A New Three Party Authenticated Key Agreement Protocol which is Defiant towards Password Guessing Attack

Soumyajit Nag
National Institute of Technology
Arunachal Pradesh, India
Email: nag.soumyajit73@gmail.com

Subhasish Banerjee
National Institute of Technology
Arunachal Pradesh, India
Email: subhasishism@gmail.com

Srijon Sen
National Institute of Technology
Arunachal Pradesh, India
Email: srijon.sen07@gmail.com

***Abstract*—In order to develop a 'common session secret key' though the insecure channel, cryptographic Key Agreement Protocol plays a major role. Many researcher's cryptographic protocol uses smart card as a medium to store transaction secret values. The tampered resistance property of smart card is unable to defend the secret values from side channel attacks. It means a lost smart card is an easy target for any attacker. Though password authentication helps the protocol to give secrecy but on-line as well as off-line password guessing attack can make the protocol vulnerable. The concerned paper manifested key agreement protocol based on three party authenticated key agreement protocol to defend all password related attacks. The security analysis of our paper has proven that the accurate guess of the password of a legitimate user will not help the adversary to generate a common session key.**

***Keywords*—Authentication, Key Agreement Protocol, Smart Card, Password Guessing Attacks, Bio-Hash**

I. Introduction

A transaction over insecure channel can only be secured if the encryption is done in a proper manner and symmetric encryption is completely dependent on key agreement protocol. Modular exponential operation [1], Bilinear pairing [2]–[4] as well as Chaotic Maps [5]–[7] are the target zone for any researcher to design a secure key agreement protocol. Key agreement protocol helps to authenticate the genuineness of the specific legitimate user and on the other hand they also allow to create a secret common session key through the insecure channel.

Most of the researchers have used the smart cards as there medium for key agreement protocol. It is observed that none of them are secured enough to resist all kind of attacks. There are some key agreement protocols [8]–[22] which uses chaotic map to generate the session key. Various protocols have used password to secure but password is an attractive target for the attacker to launch a password guessing attack where a adversary takes try and fail till he get the ultimate success. On-line and off-line are the two different types of Password guessing attack. However, many key agreement protocols are vulnerable towards the password guessing attack. It is recently found that Chen et. al's [23] protocol is unable to resist this attack [24].

The concerned paper has shown a three party authenticated key agreement protocol has been designed based on the theory of Deffie-Hellman Key exchange. Our design is not only secured against stolen smart card attack, however, it has the security against Password Guessing Attack. Another achievement of our protocol is to provide the anonymity of the user. After coming across a specific point of view, even a correct password guess of an attacker will not help him to imitate as the legitimate user. In the time of key agreement, it is not possible to understand the identity of the users as our proposed protocol is capable in order to achieve user anonymity. It is provided, a security analysis to prove that the given protocol is not prone towards the previously mentioned attacks.

II. Related Works

A. Bio Hashing

Biometric is a widely used verification parameter which offers numerous advantages over conventional authentication procedures. Password in addition with smart card are one of them. In addition to that, Biometric is known to be a property which varies on individual users and at the same time it is unable to be replaced. Therefore, the leakage of specific biometric data is an intense risk for the authentication. Various schemes have designed to protect biometric template [25], [26].

Bio-Hasing [27], [28] is a procedure to preserve the privacy of the biometric schemes. In general, the bio-hasing function is declared as $BH(K, B)$. Here, K is a secret parameter shared between user and server. B is the fingerprint of the user. The generation of bio-hasing value is done by comparison between the inner product of the random vector generated form the K and B. In the time of verification, user inputs B', performs $BH(K, B')$ and send the value to the server. Server calculates $BH(K, B)$ and checks that with the received parameter. If $BH(K, B) = BH(K, B')$, then it is confirmed that the sender is a legitimate user.

III. Our Proposed Protocol

In the concerned scenario, our three party authenticated key agreement protocol has been discussed which is defiant towards the password guessing attack. Three specific stages of our protocol are Initialization Phase, Registration Phase and Key Agreement Phase. In Table I, the required notations are provided.

TABLE I
NOTATIONS WHICH ARE USED IN THIS PAPER

Notation	Description
U_A	User A
U_B	User B
S	Secure trusted server
ID_A	Identities of User A
ID_B	Identities of User B
PW_A	Password of User A
PW_B	Password of User B
Fng_A	Fingerprint of User A
Fng_B	Fingerprint of User B
P	A large prime number
α	A primitive root of P
S_A	Secret parameter of U_A's smart card
S_B	Secret parameter of U_B's smart card
R_S	Secret parameter of Server S
R_A	Random number generated by user A
R_B	Random number generated by user B
SK	Shared common session key between U_A and U_B
$E_K(.)$	Symmetric Encryption function by Key K
$D_K(.)$	Symmetric Decryption function by Key K
$H(.)$	Single directional Hash function
$BH(.,.)$	Two factor Bio-Hasing Function
$\oplus$	Bitwise XOR operation
$\|\|$	Concatenation of the messages

A. Initialization Phase

In the initialization phase, Server(S) chooses the below parameters.

- A large prime number P.
- α is the primitive root of P.
- $E(.)$ is an encryption function.
- $D(.)$ is a decryption function.
- $H(.)$ is a single way hash function.
- $BH(.,.)$ is a Bio-Hash function.

Finally, the public parameters $\{P, \alpha, E(.), D(.), H(.), BH(.,.)\}$ are declared by the server.

B. Registration Phase

At the time of registration, any user U_i is required to compute the specific steps for registration within the server. The steps are as follows.

Step 1 : U_i selects ID_i, PW_i and Fng_i by his own. Then he sends that parameters to S through a secured channel. It is presumed that the channel is protected in the registration phase.

Step 2 : S calculates $P_S = \alpha^{R_S} \mod P$ where R_S is a secret number saved in the server. After receiving the parameters, server computes $S_i = (P_i \oplus PW_i)$ and saves Fng_i along with ID_i in it's database. Later server generates a smart card for the user where the parameters ID_i and P_S are saved.

C. Authentication and Key Agreement Phase

At the time of key agreement process, U_A and U_B have to authenticate themselves to the secure trusted server for generating a common session key. Figure 1 shows the complete and magnified steps of our scheme.

Step 1 : As a protocol initiator, U_A need to swipe the smart card in a card reader. He also requires to enter PW_A and Fng_A. Hereafter, he has to select a random number $R_A \in [1, P-1]$ and computes $P_A = \alpha^{R_A} \mod P$. After that, U_A has to calculate P_S by $P_S = (S_A \oplus PW_A)$. Again, U_A has to compute $Y_A = (P_S)^{R_A} \mod P$. By the help of the fingerprint Fng_A, U_A calculates $B_A = BH(Y_A, Fng_A)$. Then user A has to encrypt the values with the Y_A parameter. The encryption operation is $C_1 = E_{Y_A}(ID_A||ID_B||ID_S||B_A||P_A)$. At last user A has to send the message M_1 to the user B which contains the parameters P_A and C_1.

Step 2 : After receiving the message M_1, U_B inputs PW_B and Fng_B. After that, he has to select a random number $R_B \in [1, P-1]$. Then he calculates $P_B = \alpha^{R_B} \mod P$. The value P_S is calculated by $P_S = (S_B \oplus PW_B)$. Then he has to compute $Y_B = (P_S)^{R_B} \mod P$. Later he has to calculate the bio-hash value B_B by $B_B = BH(Y_B, Fng_B)$. At the end of this step, he has to encrypt the values of ID_B, ID_S, B_B, P_B by the Y_B. Mathematical expression for the operation is $C_2 = E_{Y_B}(ID_B||ID_S||B_B||P_B)$. At last user B has to send the message M_2 to the trusted server which contains the parameters P_A, C_1, P_B, C_2.

Step 3 : From the received message M_2, server has to calculate Y_A and Y_B first by it's secret value R_S. The mathematical expressions are $Y_A = (P_A)^{R_S} \mod P$ and $Y_B = (P_B)^{R_S} \mod P$. Then, S will decrypt the value C_1 by Y_A and C_2 by Y_B. After that, it has to search the values Fng_A and Fng_B with ID_A and ID_B respectively. Then server will verifies the values of B_A and B_B. If the parameters are matched then it is confirmed that the users are the legitimate users. Then server will compute two hash operation like $H_1 = H(ID_A||ID_B||ID_S||B_A||Y_A)$ and $H_2 = H(ID_A||ID_B||ID_S||B_B||Y_B||H1)$. At the final stage server has to done two encryption operations(one for user A by the value Y_A and another for user B by Y_B). The expressions are $C_3 = E_{Y_A}(ID_A||ID_B||ID_S||H_1||P_B)$ and $C_4 = E_{Y_B}(ID_A||ID_B||ID_S||H_1||H_2||C_3)$. At the end server S sends M_3 to user B which contains the parameters C_4.

Step 4 : At the current stage, U_B has to decrypt the parameter C_4 by Y_B. After decryption, he has to check the authenticity of the parameters H_2 and ID_A. If they are validated, then user B has to calculate the session key SK by $SK = (P_A)^{R_B} \mod P$. Then he will calculate the hash value $H_3 = H(SK||C_3)$ and send message M_4 to user A which contains the parameters C_3 and H_3.

Step 5 : At the ultimate step, user A has to decrypt the parameter C_3 by Y_A first. Then he has to check the authenticity of the hash value H_1. If it is validated, U_A has to calculate the session Key $SK = (P_B)^{R_A} \mod P$. After that, he is needed

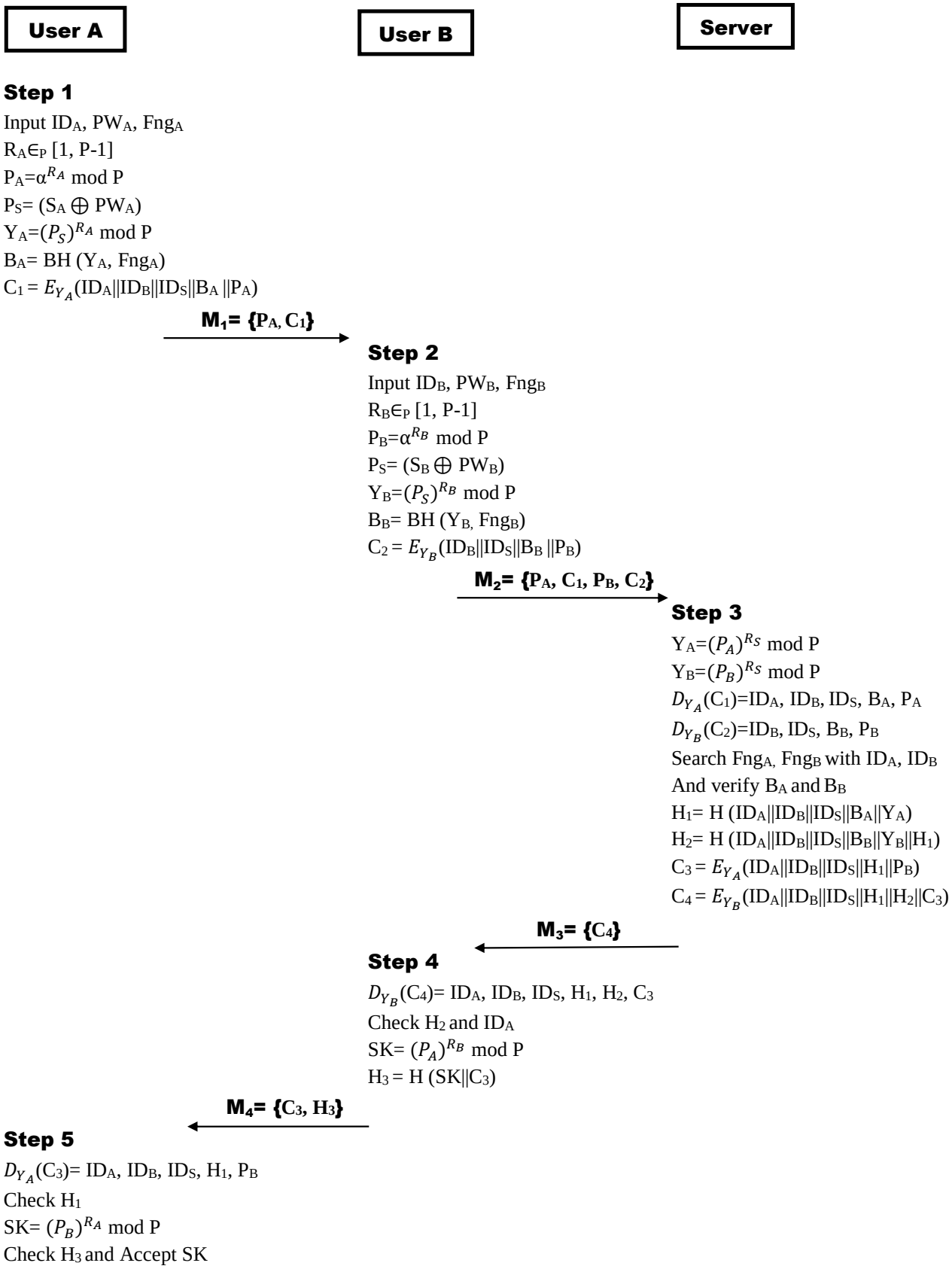

Fig. 1. The detailed steps of our proposed Protocol

to check the hash value H_3 and if it is validated then it denotes that he has established a secret common session key in addition with user B successfully.

IV. Security Analysis of our protocol

The concerned section has shown that the proposed protocol is safe against numerous types of severe attacks. Therefore, it can reach different cryptographic goals.

A. Achieving User Anonymity

There are total four communication rounds in our protocol. The communication messages are M_1, M_2, M_3, M_4 for communication rounds 1,2,3,4 respectively. M_1 contains P_A and C_1 where P_A doesn't contain any identity and C_1 is encrypted. From M_2 we can get the parameters P_A, C_1, P_B, C_2 where P_A, P_B don't hold any identity and C_1, C_2 are encrypted. M_3 contains C_4 and M_4 contains C_3, H_3. Here $C_3, C_4$are encrypted and H_3 is the hash value. So from neither of the message holds the identity publicly. As a result, users are untraceable from the attacker and the anonymity of the user identity is achieved in our protocol.

B. Defiant to On-line Password Guessing Attack

Our presented protocol needs three input parameters from each user. Those are ID_i, PW_i and Fng_i. Here PW_i and Fng_i are the authentication parameters and ID_i is the identity parameter. Now let us assume, the password is leaked for any user, then adversary can easily calculate Y_i. Still, he is unable to calculate B_i because the Fng_i is needed for the calculation of B_i. Fingerprint of any user is a unique identity for a person in this world and cannot be duplicated. Hence, it is confirmed that our protocol is defiant against on-line password guessing attack.

C. Defends Off-line Password Guessing attack

Apart from biometric, let us consider the password case only. Neither the secret parameter P_S nor the password are directly saved inside the smart card. The parameter P_S is merged with password PW_i under a XOR operation. Hence, it is not possible to reach the perfect P_S value without the help of the server. So, no off-line process will help the attacker to reach the value P_S. As a result, it could be claimed that the proposed protocol is properly secured from the off-line password guessing attack.

D. Defiant to Stolen Smart Card Attack

The authentication process of our protocol is dependent on the password and biometric of both user A and user B. None of the password or biometric are saved inside the smart card. Even, password is merged with a secured parameter and biometric is bio-hashed first, then have to send to server. As a result, if any adversary steal anyone's smart card, still he is unable to launch any kind of attack, unless and until he has a complete knowledge of password and biometric both. Hence, we can conclude that our protocol is secured enough to defend stolen smart card attack.

E. Defiant to User Impersonation Attack

Each of the communication messages have atleast a cipher text. That cipher texts are encrypted by the parameter Y_i. Now, the parameter Y_i is derived from the secret value P_S and P_S is merged with PW_i under XOR operation. Hence, to replicate the legitimate user, attacker must to be aware of users' identities. Biometric is unique for each user and password is not possible to steal normally. Hence we can conclude that our protocol is free from user impersonation attack.

F. Provides Mutual Authentication

Mutual authentication is known to be specific authentication process that takes place in between two of the legitimate parties in a key agreement protocol. In the current times, mutual authentication takes place between two individual parties with the presence of a third party. In our protocol, the secure trusted server is the third party. Now, the authentication between user A, user B and S are needed to confirm the proper mutual authentication.

Message M_1 holds the parameters of C_1 which contains the identities of U_A, U_B and S. C_1 is symmetrically encrypted by the value Y_A, which can be decrypted by Y_A only. The parameter $Y_A = (\alpha)^{R_S R_A}$. R_S is known by server only and R_A is secretly known by user A only. The untraceable property of discreet logarithm problem will not help the attacker to find R_A from P_A. Even to get the exact value of P_S, attacker is required to know the password of user A along with biometric. Hence, it is impossible to impersonate on behalf of user A as well as user B.

After receiving the parameter C_1 and C_2, server has to decrypt the parameters by Y_A and Y_B respectively. C_1 contains B_A which is the identification parameter of user A and C_2 contains user B's identification parameter B_B. If B_A and B_B are valid then the authenticity of user A and user B are confirmed by the end of server S. Server returns H_1 for U_A and H_2 for U_B. If both the parameters are properly validated by the legal parties then they are confident about the authenticity of the server. At the final message M_4, there is a parameter named H_3. If H_3 is properly validated by user A then he is assured about the successful establishment of a common session key along with user B.

G. Achieving Perfect Forward Secrecy

The term Perfect forward secrecy in cryptography signifies that the compromise of any private key will not help the attacker to compute the session key. Our protocol have encryption operations but all the encryption operations are symmetric encryption. Hence, there is no concept of public key and private key in our protocol. As the authenticity of our protocol is completely dependent on the user's biometric, then it could be guaranteed that the proposed protocol is completely achieving perfect forward secrecy.

TABLE II
PERFORMANCE COMPARISONS

	Lai et. al.	Zhao et. al.	Xie et. al.	Li et. al.	Chen et. al.	Our Protocol
Mutual Authentication among all three parties	Y	Y	Y	Y	Y	Y
Clock synchronization is needed	Y	Y	Y	N	Y	N
Perfect forward secrecy	Y	Y	Y	Y	Y	Y
Known-key-Security	Y	Y	Y	Y	Y	Y
Resist key compromise impersonation Attack	Y	Y	Y	Y	Y	Y
Resist replay attack	Y	Y	Y	Y	Y	Y
Resist man in the middle attack	Y	Y	Y	Y	Y	Y
Resist unknown key share attack	Y	Y	Y	Y	Y	Y
Resist impersonation attack	Y	Y	Y	Y	Y	Y
Resist message modification attack	Y	Y	Y	Y	Y	Y
Resist stolen smart card attack	N	N	N	N	Y	Y
Achieving user anonymity	N	Y	N	N	N	Y
Resist denial of service attack	N	N	N	N	N	Y
Resist On-line Password Guessing attack	N	N	N	N	N	Y
Resist Off-line Password Guessing attack	N	N	N	N	N	Y
Communication rounds	7	7	5	5	5	4

TABLE III
TIME COMPARISONS

	User A	User B	Server S	Total Time
Lai et. al.	$3T_P + 5T_S$	$3T_P + 5T_S$	$2T_P + 2T_F + 6T_S$	$8T_P + 2T_F + 16T_S$
Zhao et. al.	$3T_P + 1T_F + 6T_S$	$3T_P + 1T_F + 5T_S$	$2T_P + 2T_F + 8T_S$	$8T_P + 4T_F + 19T_S$
Xie et. al.	$3T_P + 2T_F + 5T_S$	$3T_P + 2T_F + 5T_S$	$2T_P + 4T_F + 4T_S$	$8T_P + 8T_F + 14T_S$
Li et. al.	$2T_P + 2T_F + 4T_S$	$2T_P + 2T_F + 4T_S$	$4T_F + 3T_H$	$4T_P + 8T_F + 11T_S$
Chen et. al.	$3T_D + 4T_S$	$3T_D + 4T_S$	$2T_D + 4T_S$	$8T_D + 12T_S$
Our Protocol	$3T_D + 2T_F + 2T_S + 1T_B$	$3T_D + 2T_F + 2T_S + 1T_B$	$2T_D + 4T_F + 2T_S + 2T_B$	$8T_D + 8T_F + 6T_S + 4T_B$

H. Maintain Known Key Security

Known key security term highlights that the disclose of old session key will not help the attacker for computing the current session key. In our protocol, the calculation of session key in user B's end is $SK = (P_A)^{R_B} \mod P$ and user A's end is $SK = (P_B)^{R_A} \mod P$. As a simplification $SK = (\alpha)^{R_A R_B} \mod P$. Here, R_A is the user A's contribution and R_B is the user B's contribution. Let us assume that the old session key was compromised, and in that time, user A and user B' random number contribution were R'_A and R'_B respectively. So, the old session key was $SK' = (\alpha)^{R'_A R'_B} \mod P$. It is obvious that the old session key $SK' \neq SK$(new session key). Hence, our protocol is able to achieve known key security.

I. Resists Man-in-the-middle Attack

Best way for preventing man-in-the-middle attack is to achieve the mutual authentication. Now the identities of the user A, user B and server S are encrypted through public channel, therefore cannot be duplicated. Every user must send their authentication parameters also along with identity for verification. None of the parameters cannot be duplicated according to the property of cryptography. Hence we can claim that our protocol can resist man-in-the-middle attack.

J. Resists Unknown Key Share Attack

Unknown key share attack is, when any user say user A end up believing that a session key is being shared with user B and also user B believes as a mistake that he shares session key with $E \neq A$. As a result, user B led to a false belief and adversary E is able to launch unknown key share attack. Mutual authentication is the best solution to prevent unknown key share attack. In the early subsection, we showed that the proposed protocol is capable enough to achieve mutual authentication. Not only that, but also we use a symmetric encryption operation to achieve user anonymity as well as authentication parameters. Hence, we can conclude that our protocol is free from unknown key share attack.

K. Resists Replay Attack

There is a certain type of attack, where any adversary is able to reply properly against a false message. Replay attack can be launched if and only if the adversary can able to impersonate on behalf of the server. In our protocol, the message M_3 contains C_4 which is encrypted by the value Y_B. The parameter Y_B cannot be calculated by adversary under various circumstances. Hence he is unable to throw a proper reply. After analysing it could be said that, our protocol is secured from replay attack.

L. Key Control is achieved

When the predictability of the session key is unaware by any user, then the protocol will be achieving the key control property. The session key generation of our protocol is done by the two random numbers where R_A is user A's contribution and R_B is user B's contribution. The selection of random number is done at the time of key agreement. Therefore, it is next to impossible in order to assume the session key prior to the activity. Hence, key control property is achieved in the proposed protocol.

V. Performance analysis of our protocol

The following section has showcased the brief illustration of our protocol's performance. The comparison between the previous protocols are done in the later stages. For the performance evaluation purpose, we have to come across with the below notations.

- T_D: Average time needed for Exponential Operation.
- T_P: Average time needed for Chebyshev polynomial.
- T_S: Average time needed for single way Hash function.
- T_B: Average time needed for Bio-Hash function.
- T_F: Average time needed for Symmetric encryption or decryption function.

For time computation, T_C is grater than T_D and both are very much greater than T_H, T_B and T_S. Hence, we can neglect the time difference for the hash operation and symmetric encryption or decryption operation. Minimizing the exponential operations are the main goal of any cryptographic efficient protocol. Table III shows the time difference between the proposed protocol and other protocols. The performance difference between the given protocol and other schemes are illustrated in Table II.

VI. Conclusion and Future Work

Many key agreement protocols was designed in the past but they are still insecure. The paper has thoroughly manifested that the proposed three party authenticated key agreement protocol is defiant towards password guessing attack. In Accordance with security analysis, given protocol is completely free from password guessing attack.

Due to the limitation of paper submission space, we are unable to serve the formal theoretical prove of our protocol by BAN logic. We will demonstrate the proof of our protocol by using ProVerif security tool also. As our main focus is to defend password related attacks, so this protocol will be implemented in the critical part of the real world where password systems are completely vulnerable. This part of our research is almost finished and we have a plan to implement it in any banking system and mobile software in near future. This work is still under progress.

References

[1] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[2] Y. J. Choie, E. Jeong, and E. Lee, "Efficient identity-based authenticated key agreement protocol from pairings," *Applied Mathematics and Computation*, vol. 162, no. 1, pp. 179–188, 2005.

[3] X. Cao, W. Kou, and X. Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges," *Information Sciences*, vol. 180, no. 15, pp. 2895–2903, 2010.

[4] T.-Y. Wu and Y.-M. Tseng, "An efficient user authentication and key exchange protocol for mobile client–server environment," *Computer Networks*, vol. 54, no. 9, pp. 1520–1530, 2010.

[5] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3d chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749–761, 2004.

[6] X.-Y. Wang, L. Yang, R. Liu, and A. Kadir, "A chaotic image encryption algorithm based on perceptron model," *Nonlinear Dynamics*, vol. 62, no. 3, pp. 615–621, 2010.

[7] L. J. Sheu, "A speech encryption using fractional chaotic systems," *Nonlinear dynamics*, vol. 65, no. 1-2, pp. 103–108, 2011.

[8] D. Xiao, X. Liao, and S. Deng, "A novel key agreement protocol based on chaotic maps," *Information Sciences*, vol. 177, no. 4, pp. 1136–1142, 2007.

[9] ——, "Using time-stamp to improve the security of a chaotic maps-based key agreement protocol," *Information Sciences*, vol. 178, no. 6, pp. 1598–1602, 2008.

[10] S. Han and E. Chang, "Chaotic map based key agreement with/out clock synchronization," *Chaos, Solitons & Fractals*, vol. 39, no. 3, pp. 1283–1289, 2009.

[11] H.-R. Tseng, R.-H. Jan, W. Yang *et al.*, "A chaotic maps-based key agreement protocol that preserves user anonymity." in *ICC*, 2009, pp. 1–6.

[12] Y. Niu and X. Wang, "An anonymous key agreement protocol based on chaotic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 4, pp. 1986–1992, 2011.

[13] Z. Tan, "A chaotic maps-based authenticated key agreement protocol with strong anonymity," *Nonlinear Dynamics*, vol. 72, no. 1-2, pp. 311–320, 2013.

[14] C.-C. Lee, C.-L. Chen, C.-Y. Wu, and S.-Y. Huang, "An extended chaotic maps-based key agreement protocol with user anonymity," *Nonlinear Dynamics*, vol. 69, no. 1-2, pp. 79–87, 2012.

[15] E.-J. Yoon and I.-S. Jeon, "An efficient and secure diffie–hellman key agreement protocol based on chebyshev chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 6, pp. 2383–2389, 2011.

[16] H. Lai, J. Xiao, L. Li, and Y. Yang, "Applying semigroup property of enhanced chebyshev polynomials to anonymous authentication protocol," *Mathematical Problems in Engineering*, vol. 2012, 2012.

[17] F. Zhao, P. Gong, S. Li, M. Li, and P. Li, "Cryptanalysis and improvement of a three-party key agreement protocol using enhanced chebyshev polynomials," *Nonlinear Dynamics*, vol. 74, no. 1-2, pp. 419–427, 2013.

[18] C.-C. Lee, C.-T. Li, and C.-W. Hsu, "A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps," *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 125–132, 2013.

[19] Q. Xie, J. Zhao, and X. Yu, "Chaotic maps-based three-party password-authenticated key agreement scheme," *Nonlinear Dynamics*, vol. 74, no. 4, pp. 1021–1027, 2013.

[20] M. S. Farash and M. A. Attari, "An efficient and provably secure three-party password-based authenticated key exchange protocol based on chebyshev chaotic maps," *Nonlinear Dynamics*, vol. 77, no. 1-2, pp. 399–411, 2014.

[21] C.-C. Lee, C.-T. Li, S.-T. Chiu, and Y.-M. Lai, "A new three-party-authenticated key agreement scheme based on chaotic maps without password table," *Nonlinear Dynamics*, vol. 79, no. 4, pp. 2485–2495, 2015.

[22] X. Li, J. Niu, S. Kumari, M. K. Khan, J. Liao, and W. Liang, "Design and analysis of a chaotic maps-based three-party authenticated key agreement protocol," *Nonlinear Dynamics*, vol. 80, no. 3, pp. 1209–1220, 2015.

[23] C.-M. Chen, L. Xu, W. Fang, and T.-Y. Wu, "A three-party password authenticated key exchange protocol resistant to stolen smart card attacks," in *Advances in Intelligent Information Hiding and Multimedia Signal Processing*. Springer, 2017, pp. 331–336.

[24] S. Nag and S. Banerjee, "Security issues of a three-party password authenticated key exchange protocol resistant to stolen smart card attacks," in *2018 2nd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*. IEEE, 2018, pp. 1–5.

[25] I. Natgunanathan, A. Mehmood, Y. Xiang, G. Beliakov, and J. Yearwood, "Protection of privacy in biometric data," *IEEE access*, vol. 4, pp. 880–892, 2016.

[26] A. Juels and M. Sudan, "A fuzzy vault scheme," *Designs, Codes and Cryptography*, vol. 38, no. 2, pp. 237–257, 2006.

[27] A. T. B. Jin, D. N. C. Ling, and A. Goh, "Biohashing: two factor authentication featuring fingerprint data and tokenised random number," *Pattern recognition*, vol. 37, no. 11, pp. 2245–2255, 2004.

[28] A. Lumini and L. Nanni, "An improved biohashing for human authentication," *Pattern recognition*, vol. 40, no. 3, pp. 1057–1065, 2007.