ISSN 2047-3338

OTPK PAKE Protocol with TRNG Based Key Generation

Rajneesh Pachouri[1], Ramratan Ahirwal[2] and Dr. Yogendra Kumar Jain[3]
[1,2,3]Department of Computer Science and Engineering, Samrat Ashok Technological Institute, Vidisha (M.P.)
[1]rajneeshrocks92@gmail.com, [2]ram2004_ahirwal2004@rediffmail.com,[3]ykjain_p@yahoo.co.in

***Abstract*– In earlier, two smart card based password authentication key exchange protocols were proposed by lee et al. and Hwang et al. respectively. But neither of them achieves two factor authentication fully since they would become complete insecure once one factor is broken. To overcome these two factor authentication problem in password authentication key exchange protocol (PAKE) proposed a new efficient PAKE protocol with the concept of one time private key (OTPK) concept, which achieves fully two factor authentications and provide forward security of session keys. And to generate more strong session keys using true random number generation method for key generation.**

***Index Terms*– OTPK, TTP, 2-Factor, Authentication, Key-Exchange and PAKE**

I. INTRODUCTION

WHENEVER numerous users need to communicate with another party in an open network, these parties should ensure about the authentication method to avoid any discrepancy or network fraud. Authentication is a process whereby a verifier is assured of the identity of a node involved in a protocol, and that den has actually participated. For example, in an email system and ATM card password is used as an authentication entity. Password authentication is the primary mechanism for verifying the identity of computer users. In the existing traditional setup the ID and PW are maintained by the remote system in a verification table. On the behalf of these table entries authentication is provided. Similarly for credit card and ATM card uses pin number as an authentication key.

A remote password authentication scheme is used to validate the legitimacy of the remote user over an insecure channel. In such types of schemes, the password is often regarded as a secret shared between the authentication server and the user. Serves are used to authenticate the identity of the individual login. Through knowledge of the password, the distant user can construct a suitable login message to the authentication server [1]. Authentication protocols make available two entities to make sure that the counterparty is the intended one whom he attempts to communicate with over an anxious network. These protocols can be considered from three dimensions: type, efficiency and security. Security in computers is information protection from unconstitutional or unintentional exposé while the information is in the transmission and while information is in storage.

In general, there are two types of authentication protocols, the password-based and the public-key based. Password based scheme is quite simple and mostly used in emails and other related login systems. In this scheme the password along with account name is registered in to the remote server. And every time when user performs login it should be verified by these remote servers. The server usually maintains a password or verification table but this will make the system easily subjected to a stolen-verifier attack. In public key based authentication, the user should register it with a third party known as a key generation centre or KGC. This key generation centre (KGC) provides a pair of public and private key to the user. Then, they can be recognized by a network entity through his public key. To simplify the key management, an identity-based public-key cryptosystem is usually adopted, in which KGC issues users ID as public key and computes corresponding private key for a user.

The PAKE protocols consent a client and a server to authenticate each other and make a stronger common session key through a pre-shared human memorable password over an insecure channel. Two-party password-based authenticated key exchange (two-PAKE) protocol is quite valuable for client-server architectures. However, in large-scale client-client communication environments where a user wants to communicate with numerous other users, Two-PAKE protocol is very problematic in key management that the number of passwords that the user would need to remember. Password-based authenticated key exchange (PAKE) protocols facilitate two users to produce a common, cryptography dependent-strong key based on an initial, low-entropy, shared secret (i.e., a password). The complexity in this setting is to prevent off-line dictionary attacks where a rival exhaustively enumerates impending passwords on its own, trying to match the truthful password to observed protocol implementations. Approximately, a PAKE protocol is protected if off-line attacks are of no use and the best attack is an on-line dictionary attack where an adversary must actively try to impersonate an honest party using each probable password. On-line attacks of this variety are inbuilt in the model of password-based authentication; more importantly, they can be detected by the server as failed login attempts and defended against. PAKE protocols are also important in practice, since passwords are perhaps the most common and widely-used means of authentication [2].

One Time Private Key is a concept of authenticating two parties via issuing a common private key at once for authenticating both parties. Whenever two parties want to securely transfer data they should have to use secure authentication protocol, so that the data can be protected from various attackers. For such strong data communication with strong authentication One Time Private Key (OTPK) concept is used. This OTPK is sent to both sender and receiver for encryption and decryption in one time only after that this key will be destroyed. This key should be either generated by the server or trusted third party. This key or code is sent to both sender and receiver. When a sender makes a request for communication to receive this key is also sent. When this request is received, the receiver compare key that comes from a sender and the server or trusted third party. If this key is matched both sender and receiver are connected and authentication using OTPK is done. After the verification is completed the key was destroyed.

II. BACKGROUND

Most password-based user authentication systems place total trust on the authentication server where passwords or easily derived password verification data are stored in a central database. These systems could be easily compromised by offline dictionary attacks initiated at the server side. Conciliation of the authentication server by either outsiders or insiders subjects all user passwords to exposure and may have serious problems. To overcome these problems in the single server system many of the systems have been proposed such as multi server systems, public key cryptography and password systems, threshold password authentication systems, two server password authentication systems. PAKE protocols facilitate two users agree on a common cryptographically-strong key that is based on an initial, low-entropy for the shared secret. PAKE is best suited for offline dictionary attack. 2-PAKE protocols are quite suitable for the client-server architecture.

III. RELATED WORK

In 2009 Jonathan Katz, Rafail Ostrovsky, Moti Yung introduced a technique is a 3-round protocol used for password-only authenticated key exchange. Technique is based on the decisional Diffie-Hellman assumption. It gives good security. It takes only public parameters- “common reference string” It doesn’t need that any party share pre-shares a public key. It is efficient, takes computation only (roughly) 4 times greater than “classical” Diffie-Hellman key exchange protocol [4]. The protocol assumes only public parameters i.e., a frequent reference string that can be hard-coded into an implementation of the protocol; in particular, and in contrast to some earlier work, this protocol does not require either party to generate and share a public key in addition to distribution a password. The protocol is also remarkably proficient, requiring computation only (roughly) 4 times greater than classical Diffie-Hellman key exchange which provides no authentication at all. This is the first protocol for password-only authentication which is both practical and provably-secure using standard cryptographic assumptions [4].

In 2006 Maryam Saeed, Ali Mackvandi, Mansour Naddafiun, Hamid reza Karimnejad introduced an enhanced PAKE protocol due to which the limitations of DH-BPAKE protocol. The author used mutual authentication Protocol with PAKE to achieve mutual verification, forward secrecy, known session key protection, and resilience to Denning-Sacco key, cooperation impersonation, Unknown Key Share, imperceptible online dictionary, off-line dictionary, ephemeral key compromise impersonation and replay attacks. The technique does not need modular multiplication, modular addition and modular inverse. Our proposed scheme provides several securities gives good computational efficiency and takes less no. of rounds [5].

In 2011 Wei-Kuo Chiang and Jian-Hao Chen invented a new protocol named TWKEAP (Three- Way Key Exchange and Agreement Protocol) to shear secret key between two communications so that they can protect their important communications. It gives mutual authentication, replay attack security and ideal forward secrecy. It takes shortest total service time and shortest queuing delay than other techniques [6].

In 2005 Shai Halevi, Hugo Krawczyk showed a review on various password authentication protocols with security on the basis of standard cryptographic assumptions. They showed that optimal resistance to off-line password for attacks of public key encryption functions. They also showed that public key techniques are unneglectable for password protocols that resist off-line guessing attacks [7].

In 2012 Jan Camenisch, Anna Lysyanskaya, Gregory Neven new introduced about the two-server case (2PASS) is universally compostable (UC) security definition for public-key setting. It gives security guarantees, efficient instantiation under DDH assumption in the random-oracle model and needs lesser than twenty elliptic-curve exponentiations on the user’s device [8]. In 2012 Kyung-kug Kim, Myung-Hwan Kim presented proposed a new technique which is an improved anonymous authentication and key exchange is secure against various well-known attacks [9].

In 2011 Dripto Chatterjee, Joyshree Nath, Suvadeep Dasgupta, Asoke Nath presented a symmetric key cryptographic technique for encryption and decryption of any file like binary file, text file. This technique uses the size of the key matrix of size 65536; each cell stores 2 characters pattern. This technique used in any communication network, business accommodation, government sectors, defense network system, and sensor networks etc [10].

In 2003 Jung Min Park, Edwin K. P. Chong, Howard Jay Sieg introduced a fair-exchange protocol by distributing the computation of RSA signatures. Using the inherent features of multi signature model, they construct protocols that necessitate no zero-knowledge proofs in the substitute protocol. Use of zero-knowledge proofs is needed only in the protocol setup phase--this is a one-time cost. Additionally, this scheme uses multi signatures that are well-matched with the basic standard (single-signer) signature scheme that makes it possible to readily integrate the fair-exchange feature with existing e-commerce systems [11].

IV. PROPOSED WORK

There are two important issues for any security protocol first is security in the form of authentication and second is speed for encryption and decryption. To achieve fully two factor authentications in PAKE protocol is great concern for the researchers; our protocol is working on fully two factor authentications for the session key exchanges. Symmetric key cryptography is 1000 times faster as compare to the asymmetric key cryptography to the encryption and decryption of a file. Proposed protocol works on symmetric key cryptography. As shown in the Fig. 1 is the architecture proposed for the 2-factor authentication protocol using OTPK, here in this technique the party p_1 generate a request to the TTP, TTP accept the request and response to the party p_1 similarly TTP accept the request from the party p_2 and send a response. After that TTP generates a unique number by the method of pseudorandom number generation and send to the both the parties via other media (mobile or email), parties enter that unique number and send to the TTP for further verification, if that unique number is verified by the TTP then at this stage 2-factor authentication is done. Authentication or the digital signatures generated is one time and as soon as the transmission is successful the key is destroyed.

The proposed protocol here works in two Stages.

Stage 1: Registration

Here in this stage the user is issued an OTP based token and if required the verification of these OTP tokens also takes place. This OTP token allows the user to authenticate to CA (certified Authority).

Stage 2: Signing

Here in this stage each of the users needs to generate digital signatures for the authentication. First of all generate a key pair of public/private key and for the authentication of the user, each user needs to provide the OTP token to the CA so that it will verifies the authenticity of the user and as soon as authentication gets success the key pair gets destroys.

As shown in the Fig. 1 is the architecture of the proposed work for 2-factor authentication. Here we are implementing the concept of 2-factor authentication protocol using OTPK. First of all for the establishment and generation of signatures both the parties request TTP for the generation of digital signatures. As soon as the signatures are generated both the parties exchange their signatures using TTP and if the signatures match the contract gets exchanges between two parties. For the fairness between the two parties the TTP is used. Full working of proposed protocol is discussed below and key generation by TRNG (true random number generation) method is discuss in section A.

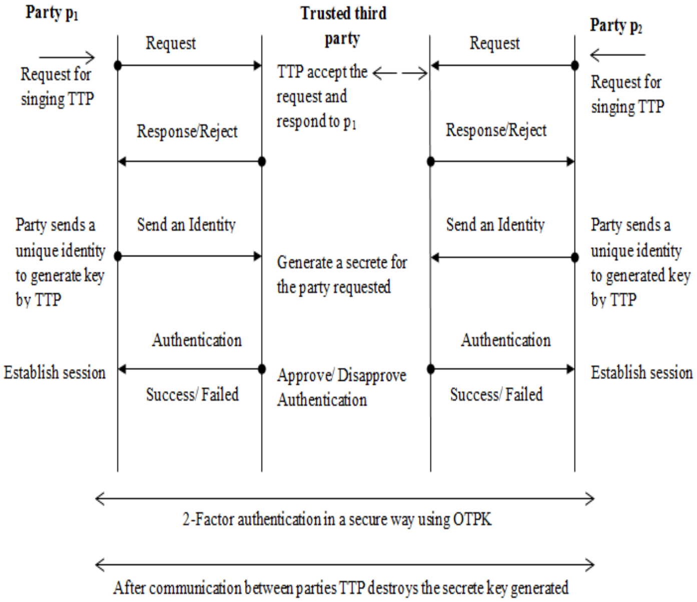

Party p1
Trusted third party
Party p2
Request
Request
Request for singing TTP
TTP accept the request and respond to p1
Request for singing TTP
Response/Reject
Response/Reject
Party sends a unique identity to generate key by TTP
Send an Identity
Generate a secrete for the party requested
Send an Identity
Party sends a unique identity to generated key by TTP
Authentication
Authentication
Establish session
Success/ Failed
Approve/ Disapprove Authentication
Success/ Failed
Establish session
2-Factor authentication in a secure way using OTPK
After communication between parties TTP destroys the secrete key generated

Fig. 1: Architecture working of 2-factor authentication using OTPK

A. *TRNG (true random number generation) based key generation*

Here give the process of generating key by using the image. Fig. 2 shows the example of image and Fig. 3 shows the corresponding binary value of the image. The corresponding M_{key} is the hash value of the image generated key. Steps of generating key by using image are discussed below:

1. Scan pixel values of the image from top to bottom and left to right.
2. Concatenating the value to generate random number consisting of 0's & 1's.
3. Random value can be generated by concatenating columns only or rows only or rows and columns. Here we give an example of concatenating row only.
4. Similarly unique value can be generated for two parties from the same image for authentication.

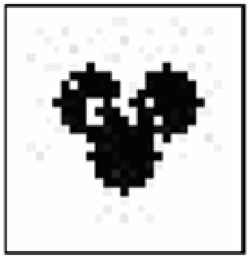
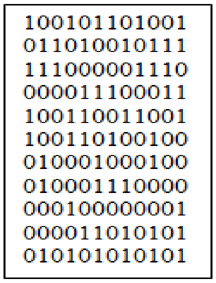


Fig. 2: Image (12×12)



100101101001
011010010111
111000001110
000011100011
100110011001
100110100100
010001000100
010001110000
000100000001
000011010101
010101010101

Fig. 3: Corresponding binary value of image

Generated key is:
X=100101101001011010010111111000001110000011100011
10011001100110011010010001000100010001000111000000
0100000001000011010101010101010101
Corresponding M_{key} is:
H(X) =502a8d2867eaa27ee99b3635c2144909

As we know that if the key length is long then the key is more secure as compare to the short key, so we have been generating a session key by using the true random generation method. Server or trusted third party randomly select the image and generated random key by this method. For example if image size is 12× 12 pixels then key size is 144 bits, which is very strong as compare to the pseudo random number. Trusted third party generated new session key in every session, the previous session key will automatically destroy.

B. Proposed protocol

Table 0: Notations used in protocol

Pw_1 & Pw_2	Password of party p_1 & p_2
H	One way hash function
r_n	Random number generated by pseudorandom generator
T_i	Random number generated by true random number generator
M_{key}	Master key

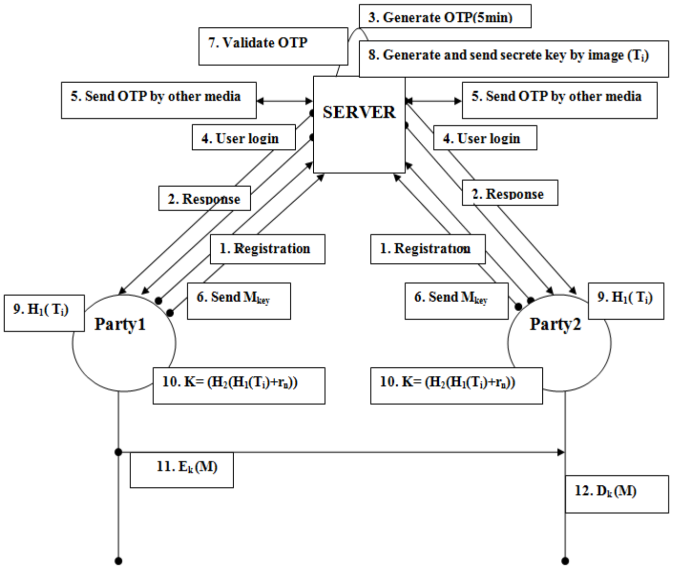


Fig. 4: Working module of proposed protocol

Steps:

1) Users register himself to the trusted third party or server. Users required username, password, mobile number, email address etc. at the time of registration.
2) Trusted third party or server get the registration form from the users and store in to the database, and send response to the users.
3) Trusted third party generates a random value r_n by the use of pseudorandom number generator that number is called one time password (OTP).
4) Users login with username and password (pw) to the trusted third party.
5) TTP verify that password if it is valid then send OTP (r_n) to the both parties via other media (email or mobile).
6) Users enter that OTP and make master key (concatenate password and hash value of r_n) and send to the TTP.
   Party $p_1 \rightarrow M_{key} = H_1(r_n + pw_1)$
   Party $p_2 \rightarrow M_{key} = H_1(r_n + pw_2)$
7) TTP verify that master keys to the own master key and if it is valid then 2-factor authentication is done at this place.
8) After this strong 2-factor authentication for generation of session keys, TTP generate a another random number T_i but this time by using true random generator (discussed in section A) and send to the both parties after calculated the hash value of this key.
9) Parties receives key (gendered by the method of true random number generation) $H_2(T_i)$ by the server or TTP.
10) Parties calculated own session keys by the concatenating image based key $H(T_i)$ and random value r_n:
    Party $p_1 \rightarrow K_1= (H_3(H_2 (T_i) + r_n))$
    Party $p_2 \rightarrow K_2= (H_3(H_2 (T_i) + r_n))$
    Where $K_1 = K_2$.
11) Party p_1 encrypted the message using own session key (K_1) and send to the party p_2.
12) Party p_2 decrypted the message using own session key (K_2), if party p_2 successfully decrypted the message then he is understood that party p_1 authenticated by the trusted third party.

V. RESULT ANALYSIS

Here in this section we have compared our protocol in three parameter basic feature and efficiency, security and performance evaluation with the referenced protocols.

A. Basic features and efficiency

In the category of basic feature and efficiency, the properties such as transparent TTP or not, offline or online TTP, TTP involvement, fairness, timeleness, additional authentication and storage cost are considered. Online TTP- A TTP involved during each session of the protocol but not during each message transmission is said to be online.

Offline TTP- A TTP involved in a protocol only in case of an incorrect behaviour of a dishonest entity or in case of network problem, is said to be offline. Timeliness- A protocol provides timeliness if and only if all honest parties always have the ability to reach, in a finite amount of time. Fairness- The protocol which guarantees the two parties involved to obtain or not obtain the others signature simultaneously is fair.

B. Security analysis

In this section, the security of the proposed scheme is analyzed and it is demonstrated that the proposed protocol has the resilience to several well-known attacks. All security parameters that are necessary for designing the PAKE protocol is taken into the proposed scheme. Such security attributes of the proposed protocol are compared with DH-BPAKE and enhance DH-BPAKE protocols in Table 2. Here, we briefly explain the security properties for proposed protocol.

Table 1: Comparison of basic features and efficiency

Parameters	Protocols				
	Verifiable Escrows Based Protocol [13]	Park et. al,'s RSA based protocol [14]	Enhance DH-BPAKE [5] protocol	DH-BPAKE [12] protocol	Proposed Protocol
Fairness	YES	YES	YES	YES	YES
Timeliness	YES	YES	YES	YES	YES
Transparent TTP	NO	YES	YES	YES	YES
TTP involvement	Off-line	Off-line	Off-line	Off-line	On-line
Additional Authentication	NO	NO	NO	NO	YES
Storage Cost	MORE	MORE	MORE	MORE	LESS

Table 2: Comparison of security attributes for the proposed protocol with the DH-BAPKE and enhanced DH-BPAKE

	DH-BPAKE protocol [5]	Enhance DH-BPAKE [12]	Our proposed protocol
Forward secrecy	YES	YES	YES
Two factor authentication	NO	NO	YES
Known session key security	YES	YES	YES
Resilience to password compromised impersonation attack	NO	YES	YES
Resilience to unknown key share(UKS) attack	YES	YES	YES
Resilience to off-line dictionary attack	YES	YES	YES
Resilience to undetectable online dictionary attack	YES	YES	YES
Resilience to replay attack	YES	YES	YES
Mutual authentication	YES	YES	YES

1) *Forward secrecy:* Even if the password of the party P_1 is disclosed, the adversary A cannot still construct the session key since the session key is generated by variable random values r_n and T_i that modify in every session. Adversary does not make session key because they do not have the value of T_i, no practical information about the session key will be leaked that is referred to as the forward secrecy property.
2) *Known session key security:* The session key is generated from two random numbers r_n and T_i that are independently selected by the trusted third party. These random numbers will change for every session which is independent of the other sessions. Consequently, the session keys of different session are independent from each other. It is impossible for adversary A to obtain one session key from the disclosed session key of the other sessions. Therefore, the proposed protocol provides the known session key security attribute.
3) *Two factor authentication:* when the user is login enters the password pw_1 for party P_1 and first time authenticate to the server, server send random number r_n via other media (mobile or email) user enter the random number and authenticate second time to the server.
4) *Resilience to password compromised impersonation attack:* Assume that the adversary A reveals party P_1 password and intercepts all the transmitted messages. He/she cannot obtain random value r_n and random key T_i, so he does not calculate the session key. Consequently, the password compromise impersonation attack cannot take place on the proposed protocol.
5)*Resilience to unknown key share (UKS) attack:* The session key is generated by the concatenation of two random values r_n and T_i which is generated and authenticated by the trusted third party, so any alteration in random numbers corresponding parties will result in different session key and fail the authentication. Consequently, success probability with UKS attack is negligible.
6) *Resilience to off-line dictionary attack:* There is not any obvious password validation information such as hash function of the password in the entire transmitted message between the parties and server. Thus, the adversary A will not be able to validate the accuracy of his/her guessed password and apply the off-line dictionary attack to the proposed protocol.
7) *Resilience to undetectable on-line dictionary attack:* Assume that the adversary A guesses pw_1 as a password of party p_1,but he cannot calculate the M_{key} because it is the combination of password and random number r_n which is comes from other secure media (email or mobile), so he/she not authenticate to the server because of M_{key} validation failed. Server notice that attack and then stop protocol run with error.
8) *Resilience to replay attack:* The trusted third party independently generate two random numbers r_n and T_i with the different method, that implicitly and explicitly are used in constructing the session key and other factors. The randomness of such changeable values guarantees the novelty and ensures us that the proposed protocol is secure against replay attack.

C. *Performance Evaluation*

This section compares the computation costs of the proposed protocol with DH-BPAKE and enhanced DH-BPAKE protocol. Table 3 shows that the proposed protocol removes the exponential operations, modular multiplication; modular inverse imposed to DH-PAKE and enhanced DH-BPAKE protocols. Among authentication plays a key role on the protocol rounds. Protocol with mutual strong 2-factor authentication complete in at least three rounds that is considered in our proposed protocol so there is decrease in the number of protocol rounds from four rounds for the DH-BPAKE protocol to three rounds for our proposed protocol. But it is one round more as compare to the enhanced DH-

BPAKE protocol because of it provides 2-factor authentication. Overall computational cost is less as compare to enhanced DH-BPAKE protocol. The comparison of computation costs of the proposed protocol with DH-PAKE and enhanced DH-PAKE protocols are summarized in table 3.

Table 3: Performance comparison of proposed protocol with DH-BPAKE and enhanced DH-BPAKE protocol

Protocols	**Participants**	**Exponentiation operation**	**Random Number Generation**	**Modular multiplication**	**Modular inverse**	**Modular addition**	**Hash Calculation**	**Number of rounds**
DH-BPAKE Protocol [12]	client	2	1	1	1	2	3	4
	server	2	1	1	1	2	3	
Enhanced DH-BPAKE Protocol [5]	client	2	1	0	0	0	3	2
	server	2	1	0	0	0	2	
Our proposed protocol	client	0	0	0	0	0	2	3
	server	0	2	0	0	0	2	

VI. CONCLUSION

The aim of this paper is construct a new and efficient 2-factor authentication PAKE protocol with TTP using one time private key (OTPK). Thus this protocol not only solve the problem of single point of failure by using one factor authentication but allow the key to always remains in client possession throughout the short lifetime, and never stored on a permanent basis so help in reducing the storage cost and thus providing security against various attacks.

REFERENCES

[1]. Amutha Prabakar Muniyandi, Rajaram Ramasamy, "Password Based Remote Authentication Scheme using ECC for Smart Card", Proceedings of the 2011 International Conference on Communication, Computing & Security, pp. 549-554, 2011.

[2]. Adam Groce and Jonathan Katz "A New Framework for Efficient Password-Based Authenticated Key Exchange", Proceedings of the 17th ACM conference on Computer and communications security, pp. 516-525, 2010.

[3]. Vinod Moreshwar Vaze "Digital Signature on-line, One Time Private Key [OTPK]", International Journal of Scientific & Engineering Research, ISSN 2229-5518, Volume 3, Issue 3, March -2012.

[4]. Jonathan Katz, Rafail Ostrovsky, Moti Yung "Efficient and Secure Authenticated Key Exchange Using Weak Passwords" Journal of the ACM, Vol. 57, No. 1, Article 3, Publication date: November 2009.

[5]. Maryam Saeed, Ali Mackvandi, Mansour Naddafiun, Hamid reza Karimnejad "An Enhanced password authenticated key exchange protocol without server public keys" 978-1-4673-4828-7/12/$31.00 ©2012 IEEE 87 ICTC 2012.

[6]. Wei-Kuo Chiang and Jian-Hao Chen "TW-KEAP: An Efficient Four-Party Key Exchange Protocol for End-to-End Communications.

[7]. Shai Halevi, Hugo Krawczyk "Public-Key Cryptography and PasswordProtocols", ACM Transactions on Information and System Security, Vol. 2, No. 3, pp. 230–268, August 1999.

[8]. Jan Camenisch, Anna Lysyanskaya, Gregory Neven "Practical Yet Universally Composable Two-Server Password-Authenticated Secret Sharing" CCS'12, October 16–18, 2012, Raleigh, North Carolina, USA. Copyright 2012 ACM 978-1-4503-1651-4/12/10 ...$10.00

[9]. Kyung-kug Kim, Myung-Hwan Kim "An Improved Anonymous Authentication and KeyExchange Scheme", CUBE 2012, September 3–5, 2012, Pune, Maharashtra, India. Copyright 2012 ACM 978-1-4503-1185-4/12/09…$10.00.

[10]. Dripto Chatterjee, Joyshree Nath, Suvadeep Dasgupta, Asoke Nath "A new Symmetric key Cryptography Algorithm using extended MSA method: DJSA symmetric key algorithm" 2011 International Conference on Communication Systems and Network Technologies, pp. 89-94, 2011.

[11]. Jung Min Park, Edwin K. P. Chong, Howard Jay Sieg "Constructing Fair-Exchange Protocols for E-commerce Via Distributed Computation of RSA Signatures", PODC"03, pp. 172-181, July 13-16, 2003, Boston, Massachusetts, USA. Copyright 2003 ACM 1-58 ! 13-708-7/03/0007...$5.00.

[12]. M.A. Strangio, "An optimal Round Two-Party password-Authenticated Key Agreement protocol," proceeding of the first IEEE International Conference on Availability, Reliability, and Security (ARES'06), pp.216-223, April. 2006.

[13]. N. Asokan, V. Shoup, and M. Waidner, "Optimistic fair exchange of digital signatures," IEEE J. Sel. Areas Commun., vol. 18, no. 4, pp. 591–606, Apr. 2000.

[14]. J. M. Park, E. Chong, H. J. Siegel, and I. Ray, "Constructing fair exchange protocols for e-commerce via distributed computation n of RSA signatures," in Proceedings of PODC'03, pp. 172–181, 2003.