A Dynamic ID Based Authenticated Group Key Agreement Protocol from Pairing

Shruti Nathani[1], B. P. Tripathi[1], and Shaheena Khatoon[2]
(Corresponding author: B. P. Tripathi)
Department of Mathematics, Govt. N. P. G. College of Science[1]
Raipur-492010, Chhattisgarh, India
S. O. S. in Mathematics, Pt. Ravi Shankar Shukla University[2]
Raipur-492010, Chhattisgarh, India
(Email: bhanu.tripathi@gmail.com)
(Received Jan. 26, 2018; revised and accepted June 3, 2018; First Online Apr. 21, 2019)

Abstract

In this paper we present an identity (ID) based dynamic authenticated group key agreement protocol. Our protocol satisfies all the required security attributes and also provide forward and backward confidentiality. The security of our protocol is based on the bilinear Diffie-Hellman(DH) assumption. We extend Lee *et al.* ID based authenticated key agreement protocol from two party to a group of users by using bilinear pairing.

Keywords: Backward Confidentiality; Bilinear Pairing; Dynamic; Forward Confidentiality; Group Key Agreement Protocol

1 Introduction

The most striking development in the history of cryptography was happened in 1976, when Diffie *et al.* [12] proposed their revolutionary concept of two party key agreement protocol whose security was based on the discrete logarithm problem. But this protocol was not suitable for group of users. Then in 1982, Ingemarsson *et al.* [16] proposed the first group key exchange protocol, but both of these schemes were vulnerable to the man in the middle attack because they did not authenticate the involved parties.

A key agreement protocol is said to provide key authentication, if each entity involved in the exchange is assured that no other entity can learn the shared secret key. A key agreement protocol which provides such a property is called an authenticated key agreement protocol (AKE) [21].

An authentication protocol allows a sender to send messages to a receiver through an insecure communication channel in such a way that the receiver can be convinced that the messages are indeed coming from the intended sender and them messages have not been modified by any adversary sitting in the middle of the communication channel. In short the aim of this type of protocols is to establish an authenticated link from the sender to the receiver. Authentication is a term which is used in a very broad sense. It is a service related to identification [21].

In 1984, Shamir [26] suggested the concept of Identity based cryptosystems where user's identities(such as email address, phone numbers, office location etc.), could be used as the public keys. Since then many identity based key agreement protocols [6, 11, 27, 29, 30, 34] have been proposed.

In the history of key agreement, a major breakthrough was happened, in 2000 when Joux [17] introduced his simple and elegant single round tripartite non-identibased key agreement protocol which makes use of bilinear pairing on elliptic curves. This was the first positive application of pairings in cryptography [13].

In 2001, Bohen *et al.* [3] proposed, a first identity based encryption scheme using weil pairing. Since then many ID based cryptographic scheme using pairing have been proposed in cryptography and is currently an area of very active research [13].

1.1 Literature Review

Based on weil and Tate pairing techniques, Smart [30] in 2002, Chen *et al.* [6] in 2003, Scott [27] in 2002, Shim [29] in 2003, Cullagh [11] in 2004, Lee *et al.* [20] in 2005 designed identity based and authenticated two party key agreement protocols. Cheng *et al.* [8] pointed out that Chen *et al.* [6] protocol is not secure against unknown key share attack. The protocol of Scott [27] is not secure against man in the middle attack. Sun *et al.* [33] showed that the protocol of Shim [29] is insecure against key compromise impersonation attack or man-in-the-middle attack. Also Choo [10] showed that protocol of Cullagh *et al.* [11] is insecure against key revealing attacks.

Since the protocol of Joux [17] was a unauthenticated

key agreement for three party using pairing on elliptic curve. So later in 2002, Nalla *et al.* [23], proposed an authenticated tripartite ID-based key agreement scheme. But this scheme of Nalla *et al.* [23] was soon cryptanalyzed by chen [5] and Shim [28]. Then again in 2002, Zhang *et al.* [37], gave an ID- based one round authenticated three party key agreement protocol the authenticity of which is assured by the Id base signature scheme of Hess [15]. Another direction of research on key agreement is to generalize the two party key agreement to multi party setting and consider the dynamic scenario where participants may join or leave a multi-cast group at any given time.

1.2 Group Key Agreement

A group key agreement is a protocol allows a group of users to exchange information over public and insecure network to agree upon a common secret key which a group session key can be derived. As, the increased popularity of group oriented applications, such as e-learning, e-conference, video-conferencing etc, the design of an efficient authenticated group key agreement protocol has recently received much attention in the current research literature.

In 1995, Burmester *et al.* [4] gave a much more efficient two round key agreement protocol in multiparty setting. In 1996, Steiner *et al.* [31] gave a group key agreement protocol based on the natural extension of the DH key agreement protocol. Later, in 1998, Steiner *et al.* [32] gave a new approach to group key agreement. They studied the problem of key agreement in dynamic peer groups(DPG).

Also, Bresson *et al.* [2] formalized the first security model for group key agreement protocol extending the group key agreement between two or three parties [25]. Then in 2002, Nalla *et al.* [22] extends the ID based two party single round authenticated protocol of Smart [30] to multiparty ID-based key agreement in a tree based setting [14]. Later, in 2003, Barua *et al.* [1], extend the basic three party protocol of Joux [17] to multiparty setting by giving a ternary tree based unauthenticated key agreement protocol. Another group key agreement protocol which is a bilinear version of BD [4] protocol, was proposed by Choi *et al.* [9], in 2004. Later in 2005, a dynamic group key agreement protocol with two constant round was propose by Dutta *et al.* [14].

Many attempts have been performed to extend the Diffie *et al.* [12] two party protocol and the Joux [17] protocol for three party to $n-$participants that means to a group key exchange. Also we seen that in the current research literature of key agreement many ID based dynamic group key agreement schemes [7, 19, 35] by using bilinear pairing have been proposed.

Above we have summerized two and three party identity based key agreement protocols employing pairing operations. Many protocols of this type were proposed [11, 22,27,30,37] analyzed and some broken [5,10,28,29,33]. In this paper we focus on the Lee *et al.* [20] two party authenticated key agreement protocol and extend this two party protocol into a dynamic ID based authenticated group key agreement(DAGKA) using bilinear pairing.

2 Preliminaries

In this section, we briefly describe the notations, definitions, preliminary concepts and properties i.e. bilinear maps, computational problems, efficiency criteria and security attributes that we used later in the paper.

2.1 Bilinear Maps

Let G_1 be an additive group of prime order l and G_2 be a multiplicative group of the same order l. We assume discrete log problems in G_1 and G_2 are hard. We consider a pairing map $e : G_1 \times G_1 \to G_2$ satisfying the following properties [20].

Bilinearity. $e(aP, bQ) = e(P, Q)^{ab}$for all $P, Q \in G_1$, and for all $a, b \in \mathbb{Z}_l^*$.

Non-degeneracy. The map does not send all pairs in $G_1 \times G_1$ to the identity in G_2. Observe that since G_1 and G_2 are groups of prime order this implies that if P is a generator of G_1, then $e(P, P)$ is a generator of G_2.

Computability. Given $P, Q \in G_1$, $e(P, Q)$ can be efficiently computable.

A bilinear map satisfying the three properties above is said to be an admissible bilinear map. We note that the Weil and Tate pairings associated with supersingular elliptic curves or abelian varieties can be modified to create such bilinear maps. We consider G_1 to be an additive abelian group defined on elliptic curves.

We consider an admissible bilinear map $e : G_1 \times G_1 \to G_2$ defined as above. Let P be a generator of G_1.

Bilinear Diffie Hellman Problem (BDHP):
The BDH problem in $< G_1, G_2, e >$ is as follows. Given $P, aP, bP, cP \in G_1$, compute $e(P, P)^{abc} \in G_2$ where a, b, c are randomly chosen from $\mathbb{Z}_l^*$. An algorithm is said to solve the BDH problem with an advantage of ϵ if

$$Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}] \geq \epsilon$$

where the probability is over the random choice of $a, b, c \in \mathbb{Z}_l^*$, the random choice of $P \in G_1^*$, and the random bits of $\mathcal{A}$. We assume that BDHP is hard, in other words, there is no polynomial time algorithm to solve BDHP with non-negligible probability.

2.2 Security Attributes

Now we give the desirable security attributes of the key agreement protocols:

Known-key security. Each run of the protocol should result in a unique secret session key. The compromise of one session key should not com- promise other session keys.

Perfect Forward secrecy. If long-term private keys of all entities are compromised, the secrecy of previously established session keys should not be affected.

Key compromise impersonation. The compromise of an entity A's long- term private key will allow an adversary to impersonate A, but it should not be able the adversary to impersonate other entities to A.

Unknown-key share. An entity A ends up believing she shares a key with B and although this is in fact the case, B mistakenly believes the key is instead shared with an entity $E \neq A$.

Message confidentiality is one of the most important feature in secure group communication. Message confidentiality ensures that the sender confidential data which can be read only by an authorized and intended receiver. Specially in DGKA protocols message confidentiality is achieved mainly by the following two components [19]:

Forward confidentiality. While a group user leaves from the current group, he should not be able to calculate the new session key.

Backward confidentiality. While a new user joins into the current group, he should not be able to calculate the previous session key.

3 Lee *et al.'s* ID Based Key Agreement

In this section, we will introduce Lee *et al.'s* [20] two party ID based key agreement.

Initialization. Let G_1 and G_2 be two groups of prime order l, where G_1 is an additive group and G_2 is a multiplicative group. The discrete logarithm problems (DLP) in both G_1 and G_2 are assumed to be hard. Let P be a generator of G_1, and $H : \{0,1\}^* \rightarrow \mathbb{Z}_l^*$ be a cryptographic hash function. The key generation center (KGC) chooses a random number $s \in \mathbb{Z}_l^*$ and set $P_{pub} = sP$. The center publishes system parameters $Params =< G_1, G_2, l, e, P, P_{pub}, H >$ and keep s as the master key, which is known only by itself.

In addition to the system initialization, KGC performs the following private key issuing process.

Private key extraction. Let A and B be the two entities who are going to agree to some session keys. The identities of A and B are ID_A and ID_B, respectively. Their public keys and private keys are as follows: $A's$ public key is $P_A = H(ID_A)$ and the private key is $S_A = sP_A$. $B's$ public key is $P_B = H(ID_B)$ and the private key is $S_B = sP_B$. The pairs (P_ID, S_ID) for A and B serve as their static public/private key pairs.

Suppose two users A and B want to share a common secret. A and B have static private keys $S_A = sP_A$ and $S_B = sP_B$ obtained from KGC. Let $kdf : G_2 \times G_1 \times G_1 \rightarrow \{0,1\}^*$ be a key derivation function which can be readily found in a number of standard documents. A and B generate ephemeral private keys a and b, respectively. The corresponding ephemeral public keys are (V_A, W_A) and (V_B, W_B) where $V_A = aP_B$, $W_A = aS_A$, $V_B = bP_A$, $W_B = bS_B$. These are the data flow between A and B.

$$\begin{aligned} A &\Rightarrow B : (V_A, W_A); \\ B &\Rightarrow A : (V_B, W_B). \end{aligned}$$

User A computes $k_A = e(aP_A + V_B, W_B)^a$. User B computes $k_B = e(bP_B + V_A, W_A)^b$. Then the shared common secret between A and B is $K = kdf(k_A, P_A, P_B) = kdf(k_B, P_A, P_B) = kdf(e(P_A, P_B)^{(a+b)abs}, P_A, P_B)$.

4 Proposed Protocol

Let $U_0 = \{U_1, U_2, \cdots, U_n\}$ be the initial set of participants that want to generate a common key. Where U_n is the group leader. And

$$ID_0 = ID_{u_1} \parallel ID_{u_2} \parallel ..... \parallel ID_{u_n}.$$

4.1 Setup

Let G_1 and G_2 be two groups of prime order l where G_1 is an additive group and G_2 is a multiplicative group. The DLP in both G_1 and G_2 are assumed to be hard. Let P be a generator of G_1 and $H : \{0,1\}^* \rightarrow \mathbb{Z}_l^*$ be a cryptographic hash function. The user U_n randomly picks a value $s \in \mathbb{Z}_l^*$ and keeps s as master private key. The user U_n computes the master public key $P_{pub} = sP$ and publishes the system parameters. $param = \{G_1, G_2, l, e, P, P_{pub}, H\}$.

Private Key Extraction. For a given user U with identity string ID, the user U_n computes the public key $PK_{ID} = Q_{ID} = H(ID)$ and distributes the corresponding static private key $SK_{ID} = sQ_{ID}$ to the user via a secure channel. Thus user $U's$ public/private key pair is defined as PK_{ID}/SK_{ID}.

Round 1.

Step 1: The group leader U_n:

1.1 chooses his ephemeral private keys $a_n \leftarrow^R \mathbb{Z}_n$.;

1.2 ephemeral public keys are (V_n, W_n)sends: where $V_n = a_n P_{pub}$ and $W_n = a_n SK_{U_n}$;

1.3 broadcast in the group:
$U_n \rightarrow$: $(\{U_1, U_2, \cdots, U_{n-1}\}, V_n, W_n)$.

Step 2: User U_1 :

2.1 chooses his ephemeral private keys $a_1 \leftarrow^R \mathbb{Z}_n$;

2.2 computes $K_1 = sQ_{U_1} + a_1$ and $M_1 = h_1(U_1, ..., U_{n-1}, a_1)$;

2.3 also ephemeral public keys are (V_1, W_1) : where $V_1 = a_1 P_{pub}$ and $W_1 = a_1 SK_{U_1}$;

2.4 sends a request:

$$U_1 \rightarrow U_n : (U_1, K_1, M_1, V_1, W_1).$$

Step 3: The user U_n :

3.1 computes $a_1 = K_1 - sQ_{U_1}$;

3.2 checks if $M_1 = h_1(U_1, ..., U_{n-1}, a_1)$; if the equality does not hold, he quits;

3.3 broadcasts:
$U_n \rightarrow^*$: (V_1, W_1).

Step 4: Each User $U_i, i = 2, ..., n-1$:

4.1 chooses his ephemeral private keys $a_i \leftarrow^R \mathbb{Z}_n$;

4.2 computes $K_i = sQ_{U_i} + a_i$ and $M_i = h_1(U_1, ..., U_{n-1}, a_i)$;

4.3 ephemeral public keys are (V_i, W_i)sends: where $V_i = a_i P_{pub}$ and $W_i = a_i SK_{u_i}$;

4.4 sends

$$U_i \rightarrow U_n : (U_i, K_i, M_i, V_i, W_i).$$

Step 5: The User U_n:$i = 2, \cdots, n-1$:

5.1 computes $a_i = K_i - sQ_{U_i}$;

5.2 checks if $M_i = h_1(U_1, ..., U_{n-1}, a_i)$; if atleast one equality does not hold, he quits;

5.3 broadcasts:
$U_n \rightarrow$: (V_i, W_i).

Round 2.

User U_1 computes

$$\begin{aligned} K_1 &= e((V_2 \times V_3 \times \cdots \times V_n)P_{pub}, \\ &\quad (W_2 \times W_3 \times \cdots \times W_n)SK_{U_1})^{a_1}. \end{aligned}$$

User U_2 computes

$$\begin{aligned} K_2 &= e((V_1 \times V_3 \times \cdots \times V_n)P_{pub}, \\ &\quad (W_1 \times W_3 \times \cdots \times W_n)SK_{U_2})^{a_2}. \\ &\vdots \end{aligned}$$

User U_{n-1} computes

$$\begin{aligned} K_{n-1} &= e((V_1 \times \cdots \times V_{n-2} \times V_n)P_{pub}, \\ &\quad (W_1 \times \cdots \times W_{n-2} \times W_n)SK_{U_{n-1}})^{a_{n-1}}. \end{aligned}$$

Key Computation. Each user $U_i, i = 1, 2, \cdots, n-1$. Let $kdf : G_2 \times \underbrace{G_1 \times G_1 \times \cdots \times G_1}_{ntimes} \rightarrow \{0,1\}^*$ be a key derivation function which can be readily found in a number of standard documents. Thus the shared common group session key,

$$\begin{aligned} K &= kdf(K_1, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(K_2, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &\vdots \\ &= kdf(K_{(n-1)}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}). \end{aligned}$$

For user U_1,

$$\begin{aligned} K &= kdf(K_1, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((V_2 \times V_3 \times \cdots \times V_n)P_{pub}, \\ &\quad (W_2 \times W_3 \times \cdots \times W_n)SK_{U_1})^{a_1}, Q_{U_1}, \\ &\quad Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((a_2 P_{pub} \times a_3 P_{pub} \times \cdots \times a_n P_{pub})P_{pub}, \\ &\quad (a_2 sQ_{U_2} \times a_3 sQ_{U_3} \times \cdots \times a_n sQ_{U_n})sQ_{U_1})^{a_1}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((a_2 \times a_3 \times \cdots \times a_n)P_{pub}^n, \\ &\quad (a_2 \times a_3 \times \cdots \times a_n)s^n \\ &\quad (Q_{U_1} \times Q_{U_2} \times Q_{U_3} \times \cdots \times Q_{U_n}))^{a_1}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((sP)^n, (Q_{U_1} \times \cdots \times Q_{U_n})s^n)^{(a_1 \times a_2 \times \cdots \times a_n)}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e(P^n, (Q_{U_1} \times ... \times Q_{U_n}))^{s^n(a_1 \times a_2 \times \cdots \times a_n)}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \end{aligned}$$

For user U_2,

$$\begin{aligned} K &= kdf(K_2, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((V_1 \times V_3 \times \cdots \times V_n)P_{pub}, \\ &\quad (W_1 \times W_3 \times \cdots \times W_n)SK_{U_2})^{a_2}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((a_1 P_{pub} \times a_3 P_{pub} \times \cdots \times a_n P_{pub})P_{pub}, \\ &\quad (a_1 sQ_{U_1} \times a_3 sQ_{U_3} \times \cdots \times a_n sQ_{U_n})sQ_{U_2})^{a_2}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((a_1 \times a_3 \times \cdots \times a_n)P_{pub}^n, \\ &\quad (a_1 \times a_3 \times \cdots \times a_n)s^n \\ &\quad (Q_{U_1} \times Q_{U_2} \times Q_{U_3} \times, ... \times Q_{U_n}))^{a_2}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e((sP)^n, (Q_{U_1} \times \cdots \\ &\quad \times Q_{U_n})s^n)^{(a_1 \times a_2 \times \cdots \times a_n)}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\ &= kdf(e(P^n, (Q_{U_1} \times \cdots \\ &\quad \times Q_{U_n}))^{s^n(a_1 \times a_2 \times \cdots \times a_n)}, \\ &\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}). \end{aligned}$$

For user U_{n-1},

$$\begin{aligned}
K &= kdf(K_{n-1}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\
&= kdf(e((V_1 \times \cdots \times V_{n-2} \times V_n)P_{pub}, \\
&\quad (W_1 \times \cdots \times W_{n-2} \times W_n)SK_{U_{n-1}})^{a_{n-1}}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\
&= kdf(e((a_1 P_{pub} \times \cdots \times a_{n-2}P_{pub} \\
&\quad \times a_n P_{pub})P_{pub}, \\
&\quad (a_1 sQ_{U_1} \times \cdots \times a_{n-2}sQ_{U_{n-2}} \\
&\quad \times a_n sQ_{U_n})sQ_{U_{n-1}})^{a_{n-1}}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\
&= kdf(e((a_1 \times \cdots \times a_{n-2} \times a_n)P_{pub}^n, \\
&\quad (a_1 \times \cdots \times a_{n-2} \times a_n)s^n \\
&\quad (Q_{U_1} \times Q_{U_2} \times \cdots \times Q_{U_n}))^{a_{n-1}}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\
&= kdf(e((sP)^n, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_n})s^n)^{(a_1 \times a_2 \times \cdots \times a_n)}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}) \\
&= kdf(e(P^n, (Q_{U_1} \times cdots \\
&\quad \times Q_{U_n}))^{s^n(a_1 \times a_2 \times \cdots \times a_n)}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_n}).
\end{aligned}$$

User n can compute the session key directly.

4.2 Join Algorithm

Let $U_{n+1}, U_{n+2}, \cdots, U_{n+m}$ be the set of users who will join the initial group U_0 , $U_j = U_1, \cdots, U_{n+m}$.

$$ID_j = ID_{U_1} \| \cdots \| ID_{U_{n+m}}.$$

As in the above protocol the user U_n is the group leader of this new group U_j also. When a new user joins the group it register itself to the group leader U_n by sending its identity $ID_{U_{n+i}}$. Then the join algorithm is executed in the following way:

Private Key Extraction. For each new registered user U_{n+i} the group leader U_n computes the public key $PK_{U_{n+i}} = Q_{U_{n+i}} = H(U_{n+i})$ and distributes the corresponding static private key $SK_{U_{n+i}} = sQ_{U_{n+i}}$ to the new joined users via a secure channel.

Round 1.

Step 1: The group leader U_n broadcasts in the group:

$$U_n \rightarrow: (\{U_1, \cdots, U_{n-1}, U_{n+1}, \cdots, U_{n+m}\}, V_i, W_i),$$

where $i = 1, 2, ..., n$.

Step 2: Each user $U_{n+i}, i = 1, ..., m$

2.1 Choose his ephemeral private keys $a_{n+i} \leftarrow^R \mathbb{Z}_n$;

2.2 Computes $K_{n+i} = sQ_{U_{n+i}} + a_{n+i}$ and $M_{n+i} = h_1(U_1, \cdots, U_{n-1}, U_{n+1}, \cdots, U_{n+m}, a_{n+i})$;

2.3 Computes $V_{n+i} = a_{n+i}P_{pub}$ and $W_{n+i} = a_{n+i}SK_{U_{n+i}}$;

2.4 Sends: $U_{n+i} \rightarrow U_n$: $(U_{n+i}, K_{n+i}, M_{n+i}, V_{n+i}, W_{n+i})$.

Step 3: The User U_n: $i = i, ..., m$

3.1 Computes $a_{n+i} = K_{n+i} - sQ_{U_{n+i}}$;

3.2 Checks if $M_{n+i} = h_1(U_1, \cdots, U_{n-1}, U_{n+1}, \cdots, U_{n+m}, a_{n+i})$;
if atleast one equality does not hold, he quits;

3.3 Broadcasts:
$U_n \rightarrow: (U_1, \cdots, U_{n-1}, U_{n+1}, \cdots, U_{n+m}, V_{n+i}, W_{n+i})$.

Round 2.

User U_1 computes $K_1 = e((V_2 \times V_3 \times \cdots \times V_{n+m})P_{pub}, (W_2 \times W_3 \times \cdots \times W_{n+m})SK_{U_1})^{a_1}$

$\vdots$

User U_{n-1} computes $K_{n-1} = e((V_1 \times \cdots \times V_{n-2} \times V_n \times \cdots \times V_{n+m})P_{pub}, (W_1 \times \cdots \times W_{n-2} \times W_n \times \cdots \times W_{n+m})SK_{U_{n-1}})^{a_{n-1}}$.

User U_{n+1} computes $K_{n+1} = e((V_1 \times \cdots \times V_n \times V_{n+2} \times \cdots \times V_{n+m})P_{pub}, (W_1 \times \cdots \times W_n \times W_{n+2} \times \cdots \times W_{n+m})SK_{U_{n+1}})^{a_{n+1}}$

$\vdots$

User U_{n+m} computes $K_{n+m} = e((V_1 \times V_2 \times \cdots \times V_{n+m-1})P_{pub}, (W_1 \times W_2 \times \cdots \times W_{n+m-1})SK_{U_{n+m}})^{a_{n+m}}$.

Key Computation. Each user $U_i, i = 1, 2, ..., n+m-1$. Let $kdf : G_2 \times \underbrace{G_1 \times G_1 \times \cdots \times G_1}_{(n+m)times} \rightarrow \{0,1\}^*$ be a key derivation function which can be readily found in a number of standard documents. Thus the shared common group session key,

$$\begin{aligned}
K &= kdf(K_1, Q_{U_1}, Q_{U_2}, ..., Q_{U_{n+m}}) \\
&= kdf(K_{n-1}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(K_{n+1}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&\quad \vdots \\
&= kdf(K_{n+m}, Q_{U_1}, Q_{U_2}, ..., Q_{U_{n+m}}).
\end{aligned}$$

For user U_1,

$$\begin{aligned}
K &= kdf(K_1, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((V_2 \times V_3 \times \cdots \times V_{n+m})P_{pub}, \\
&\quad (W_2 \times W_3 \times \cdots \times W_{n+m})SK_{U_1})^{a_1}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_2P_{pub} \times a_3P_{pub} \times \cdots \\
&\quad \times a_{n+m}P_{pub})P_{pub}, (a_2sQ_{U_2} \times a_3sQ_{U_3} \times \cdots \\
&\quad \times a_{n+m}sQ_{U_{n+m}})sQ_{U_1})^{a_1}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_2 \times a_3 \times \cdots \times a_{n+m})P_{pub}^{n+m}, \\
&\quad (a_2 \times a_3 \times \cdots \times a_{n+m})s^{n+m}Q_{U_1} \times Q_{U_2} \times \\
&\quad \cdots \times Q_{U_{n+m}}))^{a_1}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((sP)^{n+m}, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_{n+m}})s^{n+m})^{(a_1 \times a_2 \times \cdots \times a_{n+m})}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e(P^{n+m}, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_{n+m}}))^{s^{n+m}(a_1 \times a_2 \times \cdots \times a_{n+m})}, \\
&\quad Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}).
\end{aligned}$$

For user U_{n-1},

$$\begin{aligned}
K &= kdf(K_{n-1}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((V_1 \times \cdots \times V_{n-2} \times V_n \times \\
&\quad \cdots \times V_{n+m})P_{pub}, (W_1 \times \cdots \times W_{n-2} \times W_n \\
&\quad \times \cdots \times W_{n+m})SK_{U_{n-1}})^{a_{n-1}}, Q_{U_1}, \\
&\quad \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1P_{pub} \times \cdots \times a_{n-2}P_{pub} \times a_nP_{pub} \\
&\quad \times \cdots \times a_{n+m}P_{pub})P_{pub}, (a_1sQ_{U_1} \times \cdots \\
&\quad \times a_{n-2}sQ_{U_{n-2}} \times a_nsQ_{U_n} \times \cdots \\
&\quad \times a_{n+m}sQ_{U_{n+m}})sQ_{U_{n-1}})^{a_{n-1}}, Q_{U_1}, \\
&\quad \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1 \times \cdots \times a_{n-2} \times a_n \times \cdots \\
&\quad \times a_{n+m})P_{pub}^{n+m}, (a_1 \times \cdots \times a_{n-2} \times a_n \times \cdots \\
&\quad \times a_{n+m})s^{n+m}(Q_{U_1} \times \cdots \times Q_{U_{n-2}} \times Q_{U_n} \times \\
&\quad \cdots \times Q_{U_{n+m}})Q_{U_{n-1}})^{a_{n-1}}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((sP)^{n+m}, (Q_{U_1} \times \cdots \times Q_{U_{n-2}} \times Q_{U_n} \times \\
&\quad \cdots \times Q_{U_{n+m}})s^{n+m}Q_{U_{n-1}})^{a_1 \times a_2 \times \cdots \times a_{n+m}}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e(P^{n+m}, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_{n+m}}))^{s^{n+m}(a_1 \times a_2 \times \cdots \times a_{n+m})}, Q_{U_1}, \cdots, \\
&\quad Q_{U_{n+m}}).
\end{aligned}$$

For user U_{n+1},

$$\begin{aligned}
K &= kdf(K_{n+1}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((V_1 \times \cdots \times V_n \times V_{n+2} \cdots \\
&\quad \times V_{n+m})P_{pub}, (W_1 \times \cdots \times W_n \times W_{n+2} \times \cdots \\
&\quad \times W_{n+m})SK_{U_{n+1}})^{a_{n+1}}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1P_{pub} \times \cdots \times a_nP_{pub} \times a_{n+2}P_{pub} \times \cdots \\
&\quad \times a_{n+m}P_{pub})P_{pub}, (a_1sQ_{U_1} \times \cdots \times a_nsQ_{U_n} \\
&\quad \times a_{n+2}sQ_{U_{n+2}} \times \cdots \times a_{n+m}sQ_{U_{n+m}})sQ_{U_{n+1}})^{a_{n+1}}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1 \times \cdots \times a_n \times a_{n+2} \times \cdots \times a_{n+m})P_{pub}^{n+m}, \\
&\quad (a_1 \times \cdots \times a_n \times a_{n+2} \times \cdots \times a_{n+m})s^{n+m}(Q_{U_1} \\
&\quad \times \cdots \times Q_{U_n} \times Q_{U_{n+2}} \times \cdots \times Q_{U_{n+m}})Q_{U_{n+1}})^{a_{n+1}}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((sP)^{n+m}, (Q_{U_1} \times \cdots \times Q_{U_n} \times Q_{U_{n+2}} \times \\
&\quad \cdots \times Q_{U_{n+m}})s^{n+m}Q_{U_{n+1}})^{a_1 \times a_2 \times \cdots \times a_{n+m}}, Q_{U_1}, \\
&\quad \cdots, Q_{U_{n+m}}) \\
&= kdf(e(P^{n+m}, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_{n+m}}))^{s^{n+m}(a_1 \times a_2 \times \cdots \times a_{n+m})}, Q_{U_1}, \\
&\quad \cdots, Q_{U_{n+m}}).
\end{aligned}$$

For userU_{n+m},

$$\begin{aligned}
K &= kdf(K_{n+m}, Q_{U_1}, Q_{U_2}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((V_1 \times V_2 \times \cdots \times V_{n+m-1})P_{pub}, (W_1 \\
&\quad \times W_2 \times \cdots \times W_{n+m-1})SK_{U_{n+m}})^{a_{n+m}}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1P_{pub} \times a_2P_{pub} \times \cdots \times a_{n+m-1}P_{pub})P_{pub}, \\
&\quad (a_1sQ_{U_1} \times a_2sQ_{U_2} \times \cdots \\
&\quad \times a_{n+m-1}sQ_{U_{n+m-1}})sQ_{U_{n+m}})^{a_{n+m}}, \\
&\quad Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((a_1 \times a_2 \times \cdots \times a_{n+m-1})P_{pub}^{n+m}, \\
&\quad (a_1 \times a_2 \times \cdots \times a_{n+m-1})s^{n+m}(Q_{U_1} \times Q_{U_2} \\
&\quad \times \cdots \times Q_{U_{n+m}}))^{a_{n+m}}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e((sP)^{n+m}, (Q_{U_1} \times Q_{U_2} \times \cdots \\
&\quad \times Q_{U_{n+m}})s^{n+m})^{a_1 \times a_2 \times \cdots \times a_{n+m}}, Q_{U_1}, \cdots, Q_{U_{n+m}}) \\
&= kdf(e(P^{n+m}, (Q_{U_1} \times \cdots \\
&\quad \times Q_{U_{n+m}}))^{s^{n+m}(a_1 \times a_2 \times \cdots \times a_{n+m})}, Q_{U_1}, \\
&\quad \cdots, Q_{U_{n+m}}).
\end{aligned}$$

4.3 Leave Algorithm

Without loss of generality, we assume that $U_{v-1} = \{U_1, U_2, \cdots, U_n\}$ is the current group that $L = \{U_1, \cdots, U_m\}$ is the set of leaving users. Then

$$\begin{aligned}
U_v &= \{U_{m+1}, \cdots, U_{m+n}, U_n\} \\
ID_v &= ID_{U_{m+1}} \| \cdots \| ID_{U_{m+n}} \| ID_{U_m}.
\end{aligned}$$

Then the leave algorithm is executed in the following way.

Round 1:

Step 1: The group leader U_n broadcasts in the group:

$$U_n \rightarrow: (\{U_{m+1}, \cdots, U_{m+n}, U_n\}).$$

Step 2: Each user $U_{m+i}, i = 1, ..., n$

2.1 Choose his ephemeral private keys $a_{m+i} \leftarrow^R \mathbb{Z}_n$;

2.2 Computes $K_{m+i} = sQ_{U_{m+i}} + a_{m+i}$ and

$$M_{m+i} = h_1(U_{m+1}, \cdots, U_{m+n}, a_{m+i});$$

2.3 Computes $V_{m+i} = a_{m+i}P_{pub}$ and $W_{m+i} = a_{m+i}SK_{U_{m+i}}$;

2.4 Sends:

$$U_{m+i} \rightarrow U_n : (U_{m+i}, K_{m+i}, M_{m+i}, V_{m+i}, W_{m+i}).$$

Step 3: The User U_n: $i = i, ..., n$

3.1 Computes $a_{m+i} = K_{m+i} - sQ_{U_{m+i}}$;

3.2 Checks if $M_{m+i} = h_1(U_{m+1}, \cdots, U_{m+n}, a_{m+i})$; if atleast one equality does not hold, he quits;

3.3 Broadcasts: $U_n \rightarrow$: $(\{U_{m+1}, \cdots, U_{m+n}\}, V_{m+i}, W_{m+i})$.

Round 2:

User U_{m+1} computes $K_{m+1} = e((V_{m+2} \times \cdots \times V_{m+n} \times V_n)P_{pub}, (W_{m+2} \times \cdots \times W_{m+n} \times W_n)SK_{U_{m+1}})^{a_{m+1}}$

$\vdots$

User U_{m+n} computes $K_{m+n} = e((V_{m+1} \times \cdots \times V_{m+n-1} \times V_n)P_{pub}, (W_{m+1} \times \cdots \times W_{m+n-1} \times W_n)SK_{U_{m+n}})^{a_{m+n}}$

Key Computation: Each user $U_{m+i}, i = 1, 2, ..., n$. Let $kdf : G_2 \times \underbrace{G_1 \times G_1 \times \cdots \times G_1}_{(m+n+1)times} \rightarrow \{0,1\}^*$ be a key derivation function which can be readily found in a number of standard documents. Thus the shared common group session key,

$$\begin{aligned} K &= kdf(K_{m+1}, Q_{U_{m+1}}, ..., Q_{U_{m+n}}, Q_{U_n}) \\ &\vdots \\ &= kdf(K_{(m+n)}, Q_{U_{m+1}}, ..., Q_{U_{m+n}}, Q_{U_n}). \end{aligned}$$

For user U_{m+1},

$$\begin{aligned} K &= kdf(K_{m+1}, Q_{U_{m+1}}, ..., Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((V_{m+2} \times \cdots \times V_{m+n} \times V_n)P_{pub}, \\ &\quad (W_{m+2} \times \cdots \times W_{m+n} \times W_n)SK_{U_{m+1}})^{a_{m+1}}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((a_{m+2}P_{pub} \times \cdots \times a_{m+n}P_{pub} \\ &\quad \times a_nP_{pub})P_{pub}, (a_{m+2}sQ_{U_{m+2}} \times \cdots \\ &\quad \times a_{m+n}sQ_{U_{m+n}} \times a_nsQ_{U_n})sQ_{U_{m+1}})^{a_{m+1}}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((a_{m+2} \times \cdots \times a_{m+n} \times a_n)P_{pub}^{m+n+1}, \\ &\quad (a_{m+2} \times \cdots \times a_{m+n} \times a_n)s^{m+n+1}(Q_{U_{m+1}} \\ &\quad \times \cdots \times Q_{U_{m+n}} \times Q_{U_n}))^{a_{m+1}}, Q_{U_{m+1}}, \cdots, \\ &\quad Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((sP)^{m+n+1}, (Q_{U_{m+1}} \times \cdots \times Q_{U_{m+n}} \\ &\quad \times Q_{U_n})s^{m+n+1})^{a_{m+1} \times \cdots \times a_{m+n} \times a_n}, Q_{U_{m+1}}, \\ &\quad \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e(P^{m+n+1}, (Q_{U_{m+1}} \times \cdots \times Q_{U_{m+n}} \\ &\quad \times Q_{U_n}))^{s^{m+n+1}(a_{m+1} \times \cdots \times a_{m+n} \times a_n)}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}). \end{aligned}$$

For user U_{m+n},

$$\begin{aligned} K &= kdf(K_{m+n}, Q_{U_{m+1}}, ..., Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((V_{m+1} \times \cdots \times V_{m+n-1} \times V_n)P_{pub}, \\ &\quad (W_{m+1} \times \cdots \times W_{m+n-1} \times W_n)SK_{U_{m+n}})^{a_{m+n}}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((a_{m+1}P_{pub} \times \cdots \times a_{m+n-1}P_{pub} \\ &\quad \times a_nP_{pub})P_{pub}, (a_{m+1}sQ_{U_{m+1}} \times \cdots \\ &\quad \times a_{m+n-1}sQ_{U_{m+n-1}} \times a_nsQ_{U_n})sQ_{U_{m+n}})^{a_{m+n}}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((a_{m+1} \times \cdots \times a_{m+n-1} \times a_n)P_{pub}^{m+n+1}, \\ &\quad (a_{m+1} \times \cdots \times a_{m+n-1} \times a_n)s^{m+n+1}(Q_{U_{m+1}} \\ &\quad \times \cdots \times Q_{U_{m+n}} \times Q_{U_n}))^{a_{m+n}}, Q_{U_{m+1}}, \cdots, \\ &\quad Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e((sP)^{m+n+1}, (Q_{U_{m+1}} \times \cdots \times Q_{U_{m+n}} \\ &\quad \times Q_{U_n})s^{m+n+1})^{a_{m+1} \times \cdots \times a_{m+n} \times a_n}, Q_{U_{m+1}}, \\ &\quad \cdots, Q_{U_{m+n}}, Q_{U_n}) \\ &= kdf(e(P^{m+n+1}, (Q_{U_{m+1}} \times \cdots \times Q_{U_{m+n}} \\ &\quad \times Q_{U_n}))^{s^{m+n+1}(a_{m+1} \times \cdots \times a_{m+n} \times a_n)}, \\ &\quad Q_{U_{m+1}}, \cdots, Q_{U_{m+n}}, Q_{U_n}). \end{aligned}$$

User n can compute the session key directly.

5 Security Analysis

5.1 Known Key Security

From the randomness of $a_i's$ in our proposed group key agreement protocol, the session keys in different key agreements are independent of each other. The knowledge of the previous session keys does not help an adversary to derive any future session key. Hence our proposed group key agreement protocol provides known key security.

5.2 Forward Secrecy

Even if a long term private key $SK_{ID}(= sQ_{ID})$ of our proposed group key agreement is compromised , the data protected with the previous session key K is still secure because the derivation of K requires the knowledge of previous random values $a_i's$. Therefore our group key agreement protocol has the property of (perfect) forward secrecy.

5.3 Trivial Attack

An attacker may directly try to compute the group key K from the transmitted message $[U_i \rightarrow U_n :$

$(U_i, \{U_1, K_i, M_i, V_i, W_i), i = 1, ..., n-1]$ but due to difficulties of the DLP and onewayness of hash function the trivial attack is not possible in the proposed protocol.

5.4 Key Compromise Impersonation

Suppose that an adversary who know's user $U_1's$ long term private key SQ_{U_1} and want to masquerade with the group leader U_n. Then first he chooses a random value $a_1^{'} \in^R \mathbb{Z}_n$ and calculate

$$K_1 = sQ_{U_1} + a_1^{'}$$

but to verify the correspondence of his guessed random value $a_1^{'}$. He has to compute

$$M_1 = h_1(U_1, U_2, \cdots, U_{n-1}, a_1)$$

which is impossible since he requires the value of a_1 which is the ephemeral private key of the user U_1. Hence U_n will found this un-equality. So this type of attacks are also not possible.

5.5 Unknown Key Share

In our proposed GKA protocol consider the special case (i.e.for $n = 2$), the shared secret $S_{1,2} = S_{2,1} = kdf(e(P^2, (Q_{U_1} \times Q_{U_2}))^{s^{2(a_1 \times a_2)}})$, between U_1 and U_2 involves both members long term private and public keys. This ensures that only U_1 and U_2 who own the corresponding long-term private keys can obtain the same group key and can compute valid key confirmations. Any other entity cannot obtain the same group key. It is impossible that U_1 ends up believing that she/he shares a key with U_2 and although this is in fact the case, while U_2 mistakenly believes that the key is instead shared with another entity E.

5.6 Message Confidentiality

In our proposed scheme the size of shared common group session key is totally depends on the number of users in the current group and their ephemeral private keys. So when a group user want to leave or a new user want to join the group the session key size is obviously change. Also in our proposed scheme in join or leave algorithm the joining and leaving members can not know the number of participant in previous or subsequent group and they also don't know their private keys .

Hence the joining member can not compute previous session keys and leaving member can not compute the subsequent session keys .

6 Comparison

We now compare our protocol with another dynamic group key agreement protocols [18, 19, 35]. We will use the following notations.

1) *Round*: The total number of rounds.
2) Mul: The total number of scalar multiplications and modular multiplication .
3) $Msize$: The maximum number of messages sent by per user.
4) P/E: The total number of pairing computations and modular exponentiations.

Table 1: Setup algorithm -A set of users $U_{[1,...,n]}$

Protocol	Round	M size	Mul	P/E
[18]	$O(n)$	$O(n^2)$	0	$O(n^2)$
[35]	2	$O(n)$	$O(n^2)$	$O(n^2)$
[19]	1	$O(n^2)$	$O(n^2)$	O(n)
Ours	2	$O(5n)$	$O(3n)$	$O(n)$

We observe from Table 1, in our protocol the message size is $O(5n)$ which is linear as compare to [18] and [19]. Similarly, the total number of scalar multiplication is of quadratic order *i.e.* $O(n^2)$ in [35] and [19]. But in our proposed scheme Mul is $O(3n)$ which is again linear. Also in [18] and [35] the pairing computation P/E is again quadratic in order.

We observe from Table 2 the total number of users is $(n+m)$. So in our proposed scheme M size is $O(5(n+m))$, total number of scalar multiplication is $O(3(n+m))$ and the pairing computation is $O(n)$. Hence in join algorithm of our proposed scheme all cases are liner in order as compare to other recent protocols [18, 19, 35].

In Table 3 of leave algorithm the size of the resulting set of users is $(n-m)$. The total number of scalar multiplication in [35], [19] and pairing computation in [35] is of quadratic order $O(n-m)^2$. But in this table, we see that in case of our proposed scheme the M size, total number of scalar multiplication and P/E all are linear in order.

7 Conclusion

With the increasing need of authenticated and secure communication, ID based two round DAGKA protocol is presented here, which resist to all the known attacks. Our protocol also provides forward and backward confidentiality which is the important feature in case of dynamic key agreements. In the last we have given the comparison of our protocol with other recent dynamic group key agreement protocols.

References

[1] R. Barua, R. Dutta, and P. Sarkar, "Extending Joux protocol to multi party key agreement", in *Proceedings of Indocrypt 2003*, LNCS 2904, pp. 205–217, Springer-Verlag, 2003.

Table 2: Join algorithm -A set of users $U_{[n+1,\ldots,n+m]}$ join the set of users $U_{[1,\ldots,n]}$ resulting a set of size $n+m$

Protocol	Round	M size	Mul	P/E
[18]	$O(n+m)$	$O(n+m)^2$	0	$O(m(n+m))$
[35]	2	$O(m)$	$O(m(n+m))$	$O(m(n+m))$
[19]	1	$O(m(n+m))$	$O(m(n+m))$	$O(n+m)$
Ours	2	$O(5(n+m))$	$O(3(n+m))$	$O(n+m)$

Table 3: Leave algorithm -A set of users $U_{[l_1,\ldots,l_m]}$ leave the set of users $U_{[1,\ldots,n]}$ resulting a set of size $n-m$

Protocol	Round	M size	Mul	P/E
[18]	1	$O(n-m)$	0	$O(n-m)$
[35]	2	$O(n-m)$	$O((n-m)^2)$	$O((n-m)^2)$
[19]	0	0	$O((n-m)^2)$	$O(n-m)$
Ours	2	$O(5(n-m))$	$O(3(n-m))$	$O(n-m)$

[2] E. Bresson, O. Chevassut, and D. Pointcheval, "Provably authenticated group diffie-hellman key exchange - The dynamic case", In *Proceedings of Asiacrypt 2001*, LNCS 2248, pp. 290–309, Springer-Verlag, 2001.

[3] D. Bohen and M. Franklin, "Identity based encryption from the Weil pairing", *SIAM Journal of Computing*, vol. 32, no. 3, pp. 586–615, 2001.

[4] M. Burmester and Y. Desmedt, " A secure and efficient conference key distribution system", In *Proceedings of Eurocrypt 1994*, LNCS 950, pp. 275–286, Springer-Verlag, 1995.

[5] Z. Chen, "Security analysis on Nalla-Reddy's ID-based tripartite authenticated key agreement protocol", *Cryptology ePrint Archive*, Report 2003/103, 2003. (`http://eprint.iacr.org/2003/103`)

[6] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairing", in *16th IEEE Security Fudation Wrkshop* , pp. 219—233, 2003.

[7] Q. Cheng and C. Tang, "Cryptanalysis of an ID-based authenticated dynamic group key agreement with optimal round", *Internatioal Journal of Network Security*, vol. 17, no. 6, pp. 678–682, Nov. 2015.

[8] Z. Cheng and L. Chen, "On the security proof of McCullagh-Barreto's key agreement protocol and its variants", *Internatioal Journal of Security and Networks*, Special Issue on Cryptography in Network.

[9] K. Y. Choi, J. Y. Hwang, and D. H. Lee, "Efficient ID-based group key agreement with bilinear maps", in *Proceedings of PKC 2004*, LNCS 2947, pp. 130–144, Springer-Verlag, 2004.

[10] K. Choo, "Revisit of McCullagh-Barreto two party ID-based authentication key agreement protocols", 2019. (`http://eprint.iacr.org/2004/343.pdf`)

[11] N. M. Cullagh and P. Barreto, "A new two-party identity-based authenticated key agreement", Cryptology ePrint Archive, Report, 2004/122, 2004. (`http://eprint.iacr.org/2004/122.pdf`)

[12] W. Diffie and M. Hellman, " New directions in cryptography", *IEEE Transactions on Information Theory*, vol. 6, pp. 644–654, 1976.

[13] R. Dutta and R. Barua, "Overview of key agreement protocols", *IACR Cryptology ePrint Archive*, 2005.

[14] R. Dutta and R. Barua, "Constant round dynamic group key agreement", in *Proceedings of ISC 2005*, 2005. (`http://eprint.iacr.org/2005/221`)

[15] F. Hess, "Efficient identity based signature schemes based on pairings", in *Proceedings of SAC 2002*, LNCS 2595, pp. 310–324, Springer-Verlag, 2002.

[16] I. Ingemarsson, D. Tang, and C. Wang, "A conference key distribution system", *IEEE Transactions on Information Theory*, vol. 28, pp. 714–720, 1982.

[17] A. Joux, "A one round protocol for tripartite Diffie-Hellman", in *Proceedings of ANTS 4*, LNCS 1838, pp. 385–394, 2000.

[18] H. J. Kim, S. M. Lee, and D. H. Lee, "Constant-round authenticated group key exchange for dynamic groups", in *Proceedings of Asiacrypt 2004*, LNCS 3329, pp. 245–259, 2004.

[19] F. Li, D. Xie, W. Gao, J. Yan, and X. A. Wang, "Round-optimal ID-based dynamic authenticated group key agreement", *International Journal of High Performance Systems Architecture*, vol. 6, no. 3, pp. 153, 2016.

[20] H. S. Lee and Y.R.Lee, "Identity based authenticated key agreement from pairings", *Commun. Korean Math.*, Soc. 20, no. 4, pp. 849–859, 2005.

[21] A. Menezes, P. V. Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, 1996.

[22] D. Nalla and K. C. Reddy, "Identity based authenticated group key agreement protocol", in *Proceedings of Indocrypt 2002*, LNCS 2551, pp. 215–233, Springer-Verlag, 2002.

[23] D. Nalla and K. C. Reddy, "ID-based tripartite authenticated key agreement protocols from pairings",

Cryptology ePrint Archive, Report 2003/004, 2003. (http://eprint.iacr.org/2003/004)

[24] E. Okamoto, "Proposal for identity-based key distribution system", *Electronics Letters*, vol. 22, pp. 1283–1284, 1986.

[25] R. S. Ranjani, D. L. Bhaskari, and P. S. Avadhani, "An extended ID based authenticated asymmetric group key agreement protocol", *International Journal of Network Security*, vol. 17, no. 5, pp. 510–516, Sept. 2015.

[26] A. Shamir, "How to share a secret", *Communications of ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[27] M. Scott, "Authenticated ID-based key exchange and remote log-in with insecure token and PIN number", 2002. (http://eprint.iacr.org/2002/164.pdf)

[28] K. Shim, "Cryptanalysis of ID-based tripartite authenticated key agreement protocol", *Cryptology ePrint Archive*, Report 2003/115, 2003. (http://eprint.iacr.org/2003/115)

[29] K. Shim, "Efficient ID-based authenticated key agreement protocol based on the Weil pairing", *Electronics Letters*, vol. 39, no. 8, pp. 653–654, 2003.

[30] N. P. Smart, "Identity-based authenticated key agreement protocol based on Weil pairing", *Cryptology ePrint Archive*, Report 2001/111, 2001. (http://eprint.iacr.org/)

[31] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman key distribution extended to group communication", in *Proceedings of ACM CCS 1996*, pp. 31-37, 1996.

[32] M. Steiner, G. Tsudik and M.Waidner, "Cliques: A new approach to group key agreement", in *IEEE Conference on Distributed Computing Systems*, pp. 380–380, May 1998.

[33] S. Sun and B. Hsieh, "Security analysis of Shim's authenticated key agreement protocols from pairing", 2003. (http://eprint.iacr.org/2003/113.pdf)

[34] K. Tanaka and E. Okamoto, "Key distribution system for mail systems using ID-related information directory", *Computers and Security*, vol. 10, pp. 25–33, 1991.

[35] J. K. Teng, C. K. Wu and C. M. Tang, "An ID-based authenticated dynamic group key agreement with optimal round", *Science China Information Sciences*, vol. 55, no. 11, pp. 2542–2554, 2012.

[36] S.B. Wilson and A. Menezes, "Unknown key share attacks on the station-to-staion (STS) protocol", in *Proceedings of Second International Workshop on Practice and Theory in Public Key Cryptography (PKC'99)*, LNCS 1560, pp. 154–170, 1999.

[37] F. Zhang, S. Liu, and K. Kim, "ID-based one round authenticated tripartite key agreement protocol with pairings", *Crytology ePrint Archive*, 2002. (http://eprint.iacr.org/2002/122.46)

Biography

Shruti Nathani received the B.Sc., M.Sc. and M.Phil degrees in Mathematics from Pt. Ravishankar Shukla University, Raipur, Chhattisgarh, India in 2008, 2010 and 2011 respectively. She is currently a PhD candidate at the Department of Mathematics in Govt. N.PG. College of Science affiliated from Pt. Ravishankar Shukla University, Raipur, Chhattisgarh, India. Her main research interests include public key cryptography, especially in group oriented cryptography and group key establishment protocols.

B. P. Tripathi, Assistant Professor, Deptt. of Mathematics, Govt. N.PG. college of Science Raipur. The institute is affiliated to Pt. Ravishankar Shukla University, Raipur, Chhattisgarh, India. His field of interest are Non-linear Analysis, Fixed point theory and Public Key Cryptography. He has teaching experience of 24 years of undergraduate and postgraduate classes. He has written 2 books and published 35 research papers in various National and International journals. Two scholar has awarded Ph.D. degree and recently four scholars are pursuing their research work under the supervision of Dr. Tripathi.

Shaheena Khatoon received the B.Sc., M.Sc. and M.Phil degrees in Mathematics from Pt. Ravishankar Shukla University, Raipur, Chhattisgarh, India in 2005, 2007 and 2009 respectively. She joined School of studies in Mathematics, Pt. Ravi Shankar Shukla University, Raipur, Chhattisgarh, India for her research work.