ISSN 2383-630X(Print) / ISSN 2383-6296(Online)
Journal of KIISE, Vol. 46, No. 6, pp. 587-597, 2019. 6
https://doi.org/10.5626/JOK.2019.46.6.587

타원곡선 암호화와 퍼지 추출자를 이용한 강건한 삼중요소 사용자 인증 방식 연구
(A Robust Three-Factor User Authentication Scheme based on Elliptic Curve Cryptography and Fuzzy Extractor)

오중타잉[†]　　　최태영[††]
(Trung Thanh Ngo)　　(Tae-Young Choe)

요　약　삼중요소 사용자 인증은 금융 거래와 같이 사용자 인증의 보안성이 중요시되는 경우에 적용될 수 있는 인증 프로토콜이다. Fan과 Lin의 삼중요소 사용자 인증은 보안 토큰, 패스워드, 그리고 지문을 사용하여 인증을 수행하므로 이들 요소 중 하나라도 빠진 경우 인증을 할 수 없는 엄격한 인증 방식이다. 하지만, Fan과 Lin의 인증 방식은 내부자 공격, 인증 토큰 도난 공격, 그리고 메시지 변형 공격에 취약하다는 문제점을 가지고 있다. 최근에 Yeh 외 3인은 기존의 Fan과 Lin의 방식을 개선한 삼중요소 사용자 인증을 제안했으며 타원 곡선 암호화 기법을 사용하여 보안성을 높이고 성능을 향상시켰다. 하지만, 본 논문에서는 Yeh 외 3인의 인증 방식이 사용자 위장 공격과 서버 위장 공격에는 취약함을 밝힌다. 이러한 문제를 해결하기 위해 본 논문에서는 서버 스마트카드, 타원 곡선 암호화, 그리고 퍼지 추출자를 사용한 삼중요소 사용자 인증 방식을 제안한다. 또한 이 방식이 앞에서 언급한 문제점들을 모두 해결하고, 동시에 성능이 더 개선됨을 보이며, BAN 로직을 통해 안전한 통신 채널이 연결됨을 입증한다.

키워드: 삼중요소 인증, 내부자 공격, 타원 곡선 암호화, 퍼지 추출자, BAN 로직

Abstract　A three-factor user authentication is appropriate to ensure a high degree of authentication. Fan and Lin proposed a typical three-factor authentication scheme, which requires token, password, and fingerprint. The scheme does not allow authentication in the absence of any of the three factors. Unfortunately, Fan and Lin's scheme is associated with security risks such as vulnerability to insider attacks, stolen-verifier attacks, and message modification attacks. Yeh et al. proposed a three-factor user authentication, which overcomes such pitfalls and improves security and performance using elliptic curve cryptography. We found that Yeh et al.'s scheme is still vulnerable to user impersonation attacks and server masquerading attacks. We propose a robust three-factor authentication scheme entailing server smart cards, elliptic curve cryptography, and a fuzzy extractor that address the foregoing flaws and result in enhanced security. The proposed scheme is resistant to various attacks and improves system performance. BAN logic is used to prove that the scheme establishes a secure channel.

Keywords: three-factor authentication, insider attack, elliptic curve cryptography, fuzzy extractor, BAN logic

· 본 연구는 금오공과대학교학술연구비에 의하여 지원된 논문임

[†] 비 회 원 : 금오공과대학교 IT융합학과
thanh1239z@gmail.com
[††] 정 회 원 : 금오공과대학교 컴퓨터공학과 교수
(Kumoh Nat'l Institute of Technology)
choety@kumoh.ac.kr
(Corresponding author임)

논문접수 : 2019년 1월 9일
(Received 9 January 2019)
논문수정 : 2019년 3월 12일
(Revised 12 March 2019)
심사완료 : 2019년 3월 13일
(Accepted 13 March 2019)

정보과학회논문지 제46권 제6호(2019. 6)

1. Introduction

In the 21st century, Internet technologies have offered us many benefits. For example, people can stay connected everywhere at any time using social networks like Facebook. Powerful search engines like Google provide us great ways to obtain information and knowledge from giant Internet databases. However, threats of cybercrime are also increasing rapidly in terms of both quantity and the complexity level. High availability of hardware and software makes it easier for hackers to steal sensitive data. There are a number of cracking tools available on the Internet, and these tools can be used by any Internet cracker who does not know cracking mechanisms in detail. As an example, the cracker may find that a user forgets to change the default password of his/her new router, and exploit it.

In order to cope with such cybercrime threats, many countries and companies have invested a huge amount of money to improve their network security. In the past, if a user had access to a company's resource, username and password authentication was sufficient. However, now the use of username and password is no longer safe against many threats such as spyware, viruses, and keyloggers. Accordingly, new authentication schemes have been proposed to replace the traditional password authentication [1-3]. Smart cards offer strong protection of stored data against unauthorized access and have been used along with password verification in many studies [4-8]. Unfortunately, smart cards and passwords can be lost, forgotten, or even shared with other people. Thus smart cards partially provide non-repudiation, which is a method to guarantee that an individual who accesses a certain facility cannot later deny using it [9]. Recently, biometric authentications like fingerprint or iris scanning have come under the spotlight because they offer advantages such as negative recognition and non-repudiation that were not provided by tokens and passwords [10].

Combining the advantages of passwords, smart cards, and biometric authentications is considered another idea for guaranteeing both strong security and non-repudiation requirements. Accordingly, several remote authentication schemes using the three above-mentioned factors have been proposed [11-14]. For instance, Yeh et al. proposed a robust elliptic curve cryptography based authentication protocol [15] based on Fan and Lin's three-factor authentication scheme [16]. They claimed that their scheme overcomes the security pitfalls of Fan and Lin's scheme, such as being vulnerable to insider attacks, stolen-verifier attacks, and modification attacks. However, we found that Yeh et al.'s scheme still has some flaws such as vulnerability to user impersonation attacks and server masquerading attacks. In this paper, we propose a scheme that uses multiple server smart cards with a authentication protocol in order to overcome these flaws and to strengthen security. Evenmore, the proposed scheme has less computation and communication overheads compared to Yeh et al.'s scheme.

The rest of this paper is organised as follows: Section 2 briefly introduces related works on the topic. Section 3 shows the mathematical background and notations used in Yeh's scheme and the proposed scheme. The review and the analysis of Yeh's scheme are presented as a case study in Section 4 and 5, respectively. Section 6 presents the proposed scheme, which withstands the flaws discussed in Section 5. Next, the strength and validity of the proposed scheme is discussed in Section 7. Finally, we provide the concluding remarks in Section 8.

2. Previous works

Jiang et al. proposed a three-factor authentication scheme based on chaotic maps [17]. In order to reduce computation time of public key processing, the scheme uses chaotic maps which utilize Chebyshev polynomial [18]. Since Jiang et al.'s scheme relies on timestamp, reliable time server is necessary in the system. Also, the scheme requires additional computation in order to guarantee anonymity.

Another three-factor authentication scheme uses Elliptic curve cryptosystem in order to reduce encryption and decryption overheads [17]. Although the scheme uses small number of communication messages, the result of the authentication phase is an implicit shared key which requires validity codes during the main communication phase.

Chaudhry et al. proposed a three-factor authentication scheme for multi server environments [19] with ProVerif verification tool. The scheme provides user anonymity and mutual authentication. In order to provide user anonymity and small number of keys, the scheme uses public key cryptography. Unfortunately, the scheme hashes a user's biometrics which increases false rejection rate.

A lightweight three-factor authentication scheme is proposed for wireless sensor networks [20]. In order to be used in low powered system, the scheme utilizes Rabin cryptosystem which uses modular square-root. Timestamp is not fully utilized in the scheme because it opens a probability of reply attack when messages are very short and the same contents.

3. Mathematical background and notations

Elliptic curve cryptosystems (ECCs) [21] have been used in Yeh et al.'s scheme to offer better performance than other public key cryptosystems that use exponentiation modulo, since ECCs can achieve the same security level with a smaller key size. An ECC is used as an encryption method during the mutual authentication between a smart card and a server. Fuzzy extractor [22] is a method to solve the sensitive hash problem of biometric templates [11].

3.1 Elliptic curve cryptosystem (ECC)

An elliptic curve is a plane curve over a finite field, which consists of the points satisfying the equation $y^2 \equiv x^3 + ax + b(\bmod p)$ with $a,b \in F_p$ satisfying $4a^3 + 27b^2 \not\equiv 0(\bmod p)$, where F_p denotes a finite field and p represents a large prime number. Given an integer s $(0 < s < p - 1)$ and a point $P \in F_p$, the point multiplication sP is defined as $sP = P + P + ... + P$ (s times). In general, the security of an ECC is based on the complexities of the following problems:

- Problem 1 (ECDLP): Given two points P and Q over $E_p(a,b)$, the elliptic curve discrete logarithm problem (ECDLP) is to find an integer s such that $0 < s < p - 1$ and $Q = sP$.
- Problem 2 (ECDHP): Given three points P, sP, and tP over $E_p(a,b)$ for $0 < s, t < p - 1$, the elliptic curve Diffie-Hellman problem (ECDHP) is to find the point stP over F_p.

3.2 Fuzzy extractor

A fuzzy extractor extracts nearly uniform randomness R from the biometric input B. Auxiliary information Q helps to maintain the same R as long as the biometric input remains reasonably close to the original. A fuzzy extractor consists of a pair of randomized procedures *Gen* and *Rep*, which stand for "generate" and "reproduce", respectively.

- $(R, Q) = Gen(B)$ is a probabilistic generation procedure that takes a biometric input B and produces an extracted string R and an auxiliary string Q.
- $R = Rep(B', Q)$ is a deterministic reproduction procedure that allows the recovery of R from the corresponding auxiliary string Q and any vector B' close to B.

The notations used in this paper are shown in Table 1.

Table 1 Notations

Notation	Description
U	User entity
TRC	Trusted registration center
SM	Smart card of user U
SM_S	Smart card of the server
S	Server entity
A	Attacker
PW	Password of user U
ID	Identity of user U
B	Biometric template of user U
$h(.)$	A secure hash function
$a \parallel b$	a concatenates b
$a \oplus b$	a exclusive-or b
$E(M, k)$	A symmetric encryption of plaintext M using k as a symmetric key
$D(C, k)$	A symmetric decryption of cyphertext C using symmetric key k
r	A random string
F_p	A finite field of order p
$E_p(a, b)$	An elliptic curve defined on finite field F_p with $a, b \in F_p$

4. Yeh's three factor user authentication scheme

Yeh's scheme is a remote user authentication scheme that uses three authentication factors. It consists of five phases: initiation phase, registration phase, login phase, authentication phase, and password change phase.

4.1 Initiation phase

In this phase, the server sets up the following system parameters for session key generation:

1. User U and server S choose an elliptic curve $E_p(a,b)$.
2. The server randomly chooses its private key $q_s \in Z_p^*$, where Z_p^* denotes a multiplicative group and

computes the user's authentication key (public key) $Q_s = q_sP$, where P represents a point on elliptic curve $E_p(a, b)$.

3. The server chooses an one-way hash function $h()$.
4. The server stores q_s and generates message ($E_p(a, b)$, P, Q_s).

4.2 Registration phase

1. User U enters his/her identity ID and password PW, and computes $h(PW \oplus r)$, where r denotes a secret random string chosen by U.
2. U scans biometric template B and computes $\delta_r(B) = r \oplus B$ using encryption key B.
3. U submits ID, $h(PW \oplus r)$, and $\delta_r(B)$ to server S via a secure channel.
4. Server S receives U's message and computes $W = h(P \oplus h(PW \oplus r))$. Then, S stores (W, $h(.)$, P, Q_s) to a smart card SM and issues it to U via a secure channel.
5. U's smart card SM checks whether $W = h(P \oplus h(PW \oplus r))$ is correct. If this is true, U accepts the smart card. Otherwise, U rejects the smart card.
6. The sketch $E(r, B)$ is stored in the smart card using the user's biometric template B as the encryption key.

4.3 Login phase

To log in to the system, the user must adhere to the following steps:

1. U offers PW^* and his/her biometrics B^*. U's smart card retrieves random string $r = D(E(r, B), B^*)$. Then, the smart card computes $BB^* = r \oplus B^*$. A hash value $h(PW^* \oplus r)$ is sent to the server S.
2. The server S validates whether $W = h(P \oplus h(PW^* \oplus r))$ is correct. If this is true, the server accepts the login and goes to the authentication phase. Otherwise, the server rejects the login request.

4.4 Authentication phase

1. U randomly chooses a private key $q_u = r^*$ and computes $Q_u = q_uP$, where Q_u denotes U's public key and $r^* \in Z_p^*$ represents a random string.
2. U computes $Q_1 = q_uQ_s$ and $M_u = N_u + Q_u + Q_1$, where N_u as BB^*.
3. U sends $m_1 = (Q_1, Q_u, M_u)$ to the server.
4. Server S computes $Q_1 = q_sQ_u$. Then, S checks whether $N_u^* = M_u - Q_u - Q_1 = N_u$ where N_u as $\delta_r(B)$. If this is true, m_1 did come from U. Otherwise, the process is terminated.
5. S computes $Q_s^* = q_s^*P$, $T_s = N_u^* + Q_s + Q_1$ and $M_s = N_s + Q_s + Q_1 + N_u^*$, where N_s is chosen by S. Then, S sends $m_2 = (T_s, M_s, Q_s^*)$ to U_i.
6. U computes N_u^{**} and checks whether $N_u^{**} = T_s - Q_s^* - Q_1 = N_u$ is correct. If this is true, m_2 definitely comes from S. Otherwise, m_2 does not comes from S and the process is terminated. U computes $N_s^* = M_s - Q_s - Q_1 - N_u^*$, $L = N_s^* + Q_u + Q_1$ and sends m_3 (= $L = N_s + Q_u + Q_1$) to S.
7. S computes $N_s^{**} = L - Q_u - Q_1$ and checks whether $N_s = N_s^{**}$. If this is true, S accepts the login request. Otherwise, it rejects the login request.

4.5 Password changing phase

To change his/her password, the user can compute a new value $h(PW^* \oplus r)$ and send the message (ID, $h(PW^* \oplus r)$, $\delta_r(S)$) to the remote server. After receiving the demand for password change, the remote server computes the new value to update $W^* = h(P \oplus h(PW^* \oplus r))$ into the smart card in order to replace the original value of W.

5. Security analysis of Yeh's scheme

In this section, we analyse the security of Yeh's scheme on the basis of the following assumptions:

- An attacker can eavesdrop, intercept, and modify messages sent between a user and a server.

5.1 User impersonation attack

In order to impersonate a legal user, attacker A can replay messages in both the login and the authentication phases. In the login phase, A can carry out a replay attack by executing the following steps:

1. Attacker A eavesdrops login messages PW^*, B^* sent from U to S.
2. Later, A makes a connection and sends the eavesdropped login messages to S. Server S validates the messages and finds them valid. As a result, S allows A to proceed to the authentication phase.

It is known that a password is vulnerable to off-line guessing attack [17,23], and biometrics can be stolen easily [24]. The second step of this attack is possible because the sent messages do not have values like nonce or time-stamps to prevent the attacker from replaying old login messages. In the authentication phase, attacker A can carry out a replay attack by executing the following steps:

1. A eavesdrops authentication messages m_1 sent from U to server S.
2. Later, A makes a connection and sends the eavesdropped message m_1 to S. S performs Steps 4 and 5 of the authentication phase and sends m_2 to A.
3. A computes $N_s^* = M_s - Q_s^* - Q_1 - N_u$, where N_u can be computed from the message m_1 as $N_u = M_u - Q_u - Q_1$. Further, A sends $m_3 = L = N_s^* + Q_u + Q_1$ to S, as in Step 6.
4. S executes the final step 7 and grants access to A.

This attack is possible because the authentication messages m_1, m_2, and m_3 do not have values like nonce to prevent attacker A from replaying old authentication messages. Further, because Q_1 is sent in message m_1, A can use it to compute N_u^{**} and N_s^*, which are needed to compute $L = m_3$.

5.2 Server masquerading attack

Attacker A can intercept message m_1 and masquerade as the server S by executing the following steps:

1. A eavesdrops message m_2 in the previous authentication session.
2. After intercepting message m_1 from U_i, A computes $N_u = M_u - Q_u - Q_1$, $T_s = N_u + Q_s + Q_1$, and $M_s = N_s' + Q_s^* + Q_1 + N_u$, where N_s' is chosen by attacker A in Step 6 of the authentication phase. This is possible because Q_1^* is the same as Q_1.
3. U successfully validates m_2 and replies with message m_3 to A in Step 6 of the authentication phase.
4. A grants access to U_i.

This attack is possible because the sent messages do not contain values like nonce to prevent the attacker from replaying old authentication messages.

6. Enhanced scheme

In this section, we propose a new scheme that overcomes the flaws of Yeh's scheme and resists various attacks. In the proposed scheme, as depicted in Fig. 1, there are a trusted registration center (TRC), client smart cards (SM), server smart cards (SM_s), and a remote server (S). T is a smart card device that transfers data between the server and the smart cards. Note that T cannot read the secret data stored in a server smart card. It only acts as a smart card adapter between the server and the server smart cards.

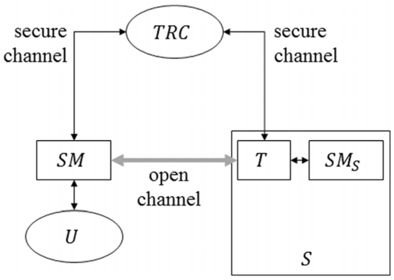

Fig. 1 Proposed system model

The enhanced scheme consists of three phases: registration phase, login phase, and server authentication phase.

6.1 Registration phase

To register at a system, a user U must carry out the following steps with TRC (Fig. 2):

1. TRC chooses an elliptic curve equation $E_p(a, b)$ and a base point P on the curve.
2. User U personally provides TRC with his/her identity ID, personal biometrics identity B, and password PW.
3. Then, TRC computes the following:
- TRC gets $(R, Q) = Gen(B)$ using a biometric device containing a fuzzy extractor and generates a random integer k.
- $E(k, R)$ is an encryption of k by secret key R, where

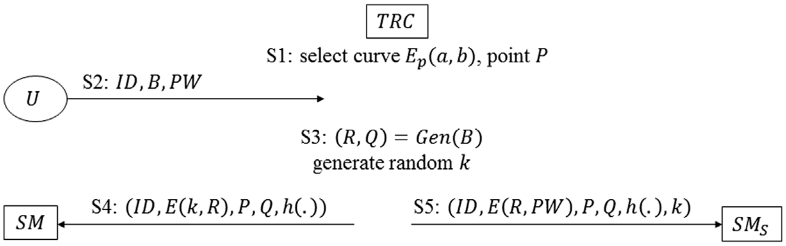
TRC
S1: select curve $E_p(a,b)$, point P
U
S2: ID, B, PW
S3: $(R,Q) = Gen(B)$
generate random k
SM
S4: $(ID, E(k,R), P, Q, h(.))$
S5: $(ID, E(R,PW), P, Q, h(.), k)$
SM_S

Fig. 2 Registration phase

k is a random number generated by TRC for user U.

- $E(R, PW)$ is an encryption of R by secret key PW.

4. TRC stores $(ID, E(k, R), P, E_p(a, b), Q, h(.))$ into smart card SM and sends the smart card to the user U via a secure channel.
5. TRC stores $(ID, E(R, PW), P, E_p(a, b), Q, h(.), k)$ into smart card SM_S and sends the smart card to the server via a secure channel.

6.2 Login and authentication phase

To login to server system S, a user U must carry out the following steps with smart card SM, and S with smart card SM_S (Fig. 3):

1. User U inserts smart card SM into a card reader and offers his/her ID' and PW'.
2. U imprints his/her biometrics identity B' on a specific device with a fuzzy extractor to get $R' = Rep(B', Q)$.
3. The smart card SM computes the following:

- $k' = D(E(k, R), R')$
- $M_1 = sP$, where $0 < s < p - 1$ is a random number generated by SM.
- $M_2 = h(k' \| M_1)$

4. SM sends (ID', M_1, M_2) to the remote server S.
5. S finds smart card SM_S which matches with ID' and forwards the message (ID', M_1, M_2) to SM_S.
6. SM_S receives the message and checks whether M_2 is equal to $h(k \| M_1)$. In the case, SM_S judges that U is the valid user and continues the phase. Otherwise, SM_S asks server S to terminate the session.
7. SM_S computes the following and sends (M_3, M_4) to SM:

- $M_3 = tP$, where $0 < t < p - 1$ is a random number generated by SM_S.
- $M_4 = h(E(R, PW) \| M_1 \| M_3)$

8. SM checks whether $h(E(R', PW') \| M_1 \| M_3)$ is equal to M_4, which means that the password and biometric information are correct. Otherwise, the protocol terminates.
9. SM and SM_S compute their session key $SK = sM_3 = tM_1$, respectively.
10. SM computes and sends $M_5 = \text{h}(SK \| M_3)$ to SM_S.
11. SM_S checks whether $M_5 = h(SK \| M_3)$. If this holds true, S successfully authenticates U. Otherwise, S rejects U's login request.

7. Security and performance analysis of the enhanced scheme

In this section, we analyze the security of the enhanced scheme based on the assumptions stated in Section 5 and show that the proposed scheme withstands all the attacks mentioned on Yeh's scheme.

7.1 Security analysis using extended BAN logic

BAN logic is a famous tool for analyzing information exchange protocol [25]. Unfortunately, it is not suitable to analyze public-key based authentication protocols [26-29]. Extended BAN logic is announced with an added functionality that can analyze public-key based authentication protocols [26]. We

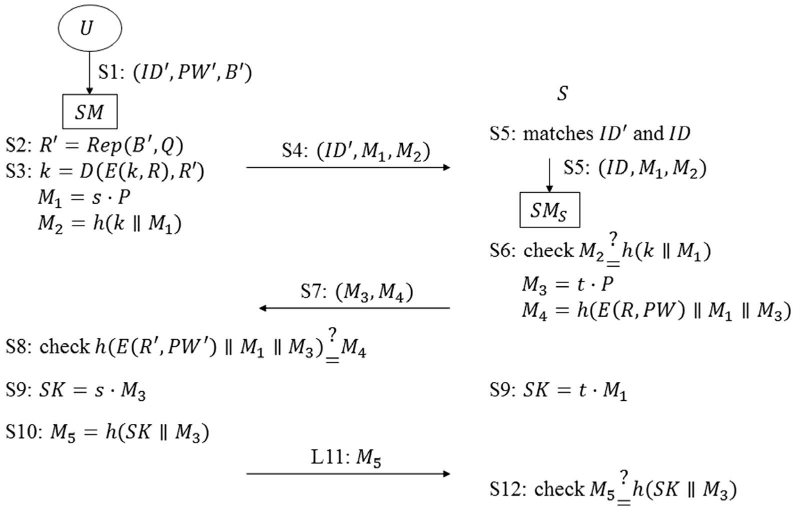

Fig. 3 Login and authentication phase

show that the login and authentication phase of the proposed scheme properly guarantees secret session key using the extended BAN logic. To apply extended BAN logic in our case, the following steps should be performed.

1. Idealize the proposed scheme in formal logic language.
2. Identify initial state assumptions of the proposed scheme.
3. Make annotated idealized protocol for each protocol statement with assertions.
4. Apply logical rules to the assumptions and assertions in order to discover the beliefs held by the parties in the proposed scheme.

Since BAN logic describes a condition or a state as logical equations, some mathematical notations are used as shown in Table 2.

Recall that a subject A generates and sends sP to a subject B, B returns tP, and $s(tP)$ (or $t(sP)$) is computed as a session key between A and B in Diffie-Hellman key-agreement on ECC. In the case, s is the A's private key-agreement key and $PK_\delta^{-1}(A)$ means that s is kept private by A. Also, sP is the A's public key-agreement key and $PK_\delta(A)$ means that B received sP safely from A.

Among rules in BAN logic [25] and its extension [26], we uses following rules:

R2 (Nonce-verification)

$$\frac{A|\equiv \#(X),\ A|\equiv B|\sim X}{A|\equiv B|\equiv X}$$

R3 (Jurisdiction rule)

$$\frac{A|\equiv B|\Rightarrow X,\ A|\equiv B|\equiv X}{A|\equiv X}$$

Table 2 BAN logic notations

Notation	Description
$A\|\equiv X$	entity A believes X
$A\overset{k}{\leftrightarrow}B$	A and B share a secret key k
$\#(X)$	data X is fresh
$A\lhd X$	A sees X
$A\|\sim X$	A said X
$A\|\Rightarrow X$	A controls X
$PK_\delta^{-1}(A)$	A's private key-agreement key is safe
$PK_\delta(A)$	A's public key-agreement key is safe

R7 (Once-said projection)

$$\frac{A|\equiv B|\sim (X,Y)}{A|\equiv B|\sim X,\ A|\equiv B|\sim Y}$$

R12 (Freshness propagation rule)

$$\frac{A|\equiv \#(X)}{A|\equiv \#(X,Y)}$$

R31 (DH key agreement)

$$\frac{A|\equiv PK_\delta^{-1}(A),\ A|\equiv PK_\delta(B),\ A|\equiv PK_\delta^{-1}(B)}{A|\equiv A\overset{k}{\leftrightarrow}B}$$

R40 (Message meaning on MAC)

$$\frac{A|\equiv A\overset{k}{\leftrightarrow}B,\ A\lhd X,\ A\lhd h(X,k)}{A|\equiv B|\sim X}$$

Rules R2-R12 are defined in [25]. Rule numbering used in [26] is applied in the paper, where Rule R31 is defined. Key k of Rule R31 is a passive session key in [26]. A session key is called *passive* if user A does not verify whether the opposite B has the key. The possession of the key is passively proved by following message exchanges encrypted by the key. If the key possession of the opposite entity is proved by the authentication protocol, the session key becomes *active*. Since our protocol does not verify key possession in authentication protocol, passive session key is sufficient. We define rule R40 in order to grant message meaning to Message Authentication Code (MAC). Given a message X and its MAC Y from sender B, where k is a secret key, a receiver A generates another MAC $h(X, k)$ using the shared secret key k and tests if Y is the same as $h(X, k)$. In the case, A assures that message X and MAC Y are sent by B. In order to apply the validity in BAN logic, rule R40 is defined.

BAN logic has been used to check whether mutual authentication is guaranteed or a fresh session key is securely shared by both sides. Although the proposed scheme uses the three factors to authenticate, goal of analysis is similar with other protocols: guaranteeing secure session key and freshness of the session key. Thus, goals for the logic are defined as follows:

G1 $U|\equiv U\overset{SK}{\longleftrightarrow}S$

G2 $U|\equiv \#(SK)$

G3 $S|\equiv U\overset{SK}{\longleftrightarrow}S$

G4 $S|\equiv \#(SK)$

Goals G1 and G3 mean that user U and server S believe a new session key SK. The goals imply that U and S are ready to authenticate each other or authenticate passively. Moreover, the session key SK should be guaranteed as a fresh key not used before, which is expressed as goals G2 and G4.

Idealized protocol of the proposed scheme is described as follows:

S4 $U \rightarrow S$: $(R_u,\ PK_\delta(U,\ R_u),\ h(PK_\delta(U,\ R_u)))$ where $R_u = sP$

S7 $U \leftarrow S$: $(R_s,\ PK_\delta(S,\ R_s),\ h(PK_\delta(S,\ R_s),\ R_u))$ where $R_s = tP$

S11 $U \rightarrow S$: $h(R_s)$

Basically, message in step S4 consists of key-agreement key sP and its MAC. Message in step S7 is similar with S4 except a response in hash value given challenge sP. Step S11 is necessary in order to check response from user U.

Before deriving the goals, we need to list initial assumptions:

A1 $U| \equiv \#(R_u)$

A2 $S| \equiv \#(R_s)$

A3 $S| \equiv U \overset{k}{\leftrightarrow} S$

A4 $U| \equiv U \xleftrightarrow{E(R,PW)} S$

A5 $U| \equiv PK_\delta^{-1}(U)$

A6 $S| \equiv PK_\delta^{-1}(S)$

A7 $U| \equiv PK_\delta^{-1}(S)$

A8 $S| \equiv PK_\delta^{-1}(U)$

A9 $U| \equiv S| \Rightarrow PK_\delta(S,R_s)$

A10 $S| \equiv U| \Rightarrow PK_\delta(U,R_u)$

Assumption A1 and A2 are reasonable because $R_u = sP$ and $R_s = tP$ are generated by U and S, respectively. Also, two values have role of nonces. Recall that key k used in A3 is a random number generated by TRC and is stored in SM_S. Although k is not possessed by user U, valid user can derive k at step S3 using the biometrics identity B. $E(R,\ PW)$ in assumption A4 is another shared key between U and S. The key is stored in smart card SM_S of server and it is generated from biometrics information R and password PW obtained from user U. Since private key-agreement keys s and t are generated by owner and are not transfer through communication channel, assumptions A5-A8 are reasonable. Assumption A9 and A10 mean that each owner of a public key-agreement key has jurisdiction over her/his key.

Proof of G1: By applying R40 (Message meaning on MAC) on A3 and S7,

$$\frac{U| \equiv U \overset{k}{\leftrightarrow} S,\ U \triangleleft (PK_\delta(S,R_s),R_u),\ h(k,PK_\delta(S,R_s),R_u)}{U| \equiv S| \sim (PK_\delta(S,R_s),R_u)}$$

and applying R7, we have

$$\frac{U| \equiv S| \sim (PK_\delta(S,R_s),R_u)}{U| \equiv PK_\delta(S,R_s)} \tag{1}$$

Assumption A1 is derives as follows by applying R12:

$$\begin{aligned} &U| \equiv \#(R_u) \\ &U| \equiv \#(PK_\delta(S,R_s),R_u) \\ &U| \equiv \#(PK_\delta(S,R_s)) \end{aligned} \tag{2}$$

Notice that above derivation is available because the values are contained in MAC. By applying R2 on Equations 2 and 1,

$$\frac{U| \equiv \#(PK_\delta(S,R_s)),\ U| \equiv S| \sim PK_\delta(S,R_s)}{U| \equiv S| \equiv PK_\delta(S,R_s)} \tag{3}$$

By applying rule R3 on Equation 3 and A9,

$$\frac{U| \equiv S| \equiv PK_\delta(S,R_s),\ U| \equiv S| \Rightarrow PK_\delta(S,R_s)}{U| \equiv PK_\delta(S,R_s)} \tag{4}$$

By applying R31 on A5, Equation 4, and A7, we get goal G1 as follows:

$$\frac{U| \equiv PK_\delta^{-1}(U),\ U| \equiv PK_\delta(S,R_s),\ U| \equiv PK_\delta^{-1}(S)}{U| \equiv U \overset{SK}{\longleftrightarrow} S}$$

Proof of G2 is the almost same as the proof of G1.

Proof of G3 starts from assumption A1, and

$$U| \equiv \#(R_u)$$

$$U| \equiv \#(sP) \tag{5}$$

$$U| \equiv \#(tsP) \tag{6}$$

$$U| \equiv \#(SK) \tag{7}$$

Equation 5 comes from $R_u = sP$, Equation 6 comes from rule R12, and Equation 7 is obtained since $SK = t(sP)$ for U. Proof of G4 is the almost same as the proof of G3. Thus, all goals are derived from our assumptions and protocol messages.

7.2 Security analysis on attack cases

Man-in-the-middle attack

In this attack, an attacker performs the following steps: during an authentication session between smart card SM and server S, the attacker intercepts the message $(M_1,\ M_2)$ sent from SM; the attacker computes $M_1' = uP$ where $0 < u < p - 1$ is a random number generated by attacker; the attacker sends message (M_1', M_2) to S. When S receives the modified message (M_1', M_2), it checks whether M_2 is the same as $h(k \| M_1')$. This is not the same. Therefore, S can detect this attack.

Server masquerading attack

In this attack, a fake server tries to masquerade a legal server by conducting the following steps: during an authentication session between smart card SM and server S, the fake server intercepts message S4: (ID', M_1, M_2) sent from SM; the fake server computes (M_3', M_4') where $M_3'=uP$ $(0 < u < p - 1)$ and $M_4' = h(r \| M_1 \| M_3')$. Since the fake server does not know R nor PW, random number r is used instead of $E(R, PW)$. In step S8, by checking $h(E(R', PW') \| M_1 \| M_4')$ and M_4', U know that the opposite is not valid server because two values are different.

User impersonation attack

In this attack, an attacker tries to impersonate a legal user by conducting the following steps: during an authentication session, the attacker eavesdrops a legal user's message (ID', M_1, M_2) sent from SM in step S4; later, the attacker initiates an authentication session with server S by sending the eavesdropped message (ID', M_1, M_2); S matches ID' with ID of a legal user in database and forwards it to SM_S at step S5; and SM_S checks MAC values and sends (M_3, M_4) at step S7; In order to get a new session key, the attacker should know s a private key-agreement key of U. Further process is blocked by the difficulty of discrete logarithm problem. Server S can detect the attacker at step S12 because the attacker cannot generate session key SK.

Stolen smart card attack

If a user's smart card is stolen by an attacker in our scheme, the attacker can breach the key for encryption and decryption by monitoring power consumption of smart card [30,31]. A decryption key can be used is R' of operation $k'=D(E(k, R), R')$ in step 3 of subsection 6.2. Since there is no process to take advantage of R', any attacks using the secret values in the stolen smart card can be prevented.

Insider attack

This attack occurs with the help of a server administrator or malicious softwares such as viruses, spywares, and trojan inside server. Although an attacker can get secret values stored in server, this attack can be prevented in the proposed scheme. Because secret values of server are stored in server smart cards and read/write accesses are restricted, they are secure against malicious softwares.

Parallel processing capability

The use of multiple server smart cards improve the server performance because authentication protocols are processed in parallel by the server smart cards and the server just transfers the data between the smart cards and the users.

In summary, Table 3 shows the comparison of functionalities between Yeh's scheme, Chaudhry's scheme, Jiang's scheme [20], and the proposed scheme.

7.3 Performance analysis

Table 4 evaluates and compares performance of our scheme with that of Yeh's scheme and Jiang's scheme in terms of computational overheads and communicational overheads. The communicational overhead is calculated in terms of total number of messages transferred and total number of information contained in them as a number within parenthesis. In the registration phase, the proposed scheme needs only two symmetric key operations ($2SK$) to complete whereas Yeh's scheme needs four hash operations ($4H$), one ECC point multiplication operation ($1PM$), and one symmetric key operation ($1SK$). In the login

Table 3 Functionality comparisons

Functionality	Yeh's scheme	Chaudhry's scheme	Jiang's scheme	Proposed scheme
Provide mutual authentication	Yes	Yes	No	Yes
Resist man-in-the-middle attack	Yes	Yes	Partial	Yes
Resist insider attack	Yes	Yes	Yes	Yes
Resist user impersonation attack	No	Yes	Yes	Yes
Resist server masquerading attack	No	Yes	Partial	Yes
Resist stolen smart card attack	Yes	Yes	Yes	Yes
Provide parallel processing capability	No	No	No	Yes

Table 4 Performance comparisons

H stands for hashing, *PM* stands for ECC point multiplication operation, *PA* stands for ECC point addition operation, *SK* stands for symmetric key operation, *RK* stands for Rabin crypto-operation and *P* stands for transferring a physical object, smart card. The numbers in the communication overhead mean the number of messages (and the number of terms in the messages).

Overheads		Yeh's scheme	Jiang's scheme	Proposed scheme
Computation	Registration phase	4*H* + 1*PM* + 1*SK*	7*H*	2*SK*
	Login & authentication phase	1*SK* + 2*H* + 20*PA* + 4*PM*	2*RK* + 26*H*	6*H* + 2*SK* + 4*PM*
Communication	Registration phase	2 (3 + *P*)	2 (2 + 2*P*)	3 (3 + 2*P*)
	Login & authentication phase	4 (8)	4 (14)	3 (6)

phase and authentication phase, the proposed scheme needs two symmetric key operations (2*SK*), six hash operations (6*H*), and four ECC point multiplication operations (4*PM*) to complete, whereas Yeh's scheme needs one symmetric key operation (1*SK*), two hash operation (2*H*), four ECC point multiplication operations (4*PM*), and twenty ECC point addition operations (20*PA*). Because Yeh's scheme needs more operations than the proposed scheme needs and ECC operations involving modulo are slower than symmetric key operations, it is obvious that the proposed scheme is more efficient than Yeh's scheme in all phases except registration phase where physical transfer of smart card is required (2*P*).

8. Conclusion

In this study, we reviewed and analysed the security of Yeh's scheme. We showed that this scheme still retains certain flaws that make it insecure against various types of attacks. Therefore, we redesigned the system model such that multiple smart cards process the authentication phase on the server side. The proposed scheme overcomes the previous security weaknesses and made the system more secure against various types of attacks. Further, it also improved the system performance significantly.

References

[1] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, Vol. 24, No. 11, pp. 770-772, 1981.

[2] L. Fan, J.-H. Li, and H.-W. Zhu, "An enhancement of timestamp-based password authentication scheme," *Computers & Security*, Vol. 21, No. 7, pp. 665-667, 2002.

[3] J.-J. Shen, C.-W. Lin, and M.-S. Hwang, "Security enhancement for the timestamp-based password authentication scheme using smart cards," *Computers & Security*, Vol. 22, No. 7, pp. 591-595, 2003.

[4] H.-M. Sun, "An efficient remote use authentication scheme using smart cards," *Consumer Electronics, IEEE Transactions on*, Vol. 46, No. 4, pp. 958-961, 2000.

[5] S.-T. Wu and B.-C. Chieu, "A user friendly remote authentication scheme with smart cards," *Computers & Security*, Vol. 22, No. 6, pp. 547-550, 2003.

[6] N.-Y. Lee and Y.-C. Chiu, "Improved remote authentication scheme with smart card," *Computer Standards & Interfaces*, Vol. 27, No. 2, pp. 177-180, 2005.

[7] E.-J. Yoon, E.-K. Ryu, and K.-Y. Yoo, "Further improvement of an efficient password based remote user authentication scheme using smart cards," *Consumer Electronics, IEEE Transactions on*, Vol. 50, No. 2, pp. 612-614, 2004.

[8] C.-T. Li and M.-S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards," *Journal of Network and Computer Applications*, Vol. 33, No. 1, pp. 1-5, 2010.

[9] A. K. Jain, P. Flynn, and A. A. Ross, *Handbook of biometrics*. Springer, 2007.

[10] S. Prabhakar, S. Pankanti, and A. K. Jain, "Biometric recognition: Security and privacy concerns," *IEEE Security & Privacy*, Vol. 1, No. 2, pp. 33-42, 2003.

[11] A. K. Das, "Analysis and improvement on an efficient biometric-based remote user authentication scheme using smart cards," *IET Information Security,* Vol. 5, No. 3, pp. 145-151, 2011.

[12] Y. An, "Security analysis and enhancements of an effective biometric-based remote user authentication scheme using smart cards," *BioMed Research International*, Vol. 2012.

[13] X. Li, J.-W. Niu, J. Ma, W.-D. Wang, and C.-L. Liu, "Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart cards," *Journal of Network and Computer Applications*, Vol. 34, No. 1, pp. 73-79, 2011.

[14] C.-H. Lin and Y.-Y. Lai, "A flexible biometrics remote user authentication scheme," *Computer Standards & Interfaces*, Vol. 27, No. 1, pp. 19-23, 2004.

[15] H.-L. Yeh, T.-H. Chen, K.-J. Hu, and W.-K. Shih, "Robust elliptic curve cryptography-based three factor user authentication providing privacy of biometric data," *IET information security*, Vol. 7, No. 3, pp. 247-252, 2013.

[16] C.-I. Fan and Y.-H. Lin, "Probably secure remote truly three-factor authentication scheme with privacy protection on biometrics," *Information Forensics and Security, IEEE Transactions on*, Vol. 4, No. 4, pp. 933-945, 2009.

[17] Q. Jiang, M. K. Khan, X. Lu, J. Ma, and D. He, "A privacy preserving three-factor authentication protocol for e-health clouds," *The Journal of Supercomputing*, Vol. 72, No. 10, pp. 3826-3849, 2016.

[18] P. Bergamo, P. D'Arco, A. De Santis, and L. Kocarev, "Security of public-key cryptosystems based on chebyshev polynomials," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 52, No. 7, pp. 1382-1393, 2005.

[19] S. A. Chaudhry, H. Naqvi, M. S. Farash, T. Shon, and M. Sher, "An improved and robust biometrics-based three factor authentication scheme for multi-server environments," *The Journal of Supercomputing*, pp. 1-17, 2015.

[20] Q. Jiang, S. Zeadally, J. Ma, and D. He, "Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks," *IEEE Access*, Vol. 5, pp. 3376-3392, 2017.

[21] D. Hankerson, S. Vanstone, and A. J. Menezes, *Guide to elliptic curve cryptography*. Springer, 2004.

[22] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *in Advances in cryptology-Eurocrypt 2004*, pp. 523-540, Springer, 2004.

[23] Q. Jiang, F. Wei, S. Fu, J. Ma, G. Li, and A. Alelaiwi, "Robust extended chaotic maps-based three-factor authentication scheme preserving biometric template privacy," *Nonlinear Dynamics*, Vol. 83, No. 4, pp. 2085-2101, 2016.

[24] A. Watson, "Biometrics: easy to steal, hard to regain identity," *Nature*, Vol. 449, No. 7162, p. 535, 2007.

[25] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proc. of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 426, pp. 233-271, The Royal Society, 1989.

[26] P. van Oorschot, "Extending cryptographic logics of belief to key agreement protocols," *Proc. of the 1st ACM Conference on Computer and Communications Security*, pp. 232-243, ACM, 1993.

[27] P. Syverson and I. Cervesato, "The logic of authentication protocols," *International School on Foundations of Security Analysis and Design*, pp. 63-137, Springer, 2000.

[28] S. Yang and X. Li, "A limitation of ban logic analysis on a man-in-the-middle attack," *Journal of Information and Computing Science*, Vol. 1, No. 3, pp. 131-138, 2006.

[29] A. Hunter and J. P. Delgrande, "Belief change and cryptographic protocol verification," *Proc. of the National Conference on Artificial Intellience*, Vol. 22, p. 427, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 2007.

[30] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," *Advances in Cryptology-CRYPTO'99*, pp. 388-397, Springer, 1999.

[31] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smartcard security under the threat of power analysis attacks," *Computers, IEEE Transactions on*, Vol. 51, No. 5, pp. 541-552, 2002.

오중타잉

2011년 미국 TROY대학교 컴퓨터과학과(학사). 2015년 금오공과대학교 컴퓨터공학과(석사). 2017년~현재 금오공과대학교 IT융합학과 박사과정. 관심분야는 VANET에서의 Traffic Light 스케줄링

최태영

1991년 고려대학교 수학교육과(학사). 1996년 POSTECH 컴퓨터공학과(석사). 2002년 POSTECH 컴퓨터공학과(박사). 2002년~현재 금오공과대학교 컴퓨터공학과 교수. 관심분야는 병렬 및 분산 알고리즘, 클라우드 컴퓨팅, IoT 시스템, 인공 지능