An Efficient Authentication Scheme

Sattar J Aboud
Information Technology Advisor
Iraqi Council of Representatives, Baghdad-Iraq

Abstract

In 2000, Peyravian and Zunic presented a simple password authentication scheme using collision-resistant hash function. Later, Hwang and Yeh denoted that Peyravian and Zunic scheme is insecure and suggested an improvement one using the server public key. However, in practice, services that do not use public keys are quite often superior to PKIs. Simultaneously, Lee, Kim and Yoo denoted that Peyravian and Zunic scheme undergoes from offline password guessing attacks and presented an improved version. However, Lee, Kim and Yoo proposed scheme is still vulnerable to the same attacks and denial-of-service attacks. Therefore, this paper presents a secure and efficient improvement . Lee, Kim and Yoo suggested a password scheme for three participants without trusted server. They claimed that the scheme can withstand different attacks and give the perfect secrecy. In this paper, we will demonstrate that their scheme undergoes from the imitation attack. Simultaneously, we will suggest an enhanced algorithm to resist the mentioned attacks.

Keyword: *password scheme, key agreement, trusted server, cryptanalysis.*

1. Introduction

If two participants need to communicate between each of them, the identity authentication of the other participant is a vital need. For improving the functioning of the scheme, Bellovin and Merrit [1] suggested an encrypted exchange protocol relied on password for authentication and key agreement. Subsequent the idea of Bellovin and Merrit scheme, several three-party schemes is suggested [2, 3, 4]. But, many of these schemes still have certain security difficulties for instance, an off-line password guessing attack [5] and an on-line password guessing attack [6]. Not just that, several schemes also needed trusted server to protect the common password [7].

To improving the efficiency and avoiding some attacks, Lee [8] suggested an effective verifier typed key agreement scheme for three participants without trusted server. Lee scheme gives the perfect secrecy by using the idea of the Diffie and Hellman protocol [9] and every user expect only requests to memorize an unforgettable password. The scheme is appropriate for some uses of low calculations. However, we will illustrate that Lee scheme is weak under the imitation attack without construction an off-line password guessing attack. Simultaneously, in this paper we will suggest an enhanced algorithm to resist the mentioned attack.

2. Related Work

Since the innovative method that withstands the password guessing attacks was presented in 1989 by Lomas, Gong, Saltzer and Needham [10], there have been a several password-typed authenticated key agreement schemes were introduced.

In 1996, Jablon [11] proposed a scheme were security relied on heuristic arguments. Also, in 1999 Halevi and Krawczyk [12] introduced another scheme, the scheme considered as inflexible for security of password-typed authenticated scheme. However, Boyarsky in 1999 [13] improved this scheme by making it secure in multi-user environment, but, this scheme is inappropriate for situation where communication has to be established between entities those sharing a common limited-entropy password. In 2000 [14], another password - typed key exchange scheme has been suggested by Boyko, MacKenzie and Patel. This scheme is relied on two-party password-typed scheme. An enhancement for this scheme was made to multi-party setting by Bresson, Chevassut and Pointcheval [15]. The security of Bresson, Chevassut and Pointcheval scheme is based on the arbitrary oracle approach and in the ideal cipher approach.

In 2004, Lee, Kim, Kim and Yoo [16] suggested a verifiable-typed key agreement scheme. In this scheme, the entity employs a document of the password, while the server keeps as a verifier for the password. Thus the scheme cannot let an opponent who able to exchange information with the server to impersonate any entity without running the dictionary attack in the password file. But, the scheme is not protected against stolen-verifier attack as Kwon, in 2004[17] have claimed. Also, Yoon and Yoo Kin 2005 [18] proposed a two-party key agreement scheme relied on Diffie and Hellman scheme. Also, in 2006, Strangio [19] presented another two-party key agreement protocol relied also on Diffie and Hellman scheme. Both schemes are not appropriate for large networks since they cannot

assume each party shares a secret password with other entity.

However, the first work that copes with off-line dictionary attacks is introduced in 2007 by Bellovin and Merritt [20]. They presented a family of encrypted key exchange to resist dictionary attack. This protocol is very important and become the foundation for future work in this area. In 2008, Shakir Hussain and Hussein Al-Bahadili [21] proposed simple authenticated key agreement protocol which is relied on Diffie and Hellman key agreement protocol. Unfortunately, this protocol is inefficient for practical use and does not allow concurrent executions. Also, this scheme is simple and cost effective. In 2009, SeongHan Shin, Kazukuni Kobara and Hideki Imai [22] introduced a scheme relied on threshold anonymous scheme. However, the scheme is complicated and costly.

In this paper, we will briefly evaluate Lee, Kim, Kim and Yoo 2004 [16] key agreement scheme and show its weaknesses to stolen-verifier attack. Then, we introduce a new scheme that verifier-typed key agreement scheme. The new scheme resists password guessing attack and stolen-verifier attack.

3. Review of Lee Scheme

In this section, we will review Lee scheme illustrate that the scheme is vulnerable. Prior to describing the scheme, we will determine the notations used which are as follows.

3.1. Notations Used

id_A : Indicates the identity of entity A

id_B : Indicates the identity of entity B

id_T : Indicates the Trusted server T

p : Represents a prime number and g is a primitive in the cyclic group Z_p^*

$h(.)$: Represents a secure one-way hash function

w_A : Represents a password for entity A

w_B : Represents a password for entity B

H : Represents the hacker

3.2. Lee Scheme Description

Prior to execution the scheme, suppose there are two entities, entity A and entity B, decide to agree a shared session key over a trusted server, named T. For enrolling T, A and B respectively, select passwords w_A for entity A and for entity B. Calculate verifiers $v_A = g^{(id_A, id_T, w_A)} \bmod p$ and $v_B = g^{(id_B, id_T, w_B)} \bmod p$ then send v_A and v_B to T through a secure channel. Trusted server T stores v_A and v_B in a password list. The steps of the scheme are as follows:

Step 1: Entity A

- Finds $y_A = g^a \bmod p$ by randomly choosing a such that $a \in Z_p^*$.
- Finds $t_A = g^{h(id_A, id_T, w_A)} \bmod p$.
- Passes y_A and the identity id_A to entity B,

Step 2: Entity B

- Finds $y_B = g^b \bmod p$ by randomly choosing a such that $b \in Z_p^*$.
- Finds $t_B = g^{h(id_B, id_T, w_B)} \bmod p$.
- Passes y_B and (id_A, y_A, id_B, y_B) to trusted server T and send y_B to entity A.

Step 3: Trusted server T

- gets back v_A and v_B from password list.
- Finds $y_{TA} = (v_A)^c \oplus v_A$ and $y_{TB} = (v_B)^d \oplus v_B$ by randomly choosing c and $d \in_R Z_p^*$.
- Passes y_{TA} and y_{TB} to entity A and entity B respectively.
- Finds $r_{TA} = (y_A)^c = g^{a*c} \bmod p$.
- Finds $r_{TB} = (y_B)^d = g^{b*d} \bmod p$.

Step 4: Entity A

- Finds $r_{AT} = (y_{TA} \oplus v_A)^{t_A^{-1}*a} = g^{a*c} \bmod p$.
- Find $v_{AT} = h(id_A, id_B, id_T, y_A, y_B, y_{TA}, r_{AT}) \bmod p$
- Passes v_{AT} to trusted server T.

Step 5: Entity B

- Finds $r_{BT} = (y_{TB} \oplus v_B)^{t_B^{-1}*b} = g^{b*d} \bmod p$
- Find $v_{BT} = h(id_B, id_A, id_T, y_B, y_A, y_{TB}, r_{BT}) \bmod p$
- Passes v_{BT} to trusted server T

Step 6: Trusted server T

- Checks if $v_{AT} \equiv h(id_A, id_B, id_T, y_A, y_B, y_{TA}, r_{AT}) \bmod p$ and $v_{BT} \equiv h(id_B, id_A, id_T, y_B, y_A, y_{TB}, r_{BT}) \bmod p$ are true or not. If they are true, T is convinced that entity A and entity B are validated.
- Calculate $v_{TA} = h(id_T, ID_A, ID_B, y_A, y_B, r_{TA})$ and $v_{TB} = h(id_T, id_B, id_A, y_B, y_A, r_{TB})$.
- Passes v_{TA} and v_{TB} to entity A and entity B respectively,

Step 7: Verification by entity A and entity B

- Entity A checks if $v_{TA} \equiv h(id_T, id_A, id_B, y_A, y_B, r_{AT})$ is true or not. If it is true, entity A is convinced that both entity B and trusted sever T are validated
- Entity B checks whether $v_{TB} = h(id_T, id_B, id_A, y_B, y_A, r_{BT})$ holds or

not. If it holds, entity B is convinced that both entity A and trusted server T are validated.

- Entity A finds $r_{AB} = (y_B)^a = g^{b*a} \bmod p$
- Entity B finds $r_{BA} = (y_A)^b = g^{a*b} \bmod p$
- Lastly, entity A and entity B compute the shared session key $S = h(id_A, id_B, id_T, r_{AB}) \equiv h(id_A, id_B, T, r_{BA}) \equiv h(id_A, id_B, id_T, g^{a*b})$ respectively.

3.3. Vulnerability in Lee Scheme

Suppose v_A has stolen by the hacker. The hacker can masquerade as entity B in order to connect with entity A by the following steps.

Step 1: Entity A

- If entity A want to connects with entity B. Entity A transmits (id_A, y_A) to entity B.
- Immediately, the hacker intercepts it and passes $(id_A, y_A^{'} = v_A)$ to entity B.

Step 2: Entity B

- Forwards the information received with (id_B, y_B) to trusted server T and also transmits y_B to the hacker respectively
- Then, the hacker passes $y_B^{'} = g^{b^{'}} \bmod p$ to entity A, such that $b^{'}$ is selected by the hacker.

Step 3: Trusted server T

- Finds $y_{TA} = (v_A)^c \oplus v_A$
- Computes $y_{TB} = (v_B)^d \oplus v_B$
- Passes the calculated values (y_{TA} and y_{TB}) to the hacker and entity B respectively.
- finds $r_{TA} = (y_A^{'})^c = (v_A)^c \bmod p$
- Lastly, computes $r_{TB} = (y_B)^d = g^{b*d} \bmod p$

Step 4: Hacker H

- Passes $y_{TA}^{'} = (v_A)^{c^{'}} \oplus v_A$ to entity A, such that $c^{'}$ is selected by the hacker.
- Finds $r_{AT}^{'} = y_{TA} \oplus v_A = (v_A)^c \bmod p$
- Passes $v_{AT}^{'} = h(id_A, id_B, id_T, y_A, y_B, y_{TA}, r_{AT}^{'})$ to trusted server T.
- computes $r_{TA}^{'} = (y_A)^{c^{'}} = g^{a*c^{'}} \bmod p$
- Computes $v_{TA}^{'} = h(id_T, id_A, id_B, y_A, y_B . r_{TA}^{'})$
- Passes $v_{TA}^{'}$ to entity A

Step 5: Entity A

- Firstly, entity A accept the received information $v_{TA}^{'}$ since $r_{AT} = r_{AT}^{'} = g^{a*d} \bmod p$
- Finds $r_{AT} = (y_{TA}^{'} \oplus v_A)^{t_A^{-1}*a} = g^{a*c^{'}} \bmod p$
- Finds $v_{AT} = h(id_T, id_A, id_B, y_A, y_B, r_{AT})$
- Passes v_{AT} to the hacker.

Step 6: Verification

- Entity A accepts the shared session key $S = h(id_A, id_B, id_T, r_{AB} = g^{b^{'}*a})$
- Entity B also accepts that the acquired session key $S = h(id_A, id_B, id_T, r_{BA} = (v_A)^b)$ is true.
- Hacker H obtains the session key $r_{BA}^{'} = (y_A)^{b^{'}} g^{a*b^{'}} \bmod p$ and impersonates entity B to interconnect with entity A.

4. Enhancement of Lee Scheme

In this section we will propose an enhanced algorithm to resist the mentioned above attack.

- Entity A and entity B must pass $v_{AT} = h(id_A, id_B, id_T, y_A, y_B, (y_{TA} \oplus v_A)^{t_A^{-1}} = g^c, r_{AT})$ and $v_{BT} = h(id_B, id_A, id_T, y_B, y_A, (y_{TB} \oplus v_B)^{t_B^{-1}} = g^d, r_{BT})$ to trusted server T respectively as in the step four of Lee scheme.
- Even if both entity A and entity B verifiers v_A and v_B are stolen by the hacker. The hacker still cannot make an impersonation attack on the proposed enhancement without $t_A = h(id_A, id_T, w_A)$, $t_B = h(id_B, id_T, w_B)$, therefore, we will illustrated the proposed enhancement by the following steps:

Step 1: Entity A

- Finds $y_A = g^a \bmod p$ such that $a \in Z_p^*$
- Passes y_A and the identity id_A to entity B,

Step 2: Entity B

- Finds $y_B = g^b \bmod p$ such that $b \in Z_p^*$
- Passes y_B to entity A
- Passes (id_A, y_A, id_B, y_B) to trusted server T

Step 3: Trusted server T

- Trusted server T gets back v_A and v_B from password list
- Finds $y_{TA} = (v_A)^c \oplus v_A$ and $y_{TB} = (v_B)^d \oplus v_B$ such that c and $d \in Z_p^*$
- Passes y_{TA} to entity A
- Passes y_{TB} to entity B
- Finds $r_{TA} = (y_A)^c = g^{a*c} \bmod p$
- Finds $r_{TB} = (y_B)^c = g^{b*c} \bmod p$

Step 4: Entity A

- Finds $r_{AT} = (y_{TA} \oplus v_A)^{t_A^{-1} * a} = g^{a*c} \bmod p$
- Finds $v_{AT} = h(id_A, id_B, id_T, y_A, y_B, g^c, r_{AT}) \bmod p$
- Passes v_{AT} to trusted server T

Step 5: Entity B

- Finds $r_{BT} = (y_{TB} \oplus v_B)^{t_B^{-1} * b} = g^{b*d} \bmod p$
- Finds $v_{BT} = h(id_B, id_A, id_T, y_B, y_A, g^d, r_{BT_s}) \bmod p$
- Passes v_{BT} to trusted server

Step 6: Trusted server T

- Checks if v_{AT} and v_{BT} are true or not. When they are true
- Computes $v_{TA} = h(id_T, id_A, id_B, y_A, y_B, r_{TA})$
- Passes v_{TA} to entity A
- Computes $v_{TB} = h(id_T, id_B, id_A, y_B, y_A, r_{TB})$
- Passes v_{TB} to entity B

Step 7: Verification

- Entity A and entity B checks if v_{TA} and v_{TB} are true or not respectively. When they are true,
- Entity A finds $r_{AB} = (y_B)^a = g^{a*b} \bmod p$
- Entity B finds $r_{BA} = (y_A)^b = g^{a*b} \bmod p$
- Lastly, entity A and entity B exchange a shared session key $S = h(id_A, id_B, id_T, r_{AB}) \equiv h(id_A, id_B, id_T, r_{BA})$.

5. Conclusion

In this paper, we illustrate that Lee scheme for three participants is weak to the impersonation attack without construction an off-line password guessing attack. Simultaneously, we suggest an enhancement to resist the above mention attack. Therefore, this article presents an improvement without, employing the server's public key. According to the analyses described above, it is unquestionable that the proposed scheme is secure, practical, and efficient. In addition, the proposed scheme is also suitable in imbalanced networks.

6. References

[1] Bellovin S and Merrit M (1992) "Encrypted key exchange: password-based protocols secure against dictionary attacks", Proceedings of the IEEE Symposium on Research in Security and Privacy, pp. 72-84.

[2] Chang C and Cheng Y (2004) "A novel three-party encrypted key exchange protocol", Computer Standards & Interfaces, volume 26, number 5, pp. 471-476.

[3] Lee T, Hwang T and Lin C (2004) "Enhanced three-party encrypted key exchange without server public keys", Computers & Security, 23, pp. 571-577.

[4] Lu R and Cao Z (2007) "Simple three-party key exchange protocol", Computers Security 26 (1), pp. 94-97.

[5] Nam J, Lee Y, Kim S, and Won D (2007) "Security weakness in a three-party pairing-based protocol for password authenticated key exchange", Information Sciences 177 (6), pp. 1364-1375.

[6] Shim K and Woo S (2007) "Cryptanalysis of tripartite and multi-party authenticated key agreement protocols", Information Sciences 177 (4), pp. 1143-1151.

[7] Sun H, Chen B and Hwang T (2005) "Secure key agreement protocols for three-party against guessing attacks", The Journal of Systems and Software, 75 (1-2), pp. 63-68.

[8] Lee S, Kim H and Yoo K (2005) "Efficient verifier-based key agreement protocol for three parties without server public key", Applied Mathematics and Computation, 167, pp. 996-1003.

[9] Diffie W and Hellman M (1976) "New directions in cryptography", IEEE Transaction on Information Theory, 22, pp. 644-654.

[10] Lomas T, Gong L, Saltzer J and Needham (1989) "Reducing Risks from poorly chosen Keys", ACM SIGOPS Operat, System Review, 23: 14-18.

[11] Jablon D (1996) "Strong password-only authenticated key exchange", SIGCOMM Computer Communication Review, vol. 26, no. 5, pp. 5-26.

[12] Halevi S and Krawczyk H (1999) "Public key cryptography and password protocols", ACM Transactions on Information and System Security, pp. 524-543.

[13] Boyarsky M (1999) "Public-key cryptography and pass-word protocols: The multi-user case", ACM Security (CCS'99), pp. 63-72.

[14] Boyko V, MacKenzie P and Patel S (2000) "Provably secure password-authenticated key exchange using Diffie-Hellman", Eurocrypt 2000, LNCS 1807, pp. 156-171, Springer-Verlag.

[15] Bresson E, Chevassut O and Pointcheval D (2004) "New security results on encrypted key exchange," in PKC 2004, LNCS 2947, pp. 145-158, Springer-Verlag.

[16] Lee S, Kim W, Kim H and Yoo K (2004) "Efficient Password-based Authenticated Key Agreement Protocol", In ICCSA, LNCS, 3046: 617-626.

[17] Kwon T, "Practical Authentication Key Agreement Using Passwords", ISC 2004, LNCS, 3255:1-12, 2004.

[18] Yoon E and Yoo K (2005) "New Efficient Simple Authenticated Key Agreement Protocol", COCOON 2005, LNCS, 3595: 945-954.

[19] Strangio M (2006) "An Optimal Round Two-Party Password-Authenticated Key Agreement Protocol", the First International Conference on Availability, Reliability and Security, p. 8.

[20] Bellovin S and Merritt M (2007) “Augmented encrypted key exchange: A password-based protocol secure against dictionary attacks and password file", International Journal of Network Security, Vol.3, No.1, PP.23-34.

[21] YShakir M. Hussain and Hussein Al-Bahadili (2008) "A non-exchanged password scheme for password-based authentication in client-server systems", American Journal of Applied Sciences, 2008.

[22] SeongHan Shin, Kazukuni Kobara and Hideki Imai (2009) “A Secure Threshold Anonymous Password Authenticated Key Exchange Protocol”, Crypto 2009, LNCS, Springer-Verlag.

