Cryptanalytic Flaws in Oh et al.'s ID-Based Authenticated Key Agreement Protocol

Meng-Hui Lim
Department of Ubiquitous IT,
Graduate School of Design and IT,
Dongseo University,
Busan, 617-716, Korea
menghui.lim@gmail.com

Sanggon Lee, Hoonjae Lee
Department of Information and Communication,
Dongseo University,
Busan 617-716, Korea
{nok60, hjlee}@dongseo.ac.kr

Abstract

A key agreement protocol is designed for two or more entities to agree upon a shared secret key, which is used to preserve confidentiality and data integrity over an open network. In 2007, Oh et al. proposed an efficient ID-based authenticated key agreement protocol on elliptic curve pairings, which is believed to be able to generate two session keys securely after a protocol execution. However, we discover that their protocol is in fact susceptible to the basic impersonation attack as well as the key compromise impersonation attack. In this paper, we present the imperfections of Oh et al.'s scheme and subsequently we suggest a slight modification to the scheme which would resolve the problems.

1. Introduction

Key agreement protocols are essential in secure communications establishment across an insecure network. Specifically, two commmunication parties would exchange some public information in prior to create a mutual secret key that is known only to themselves. A secure key agreement protocol should be designed in such a way that no single protocol entity can predetermine or predict the secret key. This secret key (also known as a *session key*) if derived securely, can subsequently be used to create a confidential or integrity-protected communication channel among the entities.

Designing a secure key agreement protocol is not an easy task. A secure key agreement protocol would need to withstand both passive and active attacks which would most probably be launched by the adversaries when the messages are exchanged in a public channel. In the former case, it is assumed that the adversary, from time to time, eavesdrops on all the messages exchanged by the honest parties. Such an attack enables the adversaries to perform offline analysis on the captured data and extract any useful information whenever is possible. On contrary, the latter case involves a more powerful adversary who is assumed to be able to intercept, delete, inject, modify and replay messages in any online instance of the key agreement protocol. These attacks are usually preceived to be the capability of an outsider adversary in a two-party communication process.

To capture the notion of security, Wilson and Menezes [10, 11] have defined a number of desirable security attributes which can be used to analyze a key agreement protocol. These security attributes are described as follows:

Known session key security. A protocol is considered to be *known session key secure* if it remains achieving its goal in the face of an adversary who has learned some previous session keys.

(Perfect) forward secrecy. A protocol enjoys *forward secrecy* if the secrecy of the previous session keys is not affected when the long term private keys of one or more entities are compromised. *Perfect forward secrecy* refers to the scenario when the long term private keys of all the participating entities are compromised.

Key-Compromise Impersonation Resilience. Suppose that a protocol entity A's long term private key has been disclosed. Obviously an adversary who knows this value can now impersonate A since it is precisely the value which identifies A. We say that a protocol is *key-compromise impersonation resilient* if this loss would not enable an adversary to masquerade as other legitimate entities to A as well or obtain other entities' secret key.

Unknown Key-Share Resilience. In an unknown key-share attack, an adversary convinces a group of entities that they share a session key with the adversary

whereas in fact, the key is shared between the group and another party. This situation can be exploited in a number of ways by the adversary when the key is subsequently used to provide encryption or integrity.

Key Control Resilience. It should not be possible for any of the participants (or an adversary) to coerce the session key to a preselected or predicted value.

Since Boneh and Franklin [1] have initiated the development of identity(ID)-based encryption scheme using the notion of bilinear mapping based on elliptic curve pairings, many ID-based key agreement protocols have been proposed thereafter. In 2002, Smart [8] has proposed a notable ID-based authenticated key agreement protocol, which combines the ideas from Boneh-Franklin's work [1] and Joux's tripartite protocol [3]. However, this protocol does not provide the forward secrecy attribute, as pointed out by Shim in [7]. Shim has then proposed an alternative ID-based authenticated key agreement protocol and claimed it to be efficient and capable of satisfying most of the desired security features. Unfortunately, Sun-Hsieh [9] have demonstrated a valid man-in-the-middle attack on Shim's scheme few months later which renders her protocol to be insecure. In 2004, Ryu et al. [6] have proposed another efficient ID-based scheme, which requires every protocol participant to perform only one pairing computation and two point-multiplications in a protocol execution. Nevertheless, Boyd-Choo [2] and Yuan-Li [12] have discovered the insecurity of Ryu et al.'s scheme against the key compromise impersonation attack and the key reveal attack. Recently, Oh et al. [5] have proposed a new ID-based protocol which is able to generate two session keys in a protocol run. They claimed that their scheme is more secure and efficient as compared to Shim's protocol [7], Ryu et al.'s protocol [6] as well as Yuan-Li.'s protocol [12]. However, we strongly disagree with their claim as we have identified some fatal flaws in their scheme which would in fact endanger the protocol participants if it is employed.

Hence, the main purpose of this paper is to demonstrate the impersonation attacks on Oh et al.'s scheme. Besides that, we also suggest a slight modification to the original scheme so as to counter the defects. The structure of this paper is generally organized as follows. In the next section, we will illustrate the basic properties of bilinear pairings and some underlying assumptions explicitly. In Section 3, we will review Oh et al.'s ID-based authenticated key agreement protocol. In Section 4, we will demonstrate the impersonation attacks on their scheme. We will then illustrate our improvement in Section 5 and the subsequent security analysis in Section 6. Last but not least, we will conclude this paper in Section 7.

2 Preliminaries

Let $\mathbf{G}_1$ be an additive group of a large prime order, q and $\mathbf{G}_2$ be a multiplicative group of the same order, q. Let $P, Q \in \mathbf{G}_1$ and $\hat{e} : \mathbf{G}_1 \times \mathbf{G}_1 \longrightarrow \mathbf{G}_2$ be a bilinear pairing with the following properties:

- **Bilinearity**: $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab} = \hat{e}(abP, Q)$ for any $a, b \in Z_q^*$.
- **Non-degeneracy**: $\hat{e}(P, Q) \neq 1$.
- **Computability**: There exists an efficient algorithm to compute $\hat{e}(P, Q)$.

A bilinear map which satisfies all three properties above is considered as *admissible bilinear*. It is noted that the Weil and Tate pairings associated with the supersingular elliptic curves or abelian varieties, can be modified to create such bilinear maps. Now, we describe some cryptographic problems:

Bilinear Diffie-Hellman Problem (BDHP). Let $\mathbf{G}_1$, $\mathbf{G}_2$, P and $\hat{e}$ be as above with the order q being prime. Given $\langle P, aP, bP, cP \rangle$ with $a, b, c \in Z_q^*$, compute $\hat{e}(P, P)^{abc} \in \mathbf{G}_2$. An algorithm α is deemed to have an advantage ϵ in solving the BDHP in $\langle \mathbf{G}_1, \mathbf{G}_2, \hat{e} \rangle$ based on the random choices of a, b, c in Z_q^* and the internal random operation of α if

$$Pr[\alpha(\langle P, aP, bP, cP \rangle) = \hat{e}(P, P)^{abc}] \geq \epsilon.$$

Computational Diffie-Hellman Problem (CDHP). Let P be an element of $\mathbf{G_1}$ as above. Given $\langle P, aP, bP \rangle$ with $a, b \in Z_q^*$, compute $abP \in \mathbf{G}_1$.

Discrete Logarithm Problem (DLP). Given two groups of elements P and Q, such that $Q = nP$. Find the integer n whenever such an integer exists.

For the remainding sections of this paper, we assume that BDHP, CDHP and DLP are hard such that there is no polynomial time algorithm to solve these cryptographic problems with non-negligible probability.

3 Review of Oh et al.'s Scheme

In this section, we revisit Oh et al's ID-based key agreement protocol [5], which consists of 3 phases, namely *Setup* phase, *Extract* phase and *Key Agreement* phase. We now describe each phases in detail.

Setup. Let E be a super-singular curve defined by $y^2 = x^3 + 1$ over F_p. The Key Generation Center (KGC) inputs a security parameter k into a BDH parameter generator $\mathcal{G}$, which returns two groups $\mathbf{G}_1$ and $\mathbf{G}_2$ of

prime order q, a generator element P of $\mathbf{G}_1$, a bilinear map $\hat{e} : \mathbf{G}_1 \rightarrow \mathbf{G}_2$. KGC then chooses a random oracle $H : \{0,1\}^* \rightarrow \mathbf{G}_1$, selects a master key $s \in Z_q^*$, computes the master public key $P_{pub} = sP$ and publishes the system parameters $\{p, q, \mathbf{G}_1, \mathbf{G}_2, E, P, P_{pub}, H, \hat{e}\}$ while keeping s secret.

Extract. For an entity with identity information $ID \in \{0,1\}^*$, the public key is given as $Q_{ID} = H(ID)$. The KGC computes the corresponding private key as $S_{ID} = sQ_{ID}$ and issues S_{ID} to the entity via a secure channel. Hence, an ID-based key pair is defined as (Q_{ID}, S_{ID}), where $Q_{ID}, S_{ID} \in \mathbf{G}_1$.

Key Agreement. Suppose that two communicating entities, A and B wish to establish a session key by carrying out an instance of the protocol run. We denote A and B's public/private key pair to be Q_A/S_A and Q_B/S_B respectively. Then, the key exchange can be performed as follows:

1. A selects a random number $a \in Z_q^*$, computes $T_A = aP$ and sends T_A to B. In a symmetrical manner, B selects a random number $b \in Z_q^*$, computes $T_B = bP$ and sends T_B to A.
2. Upon receiving T_B, A computes the shared secrets

$$K_{AB} = \hat{e}(aQ_B, P_{pub}) = \hat{e}(Q_B, P)^{as} \quad (1)$$

and

$$K'_{AB} = \hat{e}(S_A, T_B) = \hat{e}(Q_A, P)^{bs}. \quad (2)$$

After receiving T_A, B computes the shared secrets

$$K_{BA} = \hat{e}(S_B, T_A) = \hat{e}(Q_B, P)^{as} \quad (3)$$

and

$$K'_{BA} = \hat{e}(bQ_A, P_{pub}) = \hat{e}(Q_A, P)^{bs}. \quad (4)$$

3. Subsequently, A and B compute two shared session keys, which are

$$\begin{aligned} K &= kdf(ID_A, ID_B, aT_B, K_{AB}) \\ &= kdf(ID_A, ID_B, bT_A, K_{BA}) \end{aligned} \quad (5)$$

and

$$\begin{aligned} K' &= kdf(ID_A, ID_B, aT_B, K'_{AB}) \\ &= kdf(ID_A, ID_B, bT_A, K'_{BA}), \end{aligned} \quad (6)$$

where kdf is a key derivation function.

The authors claimed this protocol to be more efficient and secure than Shim's, Ryu et al.'s and Yuan et al.'s protocols by demonstrating a heuristic security and efficiency analysis. However, we discover that this is not the case where their protocol cannot withstand the basic impersonation attack and key compromise impersonation attack, and we will demonstrate these attacks in the next section.

4 Cryptanalysis on Oh et al.'s Scheme

In this section, we present two impersonation attacks on Oh et al.'s scheme in detail: the basic impersonation attack and the key compromise impersonation attack.

4.1 Basic impersonation Attack

Similar to Joux's protocol [3], Oh et al.'s does not provide adequate authentication to prevent any unauthorized entity from establishing a session key with the legal entity. Indeed, as what Oh et al. desire, their scheme is able to produce two session keys efficiently at the end of a protocol run. However, they did not manage to secure both of the keys simultaneously. Eventually, one of the keys will be exposed to a malicious outsider adversary who impersonates another legal entity and engage with the target entity in a protocol execution. We now provide the details of the basic impersonation attack as follows:

Suppose that an adversary E, pretending as B, wants to establish a protocol run with A.

1. Initially, A selects a random number $a \in Z_q^*$, computes $T_A = aP$ and sends T_A to B. E intercepts the message, selects a random number $e \in Z_q^*$, computes $T_B^\star = eP$ and sends T_E to A on behalf of B.
2. Upon receiving $T_B^\star$, A computes the shared secrets

$$K_{AB} = \hat{e}(aQ_B, P_{pub}) = \hat{e}(Q_B, P)^{as} \quad (7)$$

and

$$K'_{AB} = \hat{e}(S_A, T_B^\star) = \hat{e}(Q_A, P)^{es}. \quad (8)$$

Then, as per protocol specification, A computes two shared session keys, which are

$$\begin{aligned} K &= kdf(ID_A, ID_B, aT_B^\star, K_{AB}) \\ &= kdf(ID_A, ID_B, aeP, \hat{e}(aQ_B, P_{pub})) \end{aligned} \quad (9)$$

and

$$\begin{aligned} K' &= kdf(ID_A, ID_B, aT_B^\star, K'_{AB}) \\ &= kdf(ID_A, ID_B, aeP, \hat{e}(S_A, eP)). \end{aligned} \quad (10)$$

3. Note that E does not know a, S_A and S_B. Hence, she does not have the knowledge in computing

$$K_{BA} = \hat{e}(S_B, T_A) = \hat{e}(aQ_B, P_{pub}) = K_{AB}.$$

However, she is able to calculate

$$K'_{BA} = \hat{e}(eQ_A, P_{pub}) = \hat{e}(Q_A, P)^{es} \quad (11)$$

which is equivalent to K'_{AB} in Eq. (8). E then proceeds to calculate

$$\begin{aligned} K' &= kdf(ID_A, ID_B, eT_A, K'_{BA}) \\ &= kdf(ID_A, ID_B, aeP, \hat{e}(eQ_A, P_{pub})). \end{aligned} \quad (12)$$

Hence, E has successfully agreed on the second session key (K') with A by masquerading as B. Despite of being more efficient, Oh et al.'s scheme has failed to resist the basic impersonation attack, which may further results in other cryptanalytic attacks. In fact, this attack can also be carried out in a similar manner when E intends to cheat B by masquerading as A. If that is the case, E would be able to agree upon the first session key

$$K = kdf(ID_A, ID_B, aeP, \hat{e}(eQ_B, P_{pub})) \quad (13)$$

with A. Thus, we can conclude that Oh et al.'s scheme is insecure against the basic impersonation attack.

4.2 Key Compromise Impersonation Attack

Key Compromise Impersonation (KCI) attack is a kind of known-key attack, which can be performed by an adversary after compromising a protocol entity's private key. We often refer such entity as *corrupted*. Note that in practice, a malicious party may hack into an entity's computer in order to acquire the entity's private key, and even worse, the corrupted entity may be unaware of this intrusion. By learning this secret key, the adversary is now able to impersonate the corrupted party directly in communicating with other protocol entities. However, this is often not our interest to delve into it in a KCI analysis. On contrary, the general adversarial goal in the KCI attack is to impersonate any other legitimate party or parties and establish a common session key with the corrupted party in a protocol run.

In [5], Oh et al. have scrutinized the KCI resilience of their scheme and subsequently conjecture their scheme to be KCI secure. However, they only restrict their analysis in considering the adversary to be a passive eavesdropper, and thus they claim that the adversary is unable to compute the parameter $aT_B = bT_A = abP$ (an instance of CDHP), which is required in computing the session keys K and K'. Regrettably they have left out the most important scenario in their analysis where the adversary would impersonate any legal entity to perform the attack. Recall that in Section 4.1, we have proven Oh et al.'s scheme to be susceptible to the basic impersonation attack. In the current KCI analysis where the adversary has acquired an additional secret key of the target entity, it is easy to see that their scheme cannot resist the key compromise impersonation attack as well since the adversary can employ the same strategy as in Section 4.1 to compute one of the session keys in the KCI attack. This significantly violates the KCI security of a key agreement protocol, and renders their scheme to be KCI insecure.

5 Improvements on Oh et al.'s scheme

In the attacks which we have demonstrated previously, the adversary can only compute a shared secret out of two in each protocol execution. It is apparent that if we use the shared secrets separately to derive two distinct session keys, one of them will eventually be exposed. Hence, in order to secure the scheme, one direct way is to utilize the two shared secrets to derive only a session key, so that the adversary would not be able to compute it by learning merely a shared secret. With this, we suggest a minor modification to the session key computation of the scheme, where

$$\begin{aligned} K &= kdf(ID_A, ID_B, abP, K_{AB}, K'_{AB}) \\ &= kdf(ID_A, ID_B, abP, K_{BA}, K'_{BA}). \end{aligned} \quad (14)$$

We now provide a general overview of the enhanced protocol as follows:

1. Message Exchange:
   $A \rightarrow B : T_A = aP.$
   $B \rightarrow A : T_B = bP.$

2. Shared Secrets Computation:
   $A : K_{AB} = \hat{e}(aQ_B, P_{pub}), K'_{AB} = \hat{e}(S_A, T_B).$
   $B : K_{BA} = \hat{e}(S_B, T_A), K'_{BA} = \hat{e}(bQ_A, P_{pub}).$

3. Session Key Computation:
   $A : K = kdf(ID_A, ID_B, aT_B, K_{AB}, K'_{AB}).$
   $B : K = kdf(ID_A, ID_B, bT_A, K_{BA}, K'_{BA}).$

6 Security Analysis of the Improved Scheme

Now, we scrutinize the security of our improvement on Oh et al.'s scheme to ensure that the flaws have been overcome while the other desired security attributes are preserved.

Known session key security. In every protocol run, the ephemeral private key (a, b) that each protocol entity chooses would vary. Since these ephemeral values are involved in computing the session key of our enhanced protocol, the session key would also vary with every protocol execution. On top of that, a key deriving function (usually a one-way cryptographic hash function) is

used to derive the session key. Hence, the knowledge of some previous session keys would not allow the adversary to gain any advantage in deriving any future and other previous session keys.

Perfect Forward secrecy. Suppose that A and B's long term private key S_A and S_B have been compromised. Besides, we also assume that the adversary has obtained some previous session keys established by A and B. However, the adversary is not able to derive any other previously established session keys since the adversary does not possess their respective ephemeral private keys which are needed in constructing those session keys. Hence our enhanced protocol is able to fulfill the perfect forward secrecy property. Furthermore, if the KGC's secret s is compromised at any point, our protocol is still able to retain the confidentiality of previous session keys since the adversary is not able to compute the component abP (which is a CDHP) in recovering the session key as shown in Eq. (14). Hence, our protocol is deemed providing the KGC forward secrecy too.

Key-Compromise Impersonation Resilience. Suppose that A's static private key S_A has been compromised and the adversary wishes to impersonate B in order to establish a session with A. However, she is unable to compute the first shared secret K_{BA} in Eq. (1) or (3) as she does not know a or S_B. Notice that in our enhanced protocol, the adversary is required to derive both of the shared secrets in order to compute the session key in Eq. (14). Hence, this significantly prevents the adversary from launching her KCI attack successfully. Generally, the same situation would result when the long term key S_B is compromised (the adversary fails to compute the second shared secret K'_{AB} in this case) as our enhanced protocol is symmetric. As a result, our enhanced protocol is able to withstand the KCI attack under all circumstances.

Key Replicating Resilience The session key of our enhanced protocol is derived by using a key deriving function which takes in A and B's identity and the shared secrets. If the adversary carries out a key replicating attack on our enhanced protocol by means of choosing a random number $e \in Z_q^*$ and modifying the A's message to $e \cdot T_A$ as well as B's message to $e \cdot T_B$ during the message exchange phase, Alice's final session key $K = kdf(ID_A, ID_B, abeP, \hat{e}(Q_B, P)^{as}, \hat{e}(Q_A, P)^{bes})$ would be different from Bob's session key $K = kdf(ID_A, ID_B, abeP, \hat{e}(Q_B, P)^{aes}, \hat{e}(Q_A, P)^{bs})$ at the end of the protocol execution. With this, the adversary would not be able to force the establishment of non matching sessions to possess a common session key. As a result, if the adversary reveals A's session key, she would not be able to guess B's session key correctly with non-negligible probability and vice versa.

Basic Impersonation Resilience. Recall that our enhanced protocol mandates every protocol entity to compute both shared secrets (K_{AB} and K'_{AB} or K_{BA} and K'_{BA}) which are used subsequently to derive the session key. This has in fact prevented the adversary's attempt in launching the basic impersonation attack as described in Section 4.1 since the adversary is only able to recover one of the shared secrets. Intuitively, our enhanced protocol can resist the basic impersonation attack perfectly.

Unknown Key-Share Resilience. In our enhanced scheme, the identity of the communicating parties have been included in the session key so as to prevent the attacker from launching the unknown key-share attack in various ways on our improved protocol. Hence, a stronger sense of authentication can be achieved explicitly.

Key Control Resilience. Apparently in our protocol, no single protocol participant could force the session key to a predetermined or predicted value since the session key is derived by using both A and B's long term and ephemeral private keys.

7 Conclusion

As a conclusion, we have rebuted Oh et al's claim about the security of their protocol by mounting a basic impersonation attack and a key compromise impersonation attack on their scheme. Based on these flaws, we have subsequently suggested a slight modification to the original scheme so as to secure the scheme. Besides justifying our safeguard against the impersonation attacks, we have presented a heuristic security analysis to ensure that our enhanced scheme remains preserving all the desirable security features that the original scheme possesses.

References

[1] D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing, Advances in Cryptology-Crypto 2001, LNCS, vol. 2139, 2001, pp. 213-229.

[2] C. Boyd, K.-K.R. Choo, Security of Two-Party Identity-Based Key Agreement, Proceedings of First International Conference on Crytology in Malaysia (MyCrypt 2005), LNCS, vol.3715, 2005, pp. 229-243.

[3] A. Joux, A One-round Protocol for Tripartite Diffie-Hellman, *Proceedings of the 4th International Algorithmic Number Theory Symposium (ANTS-IV)*, LNCS, vol. 1838, July, 2000, pp. 385-394.

[4] N. McCullagh and Paulo S. L. M. Barreto, A New Two-Party Identity-Based Authenticated Key Agreement, *In Proceeding of CT-RSA 2005*, LNCS, vol. 3376, 2005, pp. 262-274.

[5] J.-B. Oh, E.-J. Yoon and K.-Y. Yoo, An Efficient ID-Based Authenticated Key Agreement Protocol with Pairings, *Parallel and Distributed Processing and Applications, In Proceeding of 5th International Symposium, ISPA 2007*, LNCS, vol. 4742, 2007, pp. 446-456.

[6] E.-K. Ryu, E.-J. Yoon and K.-Y. Yoo, An Efficient ID-Based Authenticated Key Agreement Protocol from Pairings, *NETWORKING 2004*, LNCS, vol. 3042, 2004, pp. 1458-1463.

[7] K. Shim, Efficient ID-based Authenticated Key Agreement Protocol based on Weil Pairing, *Electronics Letters*, vol. 39, no. 8, April, 2003, pp. 653-654.

[8] N.P. Smart, An Identity Based Authenticated Key Agreement Protocol Based on the Weil Pairing, *Electronics Letters*, vol. 38, 2002, pp. 630-632.

[9] H. M. Sun and B. T. Hsieh, Security Analysis of Shims Authenticated Key Agreement Protocols from Pairings, *Cryptology ePrint Archive*: Report, (113)(2003).

[10] S. B. Wilson, and A. Menezes, Authenticated Diffie-Hellman key agreement protocols, *Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC 98)*, LNCS, vol. 1999, pp. 339-361.

[11] S. B. Wilson, D. Johnson and A. Menezes, Key Agreement Protocols and their Security Analysis, *Proceedings of the 6th IMA International Conference on Cryptography and Coding*, vol. 1355, LNCS, pp. 339-361. Springer-Verlag, 1998.

[12] Q. Yuan and S.P. Li, A New Efficient ID-Based Authenticated Key Agreement Protocol, *Cryptology ePrint Archive*: Report, (309)(2005).