A Robust Remote Authentication Scheme for M-Commerce Environments

Shih-Yang Yang[1(✉)], Jian-Wen Peng[2], Wen-Bing Horng[3], and Ching-Ming Chao[4]

[1] Department of Media Arts, University of Kang Ning, Taipei 11485, Taiwan, ROC
shihyang@ukn.edu.tw
[2] Department of Commerce Technology and Management, Chihlee University of Technology, Taipei 22050, Taiwan, ROC
pchw8598@mail.chihlee.edu.tw
[3] Department of Computer Science and Information Engineering, Tamkang University, Taipei 25137, Taiwan, ROC
horng@mail.tku.edu.tw
[4] Department of Computer Science and Information Management, Soochow University, Taipei 10048, Taiwan, ROC
chao@csim.scu.edu.tw

Abstract. With the rapid growth of electronic and mobile commerce today, how to design a secure and efficient remote user authentication scheme with resource-limited devices over insecure networks has become an important issue. In this paper, we present a robust authentication scheme for the mobile device (a non-tamper-resistant device in which the secret authentication information stored in it could be retrieved) to solve the challenging lost device problem. It tries to satisfy the following advanced essential security features: (1) protecting user privacy in terms of anonymity and non-traceability, (2) supporting session keys with perfect forward secrecy, and (3) secure even for the case of lost devices, in addition to the conventional security requirements. The security of our scheme is based on the quadratic residue assumption, which has the same complexity as in solving the discrete logarithm problem. However, the computation of the quadratic congruence is very efficient. It only needs one squaring and one modular operations in the mobile device end, which is much cheaper than the expensive modular exponentiation used in those schemes based on the discrete logarithm problem. Thus, using the quadratic congruence, our scheme can achieve robustness and efficiency, even for the non-tamper-resistant mobile device.

Keywords: Authentication · Quadratic congruence · Security

J.-L. Chen et al. (Eds.): WiCON 2018, LNICST 264, pp. 119–128, 2019.
https://doi.org/10.1007/978-3-030-06158-6_12

1 Introduction

Today is at the era of the rapid growth in electronic and mobile commerce. At this age, how to design a secure and efficient remote user authentication with resource-limited portable devices, such as cellular phones or smart cards, has become an important issue to keep communications confidential and to protect user privacy. Since Lamport [1] proposed the first password authentication scheme over insecure networks, many research works [2–8] have been proposed to improve security and efficiency on the remote authentication used in, for example, electronic toll collection and online financial transactions.

According to the remote authentication schemes proposed so far, it could be summarized that a robust, complete, and efficient remote authentication scheme must satisfy the following criteria:

(1) The remote server does not need to store password or verification tables.
(2) The users can freely choose and change their own passwords.
(3) The scheme must be efficient and practical.
(4) The scheme must provide mutual authentication.
(5) The scheme must provide session key agreement with perfect forward secrecy.
(6) The scheme must protect user privacy in terms of anonymity and non-traceability.
(7) The scheme must withstand various kinds of attacks, such as replay attack, offline password guessing attack, man-in-the-middle attack, user/server impersonation attack, and so on.
(8) The scheme must be secure even for the non-temper-resistant smart card.

Recently, Chung et al. [5] proposed a remote user authentication scheme for resource-limited devices to fulfill the above criteria. However, the scheme suffers from two drawbacks. First, it does not protect user privacy, because static ID is used. Second, it is inefficient for the resource-limited devices, since the scheme is based on the discrete logarithm problem and the expensive modular exponentiation needs to be performed in such devices.

In this paper, we propose a new robust and efficient remote user authentication scheme trying to satisfy all the above security features even for the non-tamper-resistant resource-limited devices. To achieve the same security strength as those using the costly modular exponentiation while releasing the computation burden from the resource-limited devices, we utilize the quadratic congruence. In our scheme, it is very efficient in computing the quadratic congruence because the mobile device only needs to perform one squaring and one modular operations. On the other hand, solving a quadratic congruence modulo a composite needs to factorize the modulus, which has the same complexity as solving the discrete logarithm problem for the factorization. One of the distinguished features of our scheme, as compared to Chung et al.'s scheme, is that our scheme can preserve user anonymity and provide non-traceability by utilizing dynamic ID to protect user privacy. In addition, because our scheme can provide session keys with perfect forward secrecy, which makes our scheme more secure because all communications (including past ones) are confidential.

The rest of the paper is organized as follows. Section 2 presents our proposed authentication scheme. Section 3 analyzes the security of our scheme. Section 4 gives a performance comparison with other related schemes. Finally, the last section concludes this paper.

2 Our Proposed Scheme

Our proposed scheme consists of five phases: (1) initial setup phase, (2) registration phase, (3) login phase, (4) authentication phase, and (5) password change phase.

2.1 Initial Setup Phase

The server S first selects two distinct large prime numbers p and q. It then computes $n = p \times q$. In addition, S selects a random number x as its long-term secret key and a secure one-way hash function $h(\cdot)$. Note that p, q, and x are kept secretly in the server.

2.2 Registration Phase

This phase is invoked whenever a new user U initially registers to S. The user U first chooses his/her identity ID and submits it to S through a secure channel. After receiving ID, the server S first computes $Y = h(ID \,\|\, x) \oplus h(PW_o)$, where PW_o is a random default password for user U. Then, the server S stores $\{Y, n, h(\cdot)\}$ into a smart card and sends it together with the default password PW_o to U (or directly sends $\{Y, n, h(\cdot), PW_o\}$ to U's mobile device) via a secure channel. Before the user U begins to use his/her new smart card (or mobile device), he/she is requested to change his/her default password PW_o into a new one, say PW, as performed in the password change phase, described in Sect. 2.5.

2.3 Login Phase

When user U wants to login with server S, he/she inserts his/her smart card into the card reader of a terminal (or uses his/her mobile device) and inputs his/her ID and PW. The user U's portable device first generates a random number a and computes $C_0 = Y \oplus h(PW)$, $X = (T_1 \,\|\, ID \,\|\, a)^2 \bmod n$, and $C_1 = h(X \,\|\, T_1 \,\|\, C_0 \,\|\, a)$, where T_1 is the current timestamp. It then sends the login request message $\{X, C_1, T_1\}$ to server S.

2.4 Authentication Phase

User Authentication. After receiving the login request message $\{X, C_1, T_1\}$ at time T_1', the server S performs the following steps to authenticate the user U.

1. Verify whether $(T_1' - T_1) < \Delta T$, where ΔT is a predefined transmission delay. If it is not, reject the login request to avoid the replay attack.

2. Solve $X = (T_1 \| ID \| a)^2 \bmod n$ with p and q to obtain four roots (r_1, r_2, r_3, r_4) [9,10]. Check which root r_i (for $i = 1, 2, 3, 4$) containing the prefix T_1 to determine the correct root $r = (T_1 \| ID \| a)$. Then, ID and a will be determined from the correct root r.
3. Check the validity of ID. If it fails, stop the verification procedure.
4. Check if $h(X \| T_1 \| h(ID \| x) \| a) = C_1$. If they are not equal, terminate the current session. Otherwise, U is authenticated as a legal user.
5. Generate a random number b and compute $R = a \oplus b$ and $C_2 = h(R \| h(ID \| x) \| b \| T_2)$, where T_2 is the current timestamp of the server S.
6. Send the reply message $\{R, C_2, T_2\}$ to the user U's smart card.

Server Authentication. At the receipt of the message $\{R, C_2, T_2\}$ from the server S at time T_2', the user U's smart card performs the following steps to authenticate the server S.

1. Check the freshness of T_2 by verifying whether $(T_2' - T_2) < \Delta T$. If it is not, terminate the current session.
2. Compute $b = R \oplus a$, and check if $h(R \| C_0 \| b \| T_2) = C_2$. If they are not equal, terminate the current session. Otherwise, S is authenticated as the legal server.

Session Key Establishment. After mutual authentication is complete, the user U calculates the session key $SK = h(a \| C_0 \| b)$ and the server S computes the session key $SK = h(a \| h(ID \| x) \| b)$. Note that these two session keys are exactly the same since $C_0 = h(ID \| x)$ and they are used for securing transmissions by encrypting/decrypting subsequent transmitted messages during the current session.

2.5 Password Change Phase

If the user U wants to change his/her password, he/she first inserts his/her smart card into the card reader of a terminal and enters his/her ID and PW. The login and authentication phases, as described before (Sects. 2.3 and 2.4), are performed first. After successful mutual authentication, the smart card asks the user U to enter a new password PW_n. It then computes $Y_n = Y \oplus h(PW) \oplus h(PW_n)$ and replaces Y with Y_n.

3 Security Analysis

In this section, we analyze the security of our proposed authentication scheme. Note that in our scheme we allow the authentication information stored in the portable device can be retrieved (i.e., the portable device is non-tamper-resistant). Because the security of our scheme is based on the quadratic residue assumption, we describe it first.

Assumption 1. (Quadratic Residue Assumption) *Let p and q be two large primes and $n = p \times q$. If $y = x^2 \bmod n$ has a solution, then y is called a* quadratic residue *modulo n. Let QR_n denote the set of all quadratic residues in $\mathbf{Z}_n$ (i.e., $[1, n-1]$). Then, the quadratic residue assumption can be described as follows. Let $y \in QR_n$. Because of the difficulty of factoring the composite modulus n, it is computationally infeasible to find x such that $y = x^2 \bmod n$ without knowing p and q [9, 10].*

Lemma 1. *In the proposed scheme, the random numbers a (generated by the user's portable device) and b (generated by the server) cannot be derived by an adversary.*

Proof. In this scheme, we apply the quadratic residue assumption to protect the random number a in X during the login phase, where $X = (T_1 \,\|\, ID \,\|\, a)^2 \bmod n$. An adversary can eavesdrop on X from the login request $\{X, C_1, T_1\}$ transmitted over the network. Note that $n = p \times q$, where p and q are two large primes which are kept secretly in the server. Without knowing p and q, it is difficult to factor the composite modulus n by the quadratic residue assumption. Therefore, it is computationally infeasible for the adversary to solve X to obtain a even if he knows T_1 from the login request. On the other hand, due to the one-way property of the secure hash function $h(\cdot)$, it is computationally infeasible to obtain a from $C_1 = h(X \,\|\, T_1 \,\|\, C_0 \,\|\, a)$ and b from $C_2 = h(R \,\|\, C_0 \,\|\, b \,\|\, T_2)$, where C_2 can be intercepted from the reply message $\{R, C_2, T_2\}$. Although $R = a \oplus b$, without knowing a, it is impossible to derive b from R, and vice versa. Therefore, the random numbers a and b cannot be compromised by an adversary.

Proposition 1. *The proposed scheme can withstand the offline password guessing attack.*

Proof. A remote user authentication scheme which is vulnerable to the offline password guessing attack must satisfy the two conditions: (1) the users password is weak and (2) there exists a piece of password-related information used as a comparison target for the password guessing. In our scheme, only $Y = h(ID \,\|\, x) \oplus h(PW)$ stored in a user U's portable device involves the password information, PW. To launch a password guessing attack, it is assumed that an adversary has obtained Y from U's portable device and has intercepted the login request $\{X, C_1, T_1\}$ and the reply message $\{R, C_2, T_2\}$ over the network. The adversary then guesses a password PW^* and computes $C_0^* = Y \oplus h(PW^*)$. As mentioned previously, the adversary must find a piece of password-related information to verify whether the guessed password PW^* is correct or not. In our scheme, it would be $C_1 = h(X \,\|\, T_1 \,\|\, C_0 \,\|\, a)$ or $C_2 = h(R \,\|\, C_0 \,\|\, b \,\|\, T_2)$, in which a and b are two random numbers generated by the portable device and the server, respectively. By Lemma 1, the adversary cannot derive a and b. Therefore, although X, R, T_1, and T_2 are known, without correct a and b, the adversary cannot verify whether the guessed password PW^* is correct or not by comparing $h(X \,\|\, T_1 \,\|\, C_0^* \,\|\, a)$ with the intercepted C_1 (or $h(R \,\|\, C_0^* \,\|\, b \,\|\, T_2)$ with C_2). Thus, our scheme can resist the offline password guessing attack.

Proposition 2. *The proposed scheme can preserve user privacy.*

Proof. User privacy can be divided into two parts for discussion: anonymity and non-traceability. In our scheme, the login request $\{X, C_1, T_1\}$ and the reply message $\{R, C_2, T_2\}$ do not contain explicit static-ID information of the user U. Thus, an adversary cannot easily know which user is connecting with the server from these authentication messages. However, X, C_1, and C_2 indeed implicitly contain user U's ID information. Let us examine whether or not they might reveal ID information. First, consider $X = (T_1 \,\|\, ID \,\|\, a)^2 \bmod n$. By the quadratic residue assumption, it is computational infeasible to solve X. Thus, an adversary cannot obtain ID from X. Next, we consider $C_1 = h(X \,\|\, T_1 \,\|\, C_0 \,\|\, a)$ and $C_2 = h(R \,\|\, C_0 \,\|\, b \,\|\, T_2)$, where $C_0 = h(ID \,\|\, x)$. However, it is computational infeasible to derive ID from C_1 and C_2 owing to the one-way property of the hash function $h(\cdot)$. Moreover, X, R, C_1, and C_2 are different at each session due to random numbers a and b and timestamps T_1 and T_2. Hence, the adversary cannot trace a specific user just by eavesdropping on X, R, C_1, and C_2 from the exchanged login messages.

On the other hand, an adversary may want to find U's ID by performing the offline identity guessing, just like the offline password guessing. However, X, R, C_1, and C_2 contain random numbers a or b. By Lemma 1, an adversary cannot derive them without knowing p and q. Thus, the adversary cannot verify whether a guessed identity ID^* is correct or not. Hence, he cannot obtain U's ID and use it to trace the user U. Note that ID is not stored in U's portable device in our proposed scheme. Even if an adversary can extract $Y = h(ID \,\|\, x) \oplus h(PW)$ from U's portable device, he still cannot know U's ID because Y contains three unknown parameters x, ID and PW. From above observations, we conclude that our scheme can provide user anonymity and non-traceability. Hence, our scheme can protect user privacy.

Proposition 3. *The proposed scheme can withstand the insider attack.*

Proof. In our scheme, the user U only submits his/her ID to the server S during the registration phase. After U receives his/her new authentication information $\{Y, n, h(\cdot), PW_o\}$ stored in a portable device, he/she needs to change the default password, PW_o, to a new one, PW, before using it. Since the update of password is executed by the user U only after successful mutual authentication with the server S, the insider of S cannot know U's true password. Thus, the insider attack will not take place in our scheme.

Proposition 4. *The proposed scheme can provide perfect forward secrecy for session keys.*

Proof. In our scheme, we exchange the session key $SK = h(a \,\|\, h(ID \,\|\, x) \,\|\, b)$ in each session, where a and b are two random numbers generated by the user U's portable device and the server S, respectively. Perfect forward secrecy is a very important property in that if some user's long-term secret values, e.g. ID and PW in our scheme, are compromised, session keys used before still

cannot be derived. By Lemma 1, a and b cannot be obtained by an adversary. Even if both U's identity ID and password PW are compromised, the adversary still cannot derive any session keys used in previous sessions without knowing p and q. Hence, our scheme provides perfect forward secrecy for session keys.

Proposition 5. *The proposed scheme can withstand the user impersonation attack.*

Proof. If an adversary wants to impersonate the user U, he/she has to send the server S a proper login request either by replaying an old one $\{X, C_1, T_1\}$ intercepted from the network or by forging a new one $\{X^*, C_1^*, T_1^*\}$ to pass the authentication of S. In our scheme, we employ the timestamp mechanism (i.e., T_1) to prevent the replay attack. Thus, replaying previous login requests is impossible. On the other hand, to forge a new login request $\{X^*, C_1^*, T_1^*\}$, the adversary needs to have U's ID and $C_0 = h(ID \,\|\, x)$ so that he/she can counterfeit $X^* = (T_1^* \,\|\, ID \,\|\, a^*)^2 \bmod n$ and $C_1^* = h(X^* \,\|\, T_1^* \,\|\, C_0 \,\|\, a^*)$ from forged T_1^* and a^*. However, as shown in Proposition 2, the adversary cannot obtain U's ID. Furthermore, as shown in Proposition 1, the adversary cannot guess the correct password PW. Thus, he cannot compute $C_0 = h(ID \,\|\, x) = Y \oplus h(PW)$, even though Y stored in U's portable device can be obtained by the adversary. Therefore, without ID and C_0, the adversary cannot forge proper login request $\{X^*, C_1^*, T_1^*\}$. Hence, our scheme can withstand the user impersonation attack.

Proposition 6. *The proposed scheme can withstand the server impersonation attack.*

Proof. If an adversary wishes to masquerade as the server S, he/she needs to send the user U a proper reply message either by replaying an old one $\{R, C_2, T_2\}$, intercepted over the network, or by forging a new one $\{R^*, C_2^*, T_2^*\}$ to pass the authentication of U. Similarly, the timestamp mechanism is used to avoid the replay attack. On the other hand, to forge a new proper reply message $\{R^*, C_2^*, T_2^*\}$, the adversary has to know the random number a generated by U's portable device and $C_0 = h(ID \,\|\, x)$ so that he/she can fake $R^* = a \oplus b$ and $C_{2*} = h(a \,\|\, C_0 \,\|\, b^* \,\|\, T_2^*)$ from bogus b^* and T_2^*. However, as shown in Lemma 1, the adversary cannot derive a. Similarly, as demonstrated in Proposition 5, the adversary cannot compute C_0. Therefore, without knowing a and C_0, the adversary cannot forge proper reply message $\{R^*, C_2^*, T_2^*\}$. Therefore, our scheme can resist the server impersonation attack.

Corollary 1. *The proposed scheme can achieve mutual authentication.*

Proof. By Propositions 5 and 6, our scheme can withstand both the user and server impersonation attacks. Thus, the scheme can achieve mutual authentication.

Corollary 2. *The proposed scheme is secure for the non-tamper-resistant portable device.*

Proof. When the portable device is lost, the authentication information stored in the portable device can be obtained by an adversary. As stated in Propositions 1, 2, 4, 5, and 6, we have shown that our scheme can withstand the offline password guessing attack, can achieve mutual authentication, can preserve user privacy in terms of anonymity and non-traceability, and can provide session keys with perfect forward secrecy even if the adversary has obtained the secret information stored in the portable device. Therefore, we can conclude that our scheme is secure even for the non-tamper-resistant portable device.

4 Performance Comparisons

In this section, we give a comparison of our proposed scheme with two other schemes specially designed for non-tamper-resistant portable devices to provide more security in terms of security features and computation cost. The first one is Fan et al.'s scheme [3], which is also based on the quadratic congruence as ours, in addition to using some symmetric cryptosystems. The second one is Chung et al.'s scheme [5]. It is based on the modular exponentiation to achieve its security, which is the most secure scheme.

Table 1 shows the comparison of computation cost of our scheme with the other two schemes. During the login and verification phases, the computation cost in the portable device of our scheme has one additional hash operation as compared to Fan et al.'s scheme, in which the portable device only needs to perform light-weight operations such as arithmetic operation, random number generation, and one-way hash function. As compared to Chung et al.'s scheme, it needs one heavy-weight modular exponentiation in each of the login and verification phases. Because Fan et al.'s scheme requires one additional symmetric decryption in the server during the authentication phase, our scheme is more efficient than the other two schemes in terms of computation cost.

Table 1. Comparison of computation cost at the server and device end.

Phase	End	Fan et al.'s scheme	Chung et al.'s scheme	Our scheme
Registration	Server	1S+1H+1R	3H+1R+1M	2H
Login	Portable device	1R+2M	1E+2H+1R	2H+1R+2M
Verification	Portable device	3H	1E+4H+1M	2H
Verification	Server	1Q+1S+2H+1R	2E+6H+1R+1M	1Q+3H+1R

Note: E: modular exponentiation, S: symmetric encryption/decryption,
H: one-way hash function, Q: solving the quadratic congruence,
R: random number generation, M: multiplication/division.

As shown in Table 2, Fan et al.'s scheme can withstand the offline password guessing attack even if the portable device is lost. Unfortunately, Rhee et al. [6] has shown that Fan et al.'s scheme is vulnerable to the server impersonation attack. In addition, Fan et al.'s scheme does not protect user privacy because

it uses static ID. Besides, it does not provide session key agreement as well as password change procedure. On the other hand, Chung et al.'s scheme provides almost all the security features, except that is uses static ID when sending login requests. Like Fan et al.'s scheme, it also does not protect user privacy in terms of anonymity as well as non-traceability. Apparently, our proposed scheme can provide all security features listed in Table 2.

Table 2. Comparison of security properties.

Security property	Fan's	Chung's	Ours
Without verification table in the server	Yes	Yes	Yes
Resistant to offline password guessing attack	Yes	Yes	Yes
Resistant to insider attack	Yes	Yes	Yes
Resistant to user impersonation attack	Yes	Yes	Yes
Resistant to server impersonation attack	No	Yes	Yes
Providing user anonymity	No	No	Yes
Providing non-traceability	No	No	Yes
Providing mutual authentication	No	Yes	Yes
Providing session key agreement	No	Yes	Yes
Providing perfect forward secrecy for session keys	–	Yes	Yes
Providing secure password change phase	–	Yes	Yes
Secure for non-tamper-resistant portable device	No	Yes	Yes

Note: The symbol "–" stands for "not supported."

5 Conclusion

In this paper, we presented a robust and efficient remote authentication scheme with resource-limited portable devices for M-commerce environments. The security of our scheme is based on the quadratic congruence assumption, which has the same security strength as the discrete logarithm problem, while it uses much less computation cost in computing the quadratic congruence in the portable devices as comparing to the modular exponentiation. By robustness, we mean that our scheme can support the following security features: (1) without verification tables, (2) freely choosing and updating passwords, (3) providing mutual authentication, (4) providing session keys with perfect forward secrecy, (5) protecting user privacy in terms of anonymity and non-traceability, (6) withstanding various kinds of attacks, and (7) secure even for non-tamper-resistant portable devices. By efficiency, we mean that the resource-limited device in our scheme only needs to execute light-weight operations, such as bitwise exclusive-or operations, arithmetic operations, secure one-way hash functions, and pseudo random number generations, rather than the heavy-weight modular exponentiations. Because of the robustness and efficiency, these make our scheme more secure and practical.

References

1. Lamport, L.: Password authentication with insecure communication. Commun. ACM **24**(11), 770–772 (1981)
2. Hwang, M.S., Lee, C.C., Tang, Y.L.: A simple remote user authentication scheme. Math. Comput. Model. **36**(1–2), 103–107 (2002)
3. Fan, C.I., Chan, Y.C., Zhang, Z.K.: Robust remote authentication scheme with smart cards. Comput. Secur. **24**(8), 619–628 (2005)
4. Shieh, W.G., Wang, J.M.: Efficient remote mutual authentication and key agreement. Comput. Secur. **25**(1), 72–77 (2006)
5. Chung, H.R., Ku, W.C., Tsaur, M.J.: Weaknesses and improvement of Wang et al.'s remote user password authentication scheme for resource-limited environments. Comput. Stand. Interfaces **31**(4), 863–868 (2009)
6. Rhee, H.S., Kwon, J.O., Lee, D.H.: A remote user authentication scheme without using smart cards. Comput. Stand. Interfaces **31**(1), 6–13 (2009)
7. Li, X., Nju, J.W., Ma, J., Wang, W.D., Liu, C.L.: Cryptanalysis and improvement of a biometrics-based remote user authentication scheme using smart card. J. Netw. Comput. Appl. **34**(1), 73–79 (2011)
8. Wen, F., Li, X.: An improved dynamic ID-based remote user authentication with key agreement scheme. Comput. Electr. Eng. **38**(2), 381–387 (2012)
9. Patterson, W.: Mathematical Cryptology for Computer Scientists and Mathematicians. Rowman (1987)
10. Rosen, K.H.: Elementary Number Theory and its Applications. Addison-Wesley, Reading (1988)