An Efficient Bilateral Remote User Authentication Scheme with Smart Cards

Al-Sakib Khan Pathan and Choong Seon Hong
Networking Lab, Department of Computer Engineering, Kyung Hee University
spathan@networking.khu.ac.kr and cshong@khu.ac.kr

Abstract

In this paper, we propose an efficient bilateral remote user authentication scheme with smart cards. Our scheme ensures both-way authentication, so that any attempt of the adversary to affect the secure communications between the authentication server and the user could not be successful. We also present a brief analysis of our proposed scheme and show that it is well-resistant against the known attacks in remote user authentication process.

1. INTRODUCTION

The possible use of insecure channels during the communications in a remote user authentication process necessitates the exploitation of competent security mechanisms for the exchanged information between the remote system and the user. Many of the previously proposed schemes like [1], [2], [3], [4] etc. concentrate on unilateral authentication where the server messages are considered to be fully secured and only the user's validity is verified. With the increasing use of remote login systems in banking, e-commerce applications and distributed networking, use of efficient methods for remote authentication have become imperative. To deal with this issue, in this paper, we propose a new mutual authentication scheme in which both the participating entities verify each other's authenticity.

The structure of this paper is as follows: Following the Section 1, Section 2 states the basic terms and preliminaries for our scheme, Section 3 presents our bilateral authentication scheme, Section 4 presents a brief cryptanalysis of our scheme and Section 5 concludes the paper with future research directions.

2. BASIC TERMS AND PRELIMINARIES

LU-decomposition [5] underpins the authentication process of our scheme. LU-decomposition is a procedure for decomposing a square matrix A ($N \times N$) into a product of a lower triangular matrix L and an upper triangular matrix U, such that, A = LU, where, L and U have the forms,

$$L_{ij} = \begin{cases} l_{ij} & for \quad i \geq j \\ 0 & for \quad i < j \end{cases}$$

$$U_{ij} = \begin{cases} u_{ij} & for \quad i \leq j \\ 0 & for \quad i > j \end{cases}$$

So, for example, for a square matrix of dimension 4 × 4, equation A=LU, looks like:

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (1)$$

According to the definition, elementary matrix *E* is an N × N matrix if it can be obtained from the identity matrix I_n by using one and only one elementary row operation (e.g., elimination, scaling, or interchange) [7]. Elementary row operations are, $R_i \leftrightarrow R_j$, $cR_i \leftrightarrow R_i$, $R_i + cR_j \leftrightarrow R_i$. If the elementary matrices corresponding to the row operations that we use are, E_1, E_2 ….. E_k, then, $E_k \cdots .E_2E_1A$ = U. Hence, A = $(E_k \cdots .E_2E_1)^{-1}$U or L = $E_k^{-1} \cdots E_2^{-1} E_1^{-1}$.

3. OUR PROPOSED SCHEME

3.1 Pre-Processing

Before applying our scheme, a large pool of keys with size *s* is generated offline. This pool of huge number of keys is used for the formation of a secret symmetric key matrix A ($N \times N$) which is stored by the server. N also indicates the maximum number of users that could be supported by the symmetric matrix. Each element A_{ij} of A is assigned a distinct key from the key pool (generated earlier) such that A_{ij} = A_{ji} for, i,j = $\overline{1, N}$. Now, LU-decomposition is applied on matrix A to divide it into the lower triangular matrix L and upper triangular matrix U. As A is symmetric, the multiplication of the *x*th row of L and *y*th column of U (K_{xy}) generates the same value as the multiplication of *y*th row of L and *x*th column of U (K_{yx}) generates.

3.2 Details of Our Scheme

User Registration Phase. We assume that, this phase occurs over a secure channel. Let F_h be a secure one-way hash function [6] which produces 64 bit outputs. A one-way function is a transfer function *f* where given *p*, it is fairly easy to compute, $q = f(p)$ in the forward direction, but given *q*, it is computationally very difficult to find out a *p* using the inverse such that, $p = f^{-1}(q)$.

In the registration phase, the user U_a first submits his identity (ID_a) and arbitrarily chosen password PW_a to the KIC (Key Information Center, which is a part of the AS) for registration. In turn the KIC associated with the AS does the following steps:

1. Generates two random numbers *x* and *y* within the range N (the dimension of the matrices).
2. It selects the *x*th row from L matrix $L_R(x)$, *x*th column from U matrix $U_C(x)$, and *y*th column from U matrix $U_C(y)$.
3. Computes, $L_R(x) \times U_C(y) = K_{xy}$ and $\theta = F_h(ID_a \oplus K_{xy}) \oplus PW_a$ where ⊕ denotes a bitwise Exclusive-OR (XOR) operation.
4. Issues a smart card containing (F_h, K_{xy}, ν, $U_C(x)$, θ) to the user, where $\nu = (\varphi \oplus y)$ with φ is an arbitrary number which is kept secret and owned by AS. Same value of φ is used for all the users to be served by the Authentication Server.

It should be mentioned that, for all the calculations, 64 bit outputs are used and variable length inputs are made 64 bits using padding. In fact, the user can choose the password PW_a and identity ID_a according to his own will but shorter length inputs are made the same length as the key K_{xy} (64 bits) using padding.

Login & Bilateral Authentication Phase. When the user needs to login to the remote system, he attaches the smart card to the input device and keys in his identity ID_a and password PW_a.

The smart card performs the following operations:

1. Generates a random number *r* with the same length of K_{xy} and computes $H_a = K_{xy} \oplus F_h(r)$, and computes, $S_a = \theta \oplus PW_a \oplus r$.
2. Sends the login request message, M = (ID_a, H_a, ν, $U_C(x)$, S_a, T), (here, T is the current timestamp) to the Authentication Server.

After receiving the login request message M, the server performs the operations:

1. Checks the validity of ID_a. If the format is different than the allowed format, it rejects the request.
2. Tests the time interval $(T' - T) \le \Delta T$, where T′ is the timestamp of receiving the message M and ΔT is the maximum allowed time interval for transmission delay. If ΔT is greater than its boundary condition, the request is rejected.
3. Now Authentication Server computes, $(\nu \oplus \varphi)$ which eventually generates the value of *y* for that user. AS now knows which row is to be selected from the L matrix for this particular user and selects the *y*th row $L_R(y)$ and computes, $L_R(y) \times U_C(x) = K_{yx}$
4. Computes $t = F_h(ID_a \oplus K_{yx})$, and $r' = t \oplus S_a$.
5. Computes, $K_{xy}' = H_a \oplus F_h(r')$, which is expected to generate the value of $K_{xy} = K_{xy}'$ for a legitimate user, as in the previous step *r′* should be equal to the random number (*r*) generated by the user for this particular login request.
6. Now the server checks whether the condition, $K_{xy} = K_{yx}$ holds or not. If it does not hold, the server detects the user U_a as an invalid user otherwise, detects as a valid user. For the invalid user(s) the server rejects the login request and for the legitimate user(s) it proceeds to the next steps.
7. Computes $M' = F_h(K_{yx}\ \text{X-NOR}\ T'')$, where T″ is the current timestamp, X-NOR means the bitwise Exclusive-NOR (XNOR) operation and sends (M′ , T″) to the user U_a. Upon receiving the message M′ from the AS, the user U_a follows the steps:

1. Verifies the boundary condition, $T''' - T'' \le \Delta T$, where T‴ is the timestamp of receiving the message M′.
2. Then it computes, $F_h(K_{xy}\ \text{X-NOR}\ T'')$ and if it equals to the received M′, the user verifies the legitimacy of the Authentication Server.

As a symmetric matrix is used for LU-decomposition, $K_{xy}=K_{yx}$ and the procedure works for the legitimate authentication server and user. Thus, our scheme ensures bilateral verification.

Password Renewal. When a legitimate user wants to renew his password, the following operations are done.

The user enters both the old and new password for changing his password. Let, PW_a' is the new password chosen by the user instead of the old password, PW_a. Then, the server computes:

1. $\theta' = \theta \oplus PW_a \oplus PW_a'$
   $= F_h(ID_a \oplus K_{xy}) \oplus PW_a \oplus PW_a \oplus PW_a'$
2. Replaces θ with θ′ in the smart card.

It should be noted that, like the registration phase, password renewal phase takes place over a secure channel.

4 CRYPTANALYSIS OF OUR SCHEME

The underlying common secret key between the user and the authentication server is formed in our method, on the basis of LU-decomposition. The basic idea is to pre-store part of the information in the smart card so that it could be used to calculate the required information thus to help for the mutual verification in the remote authentication process. As mentioned earlier, the communications in the login and authentication phases are occurred over insecure channels. Many of the proposed schemes in this area have been found vulnerable because of the reason that, the attackers first listen to the exchanged messages, then using some computations try to deduce the secret information. Hence, the messages that are exchanged between the user and the authentication server must be well-protected or cryptic so that the adversary cannot use the information to draw any of the secret information for any sort of attack.

In the login phase in our scheme, the user sends the login request message, M = (ID_a, H_a, ν, $U_C(x)$, S_a, T). From this message an attacker cannot find any useful information that it can exploit for finding out the secret common key K_{xy} (or K_{yx}). Basically, in this message, H_a contains the information about the secret key that is to be tested, ν carries the information about the specific row information that is to be selected by the server so that the symmetric key could be generated by the multiplication operation. As the randomly generated value *r* is used to compute H_a, this value changes from session to session and it is very difficult for the adversary to derive the key from this value without knowing the exact value of *r* used for that particular session. Another parameter S_a contains the random value *r* which is computed before deriving the secret key value. In fact, it is easy to see that, $r' = t \oplus S_a = F_h(ID_a \oplus K_{yx}) \oplus F_h(ID_a \oplus K_{xy}) \oplus PW_a \oplus PW_a \oplus r = r$.

For the legitimate users, K_{xy} must be equal to K_{yx} and will eventually cancelled out to generate the value of the random number used in that particular session. The attacker can get the value of $U_C(x)$ but without knowing the corresponding row information, it has no way to generate the secret common key.

Our scheme is resistant to replay attacks [8] as replaying an old login request message will be detected in the step 2 of authentication process in the server. Masquerading or Impersonation attack could not be launched against our scheme as the adversaries have no way to derive the secret information of a valid user. The only possible limitation of our scheme is that, the number of users that could be supported by the server is restricted by the dimension of the secret symmetric matrix.

5 CONCLUSION AND FUTURE WORKS

In this paper, we have proposed a new and efficient remote user authentication scheme using smart cards which ensures bilateral authentication so that both the parties participating in the process could verify each other' s validity. Our scheme also supports password renewal by the user. As our future work, we would like to investigate how to improve the efficiency of our scheme in terms of computational costs and memory usage.

References

[1] T. C. Wu, " Remote login authentication scheme based on a geometric approach " Computer Communication, vol. 18 no. 12, 1995, pp. 959 – 963.

[2] Hwang, M.-S. and Li, L.-H., " A New Remote User Authentication Scheme Using Smart Cards" , IEEE Transactions on Consumer Electronics, Vol. 46, No. 1, February, 2000, pp. 28-30.

[3] Sun, H.-M., " An Efficient Remote User Authentication Scheme Using Smart Cards" , IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, November, 2000, pp. 958-961.

[4] Shen, J.-J., Lin, C.-W., and Hwang, M.-S., " A Modified Remote User Authentication Scheme Using Smart Cards " , IEEE Transactions on Consumer Electronics, Vol. 49, No. 2, May, 2003, pp. 414-416.

[5] Zarowski, C. J., " An Introduction to Numerical Analysis for Electrical and Computer Engineers" , Hoboken, NJ John Wiley & Sons, Inc. (US), 2004, pp. 148-151.

[6] National Institute of Standards and Technology, NIST FIPS PUB 180, " Secure hash standard," U.S. Department of Commerce, 1993.

[7] Nakos, G., and Joyner, D., Linear Algebra with Applications, Brooks/Cole USA, 1998, pp. 188-194.

[8] Syverson, P., "A Taxonomy of Replay Attacks", Proc. Computer Security Foundations Workshop VII, 1994, CSFW 7, 14-16 June, 1994, pp. 187 – 191.