Password Authenticated Key Agreement Protocol for Multi-servers Architecture

Ren-Junn Hwang Sheng-Hua Shiau

Department of Computer Science and Information Engineering, Tamkang University, Tamsui, Taipei, Taiwan 251, R.O.C.

E-mail: junhwang@ms35.hinet.net, 891190067@s91.tku.edu.tw

Abstract

This paper proposes an efficient password authenticated key agreement protocol for multi-servers architecture. The authenticated key agreement protocol is a good solution to provide authentication and confidentiality. The identity authentication and confidentiality are two important primary security services for the open network environment. The proposed scheme allows user to access multi-server securely by keeping one weak password and a smart card only. The client user and server will authenticate each other in the proposed scheme. They will agree a secret common session key for each request in the ending of the proposed scheme. Furthermore, the proposed scheme is based on straight line of geometry and symmetric cryptosystem. It does not use the overload cryptography operations, it is more efficient than the previous results.

1. Introduction

Identity authentication and confidentiality are two primary security services of the open network. It is important to authenticate each other when the client and the server want to communicate through the open network. How to protect the confidential data transmitted between the client and server is also an important issue. *Authenticated key agreement protocol* is a good solution to provide authentication and confidentiality services. By this protocol, the client and server not only authenticate each other but also generate a secret session key. They can use the session key and cryptosystem to protect the confidential transmitted data. Password-based mechanism is the most widely used method for user authentication since it allows people to choose and keep password by himself. In conventional password authenticated key agreement protocol [1,2,3,4,5,6,12, 13,14,16], a remote user use one password to login one server. The user should use different passwords to login different servers via the open network by security consideration. He should keep many different passwords secretly for the servers that he is authorized to access. It is not convenient and practical. There are some password authenticated key agreement schemes for multi-server architectures [7,10,11] are proposed. It is an important issue to enhance the security and convenience for the open network.

In 2001, Li et al. proposed a remote password authentication scheme for multi-servers using neural networks [11]. Their scheme allows a remote user to login several servers by using the same password. But in their scheme, each server needs to store the weights of the classification network, and it spends too much time on training neural networks. In 2003, Lin et al. proposed an authentication scheme based on the ElGamal digital signature scheme and the simple geometric properties on the Euclidean plane [10]. Their scheme has to take massive computation and communication costs. In 2004, Juang proposed another password authenticated key agreement scheme [7]. His scheme establishes a common session key between user and server in the final. However, in Juang's scheme, each server has to protect and keep an encrypted key table securely.

In this paper, we propose a password authenticated key agreement protocol for multi-servers architecture. In our scheme, each user only keeps one identity and password, but he can login many different servers based on it. Each server in our scheme does not need to keep a verification table or extra data to authenticate the login user. The user and server will authenticate

0-7803-9305-8/05/$20.00 ©2005 IEEE

each other, and establish a common session key in our scheme. The session key will be conformed by user and server authentication. Furthermore, the proposed scheme does not use the overload cryptography operations, it is more efficient than the previous results.

2. The proposed scheme

There are three roles in the proposed scheme: users, servers and trusted management server. The trusted management server manages a group of servers. When a user decides to login a server of this group, he should register in the trusted management server first. The proposed scheme includes three phases: the initial phase, the registration phase and the login phase. The trusted management server decides one different secret shared key to each server that he managed in the initial phase. In the registration phase, the user submits his identity, password and all of the servers that he would like to access in the future to the trusted management server via the secure channel. The trusted management server verifies the validity of the user and stores some secret parameters in the smart card. He hands out the smart card to the user. The user authorized to access each server that he specified by using the password and smart card through the login phase of the proposed scheme. The client user and login server also agree a secret session key after the login phase. They can exchange secret information confidentially based on the secret session key. The following paragraphs detail these three phases.

2.1 The initial phase

We assume that the trusted management server manage n servers. These servers are show as $S_1, S_2,\ldots, S_n$. Initially, the trusted management server chooses system parameter p, where p is a prime. He decides n secret keys for each server, show as $d_{S1}, d_{S2},\ldots, d_{Sn}$.

2.2 The registration phase

When a remote user wants to login the multi-server system, he must registers to the trusted management server first. We assume that a user U_i would like to login the servers $S_1, S_2,\ldots, S_m$, where $\{S_1, S_2,\ldots, S_m\}$ is a subgroup of $\{S_1, S_2,\ldots, S_n\}$. The registration steps are as follows:

Step 1: User U_i send his identity ID_i , password PW_i and the ID of $S_1, S_2,\ldots, S_m$ to the trusted management server.

Step 2: Trusted management server verifies the validity of the user. For each server S_j, the trusted management server computes the following values.

$d_{Uij} = h\,(ID_i \oplus d_{Sj})$,

$v^1_{ij} = h\,(PW_i \oplus ID_{Sj} \oplus T)$, where T is timestamp,

$C_{ij} = d_{Uij} \oplus h(PW_i)$,

$D_{ij} = h(v^1_{ij}) \oplus h(PW_i)$.

Step 3: Trusted management server stores the values ID_i ,C_{ij} ,D_{ij} in a smart card, and hands out the smart card to User U_i .

2.3 The login phase

Assume the authorized user U_i would like to login the server S_j at t-th times, where S_j belongs to the servers $\{S_1, S_2,\ldots, S_m\}$. The user uses his ID, password and smart card to login the server S_j as the following steps. Figure 1 shows the diagram of this phase.

Step 1: User U_i calculates X from the parameters stored in the smart card. Then he sends the message $\{ID_i, X\}$ to Server S_j.

$X = C_{ij} \oplus D_{ij}$

Step 2: Server S_j calculates d_{Uij} and $h(v^t_{ij})$ as:

$d_{Uij} = h\,(ID_i \oplus d_{Sj})$,

$h(v^t_{ij}) = d_{Uij} \oplus X$.

Server S_j randomly selects two values N_1, N_2 and constructs a straight line L_{ij} based on two points $(d_{Uij}$, $h(v^t_{ij}))$ and (N_1, N_2). L_{ij}: $y = f(x) = ax+b \bmod p$, where $a = (N_2 - h(v^t_{ij})) / (N_1 - d_{Uij}) \bmod p$, $b = N_2 - N_1\,(N_2 - h(v^t_{ij})) / (N_1 - d_{Uij}) \bmod p$,and $a,b \neq 0$.

Step 3: Server S_j uses the line equation $y = f(x) = ax+b \bmod p$ to calculate following values:

$C_1 = a \oplus d_{Uij}$,

$C_2 = b \oplus h(v^t_{ij})$,

$v^{t+1}_{ij} = f(\,h(v^t_{ij})\,)$,

$R_1 = h\,(ID_{Sj}, v^{t+1}_{ij}, a\,)$.

Then S_j returns the message $\{C_1$, C_2 , $R_1\}$ to User U_i .

Step 4: User U_i reconstructs the straight line L_{ij}' as follows.

$a' = C_1 \oplus C_{ij} \oplus h(PW_i)$,

$b' = C_2 \oplus D_{ij} \oplus h(PW_i)$,

L_{ij}': $y = f'(x) = a'x+b' \bmod p$.

He calculates $v^{t+1'}_{ij}$ based on L_{ij}' as

$v^{t+1'}_{ij} = f'(D_{ij} \oplus h(PW_i))$.

User U_i checks $h\,(ID_{Sj}, v^{t+1'}_{ij}, a)$ and R_1 are equal or not, if not, terminate the procedure.

Step 5: User U_i selects a random number N_3, and calculates the following values:

$R_2 = h\,(ID_i, v^{t+1'}_{ij}, b)$,

$K = N_3 \oplus v^{t+1'}_{ij}$,

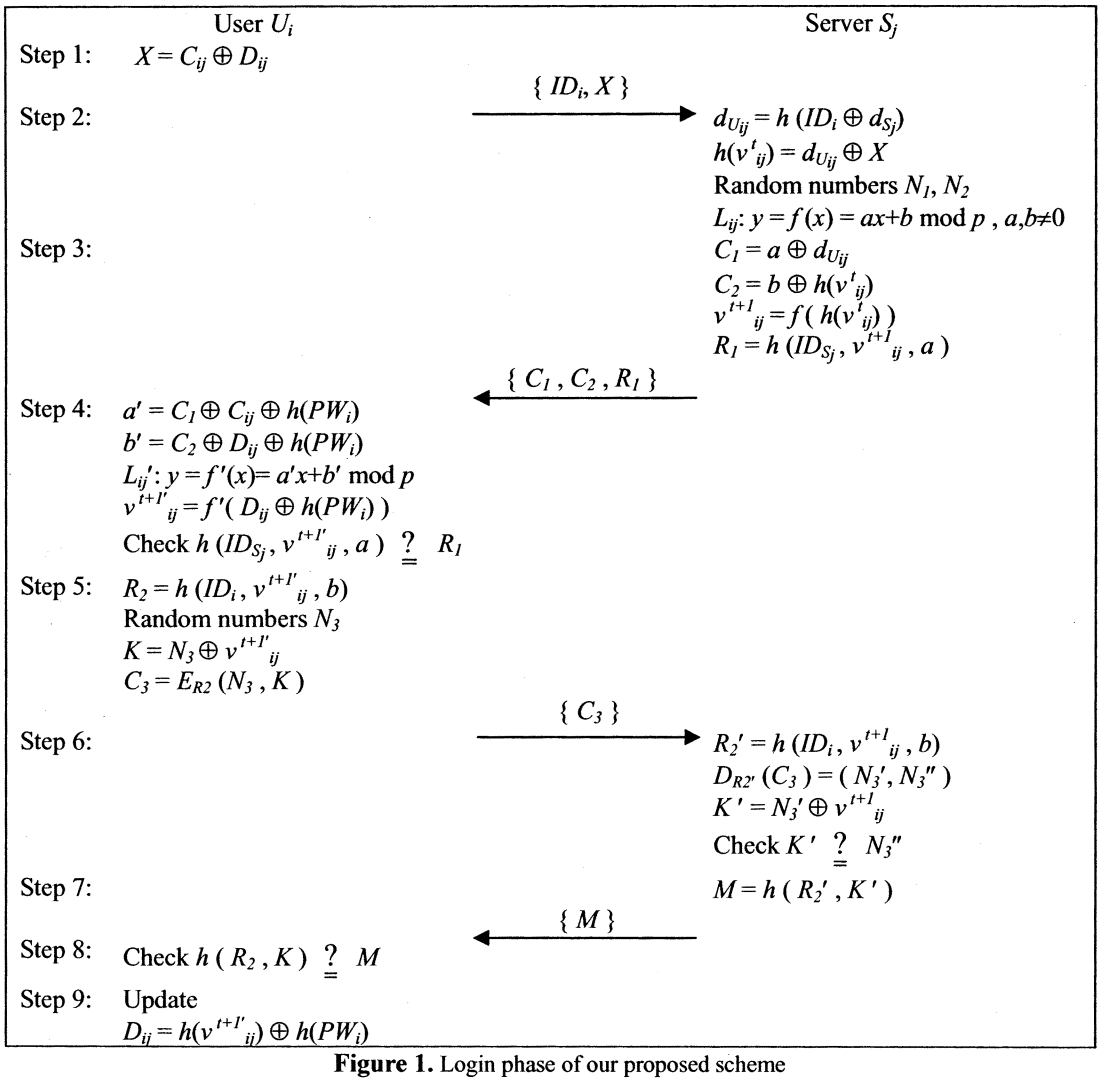

Figure 1. Login phase of our proposed scheme

$C_3 = E_{R2}(N_3, K)$, where E_{R2} shows symmetric encryption with encrypt key R_2.

User U_i sends $\{ C_3 \}$ to Server S_j.

K is the session key between User U_i and Server S_j.

Step 6: Server S_j performs the following substeps:

Step 6-1: $R_2' = h(ID_i, v^{t+1}_{ij}, b)$.

Step 6-2: $D_{R2'}(C_3) = (N_3', N_3'')$, where $D_{R2'}$ denotes symmetric decryption with decrypt key R_2'

Step 6-3: $K' = N_3' \oplus v^{t+1}_{ij}$.

Step 6-4: Check K' and N_3'' are equal or not, if not, terminate the procedure, otherwise K' is the session key between User U_i and Server S_j.

Step 7: Server S_j computes $M = h(R_2', K')$, and sends $\{ M \}$ to User U_i.

Step 8: User U_i checks $h(R_2, K)$ and M are equal or not, if not, terminate the procedure.

Step 9: User U_i updates the value D_{ij} of the smart card with $D_{ij} = h(v^{t+1'}_{ij}) \oplus h(PW_i)$.

3. Security analysis

This section analyzes the security of the proposed scheme. We demonstrate that the proposed scheme achieves the mutual authentication and explicit key authentication.

3.1 Mutual authentication

In general, the server should verify the validity of the user who would like to login. The user usually does not authenticate the server. However, there are some conspirators construct counterfeit servers in the

Internet to cheat the legal user of his secret data. It is important that the login user and the server have to authenticate each other in the open network. We show that the proposed scheme provides mutual authentication in the following paragraphs.

(a) User authenticates the server

In the proposed scheme, the user U_i reconstructed the straight line L_{ij}' based on the message $\{C_1, C_2, R_1\}$ sent by the server in Step 4 of the login phase. He also checks $h(ID_{S_j}, v^{t+1'}{}_{ij}, a)$ and R_1 are equal or not, where $v^{t+1'}{}_{ij}$ is generated from L_{ij}'. Nobody except the server keeps the secret key d_{S_j} and only the person who keeps secret key d_{S_j} can get the correct $d_{U_{ij}}$ and the same straight line L_{ij} with the user U_i. If $h(ID_{S_j}, v^{t+1'}{}_{ij}, a)$ is equal to R_1, the user U_i makes sure that the identity of server S_j and the two straight lines L_{ij} and L_{ij}' are equal.

(b) The server authenticates the login user

The server decrypts message C_3 with encryption key R_2' to get N_3'' and check K' is equal to N_3'' or not in Step 6 of the login phase. Only the people who keeps the correct password PW_i and the corresponding smart card can generate the correct encryption key $R_2 (=R_2')$ to perform $C_3 = E_{R2}(N_3, K)$ in Step 5. Nobody except the validity user keeps PW_i and the smart card. If K' and N_3'' are equal, the server S_j can make sure that the user U_i.

3.2 Explicit key authentication

Two parties in a key agreement scheme not only ensure the participant can get enough information to calculates the common session key, but also ensure the participant has calculate the correct session key between them, then we say that the key agreement scheme provides explicit key authentication.

User U_i and the server S_j compute a common session key based on N_3 and $v^{t+1}{}_{ij}$ in the login phase of the proposed scheme. By Subsection 3.1, we demonstrate that User U_i and the server S_j authenticates each other and they ensure that the other party computes the same values $v^{t+1}{}_{ij}$ ($v^{t+1}{}_{ij} = v^{t+1'}{}_{ij}$) and N_3 ($N_3 = N_3'$) in Steps 4 and 6 of the login phase respectively. It is clear that the proposed scheme provides explicit key authentication.

4. Comparison

In this section, we make the comparisons among our proposed scheme, Juang's scheme [7] and Lin et al.'s scheme [10]. The comparison is divides into three parts: the security properties, the computational cost and the communicational cost.

4.1 The security properties

This subsection discusses the security properties that are provided Juang's scheme [7], Lin et al.'s scheme [10] and our proposed scheme. Those security properties include the mutual authentication, the explicit key authentication and extra memory space require in servers. The security properties comparison results are also shown on Table 1.

In Juang's scheme, when a user registers to the registration center in registration phase, the registration center not only delivers a smart card to user, but also needs to transmit an encrypt message to each server that the user want to access. The message was encrypted by a common session key between the registration center and server, each server needs extra memory space to store this encrypted message. The user and server ware establish a common session key in Juang's scheme, but it does not provide explicit key authentication. Juang's scheme provides mutual authentication for user and server.

In Lin et al.'s scheme, each server only needs to store their key pair (public key and secret key), they don't need to store extra value. However, their scheme doesn't provide mutual authentication. When a user login a server, the server authenticates the user, but the user cannot ensure that he does not login a counterfeit server in Lin et al.'s scheme. The user and server in Lin et al.'s scheme do not establish a common session key, they should perform another method to protect the secret exchange data.

In our proposed scheme, each server doesn't need to maintain a verification table or any extra data. Our scheme provides mutual authentication for the user and the server, they establishes a secret common session key and provide explicit key authentication.

4.2 The computational cost

This section discusses the computational cost of

Table 1. The security properties comparison

	Our proposed scheme	Juang 2004	L H L 2003
Mutual authentication	Y	Y	N
Explicit key authentication	Y	N	No session key
Server has to maintain extra secret data	N	Y	N

Table 2. The computational cost and the communicational cost comparisons

	Our proposed scheme		Juang 2004		L H L 2003	
	User	Server	User	Server	User	Server
Symmetric encryption	1		2	1		
Symmetric decryption		1	1	3		
Modular exponentiation					4	6
Modular division		1			1	1
Modular multiplication	1	2			4	4
Round	2	2	2	1	1	1
Message size	ID + 3×128 bits	4×128 bits	ID + 4×128 bits	3×128 bits	ID + T + 7×1024 bits	

XOR denote the exclusive-OR operation
ID denote the bit length of identity
T denote the bit length of timestamp

Juang's scheme [7], Lin et al.'s scheme [10] and our proposed scheme. The first five rows of Table 2 show the comparison results in the computational cost. The computational cost of exclusive-OR operation, Hash function and modular addition are smaller than the symmetric encryption and the modular multiplication. We ignore the cost of exclusive-OR operation, Hash function and modular addition. In Juang's scheme, the user performs 3 symmetric encryption / decryption operations and the server performs 4 symmetric encryption / decryption operations. In Lin et al.'s scheme, the user needs 4 modular exponentiations, 1 modular division and 4 modular multiplications, and the server needs 6 modular exponentiations, 1 modular division and 4 modular multiplications. The user in our proposed scheme needs 1 symmetric encryption operations and 1 modular multiplication operations. The server needs 1 symmetric decryption operations, 2 modular multiplication operations and 1 modular division. Since it still fast than symmetric encryption and decryption. And our scheme needs 1 more round, this is because our scheme provides implicit key authentication. The computation cost of modular exponentiation is larger than the modular division. The modular division is larger than modular multiplication [8, 9]. Our proposed scheme is more efficient than the others.

4.3 The communicational cost

This subsection discusses the communicational cost of Juang's scheme [7], Lin et al.'s scheme [10] and our proposed scheme. The last two rows of Table 2 show the comparison results in the communicational cost. The security of Lin et al.'s scheme [10] is based on discrete logarithm problem. We assume that p in their scheme is 1024 bits by security consideration. In Juang's scheme [7] and our scheme, we use the one-way hash function and symmetric cryptosystem. We assume that the output size of one-way hash function, the block size of symmetric cryptosystem and the size of random number are 128 bits.

In Lin et al.'s scheme, the communicational cost between user and server is 1 ID, 1 Timestamp and 7×1024 bits. In Juang's scheme, the message size transmit from user to server is 1 ID and 4×128 bits (this include one nonce and three symmetric encryption blocks), the message size transmit from server to user is 3×128 bits (this include three symmetric encryption blocks). In our proposed scheme, the message size transmit from user to server is 1 ID and 3×128 bits (this include one 128 bits value and two symmetric encryption blocks), the message size transmit from server to user is 4×128 bits (this include two 128 bits values and two hash results). The communicational cost in our scheme is much smaller than Lin et al.'s scheme, and is the same with Juang's scheme. Although the communicational cost of Juang's scheme and our scheme are equal, our proposed scheme provides more security properties than Juang's scheme.

5. Conclusion

This paper proposes an efficient password authenticated key agreement protocol for multi-servers architecture. It is a good solution of the identity authentication and confidentiality for the open network.

There are some conspirators construct counterfeit servers in the Internet to cheat the legal user of his secret data. Especially, not only the server authenticates the client user, but also the client user authenticates the server in the proposed scheme. It provides mutual authentication and each server doesn't need to maintain a verifier table. The client user and server establish a secret session key in the ending of the proposed scheme. They use the session key to exchange secret data confidentially. Furthermore, our scheme takes less computational and communicational costs. It is more efficiency than the previous schemes.

Acknowledgements

Research supported in part by the National Science Council grant NSC-93-2213-E-032-019 Taiwan, Republic of China.

Reference

[1] C. Chang and T. Wu, "Remote Password Authentication with Smart Cards", IEE Proceeding – Computers and Digital Techniques, Vol. 138, No. 3, 1991, pp. 165-168.

[2] C. Chang and S. Hwang, "Using Smart Cards to Authenticate Remote Passwords", Computers and Mathematics with Application, Vol. 26, No. 7, 1993, pp. 19-27.

[3] H. Chien, J. Jan and Y. Tseng, "An Efficient and Practical Solution to Remote Authentication: Smart Card", Computers and Security, Vol. 21, No. 4, 2002, pp. 372-375.

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

[5] W. Juang, C. Lei and C. Chang, "Anonymous Channel and Authentication in Wireless Communications", Computer Communications, Vol. 22, No. 15-16, 1999, pp. 1502-1511.

[6] W. Juang, "Efficient Password Authenticated Key Agreement Using Smart Cards", Computers and Security, in press, 2004.

[7] W. Juang, "Efficient multi-server password authenticated key agreement using smart cards", IEEE Transactions on Consumer Electronics, Vol. 50, No. 1, 2004, pp. 251-255.

[8] T. Kobayashi and H. Morita, "Fast Modular Inversion Algorithm to Match Any Operation Unit", IEICE Transactions Fundamentals, Vol. E82-A, No. 5, 1999.

[9] Y. Lai and C. Chang, "An Efficient Multi-exponentiation Scheme Based on Modified Booth's method", INT. J. ELECTRONICS, Vol. 90, No. 3, 2003, pp. 221-233.

[10] I. Lin, M. Hwang and L. Li, "A New Remote User Authentication Scheme for Multi-server Architecture", Future Generation Computer Systems, Vol. 19, 2003, pp. 13-22.

[11] L. Li, I. Lin and M. Hwang, "A Remote Password Authentication Scheme for Multi-server Architecture Using Neural Networks", IEEE Transactions on Neural Networks, Vol. 12, No. 6, 2001, pp. 1498-1504.

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

[13] K. Tan and H. Zhu, "Remote Password Authentication Scheme with Smart Cards", Computer Communications, Vol. 18, 1999, 390-393.

[14] S. Wang and T. Chang, "Smart Card Based Secure Password Authentication Scheme", Computers and Security, Vol. 15, No. 3, 1996, pp. 231-237.

[15] T. Wu, "The secure remote password protocol", Proceeding- Internet Society Network and Distributed System Security Symposium, 1998, pp. 97-111.

[16] W. Yang and S. Shieh, "Password Authentication Schemes with Smart Cards", Computers and Security, Vol. 18, No. 8, 1999, pp. 727-733.