Mutual Authentication Scheme for Multi-server Environment Without Using Smart Cards

Jiayu WANG
School of Computer Science, Civil Aviation Flight University of China, Guanghan, Sichuan 618307, PR China
Wang.juyi@gmail.com

Fangping DENG
School of Computer Science, Civil Aviation Flight University of China, Guanghan, Sichuan 618307, PR China

***Abstract*—To authenticate a remote user over an insecure channel, which widely deployed in e-commerce, e-government, information security, business management and many more, we propose a scheme without using smartcards. Due to the elliptic curve discrete logarithm problem, the proposed scheme is safe and efficient. Moreover, it achieves the following merits: no verification table, user friendly property, mutual authentication, no time-synchronization problem, suitable for multi-server environment and simple storage device.**

Keywords-mutual authentication; smart card; nonce; multi-server

I. Introduction

A remote user authentication scheme is a method to authenticate the legitimacy of remote users over an insecure channel. This technology has been widely deployed in various kinds of authentication applications which include remote host login, online banking, access remote service, mobile buyer, e-commerce, e-government, information security, business management, activation of security devices, and many more.

In 2009, Liao et al. [1] proposed an authentication scheme which provided many advantages but employed smart cards. And if the smart card of user U_i is stolen, the secret information stored in the smart card is easy to extract. Then a legal but spiteful user U_j can successfully launch an offline password guessing attack.

Meanwhile, another scheme without using smart cards is proposed by Rhee et al. [2] . Their schemes use a common storage device instead of smart card. However, the proposed scheme employed the timestamp to overcome the replay attack. This approach cannot avoid the serious time synchronization problem. Moreover, their scheme is based on the discrete logarithm problem over finite fields, which needs considerably longer code size and more communication cost.

Basically, the Elliptic Curve Cryptosystem (ECC) based on the discrete logarithm problem over elliptic curves affords greater efficiency than either integer factorization systems or discrete logarithm systems over finite fields, in terms of computational overheads, key sizes and bandwidth. In implementations, these savings mean higher speeds, lower power consumption and code size reductions.

In this paper, we propose a secure and efficient authentication scheme based on the elliptic curve cryptosystem. The proposed scheme achieves the following merits: no verification table, user friendly property, mutual authentication, no time-synchronization problem, suitable for multi-server environment and simple storage device. Such work is inspired by Popescu's [3] identification scheme.

The remainder of this paper is organized as follows. In section 2, we describe our mutual authentication scheme. In section 3, we discuss the security of the proposed scheme. In section 4, we evaluate the efficiency and summarize the merits. Finally, we give conclusion in section 5.

II. Proposed scheme

In this section, we propose a scheme which involves three participants: the users, the registration center and the distributed service providers.

The duties of the registration center are to generate the system parameters, sent the warrant securely to the remote service providers, deal with the registration requests of new users, change passwords for registered users and update the system.

The distributed service providers are the authentication servers in our scheme. Through the user authentication process, they authenticate the legitimacy of the remote users over an insecure channel; determine if some services can be provided to the users.

We now propose our scheme which can be separated into four phases: the initialization phase, the registration phase, the mutual authentication phases (as shown in Figure 1), and the update phase.

A Initialization phase

In this phase, the registration center generates the following parameters:

1) q , a field size, where q can be a big prime or a 2 power.

2) E , an elliptic curve over finite field F_q .

3) n , a large prime number.

4) $H(\cdot)$, $h(\cdot)$, secure one-way hash functions.

Supported by NSFC(No.10871088), SRFDP(No. 200802840003).

Then, the registration center secretly keeps only one secret value s.

B Registration phase

For a service provider S_j, the registration center executes the following operations:

1) Select two points P_j, Q_j, whose orders are n in the group $E(F_q)$.

2) Compute $P_{wj} = sP_j$, on the elliptic curve E.

Then, P_{wj} with the system parameters are sent securely to the service provider S_j, as one part of his warrant by some encrypted means.

On the other hand, a new user U_i submits the registration request to the registration center in a secure channel:

1) U_i submits his identity ID_i and his choosing password PW_i to the center.

2) The center computes $V_{ij} = H(ID_i)sP_j + H(PW_i)Q_j$ on the elliptic curve E, stores ID_i and V_{ij} into the memory of the user's device.

Considering the security of our scheme, neither $H(ID_i)sP_j$, $H(PW_i)Q_j$ nor V_{ij} is equal to the infinite point of E.

Remark. The subscript i comes from the user U_i, and the subscript j comes from the service provider S_j.

C Mutual authentication phase

When a user U_i wants to login to the remote server S_j to access the services, he should input his identity id_i and password pw_i into the client terminal device. Such device takes charge of reading the stored information of the user and executing the following operations:

(1) Check if id_i is identical to ID_i stored in the memory. If not, the login request fails. Otherwise, hash id_i, proceed to the next step.

(2) Randomly select n_1, n_2 as the nonce numbers of the user.

(3) Compute $C_1 = n_1P_j + n_2Q_j$.

(4) Send the message $M_1 = \{H(id_i), V_{ij}, C_1\}$ to the remote server.

Upon receiving the login message M_1, the remote server S_j performs the following operations:

(5) Randomly select n_3 as its nonce.

(6) Compute $W = H(id_i)P_{wj}$, $C_2 = h(C_1 + W)$.

(7) Send the message $M_2 = \{n_3, C_2\}$ to the user's device.

Upon receiving the respondent message M_2, the user authenticates the service provider S_j with the following step:

(8) Verify whether $C_2 = h(C_1 + V_{ij} - H(pw_i)Q_j)$.

If it equals, the user believes that the remote server is authenticated and the password pw_i is identical to PW_i, then proceeds to the next steps. Otherwise the login request fails.

(9) Compute $y_1 = n_1 + H(id_i)n_3$, $y_2 = n_2 + H(pw_i)n_3$.

(10) Send $M_3 = \{y_1, y_2\}$ to the remote server.

Upon receiving the message M_3, the server authenticates the user with the following step:

(11) Verify whether

$$y_1P_j + y_2Q_j - C_1 = n_3(V_{ij} + H(id_i)P_j - W).$$

If it equals, the remote server believes that U_i is authenticated.

The following figure shows the complete mutual authentication phase.

User' divce		RemoteServer
Check $id_i \ ?= ID_i$ Select n_1, n_2 Compute $C_1 = n_1P_j + n_2Q_j$	$\xrightarrow{M_1=\{H(id_i),V_{ij},C_1\}}$	Select n_3
	$\xleftarrow{M_2=\{n_3,C_2\}}$	Compute $W = H(id_i)P_{wj}$ Compute $C_2 = h(C_1 + W)$
Check $C_2 \ ?=$ $h(C_1 + V_{ij} - H(pw_i)Q_j)$ Compute $y_1 = n_1 + H(id_i)n_3$ Compute $y_2 = n_2 + H(pw_i)n_3$	$\xrightarrow{M_3=\{y_1,y_2\}}$	Check $y_1P_j + y_2Q_j - C_1$ $?= n_3(V_{ij} + H(id_i)P_j - W)$

Figure 1. Mutual authentication phase

D Update phase

When the user U_i wants to change his password PW_i to the new password PW_{ij}', he may contact the registration center with his identity id_i and password pw_i. Then the center performs the password update steps:

1. Check $id_i = ID_i$.

2. Check $V_{ij} = H(ID_i)sP_j + H(pw_i)Q_j$.

3. Replace V_{ij} with $V_{ij}' = V_{ij} + (H(PW_{ij}') - H(PW_i))Q_j$.

On the other hand, the user U_i can change his password without the help of the registration center as well.

Once the step *(8)* of the mutual authentication phase

succeeds, the client terminal device performs the step *3* of the password update phase as the registration center. Consequently, the password PW_i is changed to the new password PW_{ij}' for the service provider S_j.

Moreover, if the registration center wants to change the secret s after some periods of time, it requires an update of the system involving changing the V_{ij} and the P_{wj}. But an authorized user dose not needs to change his identity and password. This is efficient and user friendly.

III. Security analysis

In this section, the security of the proposed scheme is examined under the assumption that the Elliptic Curve Discrete Logarithm (ECDLP) is intractable. The ECDLP can be stated as follows.

Fix an elliptic curve over finite field F_q. xP reprsents the point P added to itself x times. Suppose Q is a multiple of P, so that $Q = xP$ for some x. Then the ECDLP is to determine x given P and Q.

In fact, the ECDLP admit only fully exponential time algorithms. It is believed that the ECDLP is harder than the integer factorization problem and the discrete logarithm problem modulo p, which both admit general algorithms that run in sub-exponential time.

We assume that the adversary has total control over the communication channel between the users and the servers. It means the attacker may intercept, insert, delete, or modify any message in the channel. Because the user's storage devices are not designed to provide tamper resistance, the stored information is vulnerable to keep secretly. Thus we assume that the attacker may extract the user's stored information.

A *Off-line password guessing attack*

We consider that a legitimate user may choose an easy to remember or meaningful password in our scheme. It seems very easy that the attacker may launch an off-line guessing attack to find PW_i just from the stored information $\{ID_i, V_{ij}\}$. However, although the attacker may obtain the system parameters $\{H(\cdot), P_i Q_j\}$, he cannot check whether a guessed password is correct or not without the secret value s or P_{wj}, which kept secretly by the registration center and the service provider S_j respectively.

B *Registration center secret key guessing attack*

Even though an attacker knows P_{wj}, he still cannot extract the registration center secret key s. This is due to the ECDLP. Similarly, a legitimate user cannot extract s from V_{ij} as well.

C *Replay attack*

The replay attack means a malicious intruder may replay the intercepted valid message to a legitimate user U_i or a remote service provider S_j again. In our scheme, the freshness of the transmitted messages is provided by the nonce n_1, n_2, n_3 to withstand replay attack. After incepting the previous login request $\{ID_i^*, V_{ij}^*, C_1^*\}$ from the user u_i, the attacker may replay it to the service provider S_j. Then, he can receive the acknowledge message $\{n_3, C_2\}$ from S_j. However, the attacker can not compute M_3 to respond to the server S_j, since he has no previous nonce n_1^*, n_2^* and password PW_i. Similarly, when we assume that the attacker replies a previous message $\{n_3^*, C_2^*\}$ to U_i, where C_2^* is associated with C_1^*. And C_1^* is related to nonce n_1^*, n_2^*. U_i Computes $C_2^* - h(C_1)$ and checks whether it is equal to $h(V_{ij} - H(PW_i)Q_j)$. It is obvious that the equality cannot hold since random numbers n_1, n_2 is not equal to n_1^*, n_2^*.

D *Parallel session attack*

The attacker may launch a parallel session attack by replaying the server's response message as the user's login request message at a later time. It means sending $\{H(id_i), V_{ij}, C_1\}$, $C_1 = C_2^*$, to S_j. However, this attack is impossible in our scheme, because the message structure of C_2 is totally different from C_1.

E *Masquerading server attack*

If an attacker attempts to impersonate the service provider S_j, he must try to forge $M_2 = \{n_3, C_2\}$ in the mutual authentication phase. To generate M_2 that passes the verification of U_i in step (8) of the mutual authentication phase, the attacker has to prepare a valid C_2. It means he should compute W. There are two ways to get a valid W: one is from V_{ij} and another is from P_{wj}. Performing the first way needs user's password. And in the second way, P_{wj} is equal to sP_j. Suppose to the contrary that the attacker can generate a valid P_{wj} with a non-negligible probability. This means that the adversary has solved the ECDLP. This contradicts the fact that ECDLP is hard to solve in polynomial time.

F *Masquerading user attack*

If an attacker attempts to impersonate the user U_i, he must try to forge the message $M_1 = \{H(id_i), V_{ij}, C_1\}$ and $M_3 = \{y_1, y_2\}$. Because $H(id_i)$, V_{ij} can be eavesdropped from old valid messages and C_1 can be forged, anyone can generate a login message M_1 and receive the message M_2 from the remote server S_j. However, the corresponded valid

$M_3 = \{y_1, y_2\}$ can not be computed without nonce $\{n_1, n_2\}$ and user's password PW_i. If

he wants to solve the valid y_1, y_2 from the authenticate equation in the step (11) of the mutual authentication phase, he will face the ECDLP.

G Insider attack

The insider attack is defined that any manager of the system purposely leaks the secret information, and then lead to serious security flaws of authentication scheme. In our scheme, the insider can obtain $H(PW_i)Q_j$ according to the values V_{ij} and W. It is hard to get PW_i or even $H(PW_i)$ from $H(PW_i)Q_j$, since it will confront with the difficulty and the complexity of the ECDLP. Moreover, the insider cannot disclose the secrets of the registration center from P_{wj}, since it will face the ECDLP as well. Therefore, even the secret P_{wj} is leaked by the insider, it won't lead to security flaws of the other service providers. After achieving a new warrant from the registration center and updating the user's V_{ij}, the communication between the server S_j and the users recover.

H Stolen verifier attack

We do not use any verification table, so any kind of stolen verifier attack are resisted in our scheme.

I Secret key forward secrecy

In our scheme, even if the secret keys of the registration center happens to be revealed, the attacker cannot impersonate any user by using the revealed keys. It is because that he cannot compute two quantities y_1, y_2 without the knowledge of the user's nonce n_1, n_2 and the password PW_i.

J Secure password change

In the update phase, the user can freely change password with or without the help of registration center. Before accepting the new password, our scheme verifies the correctness of the old password by step 2 in the update phase or step (8) in the mutual authentication phase. Therefore, an unauthorized user can not easily change the password without knowing the corresponding old password. The legality of the user thus is assured.

IV. Efficiency and functionality Analysis

In this section, we evaluate the efficiency and summarize the merits of the proposed scheme.

A Efficiency analysis

The elliptic curve cryptosystem (ECC) affords more efficient implementations than other public key systems. In fact, the integer factorization systems (RSA) and the discrete logarithm systems (DSA) should employ a 1024 bit modulus, while a 160 bit modulus should be sufficient for the ECC. And, 300 bit ECC is a great deal more secure than 2000 bit RSA or DSA. Our scheme is exactly based on ECC, hence achieves higher speeds, lower power consumption and code size reductions than other schemes such as [2].

The storage and communication cost of our scheme is very low, however, the computation cost seems beyond the computational capability of the smart cards. Therefore, we perform the calculation on the user's terminal device instead of the smart card. Although our scheme avoids using smart card, it achieves all the benefits and does not damage the security. And using a common storage device provides additional advantages such as lower cost, convenient, practical for widespread distribution.

Moreover, algorithms for full addition on an elliptic curve (choosing and implementing the appropriate formula for the given pair of points) as well as setting the system parameters can be found in IEEE 1363 standards [4]. In fact, there are a lot of work on speeding the points multiplication and so making the ECC more efficient. For more information about the various points representations and points multiplication algorithms, see [5-8].

B Functionality analysis

(1) No verification table.

The schemes described in [9, 10], maintain the verification tables of the passwords or other personal information to authenticate the users. Considering the serious key management problems of those large systems including many entities, we should avoid any verification tables. In our scheme, it is not necessary for the registration center and the service providers to set up any verification tables.

(2) User friendly property.

It means users can be able to freely choose and change their passwords, which should not be computed by the system as in the schemes [11, 12]. As we mentioned before, the proposed scheme is user friendly, and resists off-line password guessing attack. Moreover, the password can be conveniently updated at the user's terminal. It is convenient and efficient for users.

(3) Mutual authentication.

Since several Internet frauds about unilateral authentication that the remote user cannot authenticate the system have been reported [13], mutual authentication between the user and the system is indispensable to ensure the security. The proposed scheme achieves mutual authentication and resist the impersonate remote server attack and the impersonate user attack.

(4) No time-synchronization problem.

Many schemes [14-17], employ the concept of timestamps which require the system clock synchronization to withstand the attack of replaying previously intercepted messages. However, the time synchronization is a serious problem due to the network environment and transmission delay. The proposed scheme is nonce-based. We use the nonce to keep the freshness of the transmitted message, hence overcome the

serious time synchronization problem.

(5) Suitable for multi-server environment.

In our scheme, the distributed service providers do not need the knowledge of the secret s of the registration center. Even one service provider compromised, it won't affect the security of the communication between the users and the other service providers. Besides, the proposed scheme allows the user to register only once at the registration center and then he can access all the remote service providers.

(6) Simple storage device.

The user in our scheme just needs a com mon storage device such as a USB stick, an mp3 or mp4, or a plastic card with an embedded memory chip. Compared with smart cards, such device is more convenient and less cost.

V. Conclusion

In this paper, we have proposed a rem ote mutual authentication scheme without using smart card. The proposed scheme is suitable for multi-server environment. Due to the elliptic curve discrete logarithm problem, our scheme is s afe and efficient.

Acknowledgment

The author wish to thank the referee's hard work.

References

[1] Y.Liao, S. Wang, A secure dynamic ID based remote user authentication scheme for multi-server environment, Computer standards & Interfaces 31 (2009) 24-29.

[2] H. S. Rhee, J. O. Kwon, D. H. Lee, A remote user authentication scheme without using smart cards, Computer standards & Interfaces 31 (2009) 6-13.

[3] C. Popescu, An identification scheme based on the elliptic curve discrete logarithm problem, High Performance Computing in the Asia-Pacific Region, 2000. Proceedings. The Fourth International Conference/Exhibition on, Vol. 2, (2000) 624-625.

[4] IEEE, Standard specifications for public- key cryptography, IEEE Computer Society,Vol. 29, August, (2000).

[5] S. Atay, A. Koltuksuz, H. Hybyl, S. Eren, Computational cost analysis of elliptic curve arithmetic, 2006 International Conference on Hybrid Information Technology, Vol. 1, November, (2006) 578-582.

[6] K. Jarvinen, M. Tommiska, J. Skytta, A scalable architecture for elliptic curve point multiplication, IEEE Field-Programmable Technology, (2004) 303-306.

[7] M. Morales-Sandoval, C. Feregrino-Uribe, GF(2^m) Arithmetic modules for elliptic curve cryptography, Proceedings of the IEEE International Conference on Re-ConFigurable Computing and FPGAs, IEEE Computer Society, September, (2006) 176-183.

[8] D. M. Schinianakis, A. P. Kakarountas, T. Stouraitis, A new approach to elliptic curve cryptography: an RNS architecture, IEEE MELECON, Benalmadena, Spain, May, (2006) 1241-1245.

[9] L. Lamport, Password authentication with insecure communication, Communications of ACM, Vol. 24, (1981) 770-772.

[10] H. Y. Chien, J. K. Jan, Robust and simple authentication protocol, Computer Journal, 46, (2003) 193-201.

[11] M. S. Hwang, L. H. Li, A new remote user authentication scheme using smart cards, IEEE Transactions on Consumer Electronics, Vol. 46, No. 1, February, (2000) 28-30.

[12] H. M. Sun, An efficient remote use authentication scheme using smart card, IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, November, (2000) 958-961.

[13] D. Mcelroy, E. Turban, Using smart cards in electronic commerce, International Journal of Information Management, Vol. 18, No. 1, (1998) 61-72.

[14] H. Y. Chien, J. K. Jan, Y. M. Tseng, An efficient and practical solution to remote authentication: smart card, Computer & Security, Vol. 21, No. 4, (2002) 372-375.

[15] Y. Wang, J. Liu, F. Xiao, J. Dan, A more efficient and secure dynamic ID-based remote user authentication scheme, Computer Communications 32 (2009), 583-585.

[16] H. Hsiang, W. Shihi, Weaknesses and improvements of the Yoon-Ryu-Yoo remote user authentication scheme using smart cards, Computer Communications 32 (2009), 649-652.

[17] S. Kim, M. Chung, More secure remote user authentication scheme, Computer Communications 32 (2009) 1018-1021.