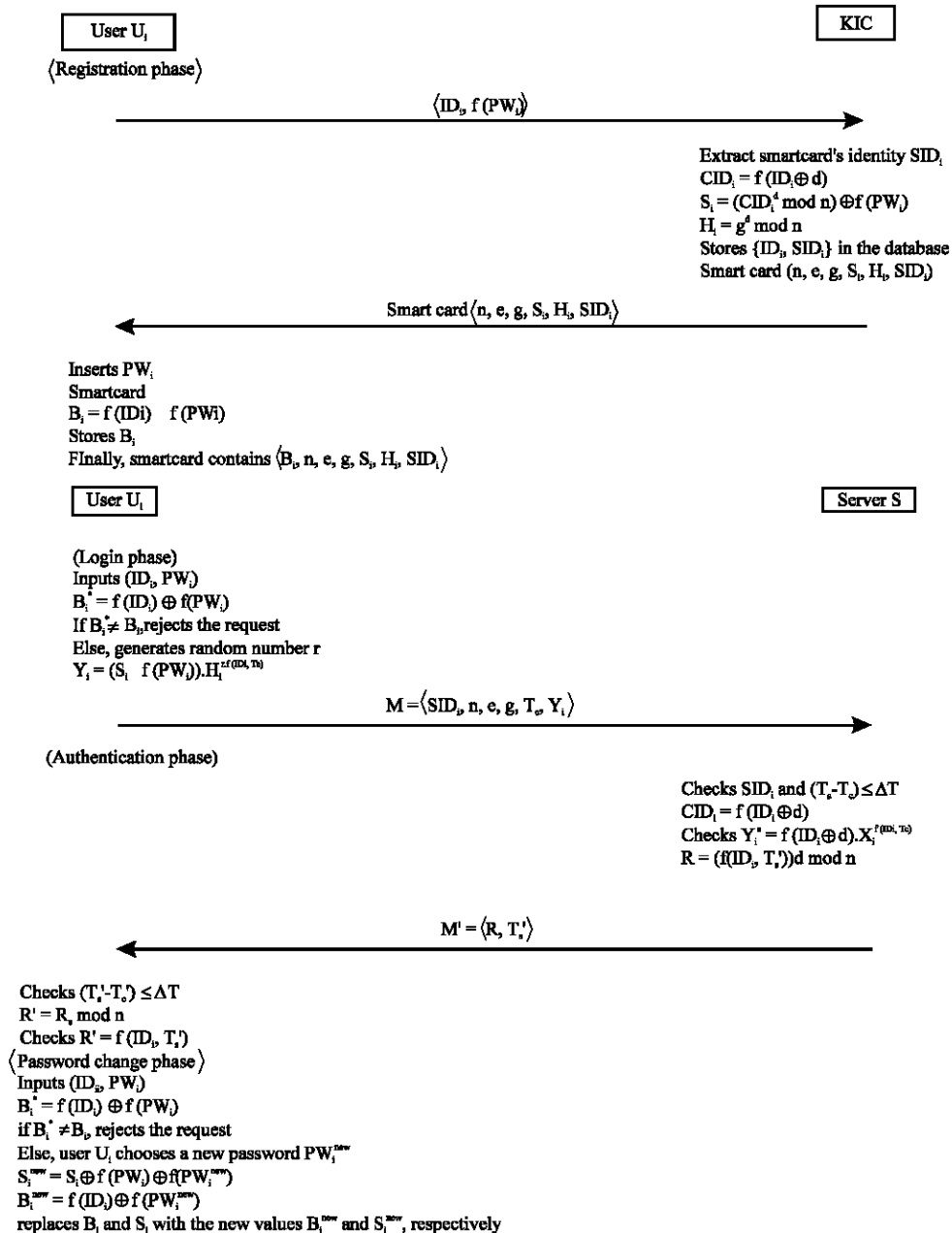


Fig. 2: Our proposed scheme

Table 2: Functionality comparison of the proposed scheme and other related scheme

Features	Shen <i>et al.</i> (2003)	Awasthi <i>et al.</i> (2011)	Huang <i>et al.</i> (2014)	Proposed scheme
F1	No	No	No	Yes
F2	Yes	Yes	Yes	Yes
F3	No	No	Yes	Yes
F4	No	No	No	Yes
F5	Yes	Yes	Yes	Yes
F6	Yes	Yes	Yes	Yes

**Authentication phase:** After receiving the login request message  $M$ , the  $S$  extracts  $ID_i$  from  $SID_i$  corresponding to  $SID_i$  in its database and verifies whether the  $ID_i$  is a legitimate identifier of user  $U_i$ . Then it checks, if  $(T_s - T_c) \leq \Delta T$ . If  $(T_s - T_c) \leq \Delta T$ , then the next step proceeds otherwise, this phase is terminated. The  $S$  computes  $CID_i = f(ID_i, \oplus d)$  and checks  $Y_i^e = f(ID_i, \oplus d). X_i^{f(ID_i, T_c)} \bmod n$ . If this is satisfied, the  $S$  accepts the login request; otherwise, the login request is rejected. The  $S$  then computes  $R = (f(ID_i, T_s'))^d \bmod n$  and sends  $M' = \{R, T_s'\}$  to user  $U_i$ .

After receiving the  $M'$  at time  $T_c'$ , the  $U_i$  checks the timestamp  $T_s'$  with the condition  $(T_s' - T_c') \leq \Delta T$ . If  $(T_s' - T_c') \leq \Delta T$ ,  $U_i$  accepts  $M'$  otherwise, the  $M'$  is rejected and this phase is terminated. The user  $U_i$  computes  $R' = R^e \bmod n$  and then checks the equation  $R' = f(ID_i, T_s')$ . If there are satisfied,  $U_i$  accepts the  $S$ , otherwise, it rejects the  $S$ .

**Password change phase:**  $U_i$  inserts  $U_i$ 's smart card into a card reader and inputs the  $ID_i$  and  $PW_i$ . The smart card computes  $B_i^* = f(ID_i) \oplus f(PW_i)$ . The smart card then verifies  $B_i^* = B_i$ . If they are not equal, the smart card rejects  $U_i$ 's request, otherwise goes to the next step.  $U_i$  chooses a new password  $PW_i^{new}$ . Then the smart card computes  $S_i^{new} = S_i \oplus f(PW_i) \oplus f(PW_i^{new}) = CID_i^d \oplus f(PW_i^{new})$  and  $B_i^{new} = f(ID_i) \oplus f(PW_i^{new})$ . The smart card replaces the existing values  $B_i$  and  $S_i$  with the new values  $B_i^{new}$  and  $S_i^{new}$ , respectively. Finally, the smart card contains the information  $\{B_i^{new}, n, e, g, S_i^{new}, H_i, SID_i\}$ .

**Security analysis of the proposed scheme:** This study is focus on the security analysis of our proposed scheme. We discuss this proposed scheme's security with regard to the security requirements which mentioned in section 1. Table 2 demonstrates the essential functionality comparison with the proposed scheme and the other associated schemes (Shen *et al.*, 2003; Awasthi *et al.*, 2011; Huang *et al.*, 2014).

**F1; User anonymity:** Suppose that the attacker has intercepted  $U_i$ 's login request  $M = \{SID_i, n, e, g, T_c, X_i, Y_i\}$ . Then, the attacker may try to analyze the login request message and retrieve any static parameter from this message. However, it is infeasible to derive  $ID_i$  from the

login request because the login request includes  $SID_i$  instead of  $ID_i$ . Using  $SID_i$ , attacker cannot acquire any information related to  $U_i$ 's identity.

**F2; Single registration:** As shown in section 5, a new user  $U_i$  just needs to register at KIC once, then KIC issues a smart card stored with the necessary secret information to user  $U_i$ . Then, the User  $U_i$  is able to use the smart card and the Password  $PW_i$  to login to all authenticated servers in the remote systems.

**F3; Freely change password:** A new user  $U_i$  is provided the freedom of password choice in our proposed scheme. The password which is chosen by the user's preference is sent to KIC and then the user  $U_i$  can use the password  $PW_i$  to login to all the legal servers. Moreover, as mentioned in section 5, a legal user can change the Password  $PW_i$  in the change password phase without communicating with the KIC. After that, the user  $U_i$  has the permission to access all the legal servers with a new password. Consequently, we maintain that our proposed scheme facilitates changing password easily and freely.

**F4; Securely change password:** The previous schemes Shen *et al.* (2003), Awasthi *et al.* (2011), Huang *et al.* (2014) should have provided a secure changing password because of guessing password attack which made the user's password revealed. Accordingly, if an attacker got the user's password it is possible to change the password arbitrarily. In this situation, unfortunately, the legitimate user must re-register at KIC. On the contrary in our proposed scheme, there is no way for the attacker to know the legitimate user's Password  $PW_i$ . Therefore, we offer a secure password change phase to legitimate users.

**F5; Provides mutual authentication:** In our proposed scheme, the  $S$  can authenticate the user by checking whether login request message  $M$  is correct, since, only legitimate users can correctly construct a login request message. Also, the  $U_i$  can authenticate the server by checking whether the authentication request message  $M'$  is correct, since, only legal servers can construct a correct response to the user's challenge.

Table 3: Security comparison of the proposed scheme and other related scheme

Attack types	Shen <i>et al.</i> (2003)	Awasthi <i>et al.</i> (2011)	Huang <i>et al.</i> (2014)	Proposed scheme
A1	Yes	Yes	Yes	Yes
A2	No	No	No	Yes
A3	Yes	Yes	Yes	Yes
A4	Yes	Yes	Yes	Yes
A5	No	No	No	Yes
A6	No	No	No	Yes
A7	No	No	No	Yes
A8	No	No	No	Yes

**F6; No verification table:** Several authentication schemes commonly demand to store a verification table in the server while in our scheme there is no need to store the nonce used for the next authentication session. Using verification tables cause several overheads problems in servers. Also, it is vulnerable to stolen-verifier attack. However, in our proposed scheme presented in section 5, this kind of verification table does not exist. Therefore, we know that our proposed scheme satisfies this requirement.

As we mentioned in section 1, a secure and efficient timestamp-based user authentication scheme should fulfill some security requirements and defend some various types of attack. So, we scrutinize our scheme under different possible attacks using informal security analysis. Table 3 indicates a security comparison of our proposed scheme and other associated schemes (Shen *et al.*, 2003; Awasthi *et al.*, 2011; Huang *et al.*, 2014).

**A1; Replay attack:** The attacker can intercept data packets for the purpose of making use of that data in some manner and then tries to login to the server using the intercepted packets transmitted between the legitimate user and the server. However, the login request message of our proposed scheme includes a current timestamp, such as  $T_c$  of  $\{SID_i, n, e, g, T_c, X_i, Y_i\}$ . Hence, our proposed scheme can defend against replay attack.

**A2; User impersonation attack:** The attacker tries to impersonate the authenticated  $U_i$  in order to cheat the other party. In order to start a new session, the attacker has to modify both  $X_i$  and  $Y_i$ . However, as discussed in section 5, to change  $Y_i$  the attacker has to guess/know the random number  $r$  which is opaque to the attacker. Thus, the probability of successfully guessing random number  $r$  which is randomly generated by the legitimate user is negligible.

**A3; Server impersonation attack:** The attacker tries to impersonate the legal  $S$  in order to cheat the authenticated user. In our proposed scheme, the legitimate  $U_i$  authenticates the  $S$  using the secret value  $R$  and this secret value is sent to the  $U_i$  after

encryption with the system's private key  $d$ . Thus, without knowledge of the server's private key  $d$ , the attacker cannot impersonate the  $S$ .

**A4; Man-in-the-middle attack:** The attacker intercepts login request and replays these intercepted messages. In our proposed scheme in order to make a successful man-in-the-middle attack, the attacker has to change  $X_i$  and  $Y_i$  properly, so that, the  $S$  can verify the message successfully. However, the attacker does not know the value of  $X_i$  or  $Y_i$ .

**A5; Off-line password guessing attack:** We deem that a stolen or lost smart-card which owned by a legitimate user provide all the secret information to the attacker as mentioned earlier. In order to make a successful password guessing attack, the attacker has to know the random number  $r$ . However, the probability of successfully guessing random number  $r$  is negligible. Also, it is hard for the attacker to solve this problem in polynomial time.

**A6; Stolen smart card attack:** The attacker can extract all the secret information from the smart card through power analysis attack (Kocher *et al.*, 1999; Choi *et al.*, 2014).. Note that  $X_i = g^r \text{ mod } n$  and  $Y_i = (S_i \oplus f(PW_i)) \cdot H_i \cdot f(ID_i, T_c) \text{ mod } n$ . In order to know the  $U_i$ 's Password  $PW_i$ , the attacker needs to guess the random number  $r$ . However, the probability of successfully guessing the  $r$  is negligible. Hence, our proposed scheme prevents stolen smart card attack.

**A7; Privileged-insider attack:** There is a possibility that the privileged insider of the KIC directly obtain the user's password and then access to the user's account of other servers by using the same password. This attack is caused by the disclosure of the user's password  $PW_i$  in the registration phase. In our proposed scheme, the  $U_i$  submits the password information to KIC in the form  $f(PW_i)$  instead of the form  $PW_i$ . Accordingly, the privileged insider as an attacker is unable to gain the user's Password  $PW_i$ .

**A8; Stolen-verifier attack:** An attacker acquires a password verifier from the server to apply to impersonate



Table 4: Efficiency comparison of the proposed scheme and other related scheme

Phases	Shen <i>et al.</i> (2003)	Awasthi <i>et al.</i> (2011)	Huang <i>et al.</i> (2014)	Proposed scheme
<b>Registration</b>				
User side	0	0	0	$3T_h+1T_x$
Server side	$1T_h+1T_x+1T_m+2T_e$	$1T_h+1T_x+1T_m+2T_e$	$2T_h+2T_x+1T_e$	$1T_h+2T_x+2T_e$
<b>Login</b>				
User side	$1T_h+3T_m+2T_e$	$1T_h+3T_m+2T_e$	$2T_h+1T_x+1T_e$	$3T_h+2T_x+2T_m+2T_e$
Server side	0	0	0	0
<b>Authentication</b>				
User side	$1T_e$	$1T_h+1T_e$	$1T_h+1T_e$	$1T_h+1T_e$
Server side	$3T_h+1T_x+1T_m+3T_e$	$3T_h+1T_x+1T_m+3T_e$	$3T_h+1T_x+3T_e$	$3T_h+1T_x+1T_m+3T_e$
<b>Authentication</b>				
User side	-	-	$2T_h+2T_x$	$4T_h+4T_x$
Server side	-	-	0	0
Total	$5T_h+2T_x+5T_m+8T_e$	$6T_h+2T_x+5T_m+8T_e$	$10T_h+6T_x+6T_e$	$15T_h+10T_x+3T_m+8T_e$

an authenticated user immediately. To succeed in a stolen-verifier attack, the attacker needs to know the user’s password. However, as shown in section 5, no verification table is stored in our proposed scheme.

**Performance analysis of the proposed scheme:** In this section, we are going to evaluate the efficiency of our proposed scheme and compare with other associated schemes (Shen *et al.*, 2003; Awasthi *et al.*, 2011; Huang *et al.*, 2014). Above all, we are going to define the following notations, so as to clarify the analysis for the computational complexity.

- $T_h$ : The time complexity of one-way hash functions
- $T_x$ : The time complexity of exclusive-OR operations
- $T_m$ : The time complexity of modular multiplication operations
- $T_e$ : The time complexity of modular exponential operations

Table 4 shows the computation cost comparison of the proposed scheme and other associated schemes. We can see that the total computation costs of the schemes of Shen *et al.* (2003), Awasthi *et al.* (2011) and Huang *et al.* (2014) and our proposed scheme in the registration, login, authentication and password change phases are  $5T_h+2T_x+5T_m+8T_e$ ,  $6T_h+2T_x+5T_m+8T_e$ ,  $10T_h+6T_x+6T_e$  and  $15T_h+10T_x+3T_m+8T_e$ , respectively. Compared with other associated schemes, our proposed scheme carries a greater computational cost. However, as shown in Table 2, the proposed scheme can ensure secure password changes. Besides, our proposed scheme is secure against password guessing attack, user impersonation attack, privileged-insider attack and stolen-verifier attack. So, it is worth spending a little more computational power to achieve better security and improved usability.

**CONCLUSION**

We proposed the authentication scheme which is improved by overcoming the security vulnerabilities of

Huang *et al.* (2014) scheme. The proposed scheme grants the permission for the user to choose and change the password, also arranges mutual authentication between the user and the server to protect from the impersonation attack. Most of all, our proposed scheme can withstand password guessing attack, stolen smart card attack and privileged-insider attack despite the challenging circumstance that the attacker acquires the user’s smart card. The proposed protocol’s safety has its roots in the RSA cryptosystem, the discrete logarithm problem and the one-way has function. The new scheme has a higher estimated time complexity compared to other associated schemes. However, this should be easily tolerated due to the higher security and it affords to resist most well-known attacks while providing the functionality for a secure timestamp-based authentication scheme.

**ACKNOWLEDGEMENT**

This research was supported by the National Research Foundation of Korea grant funded by Korea Government (Ministry of Science, ICT and Future Planning) (NRF-2017R1C1B5017492) and this research was supported by financial support of Howon University in 2019.

**REFERENCES**

Awasthi, A.K., K. Srivastava and R.C. Mittal, 2011. An improved timestamp-based remote user authentication scheme. *Comput. Electr. Eng.*, 37: 869-874.

Chan, C.K. and L.M. Cheng, 2001. Cryptanalysis of a timestamp-based password authentication scheme. *Comput. Secur.*, 21: 74-76.

Choi, Y., D. Lee, J. Kim, J. Jung and J. Nam *et al.*, 2014. Security enhanced user authentication protocol for wireless sensor networks using elliptic curves cryptography. *Sens.*, 14: 10081-10106.

- Fan, L., J.H. Li and H.W. Zhu, 2002. An enhancement of timestamp-based password authentication scheme. *Int. J. Comput. Secur.*, 21: 665-667.
- Hsu, C.L., 2004. Security of Chien *et al.*'s remote user authentication scheme using smart cards. *Comput. Stand. Interfac.*, 26: 167-169.
- Huang, H.F., H.W. Chang and P.K. Yu, 2014. Enhancement of timestamp-based user authentication scheme with smart card. *Intl. J. Network Secur.*, 16: 463-467.
- Hwang, T. and W.C. Ku, 1995. Repairable key distribution protocols for Internet environments. *IEEE. Trans. Commun.*, 43: 1947-1949.
- Kim, J., D. Lee, W. Jeon, Y. Lee and D. Won, 2014. Security analysis and improvements of two-factor mutual authentication with key agreement in wireless sensor networks. *Sens.*, 14: 6443-6462.
- Kocher, P., J. Jaffe and B. Jun, 1999. Differential power analysis. *Proceedings of the 19th Annual International Conference on Cryptology*, August 15-19, 1999, Springer, Berlin, Germany, pp: 388-397.
- Ku, W.C., C.M. Chen and H.L. Lee, 2003. Cryptanalysis of a variant of Peyravian-Zunic's password authentication scheme. *IEICE. Trans. Commun.*, 86: 1682-1684.
- Lamport, L., 1981. Password authentication with insecure communication. *Commun. ACM*, 24: 770-772.
- Liu, J.Y., A.M. Zhou and M.X. Gao, 2008. A new mutual authentication scheme based on nonce and smart cards. *Comput. Commun.*, 31: 2205-2209.
- Rivest, R.L., A. Shamir and L. Adleman, 1978. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM.*, 21: 120-126.
- Shen, J.J., C.W. Lin and M.S. Hwang, 2003. Security enhancement for the timestamp-based password authentication scheme using smart cards. *Int. J. Comput. Secur.*, 22: 591-595.
- Stallings, W., 2003. *Cryptography and Network Security: Principles and Practices*. 3rd Edn., Prentice Hall, Upper Saddle River, New Jersey, USA., ISBN:9780130914293, Pages: 681.
- Sun, D.Z., J.P. Huai, J.Z. Sun and J.X. Li, 2009. Cryptanalysis of a mutual authentication scheme based on nonce and smart cards. *Comput. Commun.*, 32: 1015-1017.
- Yang, W.H. and S.P. Shieh, 1999. Password authentication schemes with smart card. *Comput. Secur.*, 18: 727-733.