International Journal of Innovative Research in Information Security (IJIRIS) ISSN: 2349-7017(O)
Volume 1 Issue 5 (November 2014) ISSN: 2349-7009(P)
www.ijiris.com

A Robust Bilinear Pairing Based Remote Mutual Authentication Scheme

Desong Wang* Min Wan Xi Li
School of Mathematics and Computer Engineering, Xihua University

Abstract—Remote user authentication scheme is a mechanism which allows a server to authenticate a remote user over an insecure channel. Recently, Goriparthi et al. made an enhancement based on Das et al.'s remote user authentication scheme using bilinear pairings. The scheme has the merits of no verification table, freely changing password, preventing the forgery attack and the replay attack. However, we found some weaknesses of Goriparthi et al.'s scheme against the insider attack, the denial of service attack, the server spooling attack, and the time-synchronization problem. To overcome these weaknesses, we propose a bilinear pairing based robust remote mutual authentication scheme which is based on nonce instead of timestamp and fresh tag to overcome the existing time-synchronization problem and denial of service attack; our improved security patch can also perform mutual authentication between users and the remote server to prevent the server spoofing attack. The security analysis shows that our improved scheme not only inherits the merits of their scheme but also enhances the security of their scheme.

Keywords—Authentication, Security, Attack, Bilinear pairing, Smart card

I. INTRODUCTION

Remote user authentication scheme is a mechanism which allows a server to authenticate a remote user through an insecure channel. Password-based authentication scheme is the most common method to check the validity of the login message and authenticate the user. In 1981, Lamport [1] proposed a password-based authentication scheme using password tables to authenticate remote users over an insecure network. In Lamport's scheme, password table was used to verify the legitimacy of users, but if this password table is compromised, stolen, or modified by an adversary, then the system could be partially or completely compromised. Later, Shimizu [2] pointed out the weakness of Lamport's scheme [1] and proposed a modified scheme. Then, many improved remote user authentication schemes [3-13] have been proposed.

In 2006, Das et al. [14] proposed a novel remote user authentication scheme using bilinear pairings. They claimed that their proposed scheme using smart cards can prevent the replay attack, the forgery attack and the insider attack. In 2009, however, Goriparthi et al. [15] pointed out that Das et al.'s scheme suffers the replay attack and the forgery attack, and proposed an improved scheme to overcome the replay attack and the forgery attack. Juang-Nien [16] also pointed out that Das et al.'s scheme is easily vulnerable to another replay attack and the offline dictionary attack with or without a smart card. Meanwhile, Yan et al. [23] also pointed out that Goriparthi et al.'s scheme is vulnerable to the insider attack, the denial of service attack and the server spoofing attack, and the time-synchronization problem also exists in Goriparthi et al.'s scheme.

To remedy these pitfalls, we present a bilinear pairing based robust remote mutual authentication scheme. Our scheme is based on nonce instead of timestamp and fresh tag to overcome the existing denial of service attack and time-synchronization problem. Meantime, our improved security patch also establishes trust between the user and the remote system in the form of mutual authentication, so our scheme can prevent the server spoofing attack. Security analysis shows that our scheme not only inherits the merits of their scheme but also enhances the security of their scheme.

The remainder of the paper is organized as follows. In Section 2, we demonstrate our proposed scheme. In Section 3, we analyze the security of our scheme. In Section 4 demonstrates the functionality consideration of our scheme. Finally, we make a conclusion in Section 5.

II. OUR PROPOSED SCHEME

In this section, we propose a bilinear pairing based robust remote Mutual authentication scheme that can withstand the security weaknesses described in the previous research. Our proposed scheme is also composed of five phases: (1) the setup phase, (2) the registration phase, (3) the login phase, (4) the mutual authentication phase, and (5) the password-changing phase. In the setup phase, the RS (i.e. remote system) first sets up the system parameters, and then publishes the public information. In the registration phase, a user gives his/her identity information to the RS for registration. In the mutual authentication phase, a user uses his/her smart card to send a login request message to the RS, if the submitted request message is valid, then the RS and the users will perform further a series of mutual authentication operations to verify authenticity each other. In the password-changing phase, the user can freely change his/her password. The scheme is illustrated in Fig. 1. Now, we describe the five phases separately in our scheme as follows.

A. Setup phase

Let G_1 be an additive cyclic group of prime order q and G_2 be the multiplicative cyclic group of the same order. Let P be a generator of G_1, $e: G_1 \times G_1 \to G_2$ be a bilinear mapping [14, 15] with the following properties:

(1) Bilinearity: For all $Q, R, S \in G_1$, $a, b \in Z_q^*$, $e(Q+R, S) = e(Q, S)e(R, S)$,

$e(Q, R+S) = e(Q, R)e(Q, S)$, $e(aQ, bR) = e(Q, R)^{ab} = e(abQ, R) = e(Q, abR)$;

(2) Non-degeneracy: There exist $Q, R \in G_1$ such that $e(Q, R) \neq 1_{G_2}$, where 1_{G_2} is an identity element of G_2;

(3) Computability: there is an efficient algorithm to compute $e(Q, R)$ for all $Q, R \in G_1$.

Let $H: \{0,1\}^* \to G_1$ be a cryptographic one-way hash function which maps a string to a point of the additive cyclic group G_1 and $h: \{0,1\}^* \to Z_q^*$ be a secure one-way hash function [17]. The RS chooses a secret key $s \in Z_q^*$ and computes the corresponding public key $Pub_{RS} = sP$. The RS publishes the system parameters $\{G_1, G_2, e, q, P, Pub_{RS}, H, h\}$ and keeps s secret.

B. Registration phase

Before the remote user logins to the remote system, the user needs to perform the following steps.

R1. First, U_i chooses his/her ID_i, password PW_i and a random number R. RS searches user ID_i in the user ID storage table (see Table I). If it exists, then return to require U_i to re-choose his/her ID_i; otherwise, U_i interactively submits $\{ID_i, h(PW_i \oplus R)\}$ to RS through a secure channel.

R2. Next, RS creates the smart card identifier SCI_i, which is the number of smart cards that the server has issued to U_i. If ID_i is a new user, then the server will set $SCI_i = 1$ and store $\{ID_i, SCI_i\}$ in the ID storage table in the server. If the server issues a new smart card to a remote user U_i who registered before, the server can get $\{ID_i, SCI_i\}$ from the ID storage table. And then the server computes $SCI_i = SCI_i + 1$ and stores $\{ID_i, SCI_i\}$ in the ID storage table of the server.

R3. Third, RS computes $Reg_{ID_i} = sH(ID_i) + H(PW_i \oplus R)$, where $\oplus$ denotes a bit-wise exclusive-or (XOR) operation.

R4. Lastly, RS personalizes a smart card which contains ID_i, Reg_{ID_i}, $H(\cdot)$, $h(\cdot)$, SCI_i and issues it to U_i securely.

R5. After U_i receives the smart card, he/she inputs R into his/her smart card. So the memory of the smart card contains ID_i, Reg_{ID_i}, $H(\cdot)$, $h(\cdot)$, SCI_i and R.

TABLE I
ID STORAGE TABLE

ID	*SCI*
ID_1	SCI_1
ID_2	SCI_2
…	…
ID_i	SCI_i
…	…

C. Login phase

Whenever the user wants to logon to the remote server, he/she must perform the following steps.

L1. First, U_i inserts his/her smart card into the card reader and inputs his/her ID_i and password PW_i.

L2. If U_i does not pass the verification of ID_i and PW_i, then remote user authentication scheme is terminated. On the contrary, If U_i passes the verification of ID_i and PW_i, then the smart card generates a random number nonce [7, 20-22] ("Nonce" means "used only once") N_1 and computes the following messages:

$$V_i = N_1 Pub_{RS}$$
$$C_1 = N_1 \oplus h(Reg_{ID_i} - H(PW_i \oplus R))$$
$$DID_i = (N_1 + h(N_1 \| V_i \| SCI_i))(Reg_{ID_i} - H(PW_i \oplus R))$$

L3. Finally, U_i sends the login message $\{ID_i, DID_i, V_i, C_1\}$ to RS for the authentication process.

D. Mutual Authentication phase

In the mutual authentication phase, to discuss conveniences, we have given the following definition of fresh tag.

Definition 1 (Fresh tag). Any first time message sent by the users is fresh and therefore acceptable. If it is not a first time message then is not fresh and therefore rejected by the RS.

To achieve mutual authentication between the remote users and the RS, after the RS receives the login request message from the user, the user and the RS will perform the following operations:

A1. RS sets up a counter and a timestamp for the ID_i, which is used to calculate the frequency of ID_i. RS checks the session state table (see Table II) to see whether the ID_i is in the session state or not. If so, the login request is rejected; otherwise RS checks further user ID storage table to see if it has been in existence of the ID_i. If it does not exist, RS rejects the request of the user; otherwise RS checks the frequency value of the user ID_i or the fresh tag of messages $\{ID_i, DID_i, V_i, C_1\}$, if the value is more than the experience of a certain threshold or the fresh tag of messages $\{ID_i, DID_i, V_i, C_1\}$ is not fresh, then that is illegal user try to login the system or illegal to attacks on RS, so RS deletes or quarantines review of the ID_i; otherwise performs step 2 of the mutual authentication phase. In short, RS checks the validity of ID_i. If ID_i is invalid, it rejects the login request.

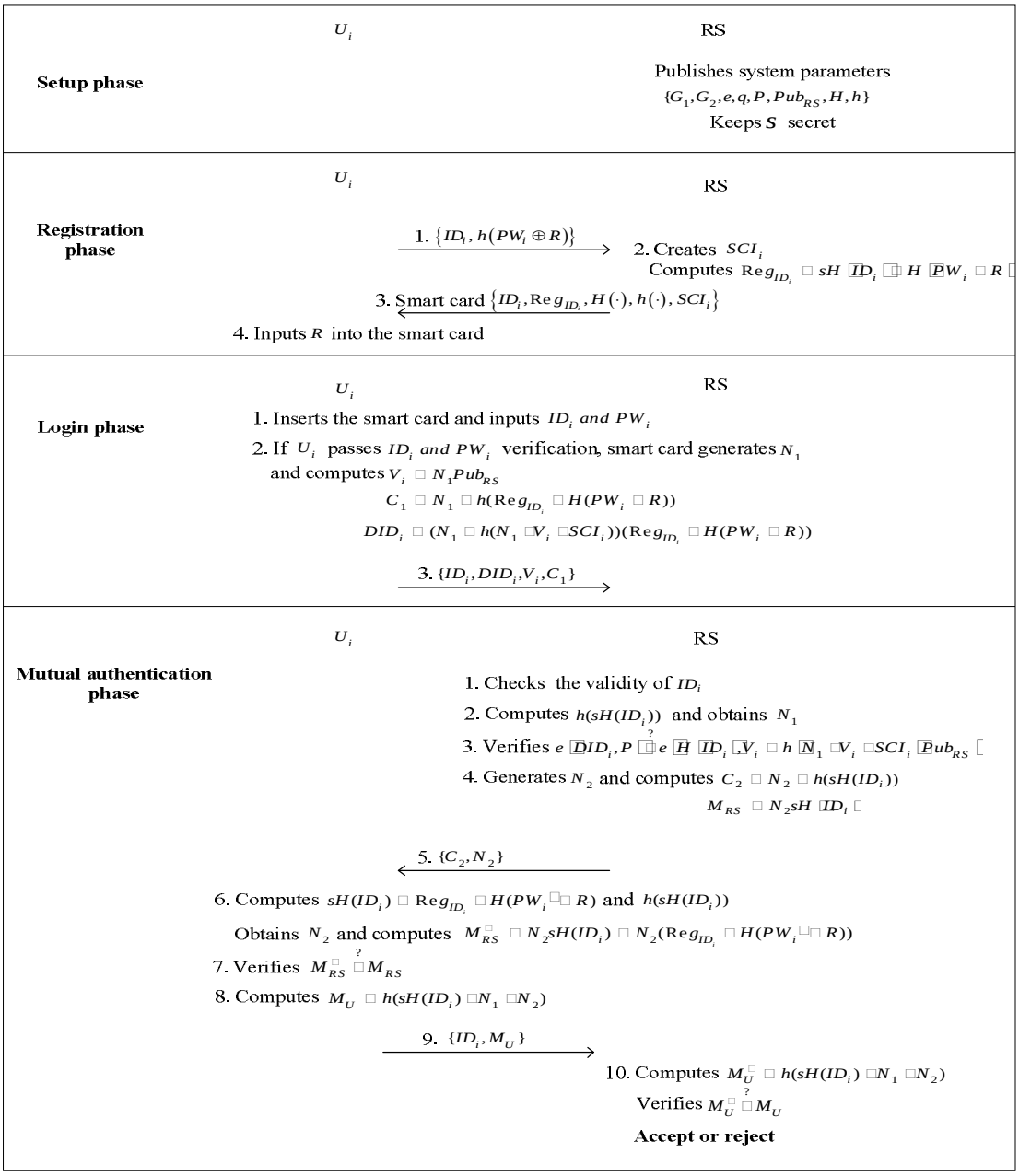

Fig. 1. Our propsed scheme

A2. RS computes $h(sH(ID_i))$ and obtains N_1 by XORing C_1 with $h(sH(ID_i))$ because $C_1 = N_1 \oplus h(Reg_{ID_i} - H(PW_i \oplus R)) = N_1 \oplus h(sH(ID_i))$.

A3. RS verifies whether $e(DID_i, P) = e(H(ID_i), V_i + h(N_1 \| V_i \| SCI_i)Pub_{RS})$ or not. If it does not hold true, login request is rejected; otherwise, U_i is authenticated and the RS further performs the following operations. Where

$$\begin{aligned} e(DID_i, P) &= e((N_1 + h(N_1 \| V_i \| SCI_i))(Reg_{ID_i} - H(PW_i \oplus R)), P) \\ &= e((N_1 + h(N_1 \| V_i \| SCI_i))(sH(ID_i)), P) \\ &= e(H(ID_i), (N_1 + h(N_1 \| V_i \| SCI_i))sP) \\ &= e(H(ID_i), (N_1 + h(N_1 \| V_i \| SCI_i))Pub_{RS}) \\ &= e(H(ID_i), N_1 Pub_{RS} + h(N_1 \| V_i \| SCI_i)Pub_{RS}) \\ &= e(H(ID_i), V_i + h(N_1 \| V_i \| SCI_i)Pub_{RS}) \end{aligned}$$

A4. RS generates a fresh random nonce [7, 20-22] ("Nonce" means "used only once.") N_2, and computes $C_2 = N_2 \oplus h(sH(ID_i))$ and $M_{RS} = N_2 sH(ID_i)$.

A5. RS sends $\{C_2, M_{RS}\}$ to U_i.

A6. Upon receiving the messages $\{C_2, M_{RS}\}$, U_i computes

$$sH(ID_i) = Reg_{ID_i} - H(PW_i^{¢} \oplus R) \text{ and } h(sH(ID_i))$$

Where Reg_{ID_i} is stored in the user's smart card and $PW_i^{¢}$ is a private password of U_i; obtains N_2 by XORing $C_2 = N_2 \oplus h(sH(ID_i))$ with $h(sH(ID_i))$ and computes $M_{RS}^* = N_2 sH(ID_i) = N_2(Reg_{ID_i} - H(PW_i^{¢} \oplus R))$.

A7. U_i validates whether $M_{RS}^* = M_{RS}$ or not. If it does not holds true, U_i terminates the connection; otherwise, RS is authenticated and U_i further performs the following operations.

A8. U_i computes $M_U = h(sH(ID_i) \| N_1 \| N_2)$.

A9. U_i sends $\{ID_i, M_U\}$ to RS.

A10. Upon receiving $\{ID_i, M_U\}$, RS computes $M_U^* = h(sH(ID_i) \| N_1 \| N_2)$ and checks if M_U^* is equal to M_U. If equivalent, mutual authentication between U_i and RS is completed and U_i may access RS is granted; otherwise the login request is rejected.

TABLE II
ID SESSION STATE TABLE

ID which is applying Conversation	Message received $\{ID_i, DID_i, V_i, C_{1i}\}$	Remote system time T
ID_1	$\{ID_1, DID_1, V_1, C_{11}\}$	T_1
ID_3	$\{ID_3, DID_3, V_3, C_{13}\}$	T_3
ID_5	$\{ID_5, DID_5, V_5, C_{15}\}$	T_5
…	…	…
ID_i	$\{ID_i, DID_i, V_i, C_{1i}\}$	T_i
…	…	…

E. Password-changing phase

Whenever user U_i wants to change the old password PW_i to the new password PW_i^*, he/she must insert his/her smart card into the card reader, and enters his/her ID_i and password PW_i. If user passes the verification of ID_i and PW_i, the smart card will perform the following operation:

$$Reg_{ID_i}^* = Reg_{ID_i} - H(PW_i \oplus R) + H(PW_i^* \oplus R) = sH(ID_i) + H(PW_i^* \oplus R).$$

Finally, replaces the previously stored Reg_{ID_i} by $Reg_{ID_i}^*$ on the smart card.

III. SECURITY ANALYSIS

The security of our improved scheme is still based on the security of one-way hash function and the difficulty of computing the discrete logarithm [15]. In the following, we will analyze the security of our improved scheme.

A. Preventing the insider attack

The insider attack is when the user's password is obtained by the server in the registration phase [8, 13, 16]. Therefore, the user must conceal his/her password from the server to prevent the insider attack. In our scheme, the user will choose a random number R and compute $H(PW_i \oplus R)$. Then he/she sends $H(PW_i \oplus R)$ to the server of the RS for registration. So the RS can not know the correct password PW_i since the entropy of R is very large.

B. Preventing the replay attack

The replay attack [16] is when an attacker tries to imitate the user to log in to the server by resending the messages transmitted between the user and the server. In the past, timestamps have been used by several schemes [3-6, 14, 15] to prevent this kind of attacks. However, this technique suffers from time-synchronization problem. For this reason, a nonce-based challenge/response scheme is more adequate for preventing the replay attack. In our improved scheme, we use random nonces to prevent this kind of attack. In our improved scheme, two nonces N_1 and N_2 are generated independently by the smart card and the RS, respectively, and both will be different in each session, and N_1 is calculated as V_i, C_1, and embedded in DID_i and N_2 is calculated as C_2 and M_{RS}. This ensures that authentication messages exposed in an unsecured channel are distinct among all sessions of authentication. Thus, an attacker has no opportunity to successfully replay used messages. Assuming an attacker intercepts the login request messages $\{ID_i, DID_i, V_i, C_1\}$ and impersonate user U_i to resend the intercepted login request messages $\{ID_i, DID_i, V_i, C_1\}$ to the RS, but at this time, the RS can detect the user who have the holder of ID_i is applying for services or has entered the session state, therefore the attacker can not be authenticated by the RS. Assuming an attacker intercepts messages $\{C_2, M_{RS}\}$ and impersonate the RS to resend the intercepted the messages $\{C_2, M_{RS}\}$ to the U_i. As the attacker can not construct the messages $\{C_2, M_{RS}\}$ before the RS send the messages $\{C_2, M_{RS}\}$ to the U_i, the U_i has achieved the RS's authentication before the U_i receive the resending messages $\{C_2, M_{RS}\}$ by the attacker, so the U_i will ignore the messages $\{C_2, M_{RS}\}$. Therefore, two nonces N_1 and N_2 used in our scheme can prevent the replay attack.

C. Preventing the denial of service attack

As the RS sets up the session state table in the authentication process, so it can effectively prevent the denial of service attack by testing the frequency value of ID_i and the fresh tag of the received messages $\{ID_i, DID_i, V_i, C_1\}$.

D. Preventing the guessing attack

It is impossible for an attacker to compute the user password PW_i from the intercepted messages $\{ID_i, DID_i, V_i, C_1\}$, $\{C_2, M_{RS}\}$ and $\{ID_i, M_U\}$, which include no any information about the password. It is also extremely hard for an attacker to derive the remote system secret key s from the eavesdropped messages $\{ID_i, DID_i, V_i, C_1\}$, $\{C_2, M_{RS}\}$ and $\{ID_i, M_U\}$ because of the property of the collision free one-way hash function and the difficulty of computing the discrete logarithm.

E. Preventing the offline dictionary attack with the smart card

The problem in this kind of attack is called the smart-card-lost problem. In this case, the attacker can obtain the secret information stored in the smart card. In order to prevent this attack, the password stored in a smart card must be controlled by the server's secret key. Even if the attacker obtains the secret information from the smart card, the attacker also can not obtain the right password. In our improved scheme, the password stored in the smart card is calculated as Reg_{ID_i}. Only the server of the RS can use the secret key s to computes $sH(ID_i)$ and obtain $H(PW_i \oplus R)$ by $\mathrm{Reg}_{ID_i} - sH(ID_i)$, so the attacker can not get the right password. It is also extremely hard for the attacker to derive the remote system secret key s from the Reg_{ID_i} because of the property of the collision free one-way hash function and the difficulty of computing the discrete logarithm. Therefore, the attacker can not obtain the right password and can not create the valid login and mutual authentication messages. Of course, when the information stored in the smart card is compromised, the offline password dictionary attack will succeed.

F. Preventing the server spoofing attack

An attacker may try to masquerade as a server such that users send confidential information to the spoofing server. In our improved scheme, a user will first authenticate the server in the registration phase. To successfully masquerade as the server, an attacker must provide the mutual authentication messages $\{C_2, M_{RS}\}$ and $\{ID_i, M_U\}$ correctly. Since C_2 is computed by $C_2 = N_2 \oplus h(sH(ID_i))$, M_{RS} is computed by $M_{RS} = N_2 sH(ID_i)$, the attacker cannot generate C_2 and M_{RS} without knowing the secret key s of the server. Thus, our improved scheme can also successfully resist the server spoofing attack.

G. Preventing the forgery attack

If an attacker wants to impersonate a legal user U_i, he/she should forge a valid login request message to pass the authentication of the RS. However, A valid user's login request message comprises ID_i, DID_i, V_i and C_1, an attacker can not compute the valid V_i, C_1 and DID_i in Step L1 of the login phase and response M_U in Step A10 of the mutual authentication phase without the information of the server's secret key s and the user's password PW_i and the random nonces N_1 and N_2.

IV. FUNCTIONALITY CONSIDERATION

A. Achieving mutual authentication

Our improved scheme can achieve mutual authentication: In step A3 of the mutual authentication phase, the remote system can authenticate the user U_i because only the valid remote system can compute and verify whether $e(DID_i, P) = e(H(ID_i), V_i + h(N_1 \| V_i \| SCI_i) Pub_{RS})$ or not; In step A10 of the mutual authentication phase, the remote system can further authenticate the user U_i because only the valid remote system can compute $M_U^* = h(sH(ID_i) \| N_1 \| N_2)$ and verify whether $M_U^* = M_U$ or not. User U_i can also authenticate the remote system because only the legitimate remote user U_i can compute $M_{RS}^* = N_2 sH(ID_i) = N_2(Reg_{ID_i} - H(PW_i \| R))$ and verify whether $M_{RS}^* = M_{RS}$ or not, where PW_i is password of the user private. Therefore, the improved scheme can achieve mutual authentication.

B. No password table

To prevent the server from holding and protecting a large password table, a password or a verification table should not be stored in the server of the RS. In our improved scheme, the hashed password with a random number $H(PW_i \| R)$ is calculated as Reg_{ID_i} which is stored in the smart card. So the server of the RS does not need to maintain a password table. In our improved scheme, the server of the RS only needs to maintain a ID storage table to store every user's ID_i and corresponding card's identifier. This table is smaller than the password table and does not need to be kept secret.

C. Choosing and changing of passwords by users

In our improved scheme, every user can also choose his/her password. Hence, the user can also easily remember the password. Furthermore, we also provide a password-changing phase for users to change or update their passwords at their own.

D. No time-synchronization problem

In our improved scheme, we use two nonces N_1 and N_2 instead of timestamp to prevent the replay attack, so no logical time clocks are needed. Therefore, time-synchronization problem does not exist in our improved scheme.

E. Revoking the lost cards without changing the user's identity

In our proposed scheme, if the user U_i loses his/her smart card, the server of the RS can revoke the lost card. When the user U_i needs to obtain a new smart card, the server of the RS will set $SCI_i = SCI_i + 1$ and issue a new smart card to the user securely.

The functionality comparison of our scheme and related schemes is summarized in Table III. In contrast with Goriparthi et al.'s scheme, our improved scheme is more secure.

TABLE III
FUNCTIONALITY COMPARISON BETWEEN OUR SCHEME AND RELATED SCHEMES

	Our scheme	Goriparthi et al.	Das et al.
Insider attack	No	Yes	Yes
Replay attack	No	No	Yes
Denial of server attack	No	Yes	Yes
Guessing attack	No	No	Yes
Offline dictionary attack with the smart card	No	No	Yes
Server spoofing attack	No	Yes	Yes
Forgery attack	No	No	Yes
Mutual authentication	Yes	No	No
No clock synchronization	Yes	No	No
No password table	Yes	Yes	Yes
Choosing and changing of passwords by users	Yes	Yes	Yes
Revoking the lost cards without changing the user's identity	Yes	No	No

V. CONCLUSION

In this paper, we have proposed a bilinear pairing based robust remote mutual authentication scheme can safely achieve mutual authentication between the users and the remote system. Moreover, our improved scheme has the important merits as follows: (1) it can prevent the insider attack; (2) it can effectively prevent the denial of service attack by testing the frequency value of ID_i and fresh tag of the login messages $\{ID_i, DID_i, V_i, C_1\}$; (3) it is a nonce-based scheme that can overcome serious time-synchronization and transmission delay problem; (4) the server of the RS and users can authenticate each other to prevent the server spoofing attack; (5) it can also prevent the replay and forgery attacks; (6) it can prevent the guessing attack; (7) it can prevent the offline dictionary attack with the smart card; 8) the server of the RS can revoke the lost cards without changing the user's identity. The security analysis shows that our scheme not only inherits the merits of their scheme but also enhances the security of their scheme.

ACKNOWLEDGMENT

The research in this paper is in part supported by the National Natural Science Foundation of China under Grant No. 61401369, Chunhui Program of Ministry of education of China under Grant No. Z2014052 and No. 12226530, Lab of Security Insurance of Cyberspace of Sichuan Province under Grant No. szjj2013-017 and No. szjj2012-028, Scientific Research Foundation of Sichuan Provincial Education Department under Grant No. 13226690, and Key Scientific Research Foundation of Xihua University under Grant No. Z1222626.

REFERENCES

[1] L. Lamport, "Password authentication with insecure communication", Communications of the ACM, 24(11), 1981, pp. 770-772.
[2] A. Shimizu, T. Horioka, and H. Inagaki, "A password authentication method for contents communication on the Internet", IEICE Transactions on Communications, E81-B(8), 1998, pp. 1666-1673.
[3] K. Tan and H. Zhu, "Remote password authentication scheme based on cross-product", Computer Communications, 22(4), 1999, pp. 390-393.
[4] M.S. Hwang and L.H. Li, "A new remote user authentication scheme using smart cards", IEEE Transactions on Consumer Electronics, 46(1), 2000, pp. 28-30.
[5] H.Y. Chien, J.K. Jan, and Y.M. Tseng, "An efficient and practical solution to remote authentication: smart card", Computers & Security, 21(4), 2002, pp. 372-375.
[6] S.T. Wu and B.C. Chieu, "A user friendly remote authentication scheme with smart cards", Computers & Security, 22(6), 2003, pp. 547-550.
[7] C. Fan, Y. Chan, and Z. Zhang, "Robust remote authentication scheme with smart cards", Computers & Security, 24, 2005, pp. 619-628.
[8] W.S. Juang, "Efficient password authenticated key agreement using smart card", Computer & Security, 23, 2004, pp. 167-173.
[9] W.C. Ku and S.M. Chen, "Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards", IEEE Transactions Consumer Electronics, 50(1), 2004, pp. 204-207.
[10] C.C. Lee, L.H. Li, and M.S. Hwang, "A remote user authentication scheme using hash functions", ACM Operating Systems Review, 36(4), 2002, pp. 23-29.
[11] W.C. Ku, "A hash-based strong-password authentication scheme without using smart cards", ACM Operating Systems Review, 38(1), 2004, pp. 29-34.
[12] W. Ku, C. Chen, and H. Lee, "Weaknesses of Lee-Li-Hwang's hash-based password authentication scheme," ACM Operating Systems Review, vol. 37, no. 4, pp. 9-25, 2003.
[13] H.A. Wen, T.F. Lee, and T. Hwang, "Provably secure three-party password-based authenticated key exchange protocol using Weil pairing", IEE Proceedings of Communications, 152(2), 2005, pp. 138-143.
[14] M. Das, A. Saxena, V. Gulati, and D. Phatak, "A novel remote user authentication scheme using bilinear pairings", Computers & Security, 25(3), 2006, pp. 184-189.
[15] T. Goriparthi, Manik L. Das, and A. Saxena, "An improved bilinear pairing based remote user authentication scheme", Computer Standards & Interfaces, 31, 2009, pp. 181-185.
[16] W.S. Juang and W.K. Nien, "Efficient password authenticated key agreement using bilinear pairings", Mathematical and Computer Modelling, 47, 2008, pp. 1238-1245.
[17] NIST FIPS PUB 180-2, Secure Hash Standard, National Institute of Standards and Technology, U.S. Department of Commerce, DRAFT, 2004.
[18] E.J. Yoon, E.K. Ryu, and K.Y. Yoo, "An improvement of Hwang-Lee-Tang's simple remote user authentication scheme", Computers & Security, 24, 2005, pp. 50-56.
[19] M.K. Khan, and J.S. Zhang, "Improving the security of 'a flexible biometrics remote user authentication scheme'", Computer Standards & Interfaces, 29, 2007, pp. 82-85.
[20] R.M. Needham and M.D. Schroeder, "Using encryption for authentication in large networks of computer", Communication of the ACM, 21(12), 1978, pp. 993-998.
[21] J.Y. Liu, A.M. Zhou, and M.X. Gao, "A new mutual authentication scheme based on nonce and smart cards", Computer Communications, 31, 2008, pp. 2205-2209.
[22] D.Z. Sun, J.P. Huai, J.Z. Sun, and J.X. Li, "Cryptanalysis of a mutual authentication scheme based on nonce and smartcards", Computer Communications, 32, 2009, pp. 1015-1017.
[23] Y.H Yan, D.S. Wang, and J. P. Li, and L.G. Li, "Cryptanalysis of a remote user authentication scheme based on bilinear pairing", the 2009 International Conference on Apperceiving Computing and Intelligence Analysis(ICACIA 2009), pp. 73-76, Chengdu, China, 2009.

