Security Weakness in a Three-Party Password-Based Key Exchange Protocol Using Weil Pairing

Junghyun Nam † Seungjoo Kim †‡ Dongho Won †§

August 15, 2005

Abstract

Recently, Wen, Lee, and Hwang proposed a three-party password-authenticated key exchange protocol making use of the Weil pairing. The protocol was claimed to be provably secure. But despite the claim of provable security, the protocol is in fact insecure in the presence of an active adversary. We demonstrate this by presenting an attack that completely compromises the authentication mechanism of the protocol. Consequently, the proof of security for the protocol is invalidated.

Keywords: Key exchange protocol, password-based authentication, Weil pairing.

1 Introduction

Bellovin and Merritt [3] was the first to consider how two parties, who only share a weak, low-entropy password, and who are communicating over an untrusted, public network, authenticate each other and agree on a high-entropy cryptographic key to be used for protecting their subsequent communication. Due in large part to the practical significance of password-based authentication, this initial work has been followed by a number of protocol proposals (e.g., [6, 2, 10, 8]) offering various levels of security and complexity.

While two-party protocols for password-authenticated key exchange (PAKE) are well suited for client-server architectures, they are inconvenient and costly for use in large scale peer to peer systems. Since two-party PAKE protocols require each pair of communication users to share a password, a large number of users results in an even larger number of potential passwords to be shared. It is due to this problem that three-party models have been often used in designing PAKE protocols [7, 12, 11, 1]. In a typical three-party setting, users do not need to remember and manage multiple passwords (one for each communicating party); rather, each user shares a single password with a trusted server who then assists users in establishing a session key by providing authentication services to them. However, this convenience comes at the price of users' complete trust in the server. Therefore, whilst the three-party model will not replace the two-party model, it offers easier alternative solutions to the problem of password-authenticated key exchange in a peer to peer system.

Recently, Wen, Lee, and Hwang [13] proposed a three-party PAKE protocol making use of the Weil pairing. Their approach seems to be quite decent and promising when one considers the contribution of the Weil/Tate pairings in constructing Joux's one round tripartite key agreement protocol [9]. Furthermore, their protocol was claimed to be provably secure in the random oracle model under a certain intractability assumption (see Section 2.1). But despite a claimed proof of security, the Wen-Lee-Hwang protocol is in fact not secure in the presence of an active adversary.

†Department of Computer Engineering, Sungkyunkwan University, 300 Cheoncheon-dong, Jangan-gu, Suwon-si, Gyeonggi-do 440-746, Republic of Korea.
E-mail: jhnam@dosan.skku.ac.kr, skim@ece.skku.ac.kr, dhwon@dosan.skku.ac.kr
‡WWW: http://dosan.skku.ac.kr/∼sjkim/
§WWW: http://dosan.skku.ac.kr/∼dhwon/

In this work, we demonstrate this by exhibiting an attack that completely compromises the authentication mechanism of the protocol. This might be seen as a paradox: How can a protocol that was proven secure later be found insecure? Our answer to this question is that the security proof of the protocol was flawed.

2 Review of the Wen-Lee-Hwang protocol

In this section, we first recall some definitions and notations and then describe the Wen-Lee-Hwang protocol.

2.1 Preliminaries

Let $\mathbb{G}_1$ be an additive group of order q for some large prime q and $\mathbb{G}_2$ be a multiplicative group of the same order q. The protocol is built making use of the modified Weil pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ which satisfies the following properties:

- *Bilinear.* $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$.
- *Non-degenerate.* There exists a $P \in \mathbb{G}_1$ such that $\hat{e}(P, P) \neq 1$.
- *Computable.* There is an efficient algorithm to compute $\hat{e}(P, Q)$ for any $P, Q \in \mathbb{G}_1$.

The security of the protocol is based on a variant of the computational Diffie-Hellman (CDH) assumption called the Weil Diffie-Hellman (WDH) assumption. Informally, the WDH assumption states that given a generator P of $\mathbb{G}_1$ and a triple $\langle aP, bP, cP\rangle$ for random $a, b, c \in \mathbb{Z}_q^*$, it is computationally intractable to compute $\hat{e}(P, P)^{abc} \in \mathbb{G}_2$. Here we simply assume the existence of $\mathbb{G}_1$, $\mathbb{G}_2$ and $\hat{e}$ that meet the three properties and the WDH assumption. Readers are referred to [4] for details on how in practice to choose the groups and to define and compute the modified Weil pairing.

One special primitive used in the protocol is a cryptographic one-way hash function G which maps an arbitrary string to an element of $\mathbb{G}_1$. G can be constructed from a typical one-way hash function H in several ways, as indicated in [5] and [4].

2.2 Protocol Description

There are three entities involved in the protocol: the authentication server S, and two users A and B who wish to establish a session key between them. We denote by ID_S, ID_A and ID_B the identities of S, A and B, respectively. Let P be a fixed generator of $\mathbb{G}_1$. The server S chooses as its secret key a random $s \in \mathbb{Z}_q^*$ and computes its public key P_S as $P_S = sP$. Let PW_A and PW_B be the passwords of A and B, respectively. Each user's password is securely shared with the server S. In describing the protocol, we assume that the public parameters $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, q, P, P_S, H, G\rangle$ have been fixed in advance and are known to all parties in the network. A high-level depiction of the protocol is given in Fig. 1, and a more detailed description follows:

1. User A chooses a random number $a \in \mathbb{Z}_q^*$ and computes aP and $k_a = H(aP\|P_S\|Q\|\hat{e}(P_S, aQ))$, where $Q = G(ID_S)$. Then A computes $c_a = \mathcal{E}_{k_a}(PW_A)$, where $\mathcal{E}_{k_a}(PW_A)$ is a symmetric encryption of PW_A under key k_a, and sends $\langle ID_A, aP, c_a\rangle$ to user B.
2. User B chooses a random number $b \in \mathbb{Z}_q^*$ and computes bP, $k_b = H(bP\|P_S\|Q\|\hat{e}(P_S, bQ))$ and $K = \hat{e}(aP, bU)$, where $U = G(ID_A\|ID_B)$. Then B computes $c_b = \mathcal{E}_{k_b}(PW_B)$ * and $\mu_b = H(ID_B\|K)$ and sends $\langle ID_A, aP, c_a, ID_B, bP, c_b, \mu_b\rangle$ † to server S.

* $\mathcal{E}_{k_b}(PW_B)$ was incorrectly stated as $\mathcal{E}_{k_a}(PW_B)$ in the fourth line of Step 2 described in Section 3 of [13]. We have corrected this typographical error.

† $\langle ID_A, aP, c_a, ID_B, bP, c_b, \mu_b\rangle$ was incorrectly specified as $\langle ID_A,\ aP,\ c_a,\ bP,\ c_b,\ \mu_b\rangle$ in the first line of Step 2 described in Section 3 of [13]; i.e., ID_B was inadvertently omitted in the message specification. This omission has been corrected to be consistent with the last sentence of the same step.

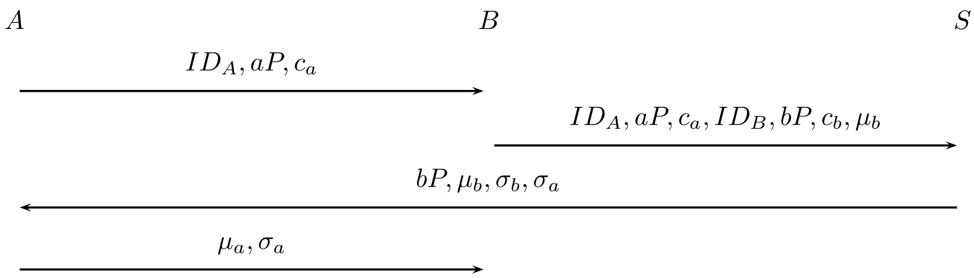

Figure 1: A high-level description of the Wen-Lee-Hwang protocol

3. S computes $k_a = H(aP\|P_S\|Q\|\hat{e}(aP, sQ))$ and $k_b = H(bP\|P_S\|Q\|\hat{e}(bP, sQ))$ to decrypt the ciphertexts c_a and c_b. S then verifies that each user's decrypted password is correct. If any of two passwords is wrong, S stops executing the protocol. Otherwise, S computes $\sigma_a = H(k_b\|aP)$ and $\sigma_b = H(k_a\|bP)$ and sends $\langle bP,\ \mu_b,\ \sigma_b,\ \sigma_a\rangle$ [‡] to user A.

4. User A computes $K = \hat{e}(bP, aU)$ and verifies that σ_b equals $H(k_a\|bP)$ and μ_b equals $H(ID_B\|K)$. If the verification fails, A aborts the protocol. Otherwise, A computes $\mu_a = H(ID_A\|K)$ and sends $\langle \mu_a, \sigma_a\rangle$ to user B. After that, A computes the session key SK as $SK = H(aP\|bP\|U\|K)$.

5. Upon receiving $\langle \mu_a, \sigma_a\rangle$, user B verifies that σ_a equals $H(k_b\|aP)$ and μ_a [§] equals $H(ID_A\|K)$. If the verification succeeds, B computes the session key $SK = H(aP\|bP\|U\|K)$; otherwise, aborts the protocol.

3 Attack on the Wen-Lee-Hwang Protocol

Unfortunately, the Wen-Lee-Hwang protocol is insecure in the presence of an active adversary. To show this, we present an attack that exploits an authentication flaw in the protocol. We assume that the adversary M is a legitimate user registered with the authentication server S and thus is able to set up normal protocol sessions with other users. Let PW_M denote M's password shared with the server S. In the attack, the goal of adversary M is to share a key with A by masquerading as B and to share another key with B by masquerading as A. The attack scenario is outlined in Fig. 2, where a dashed line indicates that the corresponding message is intercepted by M en route to its destination. A more detailed description of the attack is as follows:

1. As a preliminary step, the adversary M chooses two random numbers $m, m' \in \mathbb{Z}_q^*$ and computes mP, $m'P$, $k_m = H(mP\|P_S\|Q\|\hat{e}(P_S, mQ))$, $k'_m = H(m'P\|P_S\|Q\|\hat{e}(P_S, m'Q))$, $c_m = \mathcal{E}_{k_m}(PW_M)$ and $c'_m = \mathcal{E}_{k'_m}(PW_M)$, where $Q = G(ID_S)$.

2. When A initiates the protocol execution with the first message $\langle ID_A, aP, c_a\rangle$, M intercepts this message and instead sends $\langle ID_A, mP, c_a\rangle$ to B as if it originated from A. M then computes K' as
$$K' = \hat{e}(aP, m'U),$$
where $U = G(ID_A\|ID_B)$, and μ'_m as
$$\mu'_m = H(ID_B\|K').$$

[‡] $\langle bP, \mu_b, \sigma_b, \sigma_a\rangle$ was incorrectly specified as $\langle bP, \mu_b, \mu_b, \sigma_b, \sigma_a\rangle$ in the first line of Step 3 described in Section 3 of [13]; i.e., μ_b was inadvertently duplicated in the message specification. This duplication has been removed to be consistent with the last sentence of the same step.

[§] Due to a typographical error, μ_a was incorrectly appeared as σ_a in the fourth-to-last line of Section 3 of [13].

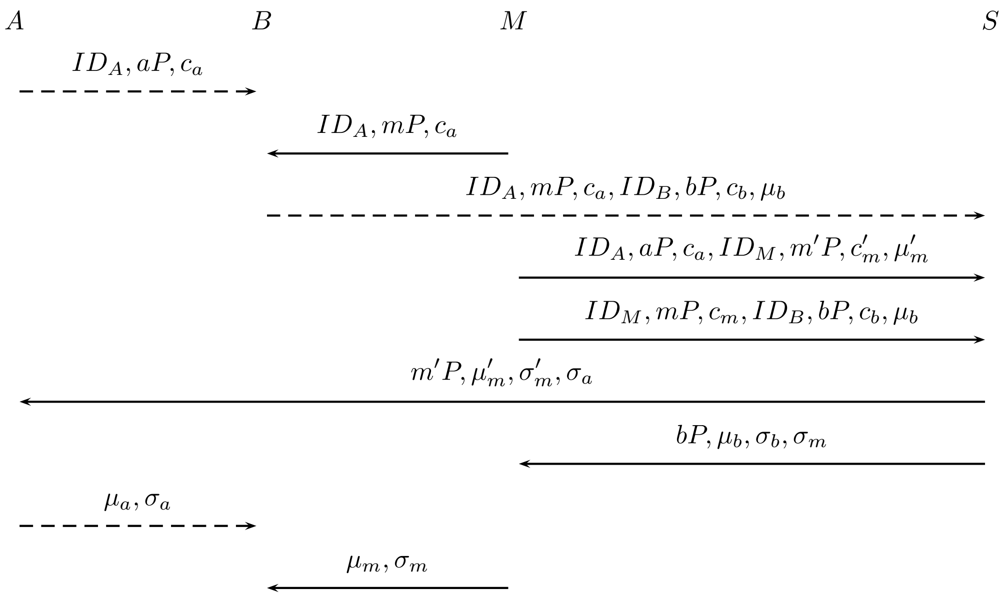

Figure 2: An attack on the Wen-Lee-Hwang protocol

3. Since, from B's point of view, mP is perfectly indistinguishable from aP of an honest execution, B believes that the message $\langle ID_A, mP, c_a \rangle$ is from A. Hence, B operates as specified in the protocol using mP in place of aP. First, B chooses a random number b and computes bP, k_b, K, c_b and μ_b. Note that K is computed as

$$K = \hat{e}(mP, bU),$$

while all other computations are done exactly as specified in the protocol. Next, B sends the message $\langle ID_A,\ mP,\ c_a,\ ID_B,\ bP,\ c_b,\ \mu_b \rangle$ to server S. But, this message is intercepted by M.

4. Using the intercepted message $\langle ID_A,\ mP,\ c_a,\ ID_B,\ bP,\ c_b,\ \mu_b \rangle$ and the values computed in previous steps, M forges two separate messages:

$$\begin{aligned} msg_{m1} &\stackrel{\text{def}}{=} \langle ID_A, aP, c_a, ID_M, m'P, c'_m, \mu'_m \rangle \text{ and} \\ msg_{m2} &\stackrel{\text{def}}{=} \langle ID_M, mP, c_m, ID_B, bP, c_b, \mu_b \rangle. \end{aligned}$$

M then sends to S the first forged message msg_{m1} alleging that it is for establishing a session key between A and herself. M also sends to S the second forged message msg_{m2} as if it originated from B who wants to establish a session key with M.

5. The forged messages will pass the verification test of S since all the decryptions of the ciphertexts c_a, c'_m, c_m and c_b will be successful producing a correct password. This is clear because, for example, S will compute the one-time key k'_m as $k'_m = H(m'P \| P_S \| Q \| \hat{e}(m'P, sQ))$ and decrypt c'_m using the key k'_m under which M's password PW_M was encrypted into c'_m.

6. Since msg_{m1} and msg_{m2} are both valid, everything proceeds as usual. In response to msg_{m1}, S computes $\sigma_a = H(k'_m \| aP)$ and $\sigma'_m = H(k_a \| m'P)$ and sends

$$msg_{s1} \stackrel{\text{def}}{=} \langle m'P, \mu'_m, \sigma'_m, \sigma_a \rangle$$

to A. Further, in response to msg_{m2}, S computes $\sigma_m = H(k_b \| mP)$ and $\sigma_b = H(k_m \| bP)$ and sends to M

$$msg_{s2} \stackrel{\text{def}}{=} \langle bP, \mu_b, \sigma_b, \sigma_m \rangle.$$

7. It is easy to see that A is unable to detect any discrepancy on msg_{s1}. First, the verification that μ'_m is equal to $H(ID_B \| \hat{e}(m'P, aU))$ will succeed since μ'_m was computed as $H(ID_B \| K')$ where $K' = \hat{e}(aP, m'U)$. Second, it is obvious that σ'_m is equal to $H(k_a \| m'P)$. Therefore, after the verifications are done, A computes $\mu_a = H(ID_A \| K')$ and sends $\langle \mu_a, \sigma_a \rangle$ to B. But, this message is intercepted by M. Finally, A computes the session key $SK' = H(aP \| m'P \| U \| K')$ without noticing that the same key is also available to M.

8. Meanwhile, adversary M upon receiving msg_{s2} from S, computes $K = \hat{e}(bP, mU)$ and $\mu_m = H(ID_A \| K)$ and sends $\langle \mu_m, \sigma_m \rangle$ to B as if it originated from A. Since σ_m equals $H(k_b \| mP)$ and μ_m equals $H(ID_A \| K)$, B will be unaware of the attack and will compute the session key $SK = H(mP \| bP \| U \| K)$ that is also known to M.

Through the attack, the authentication mechanism of the protocol is completely compromised. Indeed, the effect of our attack is the same as that of a man-in-the-middle attack. At the end of the scenario, the user A believes that she has established a secure session with B sharing a secret key SK', while in fact she has shared the key with M. Similarly, B thinks that he has shared with A a session key SK which indeed is shared with M. As a result, the adversary M can not only access and relay any confidential communications between A and B, but can also send arbitrary messages for her own benefit impersonating one of them to the other.

4 Discussion

Our attack demonstrates that the claim of provable security for the Wen-Lee-Hwang protocol was incorrect. Indeed, we found a significant gap in the reasoning of the proof given in Appendix of [13]. The proof implicitly assumed that the active adversary $\mathcal{A}$, who breaks the security of the protocol without breaking the password security, controls communication flows in a way that a session key is computed only from a pair of aP and bP returned as the response to a **Send** query. Under this assumption, they argue that an algorithm ω which breaks the WDH assumption can be constructed using $\mathcal{A}$ as a subroutine. This implicit assumption was never adequately explained or justified in the proof. Actually, as we have seen through the attack, this assumption turns out to be wrong; an active adversary is easily able to trick the users into computing their session key from an unintended pair of values, i.e., $\langle aP, m'P \rangle$ or $\langle mP, bP \rangle$.

It is not clear how to modify the protocol to make it achieve any form of provable security. Having seen our attack, one may suggest to integrate all the identities of protocol participants into the computation of k_a and k_b as part of the hash input. Although this modification seems to defeat our attack, there is no guarantee that it will prevent other potential attacks not identified here; provable security is claimed against all attacks, not just against known attacks.

References

[1] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," *PKC 2005*, LNCS 3386, pp. 65–84, 2005.

[2] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," *Eurocrypt 2000*, LNCS 1807, pp. 139–155, 2000.

[3] S. M. Bellovin and M. Merritt, "Encrypted key exchange: password-based protocols secure against dictionary attacks," *Proceedings of IEEE Symposium on Research in Security and Privacy*, pp. 72–84, 1992.

[4] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2003.

[5] D. Boneh, B. Lynn, and H. Shacham, "Short signtures from the Weil pairing," *Asiacrypt 2001*, LNCS 2248, pp. 514–532, 2001.

[6] V. Boyko, P. MacKenzie, and S. Patel, "Provably secure password-authenticated key exchange using Diffie-Hellman," *Eurocrypt 2000*, LNCS 1807, pp. 156–171, 2000.

[7] L. Gong, M. L. Lomas, R. M. Needham, and J. H. Saltzer, "Protecting poorly chosen secrets from guessing attacks," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 5, pp. 648–656, 1993.

[8] IEEE P1363.2: Password-Based Public-Key Cryptography, http://grouper.ieee.org/groups/1363/passwdPK/

[9] A. Joux, "A one round protocol for tripartite Diffie-Hellman," *Journal of Cryptology*, vol. 17, no. 4, pp. 263–276, 2003.

[10] J. Katz, R. Ostrovsky, and M. Yung, "Efficient password-authenticated key exchange using human-memorable passwords," *Eurocrypt 2001*, LNCS 2045, pp. 475–494, 2001.

[11] C.-L. Lin, H.-M. Sun, and T. Hwang, "Three-party encrypted key exchange: attacks and a solution," *ACM SIGOPS Operating Systems Review*, vol. 34, no. 4, pp. 12–20, 2000.

[12] M. Steiner, G. Tsudik, and M. Waidner, "Refinement and extension of encrypted key exchange," *ACM SIGOPS Operating Systems Review*, vol. 29, no. 3, pp. 22–30, 1995.

[13] H.-A. Wen, T.-F. Lee, and T. Hwang, "Provably secure three-party password-based authenticated key exchange protocol using Weil pairing," *IEE Proceedings — Communications*, vol. 152, no. 2, pp. 138–143, 2005.