Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2014, Article ID 423930, 6 pages
http://dx.doi.org/10.1155/2014/423930

Research Article

Two-Factor User Authentication with Key Agreement Scheme Based on Elliptic Curve Cryptosystem

Juan Qu[1] and Xiao-Ling Tan[2]

[1] *School of Mathematics and Statistics, Chongqing Three Gorges University, Chongqing 404000, China*
[2] *College of Electronic and Information Engineering, Chongqing Three Gorges University, Chongqing 404000, China*

Correspondence should be addressed to Juan Qu; qulujuan@163.com

Received 26 November 2013; Accepted 2 March 2014; Published 19 May 2014

Academic Editor: Dharma Agrawal

A password authentication scheme using smart card is called two-factor authentication scheme. Two-factor authentication scheme is the most accepted and commonly used mechanism that provides the authorized users a secure and efficient method for accessing resources over insecure communication channel. Up to now, various two-factor user authentication schemes have been proposed. However, most of them are vulnerable to smart card loss attack, offline password guessing attack, impersonation attack, and so on. In this paper, we design a password remote user authentication with key agreement scheme using elliptic curve cryptosystem. Security analysis shows that the proposed scheme has high level of security. Moreover, the proposed scheme is more practical and secure in contrast to some related schemes.

1. Introduction

Due to the rapid growth of Internet technology, more and more people use the network to acquire desired services and exchange data. Remote user authentication is one of the most important mechanisms to identify the legal user over insecure communication network. Since Lamport [1] proposed the first password-based remote user authentication scheme, many password-based single-factor authentication schemes [2–7] have been proposed in the literatures. However, most of password-based single-factor authentication schemes have various security pitfalls. In order to provide better security of the system, Hwang and Li [8] developed a two-factor authentication scheme in 2000. Two-factor authentication scheme is that the authentication schemes are based on the user's password and smart card. In the two-factor authentication scheme, when the user wants to access resources on a server, he/she inserts the smart card into a card reader and inputs his/her password. Then the smart card using the user's password generates a login request message and sends the request to the server. When receiving the login request, the server verifies the validity of the request message. In 2009, Xu et al. [9] proposed a smart-card-based password authentication scheme. They claimed that their scheme could resist stolen smart card attack. But, in 2010, Sood et al. [10] and Song [11] pointed out that Xu et al.'s scheme was vulnerable to impersonation attack and internal attack. And they proposed the improved scheme, respectively. In 2012, Chen et al. [12] analyzed Xu et al.'s scheme and pointed out that any user can impersonate other users and fool the service providing server. Meanwhile, Chen et al. [12] pointed out the security flaws of Sood et al. [10] and Song's [11] scheme. According to Chen et al. [12], Sood et al.'s scheme [10] does not guarantee mutual authentication during authentication phase, and Song's scheme is susceptible to an internal offline guessing attack. Then, an improved scheme is presented in Chen et al.'s scheme paper. Unfortunately, in 2013, Kumari and Khan [13] pointed out that Chen et al.'s scheme cannot withstand user impersonation attack, server spoofing attack, and offline password guessing attack. Besides, Chen et al.'s scheme does not provide important features such as user anonymity, confidentiality to air messages, and revocation of lost/stolen smart card. Also in 2013, Jiang et al. [14] still pointed out that Chen et al. was insecure against offline dictionary attacks and proposed an improved authentication protocol without using smart card.

In 2009, Yang and Chang [15] proposed an ID-based remote mutual authentication with key agreement scheme on ECC. In their scheme, the server and the user accomplish mutual authentication through the user's unique identity. And they claimed that the computation costs and the number of communication costs of their scheme are less than some related schemes. Nevertheless, Islam and Biswas [16] stated that Yang and Chang's scheme [15] is vulnerable to replay attack, known session-specific temporary information attack. Besides, Yang and Chang's scheme [15] does not provide user's anonymity and session key forward secrecy. Islam and Biswas further found that Yang and Chang's scheme does not define how to revoke the authentication key with same identity. Later, Truong et al. [17] pointed out that Islam and Biswas's [16] scheme still cannot resist known session-specific temporary information attack. In this paper, we present a two-factor user authentication with key agreement scheme using elliptic curve cryptosystem based on Yang and Islam's scheme. Security analysis shows that our proposed scheme can resist various attacks.

The rest of the paper is organized as follows. Section 2 introduces some preliminaries. In Section 3, the proposed two-factor authentication with key agreement scheme is described; the corresponding security analysis is given in Section 4. Finally, we conclude this paper in Section 5.

2. Preliminaries

In this section, we will introduce the basic concepts of elliptic curve cryptosystem (ECC). In all elliptic curve cryptosystem, the elliptic curve equation is defined as the form of $E_p(a,b)$: $y^2 = x^3 + ax + b(\underline{\mathrm{mod}}\, p)$. Given an integer $s \in F_p^*$ and a point $P \in E_p(a,b)$, the point multiplication sP over $E_p(a,b)$ can be defined as $s \cdot P = P + P + P + \cdots + P$ (s times). Generally, the security of ECC relies on the difficulties of the following problems.

Definition 1. Given two points P and Q over $E_p(a,b)$, the elliptic curve discrete logarithm problem (ECDLP) is to find an integer $s \in F_p^*$ such that $Q = s \cdot P$.

Definition 2. Given three points P, $s \cdot P$, and $t \cdot P$ over $E_p(a,b)$ for $s, t \in F_p^*$, the computational Diffie-Hellman problem (CDLP) is to find the point $(st)P$ over $E_p(a,b)$.

Definition 3. Given two points P and $Q = s \cdot P + t \cdot P$ over F_p^* for $s, t \in F_p^*$, the elliptic curve factorization problem (ECFP) is to find two points $s \cdot P$ and $t \cdot P$ over $E_p(a,b)$.

3. The Proposed Scheme

In this section, we will propose a two-factor user authentication with key agreement scheme based on elliptic curve cryptosystem. The notations used in proposed scheme are listed in notations. And the detailed information is described as follows and shown in Figure 1. Our scheme includes five phases: system initializing phase, the registration phase, login phase, authentication phase, and password change phase. The details of these phases are as follows.

3.1. System Initializing Phase

Step 1. The server S chooses an elliptic curve equation $E_p(a,b)$ and a base point P with the order n over $E_p(a,b)$.

Step 2. The server S selects the private key $q_s \in [1, n-1]$ and computes the corresponding public key $Q_s = q_s \cdot P$.

Step 3. The server S chooses three one-way hash functions: $H_1 : \{0,1\}^* \rightarrow G_p$, $H_2 : G_p \times G_p \rightarrow Z_p^*$, $H_3 : \{0,1\}^* \times G_p \times G_p \rightarrow \{0,1\}^k$.

Step 4. The server S publishes $\{E_p(a,b), P, Q_s, H_1, H_2, H_3\}$.

3.2. Registration Phase.

If the user U wants to become a legal user of the system, he has to submit the related information to the server S. The detail of the registration phase is described in the following steps.

Step 1. The user U generates his own identity ID_u and PW_u and a random number $b_u \in [1, n-1]$; then the user U submits ID_u and $H_1(\mathrm{PW}_u \parallel b_u) \cdot P$ to the server S over a secure communication channel.

Step 2. The server S computes $\mathrm{AID}_u = (q_s + 1) \cdot H_1(\mathrm{PW}_u \parallel b_u) \cdot P$ and $\mathrm{BID}_u = H_2(H_1(\mathrm{ID}_u) \parallel H_1(\mathrm{PW}_u \parallel b_u) \cdot P)$.

Step 3. The server S stores $\{\mathrm{AID}_u, \mathrm{BID}_u\}$ into a smart card and issues the smart card to the user U via a secure channel.

Step 4. On receiving the smart card, the user U enters the random b_u into the smart card, and the smart card contains $\{\mathrm{AID}_u, \mathrm{BID}_u, b_u\}$.

3.3. Login Phase.

When the user U wants to login to the server S, he/she inserts his smart card into the card reader of a terminal and inputs ID_u and PW_u. Then, the smart card performs the following steps for login.

Step 1. The user computes $\mathrm{BID}'_u = H_2(H_1(\mathrm{ID}_u) \parallel (H_1(\mathrm{PW}_u \parallel b_u) \cdot P))$ and checks if $\mathrm{BID}'_u = \mathrm{BID}_u$. If it holds, it means that the user U inputs the correct identity and password. Otherwise, the smart card terminates the session.

Step 2. The user selects a random $r_u \in [1, n-1]$ and computes $\mathrm{TID}_u = \mathrm{AID}_u - H_1(\mathrm{PW}_u \parallel b_u) \cdot P$, $M = r_u \cdot Q_s$, $\mathrm{CID}_u = \mathrm{ID}_u \oplus H_2(M \parallel \mathrm{TID}_u)$, $\mathrm{DID}_u = M + H_1(\mathrm{PW}_u \parallel b_u) \cdot P$, and $\mathrm{EID}_u = H_3(\mathrm{ID}_u \parallel M \parallel R)$, where $R = r_u \cdot P$.

Step 3. The user submits the login request message $M_1 = \{\mathrm{CID}_u, \mathrm{DID}_u, \mathrm{EID}_u, R\}$ to the server S.

3.4. Authentication and Key Agreement Phase.

Upon receiving the login message $M_1 = \{\mathrm{CID}_u, \mathrm{DID}_u, \mathrm{EID}_u, R\}$ from the user U, the server S performs the following steps to mutual authentication.

Step 1. The server S computes $M' = q_s \cdot R$, $H_1(\mathrm{PW}_u \parallel b_u) \cdot P = \mathrm{DID}_u - M'$, $\mathrm{TID}'_u = q_s \cdot H_1(\mathrm{PW}_u \parallel b_u) \cdot P$, and

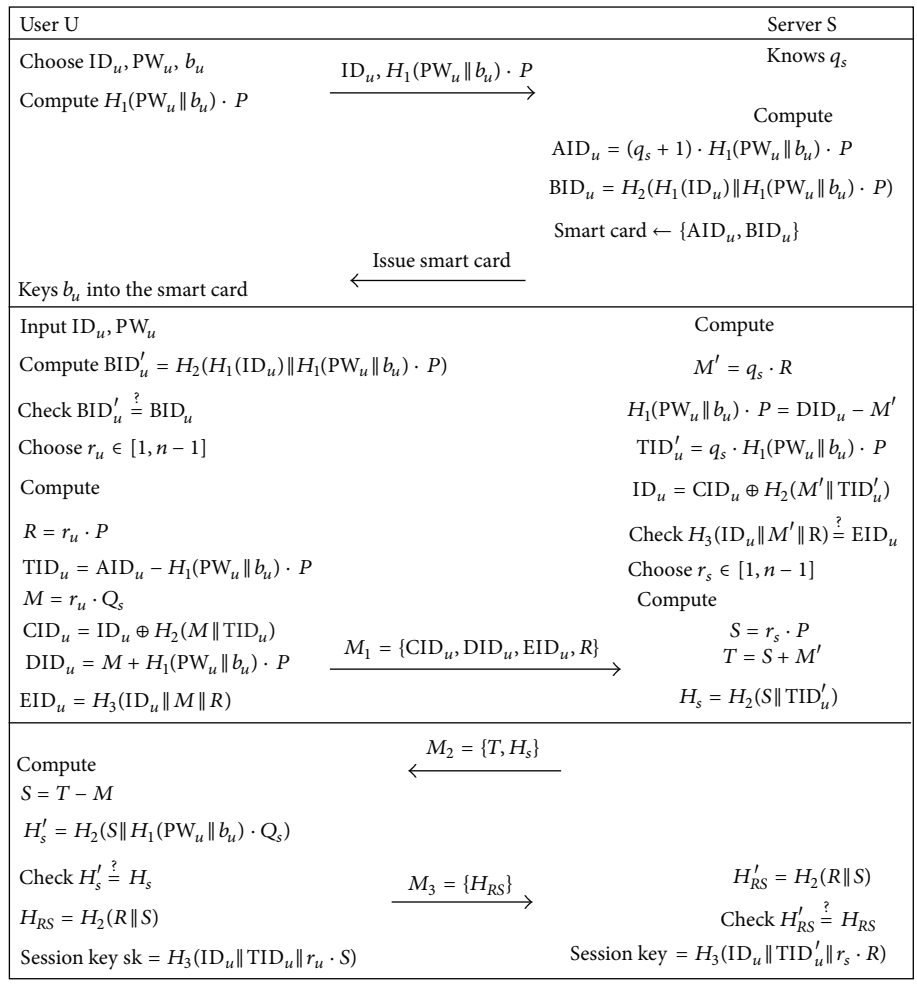

Figure 1: The proposed scheme.

$ID_u = CID_u \oplus H_2(M' \| TID'_u)$ and then checks whether $H_3(ID_u \| M' \| R) \stackrel{?}{=} EID_u$. If they are equal, the validity of the user U is authenticated by the server S. Otherwise, the session is terminated by the server S.

Step 2. The server S chooses a random number $r_s \in [1, n-1]$ and computes $S = r_s \cdot P$, $T = S + M'$, and $H_s = H_2(S \| TID'_u)$.

Step 3. The server S sends the authentication message $M_2 = \{T, H_s\}$ to the user U.

Step 4. After receiving $M_2 = \{T, H_s\}$, the user U computes $S = T - M$, $H'_s = H_2(S \| H_1(PW_u \| b_u) \cdot Q_s)$ and checks if $H'_s \stackrel{?}{=} H_s$ holds. If the equation holds, the server S is authenticated by the user U. And the user U sends the message $M_3 = \{H_{RS}\}$, where $H_{RS} = H_2(R \| S)$.

Step 5. On receiving the message $\{H_{RS}\}$, the server S computes $H'_{RS} = H_2(R \| S)$ and compares it with received H_{RS}. If it holds, the server and the user achieve mutual authentication. Otherwise, the smart card terminates the session.

Step 6. At last, the user U and the server S can compute the session key $sk = H_3(ID_u \| TID_u \| r_u \cdot S)$ and $sk = H_3(ID_u \| TID'_u \| r_s \cdot R)$.

3.5. Password Change Phase. When the user U wants to change his/her password PW_u to a new one PW_u^{new}, the user U can update his/her password by performing the following steps without the help of the server S.

Step 1. The user U inserts his smart card into a card reader and inputs ID_u and PW_u. The smart card computes $BID'_u = H_2(H_1(ID_u) \| (H_1(PW_u \| b_u) \cdot P))$ and checks if the BID'_u is the same as BID_u. If both values are the same, the user inputs a new password PW_u^{new}.

Step 2. The smart card computes $AID_u^{new} = H_1(PW_u \| b_u)^{-1} \cdot AID_u \cdot H_1(PW_u^{new} \| b_u)$, $BID_u^{new} = H_2(H_1(ID_u) \| H_1(PW_u^{new} \| b_u) \cdot P)$.

Step 3. At last, the smart card replaces AID_U and BID_U with AID_U^{new} and BID_U^{new}, respectively.

4. Security Analysis of Our Scheme

At first, we discuss the security features of the proposed authentication with key agreement scheme in this section. Then we evaluate the performance of the proposed scheme and make comparisons with some related works.

4.1. Mutual Authentication with Session Key Agreement. In the proposed scheme, the user sends the login request message $M_1 = \{\mathrm{CID}_u, \mathrm{DID}_u, \mathrm{EID}_u, R\}$ to S; after receiving the message M_1, the server authenticates the user by checking if the equation $H_3(\mathrm{ID}_u \parallel M' \parallel R) = \mathrm{EID}_u$ holds or not. If the computed value $H_3(\mathrm{ID}_u \parallel M' \parallel R)$ equals the received value EID_u, the server confirms that the user is valid. Then the server replies the message $M_2 = \{T, H_s\}$ to the user. When the user receives the message, he/she authenticates the server by comparing the computed value $H_s' = H_2(S \parallel \mathrm{TID}_u)$ with the received value H_s. If it is equal, the user confirms that the server is legitimate. At last, the server S authenticates the user U after checking if the equation $H_2(R \parallel S) = H_{RS}$ holds or not. Only when all previous equations are satisfied, the session continues and the communication parties share a session key $sk = H_3(\mathrm{ID}_u \parallel \mathrm{TID}_u \parallel r_u \cdot S) = H_3(\mathrm{ID}_u \parallel \mathrm{TID}_u \parallel r_s \cdot R)$. During the aforementioned discussion, the proposed scheme can achieve mutual authentication with session key agreement.

4.2. Forward Secrecy. Forward secrecy means that if the long-term private keys related to participating entities (e.g., the server's secret key q_s and user's password PW_u) are compromised, the secrecy of the previous session keys should not be affected. In the proposed scheme, the session key $sk = H_3(\mathrm{ID}_u \parallel \mathrm{TID}_u \parallel r_u \cdot S) = H_3(\mathrm{ID}_u \parallel \mathrm{TID}_u' \parallel r_s \cdot R)$, where $r_u \cdot S = r_s \cdot R = r_u \cdot \mathrm{r}_s \cdot P$, relies on the random values r_u and r_s. r_u and r_s are independently generated in each session and they have no relation with the server's secret key q_s and user's password PW_u. So, the attacker cannot compute any previous sk without the random value r_u chosen by the user U and the random value r_s chosen by the server S. On the other hand, even if the attacker knows $R = r_u \cdot P$ and $S = r_s \cdot P$ from the public channel, he/she still cannot get the session key sk because he/she will face solving the computational Diffie-Hellman problem. Thus, the proposed scheme provides forward secrecy.

4.3. User Anonymity. In the proposed scheme, user's identity is not stored in smart card and is also not transmitted via plain text form. In fact, user's identity is submitted with $\mathrm{CID}_u = \mathrm{ID}_u \oplus H_2(M \parallel \mathrm{TID}_u)$, which is changed for each login phase. Even if the attacker eavesdrops the login request message $M_1 = \{\mathrm{CID}_u, \mathrm{DID}_u, \mathrm{EID}_u, R\}$ and the authentication messages $M_2 = \{T, H_s\}$ and $M_3 = \{H_{RS}\}$, the attacker has no way to know the user's identity ID_u. This is because the attacker cannot procedure ID_u out of $\mathrm{CID}_u = \mathrm{ID}_u \oplus H_2(M \parallel \mathrm{TID}_u)$ without knowing the server's secret key q_s and user's password PW_u. Thus, the proposed scheme provides the user anonymity.

4.4. Resisting Server Spoofing Attack. In the proposed scheme, if the attacker wants to masquerade as the remote server S to cheat the user U, he/she has to generate a valid message $M_2 = \{T, H_s\}$, where $T = S + M'$, $H_s = H_2(S \parallel \mathrm{TID}_u')$. That is to say, the attacker must get the values M' and TID_u' to compute a valid message $\{T, H_s\}$. However, the attacker cannot compute the values $M' = q_s \cdot R$ and $\mathrm{TID}_u' = q_s \cdot H_1(\mathrm{PW}_u \parallel b_u) \cdot P$ without knowing the private key of the server S and user's password PW_u. Therefore, our scheme is secure against the server spoofing attack.

4.5. Resisting Insider Attack. Insider attack means that the user U may register to more than one server with the same identity and password; then a privileged insider of the server can impersonate the user and access the other servers by making a valid login quest. In the registration of the proposed scheme, the user U freely chooses his/her identity ID_u and password PW_u and submits ID_u and $H_1(\mathrm{PW}_u \parallel b_u) \cdot P$ to the server S. The server S cannot obtain the password PW_u from $H_1(\mathrm{PW}_u \parallel b_u) \cdot P$ since he/she will face CDL (computational discrete logarithm) problem. Therefore, the proposed scheme can resist insider attack.

4.6. Resisting Smart Card Loss Attack. Assume that the user U's smart card is lost or stolen, and the attacker can extract the information $\{\mathrm{AID}_u, \mathrm{BID}_u, b_u\}$ stored in the smart card, where $\mathrm{AID}_u = (q_s + 1) \cdot H_1(\mathrm{PW}_u \parallel b_u) \cdot P$, $\mathrm{BID}_u = H_2(H_1(\mathrm{ID}_u) \parallel (H_1(\mathrm{PW}_u \parallel b_u) \cdot P))$. On the one hand, the attacker cannot guess user's password PW_u from AID_u and BID_u since it is protected by one-way hash function. On the other hand, the attacker cannot fabricate a valid login request message or compute the session key using the stolen smart card. Besides, it is impossible for the attacker to update the user's password. This is because the attacker must have the real identity ID_u and PW_u to pass the verification $\mathrm{BID}_u' \stackrel{?}{=} \mathrm{BID}_u$. Therefore, the proposed scheme is secure against the stolen smart card attack.

4.7. Resisting Impersonation Attack. If the attacker wants to impersonate as a legitimate user U to pass the authentication of the server S, he/she has to forge a valid login request message $M_1 = \{\mathrm{CID}_u, \mathrm{DID}_u, \mathrm{EID}_u, R\}$. Assume that the attacker possesses the user's smart card and intercepts the user's previous login request message, the attacker attempts to impersonate the user U and sends the login message login message $(\mathrm{CID}_u', \mathrm{DID}_u', \mathrm{EID}_u', R')$. However, this impersonation attempt will fail in step 1 of the authentication phase, since he/she has no way to obtain the values of ID_u, q_s, and TID_u. Therefore, the proposed scheme is secure against impersonate attack.

4.8. No Key Control. In proposed scheme, the session key consists of ID_u, TID_u, and $r_u \cdot r_s \cdot P$, where r_u and r_s are, respectively, provided by the user and the server. Therefore, the fairness of the session key is guaranteed and either party is in vain attempting to preselect or control the session key.

Table 1: Computation cost comparison among our scheme and others.

	Yang and Chang [15]	Islam and Biswas [16]	Our scheme
Registration phase	$1T_m + 1T_h$	$1T_m + 1T_h$	$2T_m + 1T_a + 3T_h$
Authentication phase	$8T_m + 5T_a + 8T_h$	$7T_m + 4T_a + 6T_h$	$9T_m + 5T_a + 13T_h$
Total computation cost	$9T_m + 5T_a + 9T_h$	$8T_m + 4T_a + 7T_h$	$11T_m + 8T_a + 14T_h$

T_m: the time complexity of ECC multiplication operation; T_a: the time complexity of ECC addition/subtraction operation; T_h: the time complexity of hashing function operation.

Table 2: Security properties comparison.

	Our scheme	Yang and Chang's scheme [15]	Islam and Biswas's scheme [16]
Mutual authentication	Yes	Yes	Yes
Key agreement	Yes	Yes	Yes
Forward secrecy	Yes	No	Yes
User anonymity	Yes	No	Yes
Server spoofing attack	Yes	Yes	Yes
Insider attack	Yes	Yes	Yes
Impersonation attack	Yes	Yes	Yes
Stolen smart card attack	Yes	N/A	N/A
No key control	Yes	Yes	Yes
Replay attack	Yes	No	Yes
No key control	Yes	No	No

4.9. Resisting Replay Attack. Replay attack means that the attacker may impersonate the legitimate user by reusing the information obtained from the previous run protocols. In the proposed scheme, r_u and r_s are the random numbers that are selected by the user U and the server S, respectively. And they are different for each session. So, the messages exposed in public channel are different in each session. Thus, the proposed scheme can prevent replay attack.

4.10. Known Session-Specific Temporary Information Attack. Known session-specific temporary information attack means that if the session ephemeral secrets are exposed to an adversary accidentally, this exposure should not compromise the generated session key. In the proposed scheme, if the session ephemeral secrets r_u and r_s are leaked, the adversary cannot obtain the session key $sk = H_3(\mathrm{ID}_u \parallel \mathrm{TID}_u \parallel r_u \cdot r_s \cdot P)$. This is because the adversary has no way to know ID_u and TID_u. Hence, the proposed scheme can resist known session-specific temporary information attack.

5. Performance and Functionality Analysis

In this section, we compare the efficiency and security properties of the proposed scheme with related schemes proposed by Yang and Chang [15] and Islam and Biswas [16].

Table 1 is about the computation cost comparison between our proposed scheme and other related schemes. We only consider ECC multiplication operation, ECC addition/subtraction operation, and hash operation. And the computation cost of XOR operation can be ignored when compared to these operations. According to Table 1, the cost of our proposed scheme is slightly higher than other schemes. However, our proposed scheme can achieve all security properties as mentioned in Table 2. We summarize security properties comparisons between the proposed scheme and two previous schemes in Table 2. It is easy to draw that our proposed scheme can achieve all security requirements. So, the proposed scheme has stronger security.

6. Conclusion

In this paper, we have proposed a two-factor user authentication with key agreement scheme based on elliptic curve cryptosystem. The analysis shows that the computation costs of our proposed scheme are slightly higher than other schemes; however, our scheme can accomplish most desired security goals compared with some related schemes. As a result, our scheme is more secure and practical for real-life use.

Notations

S: Sever
U: User
(q_s, Q_s): The server S's private/public key pair, where $Q_s = q_s \cdot P$
ID_u: Identity of the user U
PW_u: Password of the user U
$H_1(\cdot)$: A secure one-way hash function, where $H_1 : 0, 1^* \rightarrow G_p$
$H_2(\cdot)$: A secure one-way hash function, where $H_2 : G_p \times G_p \rightarrow Z_p^*$
$H_3(\cdot)$: A secure one-way hash function, where $H_3 : \{0,1\}^* \times G_p \times G_p \rightarrow \{0,1\}^k$
r_u: A secret number chosen by the user U
r_s: A secret number chosen by the server S
$\parallel$: Message concatenation operation
F_p: A finite field
$E_p(a, b)$: An elliptic curve defined on finite field F_p with prime order n
G: Additive group of points on E over a finite field F_p
P: Generator of G.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

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

[2] M. Peyravian and N. Zunic, "Methods for protecting password transmission," *Computers and Security*, vol. 19, no. 5, pp. 466–469, 2000.

[3] C. C. Lee, L. H. Li, and M. S. Hwang, "A remote user authentication scheme using hash functions," *ACM SIGOPS Operating Systems Review*, vol. 36, no. 4, pp. 23–29, 2002.

[4] KuWC, C. M. Chen, and H. L. Lee, "Weaknesses of Lee-Li-Hwang's hash-based password authentication scheme," *Operating Systems Review*, vol. 37, no. 4, pp. 19–25, 2003.

[5] E. J. Yoon, E. K. Ruy, and K. Y. Roo, "A secure user authentication scheme using hash functions," *ACM Operating Systems Review*, vol. 38, no. 2, pp. 62–68, 2004.

[6] Qi Jiang, Jianfeng Ma, Guangsong Li, and Li Yang, "Robust two-factor au- thentication and key agreement preserving user privacy," *InternatioNal Journal of Network Security*, vol. 16, no. 3, pp. 229–240, 2014.

[7] H.-H. Ou, I.-C. Lin, M.-S. Hwang, and J.-K. Jan, "TK-AKA: using temporary key on authentication and key agreement protocol on UMTS," *International Journal of Network Management*, vol. 19, no. 4, pp. 291–303, 2009.

[8] M.-S. Hwang and L.-H. Li, "A new remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 1, pp. 28–30, 2000.

[9] J. Xu, W.-T. Zhu, and D.-G. Feng, "An improved smart card based password authentication scheme with provable security," *Computer Standards and Interfaces*, vol. 31, no. 4, pp. 723–728, 2009.

[10] S. K. Sood, A. K. Sarje, and K. Singh, "An improvement of Xu et al.'s authentication scheme using smart cards," in *Proceedings of the 3rd Annual ACM Bangalore Conference (COMPUTE '10)*, pp. 1–5, Karnataka, India, January 2010.

[11] R. Song, "Advanced smart card based password authentication protocol," *Computer Standards and Interfaces*, vol. 32, no. 5-6, pp. 321–325, 2010.

[12] B. L. Chen, W. C. Kuo, and L. C. Wuu, "Robust smart-card-based remote user password authentication scheme," *International Journal of Communication Systems*, vol. 27, no. 2, pp. 377–389, 2014.

[13] S. Kumari and M. K. Khan, "Cryptanalysis and improvement of a robust smart-card- based remote user password authentication scheme," *International Journal of Communication Systems*, 2013.

[14] Q. Jiang, J. F. Ma, G. S. Li, and Z. Ma, "An improved password-based remote user authentication protocol without smart cards," *Information Technology and Control*, vol. 42, no. 2, pp. 150–158, 2013.

[15] J.-H. Yang and C.-C. Chang, "An ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Computers and Security*, vol. 28, no. 3-4, pp. 138–143, 2009.

[16] S. H. Islam and G. P. Biswas, "A more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *The Journal of Systems and Software*, vol. 84, no. 11, pp. 1892–1898, 2011.

[17] T. T. Truong, M. T. Tran, and A. D. Duong, "Improvement of the more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on ECC," in *Proceedings of the 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA '12)*, pp. 698–703, Fukuoka, Japan, 2012.