A Symmetric Two-Server Password based Authentication and Key Exchange Protocol Deployed in PaaS

Anitha Kumari K
Assistant Professor,
Department of IT,
PSG College of Technology

Sudha Sadasivam G
Professor
Department of CSE,
PSG College of Technology

Rohini L
PG Scholar
Department of IT,
PSG College of Technology

ABSTRACT
In real time applications, more number of servers and data centers are needed for fast processing in the required time and to provide high level of security in communication due to rapid growth of data. Password Authenticated Key Exchange (PAKE) protocol is used to verify the authentication of the communicating parties and then secret key is generated based on their passwords. Mostly in single server environment the users share a password with a trusted single server. If the single server is compromised, then the environment is prone to many attacks such as online dictionary attacks, server spoofing attack and stolen verification attacks. The proposed system is built based on ElGamal encryption scheme and Diffie-Hellman Key Exchange algorithm in the two-server password based authentication and key exchange protocol. Discrete logarithm in f*p is used in ElGamal encryption to provide additional security. Discrete logarithm problem would render the ElGamal cryptosystem, secure against the man in the middle attack and other cryptographic attacks. The proposed scheme is provided with additional security and also its resistance against attacks.

Keywords
Password Authenticated Key Exchange, Two-Server, Diffie-Hellman Key Exchange.

1. INTRODUCTION
A basic concern in Distributed Systems Environment is security that is providing authentication of local and remote entities in the system. Distributed systems use different password techniques such as: i) Simple text password ii) Graphical password and iii) 3D password object. But each of this has its own advantages and drawbacks. The problem of textual password authentication system is that, it is easy to crack and defenseless to dictionary or brute force attacks. Graphical passwords demand memory space similar to that of the textual password. However, some of the graphical password schemes take a long duration. Thus time complexity is a barrier here. Similarly 3D- password authentication has its own limitations. Like data privacy another major problem is confidentiality and is merely associated with the occurrence of an encryption scheme for securing message exchanges. They must establish system parameters to encrypt messages to be sent and decrypt messages received if sender and receiver wish to exchange encrypted messages. Both will need a copy of the same key if the cipher is a symmetric key cipher. Both will require the other's public key if an asymmetric key cipher with the public/private key property. The key exchange problem is how to securely exchange keys or other information needed such that only the communicating parties obtain a copy. Password Authenticated Key Exchange (PAKE) is a method to establish a secret key between two communicating parties based upon their knowledge of confidential information like a password. Established secret key can be used for secure exchange of messages such that without the knowledge of the secret key no information regarding the messages exchanged can be obtained by an unauthorized party. A crucial property of PAKE is that an attacker or man in the middle cannot guess a password without further interactions with communicating parties. Thus PAKE furnishes strong security with the help of low entropy passwords. In a single server scenario, the user's low entropy passwords will be maintained as plain text or encrypted text in a trusted single server that leads to various attacks like online dictionary attack, offline dictionary attack, spoofing attack etc. Then the server will be compromised because of these attacks. The user credentials are shared between multiple servers in a multi-server model and authentication of a user relies upon the authentication results the entire server. Disadvantage of such a multi-server model is that the users have to communicate with all the servers in the system which results in communication complexity. A two-server model overcomes the drawback of multi-server model. It guarantees that the system is resistant to cryptographic attacks, such that the compromise of a single server does not reveal any useful information regarding the user's password. Thus, Two Server model improves the security of user's low entropy password.

2. RELATED WORK
In 2006 the basic two server model to safeguard a system against a single point of threat and a practical authenticated key exchange protocol upon the two server model got proposed in [3]. That system involved three entities namely users, a service server (SS) which is a public server and a control server (CS) a so called backend server. The primary goal is to make the system block offline dictionary intrusion activities on the two servers, in which CS and SS are controlled by passive and active adversaries respectively. This was made successful by strengthening the user's short password π into two long shares π_1 and π_2 in such a way that they are no more vulnerable subject to offline dictionary attack and distribute them to the two servers. As a result an attacker has to compromise both the servers in order to get hold of the user's password π.During authentication, when the user U provides his/her password π to the service server SS which is using its share π_1 and takes the help of the control server CS that provide its share π_2 for user authentication. Once the service server SS and the user U authenticate with each other, they negotiation happened with a secret session key to secure their further communications. The protocol is secure against offline dictionary attacks by CS as a passive adversary when it tries to bug into the communication channels, because CS will not be able learn anything on π_1. It is also proven that the protocol is more robust against offline dictionary attacks by SS as an active adversary as it is not possible for SS to manipulate the parameters and also have CS to authenticate U. Because of this, as an active attacker, SS is left

ineffective in offline dictionary attack.

To overcome these drawbacks of the basic model, Yang et al. came up with an improved model [3] by introducing an extra parameter g_3 for the usage of user authentication. On removal of the secret channel it does not enable outside attackers who have no control on any server to get the session key used between U and SS and at the same time CS will not be able to compute the session key which is shared between U and SS.

A two server authentication and key exchange protocol which do support multiple service servers SS_j and a Single Control server CS is given in [4]. Each of this service server SS_j has its own secret key $KS_j = h(SS_j, x)$ which is computed by CS. This protocol is robust against the stolen verification attack without the assumption of implementing a secure database in the service server. The user U should register himself initially with CS using his/her identifier and password. While in the authentication phase user U requests the particular SS_j with the message < UID || SS_j || Req>. The service server SSj calculates its password share πj for the user with the identifier UID and also passes on the request to the CS. Once the SS_j and CS authenticate with each other SS_j and U negotiate within themselves with a secret session key K. When an adversary attempts to masquerade as one of the service servers he/she will not be able to make it successful. If one of the legitimate users tries to spoof a server by using his/ her knowledge about the server which they got from prior communication with that server, it is impossible to succeed without knowing the user password π of any other user. In case a legitimate server SS_i tries to spoof another server SS_j, SS_i would have no knowledge about the password share π_j= h (UID || KS_j) of SS_j. Therefore this protocol is proved to be robust against several spoofing attacks. Since none of the service servers SS_j does store any data related to user's password, the protocol is more robust against stolen verification attack.

A Novel Two-Server Password Authentication Scheme [5] with Provable Security focuses on the way to protect the password data from the compromise of a server and the compromising server does not help an adversary to authenticate with the other server. This protocol is more robust against off-line dictionary attacks tried by an active adversary.

3. FUNDAMENTALS

The projected protocol is developed upon two cryptographic algorithms namely Diffie-Hellman (DH) Key Exchange and ElGamal encryption scheme.

3.1 Diffie Hellman Key Exchange Protocol

DH Key exchange is one of the bases of various authenticated key exchange protocols. DH key exchange can be used in a state where two parties' user1 and user2, who has no information about each, but wish to establish a secret key over a public channel.

3.1.1 Basic Steps

- User1 and User2 consent upon on cyclic group with a large prime order q with a generator g.
- User1 can arbitrarily chooses an integer a $\leftarrow Z_q^*$ and computes $X=g^a$, while User2 chooses an integer b from Z_q^* and computes $Y = g^b$.
- Then both user exchanges X and Y.
- User1 computes the secret key $K1 = Y^a = g^{ba}$.
- User2 computes the secret key $K2 = X^b = g^{ab}$.

3.2 ElGamal Encryption Scheme

ElGamal encryption scheme was developed on the basis of DH Key exchange. It consists of three phases namely, Key Generation, Encryption and Decryption.

3.2.1 Key Generation

A cyclic group of large prime order q is chosen with a generator g. Then a randomly chosen number x from Z_q^* is considered as decryption key and it is used to calculate the encryption key as $y = g^x$. , the public parameters of the encryption scheme are g and y.

3.2.2 Encryption

On input of a original plain text message m, it chooses an integer r at random from Z_q^* and outputs the cipher text C = E(m, y) = (A, B) = (g^r, $m.y^r$).

3.2.3 Decryption

On input of a cipher text (A, B) and the decryption key x, it outputs the plain text message $m = D(C, x) = B/A^x$.

3.3 Probabilistic Encryption Scheme

ElGamal encryption is one of the type probabilistic encryption schemes. If the same message is encrypted several times, it will yield different cipher texts. It is proved that ElGamal encryption is semantically secure under DDH assumption. ElGamal encryption scheme also possess useful homogenous properties as given below.

- Given an encryption of message m as (A, B), one can compute (A, α B), encryption of ám and one can also compute (A^α , B^α), an encryption of $m^á$.
- Given encryptions of m1 and m2 as (A1, B1) and (A2, B2) respectively, one can compute (A1A2, B1B2), an encryption of m1m2.

4. PROPOSED PROTOCOL

The proposed system has two servers S1 and S2 which run in parallel to authenticate the clients and to provide services to authenticated clients. There are three phases in the proposed system design which are initialization, registration, authentication and key exchange. The public parameters required for registration and authentication are established and published in the initialization phase. Prior to authentication each client C decides a password $PSWD_c$ and generates password authentication information $auth^1$ and $auth^2$ for S1 and S2 respectively and transmission occurs through different secure channels. The client remembers only the password for authentication after successful registration. The client establishes different secret keys with the server S1 and S2during the key exchange phase. The client and the two servers communicate via a public channel that could be eavesdropped, delayed, replayed or tampered by an attacker during authentication and key exchange phases. Since the two servers S1 and S2 co-operate and contribute equally to the authentication in terms of computation and communication the proposed protocol is symmetric.

4.1 Initialization Module

The peer servers S1 and S2 jointly publish the public parameters of the system in the initialization module. A cyclic group of larger prime order q with a generator g_1 is chosen by the two servers S1 and S2. An integer $s_1$is randomly chosen by server S1 from Z_q^* and calculates g_1^{S1}, while server S2 randomly chooses an integer s_2 from Z_q^* and calculates g_1^{S2}. The values g_1^{S1} and $g_1^{S2}$are exchanged by S1 and S2and then calculate g_1^{S1S2}. A hash function H is agreed upon by the two servers S1 and S2. As depicted in figure 1, S1 and S2 jointly publish the public parameters q,g1, g2 and H. Initialization process ensures that until the two servers S1 and S2 collude nobody will be able to know the discrete logarithm of g2 to the base g1. 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.

4.2 Registration Module

It is necessary that every client C registers with the server S1 and S2 prior to authentication. A password PSWD $_c$ is chosen by the client C. Then the decryption key x_i is randomly chosen by C from $Z_q^*$and calculates the encryption key $y_i = g_1^{xi}$ for server Si (i = 1, 2). Then client encrypts the password $PSWD_c$ with y_i as given in equation 1, where a_i is randomly chosen from Z_q^*.

$$(A_i, B_i) = g^{ai}{}_1(\text{mod } q),\ g_2^{PSWDc}.\ y_i^{ai}\ (\text{mod } q) \qquad (1)$$

Then an integer $b_1$is randomly chosen by the client from Z_q^* and calculates b_2 as given in equation 2.

$$b_2 = b_1 \oplus H(PSWD_c) \qquad (2)$$

Finally the client C delivers the authentication information $auth^1 = \{x_1, a_1, b_1, A_2, B_2\}$ to the server S1 and authentication information $auth^2 = \{x_2, a_2, b_2, A_1, B_1\}$ to the server S2 through two different secure channels. The client then remembers the password $PSWD_c$ alone for authentication and key exchange. In figure 2 registration process is depicted.

4.3 Authentication and Key Exchange Module

A client and two servers can mutually authenticate each other and then the client and two servers can generate the secret keys on successful registration. In figure 3, authentication and key exchange procedure consists of five steps in terms of parallel computation of servers S1 and S2.

5. RESULT ANALYSIS

All possible types of passwords are tested using the protocol, out of which a few is listed in table 1.Table 1 provides the authentication and key exchange results for a set of legal and illegal clients.

5.1 SECURITY ANALYSIS

Do not include headers, footers or page numbers in your submission. These will be added when the publications are assembled.

5.1.1 Security against Stolen-verified Attack

The database of the servers S1 and S2 can be the target of an adversary. Obtained database information is of no use if the adversary tries to manage to get the database of one server Si. This is because the server Si stores the password of the client that is ElGamal encrypted with Sj's encryption key y_j along with the Si's decryption key x_i. Unless the adversary has obtained the database of the both the servers S1 and S2 the server will not be able to decrypt the ElGamal encrypted password. The adversary will be provided with the exponential of the password even if he/she manages to obtain both the databases and decrypts the password. It is known fact that the discrete logarithm problem is a NP Hard problem. Even if both the servers' database is stolen, the adversary will not be able to obtain the password $PSWD_c$ of the client.

5.1.2 Security against Man in the middle Attack

If an adversary is trying to obtain the communication information, and with that information can make independent connections with the victims and relays messages between them to make them believe that they are talking directly to each other over a private connection, in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all relevant messages passing between the two victims and can inject new ones. But in this protocol all the password details which passed from client to server side are converted with help of discrete logarithm problem. So the adversary obtained information cannot be provided with the exponential of the password even if he/she manages to obtain both the databases and decrypted password they obtain only it's the mathematical value. Even if both the servers' communication is stolen, the adversary will not be able to obtain the password $PSWD_c$ of the client.

5.1.3 Security against Server Spoofing Attack

To obtain the secret key established between the client and the server, if an adversary a tries to masquerade as one of the servers or both, he/she will not make it successful. With reference to figure 3, when the client tries to verify the authenticity of the server in step 6, the authentication fails because the adversary A does not have any knowledge about the A_i, B_i. And thus a secret will not be established between the client and the malicious server. A legitimate server will not be successful as the ElGamal decryption key x_i of Si is not known to Sj even if it masquerades as another server. Thus the protocol is secured against all types of server spoofing attack.

6. PERFORMANCE ANALYSIS

In this section we examine the performance of the protocol. As stated earlier, both the servers S1 and S2 do equally contribute to authentication and key exchange and have got same communication and computation complexity. Therefore it is sufficient to analyze the performance of any one server.

6.1 Communication Performance in terms of Bits

Communication performance would be measured in terms of *L* and *l*, where *L* is the bit size of an element from Z_q* and *l* is the bit size of the hash value. The server S1 which receives M1 (contains one element R Z_q*) from client in the first round and the server exchanges M2 (contains two elements A2' and B2' Z_q *) and M3 (contains two elements A1' and B1' Z_q *) with S2 in the second round. Then the server S1 delivers M4 (contains one element R1 Z_q* and one hash value h1) to client in third round and finally receives M6 (contains two hash values h1' and h2') from the client. Hence the communication complexity of S1 is given by *6L+3l*.

As far as the client is concerned, it broadcasts M1 (which contains one element R Z_q*) in the first round and receives M4 (containing one element R1 Z_q* and one hash value h1) and M5 (contains one element R1 Z_q* and one hash value h1) in the third round. Finally the client does broadcast M6 (which contains two hash values h1' and h2'). Henceforth the communication complexity of the client is given by *3L+4l*.

6.2 Communication Performance in terms of Rounds

While talking about parallel computation, one communication round is supposed to be a two way transmission of messages. With reference to figure 3, it is made clear that the client is involved in 3 communication rounds and each of the servers are involved in 4 communication rounds. Totally the protocol authenticates and does exchange secret key within 4 communication rounds.

6.3 Computation Performance

Since each one of the computations is dominated by modular exponentiation, only the number of modular exponentiations is considered as computation performance for each party. With reference to figure 3, the client does have a computation complexity of 3 modular exponentiations and each one of the servers have a computation complexity of 4 modular exponentiations. The performance comparison of the protocol with Yang et al.'s protocol [3] is given in table 2. This can be seen as the proposed protocol is more efficient than Yang et al.'s protocol. In this proposed protocol, one of the two servers is better than service server (SS) of Yang et al.'s protocol. But another server of the proposed protocol is little 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 is establishing only a secret key with the SS in the end. But the proposed protocol is symmetric, where the two servers S1 and S2 are public and the client establishes a secret key with each one of the servers respectively. In addition Yang et al.'s protocol does run in series, and the proposed protocol runs in parallel. Therefore the total running time of the proposed symmetric protocol is actually equal to the running time of one server. While in the asymmetric Yang et al.'s protocol the total running time is equal to the sum of the running time of the two servers, One of the drawbacks of the proposed protocol is, that the storage space needed is approximately 5L for each one of the registered client, which is really greater than that of the storage space which is required for other two server protocols like Yang et al.'s protocol.

7. DEPLOYMENT

Deployment is done through the cloud computing services there are three types' services namely Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS) and this protocol has been deployed in Jelastic Cloud. Jelastic is a Platform as Infrastructure (PAI) cloud computing service that provides networks, servers, and storage solutions to software development clients, enterprise businesses etc.

8. CONCLUSION

A symmetric protocol for two-server based password-only authentication and key exchange is implemented and the results are analyzed. The proposed protocol has been developed with the ElGamal encryption scheme and Diffie-Hellman Key exchange algorithm. Security of the protocol lies in the strength of the cyclic group, modular arithmetic and Elgamal encryption. Security analysis results,that the protocol is secured against various cryptographic attacks such as server spoofing attack, stolen verification attacks, etc. Performance analysis has been shown that the protocol is better than the existing protocols and also has been developed in PaaS. As a future work, this symmetric protocol can be converted to an asymmetric protocol such that it has the security advantages of both asymmetric protocols and ElGamal encryption technique.

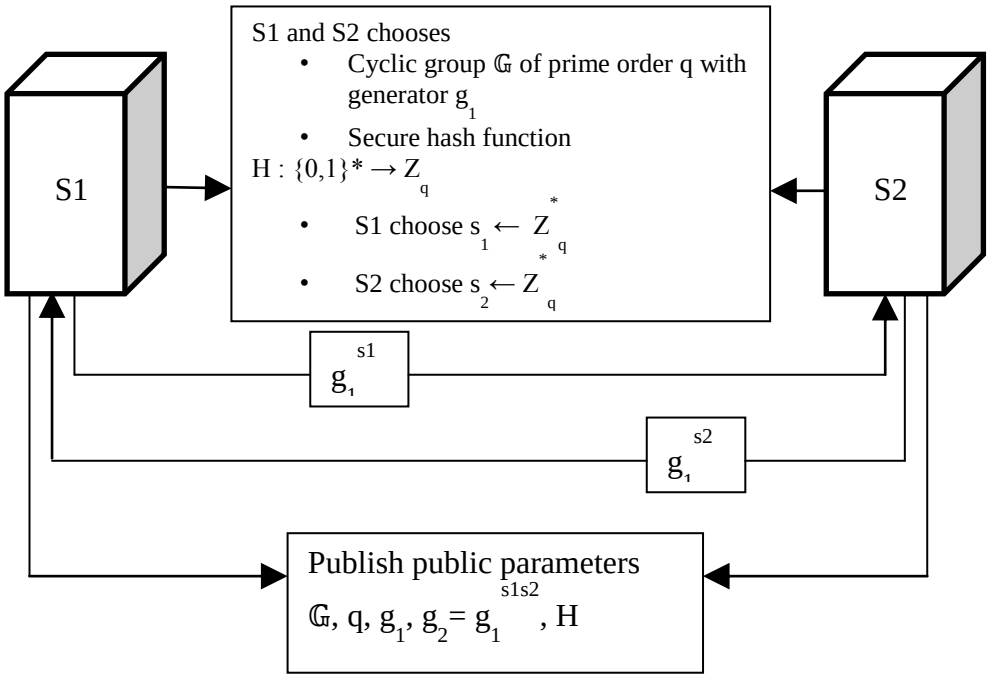

Fig 1: Initialization module

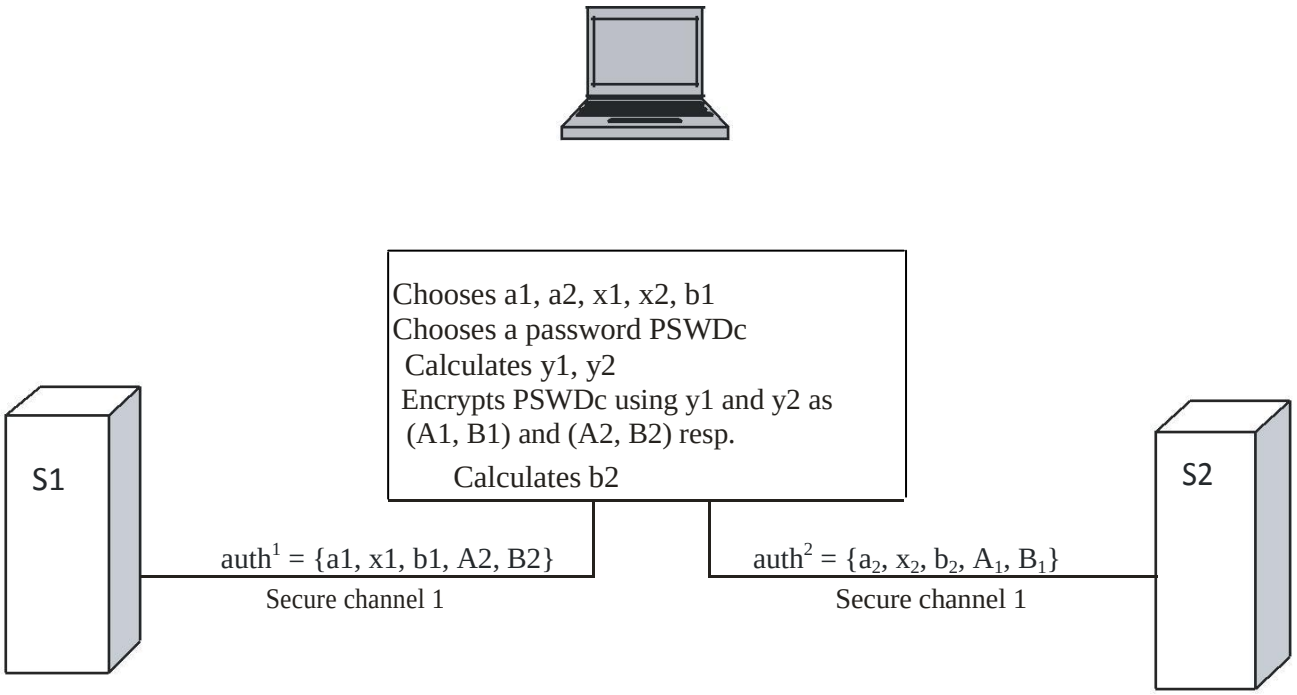

Fig 2: Registration module

Table 1. Performance comparison with Yang et al.'s protocol

Participants	Proposed protocol	Yang et al.'s protocol
Client C		
Communication (bits)	3L + 4l	4L + 2l
Communication (rounds)	3	6
Computation	3	5
Server S1 / SS		
Communication (bits)	6L + 3l	8L + 3l
Communication (rounds)	4	10
Computation	4	6
Server S2 / CS		
Communication (bits)	6L + 3l	4L + 1l
Communication (rounds)	4	4
Computation	4	3

Fig 1: Output of deployment

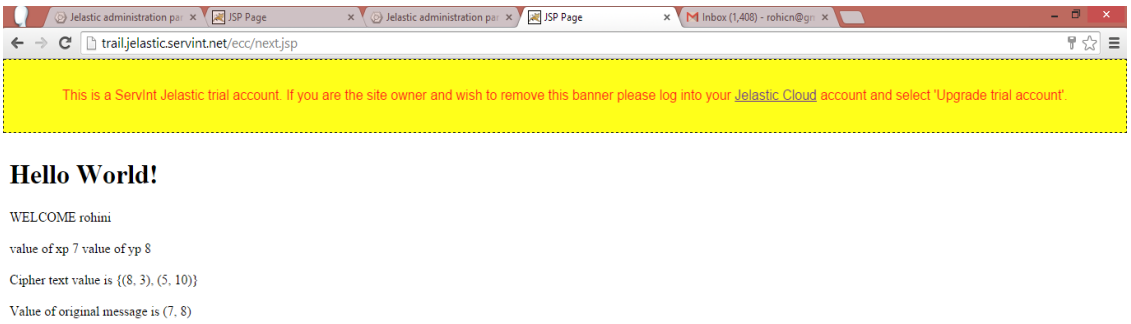

Fig 2: Final Result

9. REFERENCES

[1] Xun Yi., and San Ling, Huaxiomg, "Efficient Two-Server Password Only Authenticated Key Exchange", IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 9, pp. 1773- 1782, 2013.

[2] Hung-Yu Chien., and Tzong-Chen Wu, Ming- KueiYeh, "Provably Secure Gateway-Oriented Password-Based Authenticated Key Exchange Protocol Resistant to Password Guessing Attacks",Journal Of Information Science And Engineering, Vol. 29, No. 2, pp. 249-265, 2013.

[3] Yanjiang Yang., and Deng R.H, FengBao, "A Practical Password-Based Two-Server Authentication and Key Exchange System", IEEE Transactions on Dependable and Secure Computing, Vol. 3, No. 2, pp. 105-114, 2006.

[4] Jun Ho Lee., and Dong Hoon Lee, "Secure and Efficient Password-Based Authenticated Key Exchange Protocol for Two-Server Architecture", International Conference on Convergence Information Technology, 2007, Vol. 21, No. 23, pp. 2102-2107, 2007.

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

[6] Her-TyanYeh., and Hung-Min Sun, "Simple Authenticated Key Agreement Protocol Resistant to Password Guessing Attack", ACM SIGOPS Operating Systems Review, Vol. 36, No. 4, pp. 14-22, 2002.

[7] Anamika Chouskey., and YogadharPandey, "An Efficient Password Based Two-Server Authentication and Pre-shared Key Exchange System using Smart

Cards", International Journal of Computer Science and Information Technologies,Vol. 4, No. 1, pp. 117-120, 2013.

[8] Katz J., and MacKenzie P, Taban G, Gligor V, "Two-server password-only authenticated key exchange", Proc. ACNS'05, pp. 1-16, 2009.

[9] Lishan Kang, Xuejie Zhang(2010), "Identity - Based Authentication in Grid Storage Sharing", 2010 International Conference on Multimedia Information Networking and Security.

[10] Dinesha H A, Agrawal V K, "Multi-Dimensional Password Generation Technique for Accessing Cloud Services", International Journal on Cloud Computing: Services and Architecture, 2012, Vol.2, No.3. pp.31.

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