Provably Secure Group Key Exchange Protocol in the Presence of Dishonest Insiders

Ziba Eslami, Mahnaz Noroozi, and Saideh Kabiri Rad
(Corresponding author: Ziba Eslami)

Department of Computer Science, Shahid Beheshti University
G. C., Tehran, Iran
(Email: z_eslami@sbu.ac.ir)
(Received July 16, 2013; revised and accepted Oct. 10 & Dec. 20, 2013)

Abstract

The most important security concern in group key exchange protocols is the semantic security of the produced shared key which dictates that outsiders should not be able to learn anything about the key. It is also challenging for these protocols to retain their security even in the presence of dishonest insiders who do not follow the protocol specifications. In this paper, we propose an identity-based group key exchange protocol which addresses these security concerns. We prove that our scheme achieves semantic security in a well-known adversarial model. We then show that the success probability of recognizing dishonest insiders in the proposed scheme is almost one. We further provide a comparison between our protocol and some other schemes in terms of computation and communication cost, as well as security properties.

Keywords: Bilinear pairing, dishonest insiders, elliptic curve, group key exchange, provable security

1 Introduction

In an Internet conference the participants communicate with each other over an insecure network. In order to prevent the conference contents to be revealed, their communications must be encrypted. Therefore, the participants should agree upon a common key and use it for encrypting the messages. One solution to establish such a common key is using group key exchange protocols in which the group members compute a common key via an insecure public channel cooperatively [16, 20, 21, 36].

The first key exchange protocol was proposed in 1976 by Diffie and Hellman [15]. The scheme enabled two participants to establish a common key and its security was based on the discrete logarithm problem. But it was not suitable for groups of users. In 1982, Ingemaresson et al. [23] proposed the first group key exchange protocol. Both of these schemes were vulnerable against the man-in-the-middle attack, i.e., an adversary could impersonate the participants without being detected. So the authentication property was added to the key exchange protocols [25] and therefore, one could assure that he is establishing the key with legitimate participants [19, 30]. In other words, while security in key exchange protocols is considered against a passive adversary who only eavesdrops, in authenticated key exchange protocols, a stronger class of adversaries is involved who is capable of controlling all communication in the network.

In 1984, Shamir [32] introduced the concept of identity-based cryptosystems. In this setting, a user's private key is generated by a trusted key generator center (KGC), enabling any party to derive the user's public key from his identity and thereby removing the need for public-key certificates. An authenticated key exchange protocol is called identity-based if the users use an identity-based asymmetric key pair instead of a traditional public/private key pair for authentication and determination of the established key. Since identity-based systems simplify the process of key management (compared to the traditional public key systems), they have been considered extensively for designing key exchange protocols [11, 33, 35].

In real world, it is not reasonable to assume that all participants are honest and in fact an important issue in key exchange protocols is the presence of malicious insiders. Two types of malicious behavior are considered in the literature. One is impersonation in which the malicious participant impersonates another entity in the group who is not present [13]. Another type of malicious act which we consider in the paper as of this point, pertains to participants who prohibit the group from computing the same shared key by broadcasting fake information, i.e., values which are not produced according to the protocol specifications [18]. The word "dishonest" refers to this kind of users throughout this paper. To deal with dishonest participants, the fault-tolerance property was introduced for group key exchange protocols in 2002 [34]. This property ensures that honest participants are able to acquire a conference-key; no matter how many dishonest participants exist. Since then, some other fault-tolerant group

key exchange protocols have been proposed [22, 37].

So far and to the best of our knowledge, no formal security proof exists for the fault-tolerance property of these schemes. It is one of the goals of this paper to introduce a formal proof for security against dishonest insiders (see Section 4.1). In fact, until recently, 'informal' definitions and proofs were widely used for group key exchange protocols. Most of such definitions were originally stated for two-party protocols and then adapted to a group setting. The informal definitions serve as foundations for the subsequent formal security models for group key exchange protocols. Examples are notions like key privacy, known-key security, key freshness, forward and backward secrecy, entity authentication, unknown key-share resilience, key confirmation, and key control. For more details, the interested reader is referred to [31].

In the paradigm of provable security for key exchange protocols, a 'formal model' must be defined. In this model, the capabilities of the adversary as well as the players should be captured. It has to be clearly stated what it means for the scheme to be secure and provide a proof of its security. The security proof aims to show that the scheme actually achieves the claimed security goals under computational assumptions. The proof usually works via reduction to an underlying hard problem.

Bellare and Rogaway [3] in 1993 proposed the first computational security model for authenticated two-party key exchange protocols. Since then some other security models were proposed [1, 10]. In 2001, Bresson et al. [8] proposed the first computational (game-based) security model for group key exchange protocols (referred to as the BCPQ model) which builds on prior work from the 2-party setting [2, 3, 4]. This model has been widely used to analyze group key exchange protocols [6, 7].

In 1994, Burmester and Desmedt [9] proposed an efficient group key exchange protocol. A variant of this protocol was later proposed in 2008 by Dutta and Barua [17]. The variant has a lower communication cost and can handle the joining and leaving processes of the participants. However, this scheme has some weaknesses: (1) The authors claimed that their scheme could detect the presence of dishonest participants. However, Eslami and Kabiri in [18] designed an attack to prove that this claim was not true and showed how two dishonest insiders could prohibit legitimate participants from obtaining the same shared key; (2) It is unable to identify dishonest participants.

In this paper, we employ elliptic curves and bilinear pairings to propose a group key exchange protocol which does not suffer from these weaknesses. The advantage of elliptic curve-based cryptosystems is their short key size, high processing throughput, and low bandwidth. We prove that our proposed scheme is secure in BCPQ model. Since this model does not consider dishonest participants, we design an experiment to formally prove the security of our protocol in the presence of dishonest participants.

The rest of this paper is organized as follows: Section 2 briefly explains preliminary concepts. Our proposed protocol is described in Section 3. In Section 4, we discuss the security concepts of the proposed protocol. Section 5 is devoted to performance analysis and comparisons. Finally, a conclusion is drawn in Section 6.

2 Preliminaries

We briefly describe the preliminaries needed in the paper. First, the definition of bilinear pairings is given. Then we introduce elliptic curves. The computational problems used for security analysis are listed afterwards. Finally, we describe the security model in which the security of our group key exchange protocol is proven.

2.1 Bilinear Pairings

Let P denote a generator of G_1 where G_1 is an additive group of large order q and let G_2 be a multiplicative group. A pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ which has the following properties:

Bilinearity. For every $S, Q, R \in G_1$ we have:

$$e(S+Q,R) = e(S,R) \cdot e(Q,R).$$

$$e(S,Q+R) = e(S,Q) \cdot e(S,R).$$

Non-degeneracy. There exist $R, Q \in G_1$ such that $e(R,Q) \neq 1_{G_2}$, where 1_{G_2} is the identity element of G_2.

Computability. There exists an efficient algorithm to compute $e(R,Q) \in G_2$ for any $R, Q \in G_1$.

Note that non-degeneracy means that if $e(R,Q)$ is the identity element of G_2, then either R is the identity of G_1 or Q is the identity of G_1. (See [12], pp. 29)

2.2 Elliptic Curves

An elliptic curve defined over $GF(q)$ is given by the equation: $E : y^2 = x^3 + ax + b$, where $a, b \in GF(q)$ and $4a^3 + 27b^2 \neq 0$. The points of E (plus an infinite point O) together with a special operator "+", form a finite Abelian group.

2.3 Cryptographic Assumptions

Definition 1. *(Elliptic Curve Discrete Logarithm (ECDL) Problem) Given kA where A is a point on elliptic curve E, find k.*

The advantage of a distinguisher $\mathcal{A}$ against the $ECDL$ problem is defined as

$$adv_{\mathcal{A},E}^{ECDL} = Pr[\mathcal{A}(kA) = k].$$

Definition 2. *(Elliptic Curve Discrete Logarithm (ECDL) Assumption) Given $kA \in$ elliptic curve E, $adv_{\mathcal{A},E}^{ECDL}$ of a distinguisher $\mathcal{A}$ whose goal is to solve the $ECDL$ problem is negligible.*

Note that a negligible function f has the property that for every polynomial $p(.)$ there exists an N such that for all integers $n > N$ it holds that $f(n) < \frac{1}{p(n)}$.

Definition 3. *(Computational Diffie-Hellman (CDH) Problem) Given (P, aP, bP) where P is the generator of additive group G from order q and $a, b \in Z_q^*$, find abP.*

The advantage of a distinguisher $\mathcal{A}$ against the CDH problem is defined as

$$adv_{\mathcal{A},G}^{CDH} = Pr[\mathcal{A}(P, aP, bP) = abP].$$

Definition 4. *(Computational Diffie-Hellman (CDH) Assumption) Given (P, aP, bP), $adv_{\mathcal{A},G}^{CDH}$ of a distinguisher $\mathcal{A}$ whose goal is to solve the CDH problem is negligible.*

2.4 Security Model

We describe below the adversarial model following Bresson et al.'s definition [8], denoted by $BCPQ$. We then adopt it for the security analysis of our protocol. In this model, the process controlled by a player running on some machine is modeled as an instance of the player. The various types of attacks are modeled by queries to these instances and the security of the session key is modeled through semantic security. In the model, it is assumed that players are honest. Therefore, the security in the presence of dishonest participants is studied separately in Section 4.1. Note that we describe the model in the case of a passive adversary.

Notations: Throughout this section, we use the following notations:

n:	The number of participants;
U_i:	The ith participant;
Π_i^s:	Instance s of U_i;
LL_i:	The long-lived key of player U_i;
SK_i^s:	The session key related to instance s of U_i.

Description. The model consists of protocol participants (n players that should agree on a common key in different sessions through protocol P) and an adversary $\mathcal{A}$ (which is not a player in the formalization, and is given enormous capabilities). A player U_i can have many instances called oracles, involved in distinct concurrent executions of P. As defined in notations, we denote instance s of player U_i as Π_i^s with $s \in \mathbb{N}$. Each player U_i holds a long-lived key LL_i which is a pair of matching public/private keys. The adversary $\mathcal{A}$ controls all communications in a game $Game^{ke}(\mathcal{A}, P)$. This is a game between $\mathcal{A}$ and the oracles Π_i^s involved in the executions of P. During the game, $\mathcal{A}$ can ask a set of queries $(Execute, Reveal, Corrupt, Test)$ defined below with the restriction that the $Test$-query $Test(\Pi_i^s)$ must be asked only once. Moreover, it is only available if Π_i^s is fresh which means that Π_i^s or its partners involved in the execution of P, has not been asked for a $Reveal$-query, and none of them has been asked for a $Corrupt$-query. We now describe the queries that $\mathcal{A}$ can ask in $Game^{ke}(\mathcal{A}, P)$:

- $Execute(U_{l_1}, s_1, \cdots, U_{l_m}, s_m)$: This query models adversary $\mathcal{A}$ initiating an execution of protocol P. This executes the protocol between the (unused) instances $\{\Pi_{l_i}^{s_i}\}_{1 \le i \le m \le n}$, and outputs the transcript of the execution.
- $Reveal(\Pi_i^s)$: This query models the attacks resulting in the session key being revealed. The $Reveal$-query unconditionally forces Π_i^s to release session key SK_i^s.
- $Corrupt(U_i)$: This query models the attacks resulting in the player U_i's LL-key being revealed. Adversary $\mathcal{A}$ gets back LL_i but does not get the internal data of any instances of U_i executing P.
- $Test(\Pi_i^s)$: This query models the semantic security of the session key SK_i^s. After flipping a coin b, $\mathcal{A}$ is given SK_i^s if $b = 1$ or a random string if $b = 0$.

At the end of the game, adversary $\mathcal{A}$ outputs a bit b' and wins the game if $b = b'$.

The security is now defined as follows:

Definition 5. *The key exchange protocol P is $\mathcal{A}$-secure if adversary $\mathcal{A}$ succeeds in $Game^{ke}(\mathcal{A}, P)$ with probability that is at most negligibly greater than $\frac{1}{2}$.*

For more details, the interested reader is referred to [8].

3 The Proposed Scheme

In this section, we propose an identity-based group key exchange protocol based on elliptic curves. Suppose a set of n users $U = \{U_1, U_2, \cdots, U_n\}$ wish to establish a common session key among themselves. We assume that ID_i is the identity of U_i. Our protocol involves four phases:

1) The **initialization** phase in which the Key Generation Center (KGC) outputs the public parameters and generates the private key of each user.
2) The **information exchange** phase which consists of two rounds.
3) The **dishonest user elimination** phase which eliminates the dishonest user if such a person exists.
4) The **key computation** phase in which each user computes the common key.

Now, we explain these phases in details.

The initialization phase. In this phase which is executed once, the KGC performs the following steps:

1) Chooses an elliptic curve group G_1, a multiplicative group G_2 (both of prime order q), a generator $P \in G_1$, a bilinear pairing $e : G_1 \times G_1 \to G_2$, and a hash function $H : \{0,1\}^* \to Z_q^*$.

2) Chooses randomly $s \in Z_q^*$ (KGC's private key).

3) Computes $P_{pub} = sP$ (KGC's public key).

4) Outputs the public parameters $params = \{G_1, G_2, q, P, H, P_{pub}\}$.

5) Computes U_i'private key $S_i = \frac{1}{s+H(ID_i)}P$ and sends it to him securely.

The identity-based key pair produced in this phase will be used in Section 4 to authenticate the exchanged information. This is done by applying the compiler proposed in [27].

The information exchange phase (round 1). Each user $U_i(i = 1, \cdots, n)$ performs the following steps:

1) Chooses randomly $k_i \in Z_q^*$.

2) Computes P_i as follows:

$$P_i = k_i P.$$

3) Sends P_i to all other users.

The information exchange phase (round 2). Each user $U_i(i = 1, \cdots, n)$ performs the following steps:

1) Computes Y_i using P_{i-1} and P_{i+1}.

$$Y_i = k_i(P_{i+1} - P_{i-1}).$$

2) Sends Y_i to all other users.

The dishonest user elimination phase. Each user $U_i(i = 1, \cdots, n)$ verifies the following equation:

$$e(Y_j, P) = e(P_{j+1} - P_{j-1}, P_j), (j = 1, \cdots, i-1, i+1, \cdots, n). \quad (1)$$

If the equation does not hold for some j, then U_j will be considered dishonest. After eliminating dishonest participants from the session, the honest participants restart the protocol.

The key computation phase. Each user $U_i(i = 1, \cdots, n)$ obtains the common key using the following equation:

$$\mathrm{K_i} = \mathrm{nk_iP_{i-1}} + (\mathrm{n}-1)\,\mathrm{Y_i} + (\mathrm{n}-2)\,\mathrm{Y_{i+1}} + \ldots + \mathrm{Y_{i+n-2}}.$$

It may be easily verified that all users compute the same key: $(k_1k_2 + k_2k_3 + \cdots + k_nk_1)P$.

4 Security Analysis

In this section, we analyze security of the proposed scheme, denoted throughout this section by Π. The proof is presented in two parts. In Section 4.1, we prove that Π is capable of identifying dishonest participants, i.e., participants who prohibit the group from computing a common key by broadcasting fake values which are not produced according to the protocol specifications. Recall that the $BCPQ$ model does not support such participants. Therefore, in this part of the proof, we design an experiment which helps us formulate a suitable definition of security for this purpose.

The second part of the proof is devoted to showing that Π achieves security following the $BCPQ$ model, i.e., in the sense of Definition (5). This part is essentially adopted from [27]. Recall from Section 1, that security in key exchange protocols is considered against a passive adversary while in authenticated key exchange protocols, we must consider an active one. In [27], Katz and Yung presented an efficient compiler that transforms any group key exchange protocol secure against a passive eavesdropper to an authenticated protocol which is secure against an active adversary. This is achieved by adding a signature scheme which is strongly unforgeable under adaptive chosen message attack. Therefore, we show in Section 4.2, that our protocol is a secure key exchange protocol following Definition 5. Then applying Katz and Yung's compiler with values produced in initialization phase, we conclude that our scheme is a provably secure protocol for authenticated group key exchange.

4.1 Identification of Dishonest Participants

The aim of this section is to formally prove that in our scheme it is impossible for a participant to prohibit group members from obtaining a common key and remain unnoticed. Although, there exist protocols having formal proof for detecting dishonest participants [26], to the best of our knowledge, the present paper is the first to demonstrate a formal proof of security for the purpose of identifying dishonest participants.

In existing research papers where only the detection of dishonest participants is concerned, the approach is to show that the success probability of an insider in disrupting establishment of a key among honest participants is negligible. However, since we are after identifying dishonest insiders, we prefer to adopt the following definition: *Identification of dishonest insiders is achieved if the success probability of an honest participant in recognizing dishonest insiders is negligibly less than 1.*

We now propose the following experiment which is a game played between a user U_i and an imaginary tester who wishes to see if U_i succeeds in distinguishing protocol values from fake ones. Therefore, a protocol has security against dishonest insiders if the success probability of any honest participant playing this experiment is at most neg-

ligibly less than 1.

Formalizing the above notions, let Π be a group key exchange protocol containing n users $\{U_i\}_{i=1}^n$, R rounds of information exchange, and k as the security parameter. We define the following experiment:

Capability of identifying dishonest participants experiment $GKE_{U_i,\Pi}^{disP}(k)$**:**

- A matrix of $R\times(n-1)$ random bits $b_{r,j} \leftarrow \{0,1\}, r \in \{1,\cdots,R\}, j \in \{1,2,\cdots,i-1,i+1,\cdots,n\}$ is chosen by tester.
- The tester and U_i execute protocol Π where the tester essentially plays the role of all the users except U_i. This execution of the protocol results in a transcript $Trans$ containing all the messages exchanged among participants during different rounds of the information exchange phase of the protocol. Let $M_{r,j}^0$ denote the value produced honestly by U_j in round r (for every $r \in \{1,\cdots,R\}$, $j \in \{1,2,\cdots,i-1,i+1,\cdots,n\}$) and let $M_{r,j}^1$ be the corresponding fake value, i.e., $M_{r,j}^1$ is not computed following the protocol's steps. The tester computes $M_{r,j}^{b_{r,j}}$ as the value of round r produced by U_j.
- At the end, U_i outputs a matrix containing $b_{r,j}^{'}, r \in \{1,\cdots,R\}, j \in \{1,2,\cdots,i-1,i+1,\cdots,n\}$.
- The output of the experiment is defined to be 1 if $b_{r,j}^{'} = b_{r,j}, \forall r \in \{1,\cdots,R\}, \forall j \in \{1,2,\cdots,i-1,i+1,\cdots,n\}$ and 0 otherwise. If $GKE_{U_i,\Pi}^{disP}(k) = 1$, we say that U_i succeeds.

Definition 6. *A group key exchange scheme* Π *is capable of identifying dishonest insiders if for all probabilistic polynomial-time honest users* U_i*, there exists a negligible function* $negl$ *such that:*

$$Pr[GKE_{U_i,\Pi}^{disP}(k) = 1] \geq 1 - negl(k).$$

We now show that the protocol is secure against dishonest participants with respect to this definition. The following two lemmas form the basis of the proof.

Lemma 1. *In round 1 of the information exchange phase of the proposed protocol, the following holds for an honest participant* U_i *and for all* $j(\neq i)$*:*

$$Pr[U_i(Trans, M_{1,j}^0) = 0] - Pr[U_i(Trans, M_{1,j}^1) = 0] = 1.$$

In other words, U_i *is able to differentiate between* $M_{1,j}^0$ *and* $M_{1,j}^1$ *in Trans with probability 1.*

Proof. Suppose that U_i wants to determine the value of $b_{1,j}$ by observing $M_{1,j}^{b_{1,j}}$ after executing round 1 of the information exchange phase. Note that $M_{1,j}^{b_{1,j}}$ is the value produced honestly in round 1 by U_j if and only if $M_{1,j}^{b_{1,j}}$ is in G_1. Therefore, U_i checks the membership of $M_{1,j}^{b_{1,j}}$ in G_1 and returns $b_{1,j}^{'} = 0$ if it is a member of that group, and $b_{1,j}^{'} = 1$ otherwise. So U_i can distinguish $M_{1,j}^0$ from $M_{1,j}^1$ with probability exactly 1. □

Lemma 2. *In round 2 of the information exchange phase, the following holds for an honest participant* U_i *and for all* $j(\neq i)$*:*

$$Pr[U_i(Trans, M_{2,j}^0) = 0] - Pr[U_i(Trans, M_{2,j}^1) = 0] = 1.$$

In other words, U_i *is able to differentiate between* $M_{2,j}^0$ *and* $M_{2,j}^1$ *with probability equal to 1.*

Proof. Suppose that U_i wants to determine the value of $b_{2,j}$ by observing $M_{2,j}^{b_{2,j}}$ after executing round 2 of the information exchange phase. We show that $M_{2,j}^{b_{2,j}}$ satisfies Equation (1) if and only if it is the value produced honestly in round 2 by U_j.

First assume that $M_{2,j}^{b_{2,j}}$ satisfies Equation (1). Then we have: $e(M_{2,j}^{b_{2,j}}, P) = e(P_{j+1} - P_{j-1}, P_j) = e(P_{j+1} - P_{j-1}, k_jP) = e(k_j(P_{j+1} - P_{j-1}), P)$. Therefore, we have two possibilities: either $M_{2,j}^{b_{2,j}} = k_j(P_{j+1} - P_{j-1}) = Y_j$ or there exists α such that $M_{2,j}^{b_{2,j}} = Y_j + \alpha$ where $e(\alpha, P) = 1$. But as mentioned in Section 2, $e(\alpha, P) = 1$ if and only if α is the identity element of G_1. So $M_{2,j}^{b_{2,j}} = Y_j + identity = Y_j$. It means that $M_{2,j}^{b_{2,j}}$ is the value produced honestly in round 2 by U_j.

Now, suppose that $M_{2,j}^{b_{2,j}}$ is the value produced honestly in round 2 by U_j, i.e., Y_j. Then we have: $M_{2,j}^{b_{2,j}} = Y_j = k_j(P_{j+1} - P_{j-1})$. So we have: $e(M_{2,j}^{b_{2,j}}, P) = e(k_j(P_{j+1} - P_{j-1}), P) = e(P_{j+1} - P_{j-1}, k_jP) = e(P_{j+1} - P_{j-1}, P_j)$, which means that $M_{2,j}^{b_{2,j}}$ satisfies Equation (1).

Therefore, it is enough for U_i to check if $M_{2,j}^{b_{2,j}}$ satisfies equation $e(M_{2,j}^{b_{2,j}}, P) = e(P_{j+1} - P_{j-1}, P_j)$ and if so returns $b_{2,j}^{'} = 0$, otherwise $b_{2,j}^{'} = 1$ is returned. So U_i can distinguish $M_{2,j}^0$ from $M_{2,j}^1$ with probability equal to 1. □

Theorem 1. *The proposed protocol (denoted by* Π*) is capable of identifying dishonest participants.*

Proof. Let U_i be a probabilistic polynomial-time honest user. Using the definition of experiment $GKE_{U_i,\Pi}^{disP}(k)$ we have:

$$Pr[GKE_{U_i,\Pi}^{disP}(k) = 1] =$$
$$Pr[(b_{1,j} = b_{1,j}^{'} \forall j \neq i) \wedge (b_{2,j} = b_{2,j}^{'} \forall j \neq i)].$$

So it's sufficient to prove that U_i's guesses are correct in round 1 and 2 of the information exchange phase. In other words, we should prove the following two claims:

Claim 1. *In round 1 of the information exchange phase, we have:* $\forall j \neq i : Pr[b_{1,j} = b_{1,j}^{'}] = 1$.

Proof. Since $\forall j, r : Pr[b_{r,j} = 0] = Pr[b_{r,j} = 1]$, we have $\forall j \neq i$:

$$\begin{aligned} Pr[b_{1,j} = b'_{1,j}] &= \frac{1}{2}Pr[b'_{1,j} = 0 | b_{1,j} = 0] \\ &\quad + \frac{1}{2}Pr[b'_{1,j} = 1 | b_{1,j} = 1] \\ &= \frac{1}{2}Pr[U_i(Trans, M^0_{1,j}) = 0] \\ &\quad + \frac{1}{2}Pr[U_i(Trans, M^1_{1,j}) = 1] \\ &= \frac{1}{2}Pr[U_i(Trans, M^0_{1,j}) = 0] \\ &\quad + \frac{1}{2}(1 - Pr[U_i(Trans, M^1_{1,j}) = 0]) \\ &= \frac{1}{2} + \frac{1}{2}(Pr[U_i(Trans, M^0_{1,j}) = 0] \\ &\quad - Pr[U_i(Trans, M^1_{1,j}) = 0]) \\ &= \frac{1}{2} + \frac{1}{2}(1) \\ &= 1, \end{aligned}$$

where the last line follows from Lemma 1. □

Claim 2. *In round 2 of the information exchange phase, we have:* $\forall j \neq i : Pr[b_{2,j} = b'_{2,j}] = 1$.

Proof. Since $\forall j, r : Pr[b_{r,j} = 0] = Pr[b_{r,j} = 1]$, we have $\forall j \neq i$:

$$\begin{aligned} Pr[b_{2,j} = b'_{2,j}] &= \frac{1}{2}Pr[b'_{2,j} = 0 | b_{2,j} = 0] \\ &\quad + \frac{1}{2}Pr[b'_{2,j} = 1 | b_{2,j} = 1] \\ &= \frac{1}{2}Pr[U_i(Trans, M^0_{2,j}) = 0] \\ &\quad + \frac{1}{2}Pr[U_i(Trans, M^1_{2,j}) = 1] \\ &= \frac{1}{2}Pr[U_i(Trans, M^0_{2,j}) = 0] \\ &\quad + \frac{1}{2}(1 - Pr[U_i(Trans, M^1_{2,j}) = 0]) \\ &= \frac{1}{2} + \frac{1}{2}(Pr[U_i(Trans, M^0_{2,j}) = 0] \\ &\quad - Pr[U_i(Trans, M^1_{2,j}) = 0]) \\ &= \frac{1}{2} + \frac{1}{2}(1) \\ &= 1, \end{aligned}$$

where the last line follows from Lemma 2. □

Combining these two claims, we conclude that

$$Pr[GKE^{disP}_{U_i,\Pi}(k) = 1] = 1 \geq 1 - negl(k).$$

completing the proof. □

4.2 BCPQ-Security of the Proposed Scheme

In this section, we consider semantic security of the proposed protocol following BCPQ-model described in Section 2.4. We will show that the success probability of an attacker to learn anything about the shared key produced by the proposed scheme is negligible. Our proofs are inspired by [27].

Theorem 2. *The proposed protocol (denoted here by* Π*) is a secure group key exchange protocol in the sense of Definition 5.*

Proof. Let $\mathcal{A}$ be an adversary attacking the security of Π. We are going to show that his success probability is at most negligibly greater than $\frac{1}{2}$ in the game $Game^{ke}(\mathcal{A}, \Pi)$ defined in Section 2.4. So we claim that the value of the session key is indistinguishable for $\mathcal{A}$ from a random value in G_1.

First we assume that $\mathcal{A}$ eavesdrops on a single execution of the protocol. So he uses just one *Execute* query and he cannot use any *Reveal* query (because there is only one session key and it must be fresh for meaningful *Test* query). Since the participants have no LL-keys, there is no Corrupt query here. Thereupon, $\mathcal{A}$ can only use the knowledge of the transcript of this execution to output b'.

Consider the distribution $D = (Trans, sk)$, where $Trans = (\{P_i\}, \{Y_i\})$ is the transcript of an execution of the protocol and sk is the resulting session key. Let D' be another distribution in which (as in D) all the $\{P_i\}$ are uniformly distributed in G_1, but in which (in contrast to D) all the $\{Y_i\}$ are uniformly distributed in G_1 subject to the constraint $\sum_i Y_i = 0$. We have the following two claims:

Claim 3. *No efficient adversary can distinguish between the distributions* D *and* D'.

Proof. In order to show that the distributions D and D' are computationally indistinguishable, hybrid argument can be used. We do this by defining a sequence of n hybrid distributions in which one Y_i at a time is replaced with a random group element (subject to the above-mentioned constraint). Since adjacent distributions in this definition differ by only one Y_i, it is computationally easy to conclude their indistinguishability. Then since computational indistinguishability is transitive across a polynomial number of distributions, we conclude that D and D' are computationally indistinguishable. □

Claim 4. *In distribution* D'*, the value of the session key is uniformly distributed in* G_1*, independent of the value of the transcript.*

Proof. Let $c_{i,i+1} := k_i k_{i+1}$ for $1 \leq i \leq n$. Given $Trans = (\{P_i\}, \{Y_i\})$, the values $c_{1,2}, \cdots, c_{n,1}$ are constrained by the following n equations (only $n-1$ of which are linearly

Table 1: Computation cost of our protocol and the protocols proposed in [17, 22], and [37]

	Dutta and Barua [17]	**Huang et al. [22]**	**Zhao et al. [37]**	**Ours**
T_M	$2n-2$	$2n-2$	-	-
T_D	1	-	-	-
T_{exp}	3	$2n-1$	$3n-2$	-
T_A	-	-	-	$2n-1$
T_{SM}	-	-	-	$n+1$
T_{BP}	-	-	-	$2n-2$
T_H	-	-	n	-
T_{Sign}	2	1	-	2
T_{Vrfy}	$n+1$	$n-1$	-	$2n-2$
Total	$(2n-2)T_M+1T_D+3T_{exp}+2T_{Sign}+(n+1)T_{Vrfy}$	$(2n-2)T_M+(2n-1)T_{exp}+1T_{Sign}+(n-1)T_{Vrfy}$	$(3n-2)T_{exp}+nT_H$	$(2n-1)T_A+(n+1)T_{SM}+(2n-2)T_{BP}+2T_{Sign}+(2n-2)T_{Vrfy}$

independent):

$$\frac{1}{P}Y_1 = c_{1,2} - c_{n,1}$$
$$\vdots$$
$$\frac{1}{P}Y_n = c_{n,1} - c_{n-1,n}$$

Furthermore, $sk = (c_{1,2} + c_{2,3} + \cdots + c_{n,1})P$; equivalently, we have

$$\frac{1}{P}sk = c_{1,2} + c_{2,3} + \cdots + c_{n,1}.$$

Since this final equation is linearly independent from the set of equations above, sk is independent of $(\{P_i\}, \{Y_i\})$. This implies that even for a computationally-unbounded adversary $\mathcal{A}$, we have

$$Pr[(\{P_i\}, \{Y_i\}, sk_0) \leftarrow D^{'}; sk_1 \leftarrow G_1; b \leftarrow \{0,1\} : A(\{P_i\}, \{Y_i\}, sk_b) = b] = \frac{1}{2}.$$

□

The above two claims, prove the security of the protocol for an adversary making only a single Execute query. The case of multiple Execute queries can be dealt with using a straight forward hybrid argument. □

5 Performance Analysis and Comparison

In this section, we first consider the communication and computation cost of the proposed protocol. The analysis of the communication cost is done in terms of the number of messages sent by any single user. Note that we consider the communication cost in the point-to-point model in which each message sent to a different party is counted separately. Computation complexity analysis is also done in terms of the basic time-consuming operations. We then provide a comparison between our protocol and some other schemes in terms of computation cost, communication cost, and security properties. In order to evaluate the computation cost, we use the following notations:

- T_M: Execution time for one multiplication operation in multiplicative groups;
- T_D: Execution time for one division operation in multiplicative groups;
- T_{exp}: Execution time for one exponentiation operation in multiplicative groups;
- T_A: Execution time for one addition operation in elliptic curve groups;
- T_{SM}: Execution time for one scalar multiplication operation in elliptic curve groups;
- T_{BP}: Execution time for one bilinear pairing operation of two elements over an elliptic curve;
- T_H: Computation time of a hash function;
- T_{Sign}: Generation time of one signature;
- T_{Vrfy}: Execution time for one signature verification.

Here, we first analyze the communication and computation cost of the proposed protocol with n participants. In round 1 and 2 of the information exchange phase, each user should send P_i and Y_i to $n-1$ other users. Therefore, $2n-2$ messages are communicated in total by each participant. Moreover, in round 1, one T_{SM} is required for computing P_i and in round 2, each participant requires $1T_A+1T_{SM}$ to compute Y_i. In the dishonest user elimination phase $(n-1)(2T_{BP}+1T_A)$ is needed as well. Finally, in the key computation phase, we require $(n-1)T_{SM} + (n-1)T_A$ to compute the common key. As a result, $(n+1)T_{SM} + (2n-2)T_{BP} + (2n-1)T_A$ is required for each participant in our protocol. Note that by applying the compiler of Katz and Yung to this protocol, an extra message has to be sent by each user to the others which can be sent along with the first broadcasted

Table 2: Comparison between our protocol and the protocols proposed in [17, 22], and [37]

	Dutta and Barua [17]	**Huang et al. [22]**	**Zhao et al. [37]**	**Ours**
Communication cost	$n+1$	$2n-2$	$2n-2$	$2n-2$
Computation cost	$(2n+728)T_M + 2T_{Sign} + (n+1)T_{Vrfy}$	$(482n-242)T_M + 1T_{Sign} + (n-1)T_{Vrfy}$	$(720.4n-480)T_M$	$(189.1n-125.9)T_M + 2T_{Sign} + (2n-2)T_{Vrfy}$
	$(352.9n+1618.7)T_M$	$(832.9n-323)T_M$	$(720.4n-480)T_M$	$(890.9n-287.9)T_M$
Formal proof of semantic security	Yes	No	No	Yes
Formal proof of security against dishonest participants	No	No	No	Yes
Detecting the presence of dishonest participants	No	Yes	Yes	Yes
Identifying dishonest participants	No	Yes	Yes	Yes

message. Moreover, one signature generation and $n-1$ signature verifications (for each of the two messages sent in round 1 and 2) are added to the computation complexity of the proposed protocol.

Now, we compare our protocol with those of Dutta and Barua [17], Huang et al. [22] and Zhao et al. [37]. The computation cost of these protocols is shown in Table 1. Table 2 lists the comparison between our protocol and group key exchange protocols [17, 22, 37] in terms of performance and security properties.

In order to make comparison more clear, we have used the relationship between the execution times of operations as in [5, 24, 28, 29]. We assume that $T_{exp} \cong 8.24T_{SM}$, $T_{exp} \cong 240T_M$, $T_{exp} \cong 600T_H$, $T_{exp} \cong 3.2T_{BP}$, $T_A \cong 5T_M$, and $T_D \cong 10T_M$. Besides, we have set the execution times of the signature and verification algorithms used in [14] as T_{Sign} and T_{Vrfy} respectively to simplify cost relations. The results are summarized in Table 2. According to Table 2, the scheme of Dutta and Barua outperforms the others in communication costs; however the detection and identification of dishonest participants are not achieved. The schemes of [22] and [37] both detect and identify dishonest participants but no formal proofs are provided. In our scheme, capabilities of the method against dishonest participants are formally proved. Moreover, detection and identification of dishonest participants is achieved at reasonable cost.

6 Conclusions

In this paper, elliptic curves are employed to propose an identity-based group key exchange protocol. It is proved the proposed protocol achieves security following the adversarial model of Bresson et al. The security of the protocol in the presence of dishonest participants is proved formally as well. The performance of the scheme in terms of computation cost, communication cost, and security properties is also considered.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Bellare, R. Canetti, and H. Krawczyk, "A modular approach to the design and analysis of authentication and key exchange protocols," in *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing*, pp. 419–428, New York, NY, USA, 1998.

[2] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Proceedings of the 19th International Conference on Theory and Application of Cryptographic Techniques*, vol. 1807, pp. 139–155, Berlin Heidelberg, 2000.

[3] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology (Crypto'93)*, LNCS 773, pp. 232–249, Springer, 1993.

[4] M. Bellare and P. Rogaway, "Provably secure session key distribution - the three party case," in *Proceedings of the 27th ACM Symposium on the Theory of Computing*, pp. 57–66, New York, NY, USA, 1995.

[5] G. M. Bertoni, L. Breveglieri, L. Chen, P. Fragneto, K. A. Harrison, and G. Pelosi, "A pairing SW implementation for smart-cards," *Journal of Systems and Software*, vol. 81, no. 7, pp. 1240–1247, 2008.

[6] C. Boyd and J. M. G. Nieto, "Round-optimal contributory conference key agreement," in *Proceedings of the 6th International Workshop on Theory and Practice in Public Key Cryptography*, LNCS 2567, pp. 161–174, Springer, 2003.

[7] E. Bresson, O. Chevassut, and D. Pointcheval, "Dynamic group diffie-hellman key exchange under standard assumptions," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, LNCS 2332, pp. 321–336, Springer, 2002.

[8] E. Bresson, O. Chevassut, D. Pointcheval, and J. J. Quisquater, "Provably authenticated group Diffie-Hellman key exchange," in *Proceedings of the 8th ACM Conference on Computer and Communications Security*, pp. 255–264, 2001.

[9] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques*, LNCS 950, pp. 275–286, Springer, 1995.

[10] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques*, LNCS 2045, pp. 453–474, Springer, 2001.

[11] 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.

[12] S. Chatterjee and P. Sarkar, *Identity-Based Encryption*, Springer, 2011.

[13] K. Y. Choi, J. Y. Hwang, and D. H. Lee, "Id-based authenticated group key agreement secure against insider attacks," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, no. 7, pp. 1828–1830, 2008.

[14] S. Cui, P. Duan, C. W. Chan, and X. Cheng, "An efficient Identity-based signature scheme and its applications," *International Journal of Network Security*, vol. 5, no. 1, pp. 89–98, 2007.

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

[16] R. Dutta and R. Barua, "Password-Based encrypted group key agreement," *International Journal of Network Security*, vol. 3, no. 1, pp. 23–34, 2006.

[17] R. Dutta and R. Barua, "Provably secure constant round contributory group key agreement in dynamic setting," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2007–2025, 2008.

[18] Z. Eslami and S. Kabiri Rad, "Another security weakness in an authenticated group key agreement," *Journal of Internet Technology*, vol. 11, no. 4, pp. 573–576, 2010.

[19] A. A. Hafez, A. Miri, and L. O. Barbosa, "Authenticated group key agreement protocols for Ad-hoc wireless networks," *International Journal of Network Security*, vol. 4, no. 1, pp. 90–98, 2007.

[20] M. Hietalahti, "A clustering-based group key agreement protocol for Ad-hoc networks," *Electronic Notes in Theoretical Computer Science*, vol. 192, no. 2, pp. 43–53, 2008.

[21] S. Hong, "Queue-based group key agreement protocol," *International Journal of Network Security*, vol. 9, no. 2, pp. 135–142, 2009.

[22] K. H. Huang, Y. F. Chung, H. H. Lee, F. Lai, and T. S. Chen, "A conference key agreement protocol with fault-tolerant capability," *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 401–405, 2009.

[23] I. Ingemaresson, D. T. Tang, and C. K. Wong, "A conference key distribution system," *IEEE Transaction on Information Theory*, vol. 28, no. 5, pp. 714–720, 1982.

[24] W.-S. Juang, "Ro-cash: An efficient and practical recoverable pre-paid offline e-cash scheme using bilinear pairings," *The Journal of Systems and Software*, vol. 83, no. 4, pp. 638–645, 2010.

[25] M. Just and S. Vaudenay, "Authenticated multi-party key agreement," in *Proceedings of The International Conference on the Theory and Applications of Cryptology and Information Security*, pp. 36–49, London, UK, 1996.

[26] J. Katz and J. S. Shin, "Modeling insider attacks on group key-exchange protocols," in *Proceedings of the 12th ACM conference on Computer and communications security*, pp. 180–189, New York, NY, USA, 2005.

[27] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," *Journal of Cryptology*, vol. 20, no. 1, pp. 85–113, 2007.

[28] K. Lauter, "The advantages of elliptic curve cryptography for wireless security," *IEEE Wireless Communications*, vol. 11, no. 1, pp. 62–67, 2004.

[29] A. Lenstra, E. Tromer, A. Shamir, W. Kortsmit, B. Dodson, J. Hughes, and P. Leyland, "Factoring estimates for a 1024-bit RSA modulus," in *Proceedings of the 9th International Conference on the Theory and Application of Cryptology and Information Security*, LNCS 2894, pp. 55–74, Springer, 2003.

[30] J. P. Lin and J. M. Fu, "Authenticated key agreement scheme with Privacy-Protection in the Three-party setting," *International Journal of Network Security*, vol. 15, no. 3, pp. 179–189, 2013.

[31] M. Manulis, *Survey on Security Requirements and Models for Group Key Exchange*, Technical Report TR-HGI-2006-002, Ruhr-Universität Bochum, Jan. 5, 2008.

[32] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of Advances in Cryptology (CRYPTO'84)*, LNCS 196, pp. 47–53, Springer, 1985.
[33] Z. Tan, "Efficient identity-based authenticated multiple key exchange protocol," *Computers & Electrical Engineering*, vol. 37, no. 2, pp. 191–198, 2011.
[34] W. G. Tzeng, "A secure fault-tolerant conference key agreement protocol," *IEEE Transactions on Computers*, vol. 80, no. 4, pp. 373–379, 2002.
[35] S. Wang, Z. Cao, and F. Cao, "Efficient Identity-based authenticated key agreement protocol with PKG forward secrecy," *International Journal of Network Security*, vol. 7, no. 2, pp. 181–186, 2008.
[36] Z. You and X. Xie, "A novel group key agreement protocol for wireless mesh network," *Computers & Electrical Engineering*, vol. 37, no. 2, pp. 218–239, 2011.
[37] J. Zhao, D. Gu, and Y. Li, "An efficient fault-tolerant group key agreement protocol," *Computer Communications*, vol. 33, no. 7, pp. 890–895, 2010.

Ziba Eslami received her B.S., M.S., and Ph.D. in Applied Mathematics from Tehran University in Iran. She received her Ph.D. in 2000. From 1991 to 2000, she was a resident researcher in the Institute for Studies in Theoretical Physics and Mathematics (IPM), Iran. During the academic years 2000-2003, she was a Post Doctoral Fellow in IPM. She served as a non-resident researcher at IPM during 2003-2005. Currently, she is associate professor in the Department of Computer Sciences at Shahid Beheshti University in Iran. Her research interests include design theory, combinatorial algorithms, cryptographic protocols, and steganography.

Mahnaz Noroozi received her B.S. degree in Computer Sciences in 2010 from Sharif University of Technology, Tehran, Iran. In 2012, she received her M.S. degree in Computer Sciences from Shahid Beheshti University, Tehran, Iran. She is currently doing research on cryptographic protocols and their security.

Saideh Kabiri Rad received the B.S. degree from Shahid Bahonar University, in Kerman and the M.S. degree from Shahid Beheshti University, in Tehran, both in Computer Science, Iran. She received the B.S. in 2007 and M.S. in 2010. She is currently doing research on information security and cryptography.