ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

A Dexterous Asymmetric Two-Server Password Authentication Scheme using ECC with Provable Security

[1]Anitha Kumari, K., [2]Dr. Sudha Sadasivam, G., [3]Rajesh, S.

[1]Assistant Professor,[3]Pg Scholar, Department of IT,[2]Professor, Department of CSE,PSG College of Technology, Coimbatore – 641004.

ARTICLE INFO

Article history:
Received 20 January 2015
Accepted 02 April 2015
Published 20 May 2015

Keywords:
Authentication, two-server, key exchange.

ABSTRACT

An emerging technology that provides people a way to share large amount of hardware and software resources that belong to different organizations is cloud computing. The main problem of cloud computing is management of the public concerns such as the confidentiality and privacy issues. Authentication of entities is a fundamental concern in the cloud based system. Most systems lack a security model that guarantees an end-to-end security and confidentiality. Adopting a cloud computing paradigm may have positive as well as negative effects on authenticating a user and exchanging the data. Most password-based user authentication systems place total trust on a single authentication server where clear text passwords or easily derived password verification data are stored in a central database. Compromise of the authentication server by either outsiders or insiders subjects all user passwords to exposure and may have serious legal and financial repercussions to an organization.Multi server systems are difficult to deploy and operate in practice since either a user has to communicate simultaneously with multiple servers or the protocols are quite expensive. The adversaries get overcome by making use of two server authentication protocol. **Password authenticated key exchange** (PAKE) is a password-only system in the sense that it requires no public key cryptosystem and, thus, no PKI. This makes the system very attractive considering PKIs are proven notoriously expensive to deploy in the real world. Usually with the help of single trusted server users share a password for authentication purposes and also in PAKE with single server is vulnerable to various attacks such as offline dictionary attacks, server spoofing attack and stolen verification attacks, in which server is getting compromised. Therefore the proposed protocol makes use of two-server password based authentication and key exchange scheme for authentication building upon Elliptic Curve Cryptography (ECC) encryption scheme and Diffie-Hellman Key Exchange algorithm. By using a solution of the discrete logarithm in the Fp security of ECC encryption of finding a password has been strengthened. Thus the discrete logarithm problem facilitates the ECC cryptosystem to secure against the offline dictionary attack and other cryptographic attacks. Implementation has been carried out in a cloud environment using Eucalyptus and performance has been analyzed with other similar protocols.

ToCite ThisArticle: Anitha Kumari, K., Dr. Sudha Sadasivam, G., Rajesh, S., A Dexterous Asymmetric Two-Server Password Authentication Scheme using ECC with Provable Security. ***Aust. J. Basic & Appl. Sci.,*** *9(16): 1-7, 2015*

INTRODUCTION

Most password-based user authentication systems place total trust on a single authentication server where clear text passwords or easily derived password verification data are stored in a central database. Compromise of the authentication server by either outsiders or insiders subjects all user passwords to exposure and may have serious legal and financial repercussions to an organization.Multi server systems are difficult to deploy and operate in practice since either a user has to communicate simultaneously with multiple servers or the protocols are quite expensive. Authentication of entities is a fundamental concern in the distributed system. Distributed authentication systems can use different password techniques like: i) Simple text password ii) Graphical password and iii) 3D password object. But each of this has its own merits and drawbacks. Textual password authentication systems are easy to break and much vulnerable to brute force attacks. Graphical and 3D passwords require larger memory space. Thus, they are also constrained by time complexity. Similarly, another major problem is confidentiality, which is often treated as synonymous with data privacy and, as such, merely associated

Corresponidng Author: Anitha Kumari, K., Assistant Professor, Department of IT, Department of CSE, PSG College of Technology, Coimbatore – 641004
E-mail: anitha.psgsoft@gmail.com

with the existence of an encryption scheme for securing message exchanges. If the sender and receiver wish to exchange encrypted messages, then they must establish system parameters to encrypt messages to be sent and decrypt messages received. In case of symmetric ciphers both the sender and receiver will need the same secret key. In case of an asymmetric key cipher both sender and receiver will need each other's public key. The key exchange problem is to securely exchange keys, or other information securely, such that no one else other than the communicating parties obtains a copy. Whitfield Diffie and Martin Hellman were the first to establish such a method named as Diffie-Hellman Key Exchange (DH-Key Exchange). However, DH key exchange did not address the problem authenticity of the persons involved inthe key exchange process. To ensure authenticity, the communicating parties must hold some secret information. Otherwise, there is nothing that could prevent man-in-the-middle attack. Commonly used secret information is (i) high entropy secret key; and (ii) the case considered in this work: a low entropy password that is more user friendly. PAKE is a method to establish a secret key between two communicating parties based upon their knowledge of secret information like a password. Established secret keys can be used to secure exchange of messages such that an unauthorized party can obtain no information regarding the messages exchanged without the knowledge of the secret key. An important property of PAKE is that an intruder or man in the middle cannot brute force guess a password without further interactions with communicating parties. Thus, PAKE provides strong security using low entropy passwords.In a single server environment, the user's low entropy passwords will be stored as clear or encrypted text in a trusted single server. Compromise of the server leads to several attacks such as online dictionary attack, offline dictionary attack, spoofing attacks, etc.,to make the PAKE protocol resist against such vulnerable attacks, two-server based PAKE protocol is proposed. In the two-server model, user's low entropy password and other authentication information are distributed between two servers, such that the compromise of a single server does not provide any useful information about the user's password. Thus, two server model improves the security of user's low entropy password.

2. Related Work:

A practical authenticated key exchange protocol upon the two server model was proposed in 2006(Yanjiang Yang., Deng R.H and FengBao, 2006). Three entities are involved in this system,namely users, a service server (SS) that is a public server and a control server (CS) that is the backend server. The system should resist against offline dictionary attacks on the two servers is their primary goal, where passive and active adversaries control CS and SS respectively. It has been proved that SS is not effective in offline dictionary attack as an active attacker. Yang et al. proposed an improved model to overcome the drawbacks of basic model by introducing an extra parameter g3 for the purpose of user authentication. By removing the secret channel does not facilitate outside attackers to derive the session key used between U and SS, who have no control on any server and at the same time CS cannot compute the session key shared between U and SS.

In (Jun Ho Lee., and Dong Hoon Lee,2007) a two server authentication and key exchange protocol that supports multiple service servers SSj and a Single Control server CS is given. CS computes each service server SSj with its own secret key KSj = h(SSj, x). Their protocol is strongly against the stolen verification attack without the assumption of deploying a secure database in the service server. Using his/her identifier, and password the user U must initially register himself with CS. User U requests the particular SSj with the message < UID || SSj || Req> during the authentication phase. The password share πj is calculated by the service server SSj for the user with the identifier UID and also passes the request to the CS. SSj and U negotiate a secret session key K once the SSj and CS authenticate each other. While an adversary tries to masquerade as one of the service servers he/she will not triumph. By using his/her knowledge about the server obtained from prior communication with that server if one of the legitimate user tries to spoof a server, it's impossible to succeed without knowing the user password π of another user. If a legitimate server SSi tries to spoof another server SSj, SSi has no knowledge about the password share πj = h(UID || KSj) of SSj. Thus, this protocol has proven to be secured against sever spoofing attack. The protocol is secure against stolen verification attack since none of the service servers SSj stores any information related to the user's password. A novel two-server password authentication scheme with provable security (Dexin Yang., and Bo Yang., 2007) concentrates to protect the password information from the compromise of a server and the compromising server does not help an adversary to get authenticated by the other server. Their protocol is resistant against off-line dictionary attacks done by an active adversary.

3. Proposed Protocol:

Proposed protocol is built upon two cryptographic algorithms, namely Diffie-Hellman (DH) key exchange algorithm and ECC.In the proposed system authentication of client services is done by two servers S1 and S2 that run in parallel to authenticate the clients. The proposed system design consists of three phases, namely initialization, registration, authentication and key exchange. Initialization phase is where the publicparameters required for registration and authentication are established and published. Prior to authentication

each client C chooses a password PSWD and generates password authentication information auth1 and auth2 for S1 and S2 respectively and transmits it through different secure channels. After successful registration, the client remembers only the password for authentication. Finally, during the key exchange phase the client establishes different secret keys with the server S1 and S2. During authentication and key exchange phases the client and the two servers communicate via a public channel that could be eavesdropped, delayed, replayed or tampered by an attacker. The proposed protocol is asymmetric where the two servers S1 and S2 co-operate and contribute to authentication in terms of computation and communication.

3.1 Initialization Module:

In the initialization module, the peer servers S1 and S2 jointly publish the public parameters of the system. The two servers S1 and S2 choose a cyclic group of larger prime order q with a generator g1 and an elliptic curve equation $E_p(a, b)$ with order n where p is a base point with order n satisfying the condition $4a^3 + 27b^2 \bmod p \neq 0$. Server S1 randomly chooses an integer x1 from Zq* and calculates$G_1 \times x_1$, while server S2 randomly chooses an integer x_2 from Z_q^* and calculates$G_1 \times x_2$ where $G_1$is apoint on elliptic curve E. The two servers S1 and S2 also agree upon a hash function H and then S1 and S2 jointly publish the public parameters $\{\mathbb{G}, \boldsymbol{q}, g_2, \mathrm{Ep(a,b)}, G_1, G_2 \mathrm{Hash}\}$.Then the C arbitrarily chooses the decryption key xi from Zq* and calculate the encryption key $y_i = G_1 \times x_i, i = 1,2$ for server Si (i = 1, 2). The initialization process ensures that nobody will be able to know the value of P from g_2^P and value of r from scalar multiplication of rXG_1 as it is a discrete logarithmic problem as shown in figure 1.

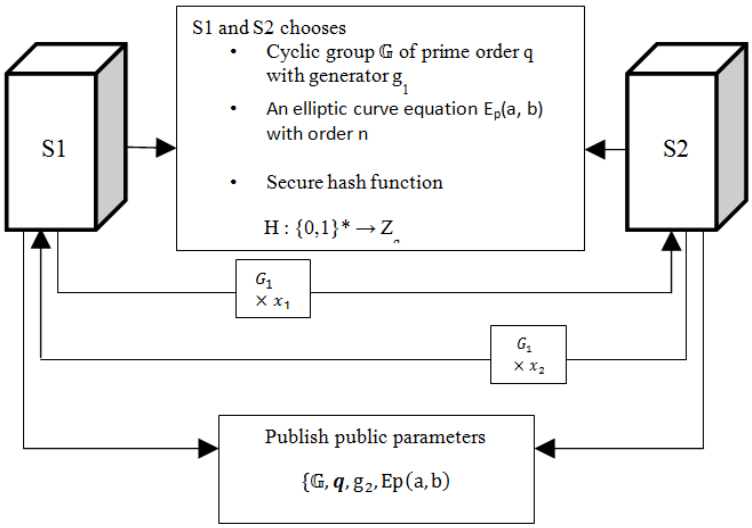

Fig. 1: Initialization module.

3.2 Registration Module:

On beforehand to authenticate it is necessary that every client C needs to get registers with the server S1 and S2 by choosing a password PSWD. The proposed model assumes that the two servers never collude and it is a well-known fact that the discrete logarithm problem is a hard. Then the client encrypts the password PSWD with y_i as given in equation 1, where a_i is randomly chosen from Zq* and PSWD mapped to points$M_1 and M_2$ on the elliptic curve E.

$$(A_2, B_2) = (G_1 \times a_2 , M_1 + y_2 \times a_2)$$
$$(A_1, B_1) = (G_1 \times a_1 , M_2 + y_1 \times a_1) \quad (1)$$

b_1 is randomly chosen integer from Zq* by client C and calculates b2 as given in the equation 2.

$$b_2 = b_1 \oplus H(PSWD) \quad (2)$$

At last the client C delivers the information of auth$^1\{A_2, B_2, x_1, a_1, b_1\}$ to the server S1 and authentication information auth$^2 = \{A_1, B_1, x_2, a_2, b_2\}$ to the server S2 through different secure channels. During registration client C only remembers the password PSWD alone for authentication and key exchange.

3.3 Authentication and Key Exchange Module:

Once the registration process is successful, mutual authentication between a client and two servers will take place and on successful authentication, generation of secret key is generated for further communication. Authentication and key exchange procedure consist of five steps in terms of parallel computation of servers S1 and S2 as shown in figure 2.

4.Security and Result Analysis:

The protocol is tested with all possible types of passwords, out of which a few are listed in table 1.

Table 1 shows the authentication results for a set of legal and illegal client.

Remark 1:

Using openstack, cloud environment has been set up. In cloud, two Virtual Machine (VM) instances for Server S1 and Server S2 are started. The client runs by using the IP 172.16.32.180 and uses the key 'psg1key0' to communicate with the servers. Server S1 VM uses the IP 172.16.32.181 and Server S2 VM uses the IP 172.16.32.182 with the key 'psg1key0' for communication among the servers and with the client. Figure 3 shows the variations in the session key generated for each and every session of different users. It is apparent that it is impracticable to guess or obtain the session key with the perception of previously established session keys.

Table 1: Authentication and key exchange results.

Username	Password	Authentication Outcome	Run Time	Authentication Time	Key Length (bits)
User35	amm45ie&	Valid	4.95569569E8	3.51073654E8	160
User108	delic0i7ie$	Valid	5.00543845E8	3.27314065E8	160
User35	amm45ieee	Invalid	-	-	-
User54	Password	Invalid	-	-	-
User56	Mr_user12_6784	Valid	2.11211023E8	3.11882428E8	160

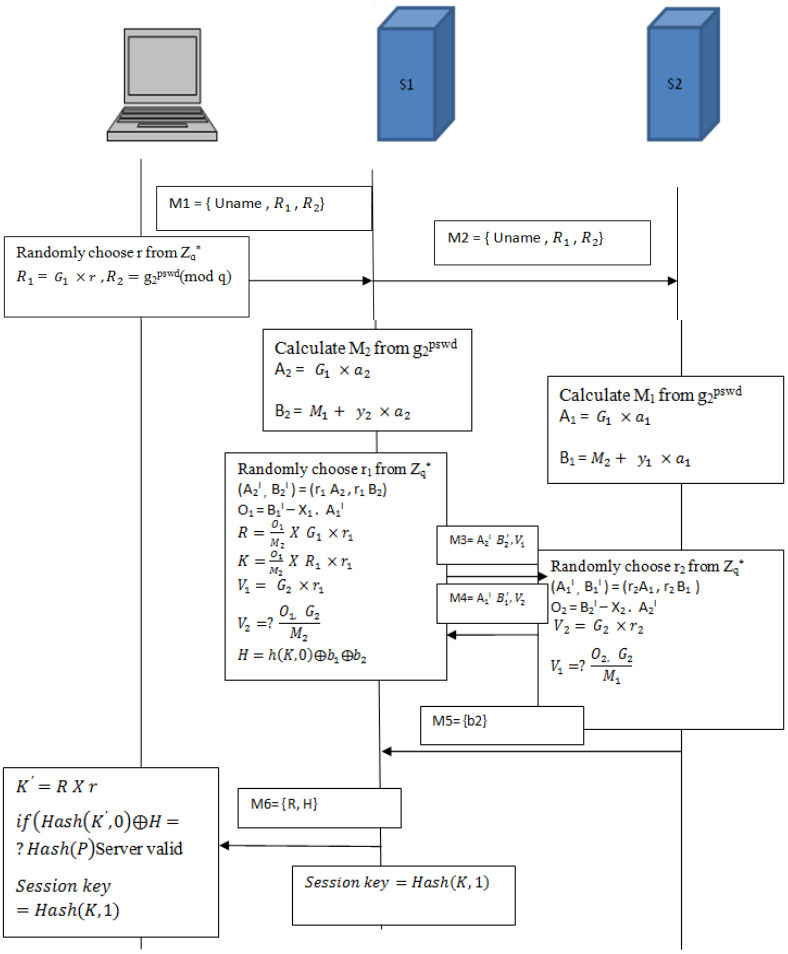

Fig.2: Authentication module.

4.1 Security Analysis:

Claim 1: Protocol Resistance against Online Dictionary Attack:

An adversary mounts an online dictionary attack by trying to login with all possible passwords. Such type of attack can be restricted by limiting the number of incorrect logins for a specific user account. Thus the protocol is resistant to online dictionary attack.

Claim 2: Proposed Protocol Affirms Known-Key Security:

Proof:

Known-Key security refers to an attempt made to obtain the session key SK with the knowledge of

another session key Sk1. Since the key-generation procedure of the proposed protocol involves the use of random numbers and generated for each and every session of users, it is infeasible to guess or obtain the session key with the knowledge of previously established session keys. Thus the proposed protocol satisfies Known-Key security property.

Claim 3: Proposed 3D PAKE Protocol Assures Key Control Property:

Proof:

Key Control property states that none of the communicating parties should force the other secret key generation process. Here the secret key is generated using the random numbers r, r_1, r_2 chosen by the client, server S1 and S2 respectively. Thus, all the communicating parties equally contribute to generate the session keys with none of the parties forcing the other. Thus the proposed protocol satisfies key control property.

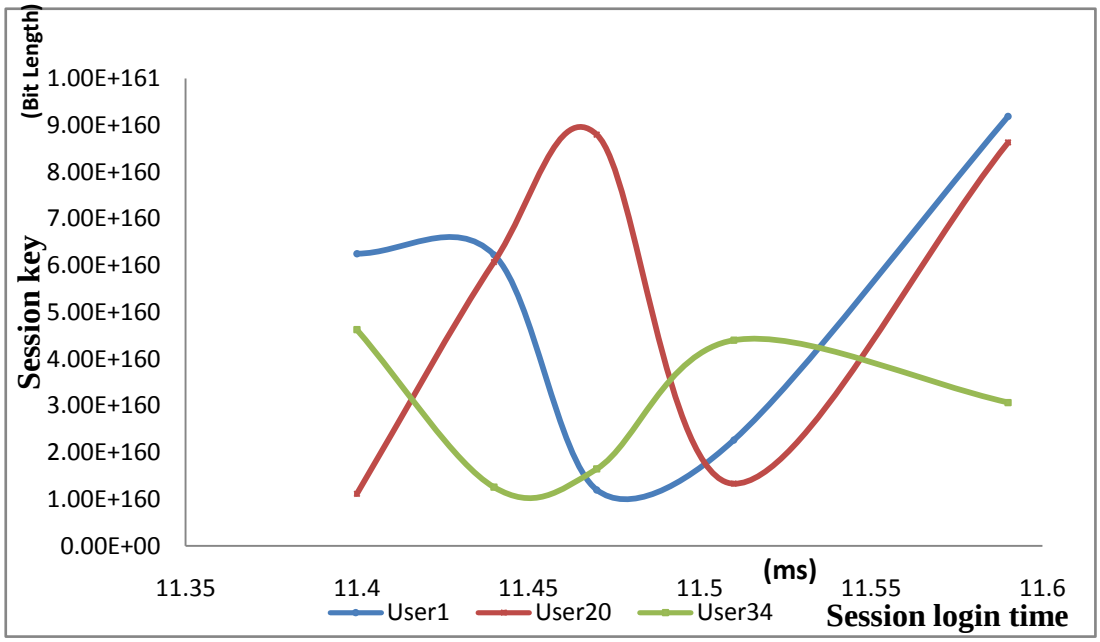

Fig. 3: Session key Vs session login time for multiple users.

5. Performance Analysis:

The performance of the protocol is examined in this section. As stated earlier, both the servers S1 and S2 equally contribute to authentication and key exchange and have same communication and computation complexity. Hence it is sufficient to analyze the performance of one server.

5.1 Communication Performance in terms of Bits:

Communication performance is measured in terms of *L* and *l*, where *L* is the bit size of an element from Zq* and *l* is the bit size of the hash value. The server S1 receives M1 (contains R1 and R2) from a client in the first round and the server S1 sends M2 (contains R1 and R2) to S2. S1 and S2 exchanges M3 (contains two elements A_2', B_2' V_1) and M4 (contains A_1', B_1',V_2). Then the server S1 receives M5 (contains one element $b_2 \in$ Zq*) from S2 and finally S1 sends M6 (contains one hash value H and R) to the client. Thus the communication complexity of S1 is given by $11L+1l$ and S2 is given by $6L$. As far as the communication complexity of the client is given by $3L+1l$.

6.2 Communication Performance in terms of Rounds:

With reference to figure 2, it is clear that the client is involved in 2 communication rounds and server S1 is involved in 6 communication rounds where server S2 is involved in 4 communication rounds.

6.3 Computation Performance:

Only the number of scalar multiplication is considered as computation performance for each party since each computation is dominated by scalar multiplication. The client has a computational complexity of 3 scalar multiplications and server S1 has a computational complexity of 7 scalar multiplications where server S2 has a computational complexity of 5 scalar multiplications with reference to figure 2. The performance comparison of the proposed ECC protocol has been compared with Yang et al.(Yanjiang Yang., and Deng R.H, FengBao., 2006), Jin et al (Haimin Jin., Duncan S., Wong and Yinlong Xu., 2007) and Yi et al.'s protocol (Xun Yi., San Ling and Huaxiomg, 2013) as given in table 2. It can be seen that the proposed

protocol is more efficient than Yang et al.'s, Jin et al.'s and Yi et al.'s protocol. One of the two servers is more efficient than service server (SS) of Yang et al.'s protocol in the proposed protocol. But another server of the proposed protocol is slightly less efficient than the control server (CS) of Yang et al.'s protocol. Since Yang et al.'s protocol is asymmetric, where only SS is known to be the public server and CS is hidden and the client establishes a secret key only with the SS at the end. The proposed protocol is also asymmetric, where server S1 is public and the client establishes a secret key with server S1.

Table 2: Performance comparison.

Participants	ECC protocol	Yi et al.'s protocol [11]	JWX protocol [5]	Yang et al.'s protocol [12]
Client C				
Communication (bits)	3L + 1l	3L + 4l	6L + 2l	4L + 2l
Communication (rounds)	2	3	3	6
Computation	3	4	6	5
Server S1 / SS				
Communication (bits)	11L + 1l	6L + 3l	11L + 3l	8L + 3l
Communication (rounds)	6	4	6	10
Computation	7	5	8	6
Server S2 / CS				
Communication (bits)	6L	6L + 3l	5L + 1l	4L + 1l
Communication (rounds)	4	4	3	4
Computation	5	5	4	3

Remark 2:

Since the password is sent to server as g_2^{pswd} obtaining PSWD from g_2^{pswd} is infeasible which is a discrete logarithm problem. Thus, it overcomes the weakness of Yang et al protocol. Also, when compared with the Yi et al protocol, proposed protocol's key length is less without violating the security measures. Thus the speed is more and bandwidth consumption is very low. Thus, it prevails over the limitation (i.e.,) storage complexity of the Yi et al protocol.

Conclusion:

An asymmetric protocol for two-server based, password-only authentication and key exchange is implemented and the results are analyzed. The protocol is built upon the ECC encryption scheme and Diffie- Hellman Key exchange algorithm. Security of the protocol lies in the strength of ECC encryption technique that uses cyclic group and scalar multiplication. Security analysis has shown that the protocol is secured against various cryptographic attacks such as server spoofing attack, stolen verification attack, dictionary attack etc. The protocol also satisfies forward secrecy. Performance analysis has shown that the protocol is more efficient than the existing asymmetric and symmetric two-server PAKE protocols. A great phenomenon of this protocol is fully utilized the virtues of ECC encryption technique. As a future work for a high speedreliable connection the request can be made to handle by multiple front end servers.

REFERENCES

Anamika Chouskey and Yogadhar Pandey, 2013.An Efficient Password Based Two-Server Authentication and Pre-shared Key Exchange System using Smart Cards.International Journal of Computer Science and Information Technologies, 4(1):117-120.

Bhavana, A., V.Alekhya, K.Deepak and V.Sreenivas, 2013.Password Authentication System (PAS) for Cloud Environment.International Journal of Advanced Computer Science and Information Technology, 2:29-33.

Dexin Yang and Bo Yang, 2010.A Novel Two-Server Password Authentication Scheme with Provable Security. IEEE 10th International Conference on Computer and Information Technology (CIT), pp: 1605-1609.

Dinesha, H.A., V.K.Agarwal, 2012.Multi-Dimensional Password Generation Technique for Accessing Cloud Services.International Journal on Cloud Computing: Services and Architecture,2(3):31.

Haimin Jin, S.Duncan, Wong and Yinlong Xu, 2007.An Efficient Password-Only Two-Server Authenticated Key Exchange System.In proceeding of: Information and Communications Security, 9th International Conference, ICICS 2007.

Her-TyanYeh and Hung-Min Sun, 2002.Simple Authenticated Key Agreement Protocol Resistant to Password Guessing Attack.ACM SIGOPS Operating Systems Review, 36(4):14-22.

Hung-Yu Chien and Tzong-Chen Wu, Ming-KueiYeh, 2013.Provably Secure Gateway-Oriented Password-Based Authenticated Key Exchange Protocol Resistant to Password Guessing Attacks.Journal Of Information Science And Engineering, 29(2):249-265.

Jun Ho Lee and Dong Hoon Lee, 2007.Secure and Efficient Password-Based Authenticated Key Exchange Protocol for Two-Server Architecture.International Conference on Convergence Information Technology, 21(23):2102-2107.

Katz, J.,P.Mackenzie, G.Taban, V.Gligor, 2009.Two-server password-only authenticated key exchange. Proc. ACNS'05, pp: 1-16.

Lishan Kang, Xuejie Zhang, 2010.Identity - Based Authentication in Grid Storage Sharing.2010 International Conference on Multimedia Information Networking and Security.

Xun Yi and San Ling, Huaxiomg, 2013. Efficient Two-Server Password Only Authenticated Key Exchange. IEEE Transactions on Parallel and Distributed Systems, 24(9):1773-1782.

Yanjiang Yang and R.H. Deng FengBao, 2006.A Practical Password-Based Two-Server Authentication and Key Exchange System.IEEE Transactions on Dependable and Secure Computing, 3(2):105-114.