A Secure Deniable Authentication Protocol based on Bilinear Diffie-Hellman Algorithm

Jayaprakash Kar
Department of Information Technology
Al Musanna College of Technology
Sultanate of Oman
jayaprakashkar@yahoo.com

Banshidhar Majhi
College of Computer Science
King Khalid University
Saudi Arabia
bmajhi@nitrkl.ac.in

Syed Hamid Hasan
Department of Information System
King Abdul Aziz University
Saudi Arabia
shh786@hotmail.com

***Abstract*—This paper describes a new deniable authentication protocol whose security is based Diffe-Hellman (CDH) Problem of type Decisional Diffie-Hellman(DDH) and the Hash Diffie-Hellman (HDDH) problem.This protocol can be implemented in low power and small processor mobile devices such as smart card, PDA etc which work in low power and small processor. A deniable authentication protocol enables a receiver to identify the true source of a given message, but not to prove the identity of the sender to a third party. This property is very useful for providing secure negotiation over the internet. Our proposed protocol will be achieving the most three security requirement like deniable authentication, Confidentialities and also it is resistant against Man-in middle Attack.**

Keywords: deniable authentication, ECDLP, ECDHP, HDDH, Bilinear pairing.

I. Introduction

Nowadays, authentication had emerged to be an essential communication process in key establishment. In fact, the aim of this process is to assure the receiver by verifying the digital identity of the sender, especially when communicating via an insecure electronic channel. Authentication can be realized by the use of digital signature in which the signature (signers private key) is tied to the signer as well as the message being signed. This digital signature can later be verified easily by using the signers public key. Hence, the signer will not be able to deny his participation in this communication. Generally, this notion is known as non-repudiation. However, under certain circumstances such as electronic voting system, online shopping and negotiation over the internet, the non-repudiation property is undesirable. It is important to note that in these applications, the senders identity should be revealed only to the intended receiver. Therefore, a significant requirement for the protocol is to enable a receiver to identify the source of a given message and at the same time, unable to convince to a third party on the identity of the sender even if the receiver reveal his own secret key to the third party.

II. Background

In this section we brief overview of Computational Diffie-Hellman (CDH) problem, Decisional Diffie-Hellman and Hash Diffie-Hellman problem in $\mathbb{G}$ and subsequently describe about the deniable property.

III. Bilinear Pairings and Diffie-Hellman Problems

A. Bilinear Pairings

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups of the same prime order q. Let e be a computable bilinear map $e : \mathbb{G}X\mathbb{G} \rightarrow \mathbb{G}_T$, which satisfies the following properties:

- Bilinear: $e(aP, bQ) = e(P, Q)^{ab}$, where $P, Q \in \mathbb{G}$ and $a, b \in \mathbb{Z}_q^*$.
- Non-degenerate: There exists $P, Q \in \mathbb{G}$ such that $e(P, Q) \neq 1_{\mathbb{G}_T}$
- Computability: There exists an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in \mathbb{G}$.

We call such a bilinear map e is an admissible bilinear pairing, and the Weil or Tate pairing in elliptic curve can give a good implementation of the admissible bilinear pairing.

Definition 1. Bilinear Parameter Generator *: A bilinear parameter generator $\mathcal{G}$ is a probabilistic polynomial time algorithm that takes a security parameter k as input and outputs a 5-tuple $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ as the bilinear parameters, including a prime number q with $|q| = k$, two cyclic groups $\mathbb{G}, \mathbb{G}_T$ of the same order q, an admissible bilinear map $e : \mathbb{G}X\mathbb{G} \rightarrow \mathbb{G}_T$ and a generator P of $\mathbb{G}$*

Definition 2. Bilinear Diffie-Hellman Problem*: Let $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ be a 5-tuple generated by $\mathcal{G}(k)$, and let $a, b, c \in \mathbb{Z}_q^*$. The BDHP in $\mathbb{G}$ is as follows: Given Given (P, aP, bP, cP) with $a, b, c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc} \in \mathbb{G}_T$. The (t, ϵ)-BDH assumption holds in $\mathbb{G}$ if there is no algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{BDH}(\mathcal{A}) = Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}] \geq \epsilon$$

where the probability is taken over all possible choices of (a, b, c). Here the probability is measured over random choices of $a, b, c \in \mathbb{Z}_q^*$ and the internal random operation of A. More formally, for any PPT algorithm $\mathcal{A}$ consider the following experiment:
Let $\mathcal{G}$ be an algorithm which on input 1^k outputs a (description of a) group G of prime order q (with $|q| = k$) along with a generator $P \in \mathbb{G}$. The computational Diffie-Hellman (CDH) problem is the following:

978-1-4244-8767-7 /11/$26.00 ©2011 IEEE

$\underline{\mathbf{Exp}_{\mathcal{G}(k)}^{CDH}}$

1) $(\mathbb{G}, q, P) \leftarrow \mathcal{G}(1^k)$
2) $a, b, c \leftarrow \mathbb{Z}_q^*$
3) $U_1 = aP, U_2 = bP, U_3 = cP$
4) if $W = e(P,P)^{abc}$ return 1 else return 0

We assume that BDHP is a hard computational problem: letting q have the magnitude $2k$ where k is a security parameter, there is no polynomial time (in k) algorithm which has a non-negligible advantage (again, in terms of k) in solving the BDHP for all sufficiently large k.

Definition 3. Decisional Diffie-Hellman Problem *:Let $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ be a 5-tuple generated by $\mathcal{G}(k)$,and let $a, b, c \in \mathbb{Z}_q^*$, the DDH problem in $\mathbb{G}$ is as follows: Given (P, aP, bP, cP), decide whether it is a Diffie-Hellman tuple.*

Definition 4. Hash Decisional Diffie-Hellman Problem *:Let $(q, \mathbb{G}, \mathbb{G}_T, e, g)$ be a 5-tuple generated by $\mathcal{G}(k)$,$\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^l$ be a secure cryptographic hash function, whether l is a security parameter, and let $x, y \in \mathbb{Z}_q^*, h \in \{0,1\}^l$, the HDDH problem in $\mathbb{G}$ is as follows: Given (P, aP, bP, h), decide whether it is a hash Diffie-Hellman tuple $((P, aP, bP, \mathcal{H}(e(P,P)^{ab}))$. If it is right, outputs 1; and 0 otherwise. The (t, ϵ)-HDDH assumption holds in $\mathcal{G}$ if there is no algorithm $\mathcal{A}$ running in time at most t such that*

$$\mathbf{Adv}_{\mathbb{G}}^{HDDH}(\mathcal{A}) = |Pr[\mathcal{A}(P, aP, bP, \mathcal{H}(e(P,P)^{ab})) = 1] - Pr[\mathcal{A}(P, aP, bP, h) = 1]| \geq \epsilon$$

where the probability is taken over all possible choices of (a, b, h).

IV. Deniable property

Deniable authentication protocol is a new security authentication mechanism. Compared with traditional authentication protocols, it has the following two features:

1) It enables an intended receiver to identity the source of a given message.
2) However, the intended receiver can not prove to any third party the identity of the sender

In 1998, Dwork et al. [11] developed a notable deniable authentication protocol based on the concurrent zero-knowledge proof, however the protocol requires a timing constraint and the proof zero-knowledge is subject to a time delay in the authentication process. Auman and Rabin [12] proposed some other deniable authentication protocols based on the factoring problem. In 2001, Deng et al. [**?**] also proposed two deniable authentication protocols based on the factoring and the discrete logarithm problem respectively. Our proposed protocol will be achieving the following properties.

- **Deniable authentication**: The intended receiver can identify the source of a given message, but cannot prove the source to any third party.
- **Authentication**: During the protocol execution, the sender and the intended receiver can authentication each other.
- **Confidentialities**: Any outside adversary has no ability to gain the deniable authentication message from the transmitted transcripts.

An IDbased deniable authentication protocol (IBDAP) consists of the following four algorithms: **Setup**, **Extract**, **Send** and **Receive**. We describe the functions of each as follows.

- **Setup**: On input of the security parameter 1^k the PKG (Private Key Generator) uses this algorithm to produce a pair (params, master-key), where params are the global public parameters for the system and master-key is the master secret key kept secretly by PKG. We assume that params are publicly known so that we do not need to explicitly provide them as input to other algorithms.
- **Extract**: On input of an identity i and the master secret key master-key, the PKG uses this algorithm to compute a public-secret key pair (pk_i, sk_i) corresponding to i.
- **Send**: The sender S uses this algorithm with input (m, sk_S, pk_R) to output a deniable authentication message $\tilde{m}$, where pk_R is the public key of the receiver R.
- **Receive**: The receiver R uses this algorithm with input $(\tilde{m}, m, pk_S, pk_R)$ to output 1 if the deniable authentication message $\tilde{m}$ is valid or 0 otherwise. The above algorithms must have the following consistency requirement. If

$$\tilde{m} \leftarrow \textbf{Send}(m, sk_S,\ pk_R), \text{ then we must have } 1 \leftarrow \textbf{Receive}(\ \tilde{m},\ m,\ pk_S,\ pk_R).$$

V. Security model

Security Notions In this subsection, we explain the security notions of ID-based deniable authentication protocol. We first recall the usual security notion: the unforgeability against chosen message attacks (Goldwasser et al., 1988), then we consider another security notion: the deniablity of deniable authentication protocol. **Player**. Let $P = \{\mathcal{P}_0, \mathcal{P}_1, \ldots \mathcal{P}_n\}$ be a set of players who may be included in the system. Each player $\mathcal{P}_i \in P$ get his public-secret key pair (pk_i, sk_i) by providing his identity i to the **Extract** algorithm. A player $\mathcal{P}_i \in P$ is said to be fresh if $\mathcal{P}_i$'s secret key sk_i has not been revealed by an adversary; while if $\mathcal{P}_i$s secret key sk_i has been revealed, $\mathcal{P}_i$ is then said to be corrupted. With regard of the unforgeability against chosen-message attacks, we define the security notion via the following game played by a challenger and an adversary. [**Game 1**]

- Initial: The challenger runs Setup to produce a pair $(params, master - key)$, gives the resulting $params$ to the adversary and keeps the master-key secretly.
- Probing: The challenger is probed by the adversary who makes the following queries.
- Extract: The challenger first sets $\mathcal{P}_0, \mathcal{P}_1$ to be fresh players, which means that the adversary is not allowed to make Extract query on $\mathcal{P}_0$ or $\mathcal{P}_1$. Then, when the adversary submits an identity i of player $\mathcal{P}_i, (i = 0, 1)$, to the challenger. The challenger responds with the public-secret key pair (pk_i, sk_i) corresponding to i to the adversary.

- Send: The adversary submits the requests of deniable authentication messages between $\mathcal{P}_0$ and $\mathcal{P}_0$. The challenger responds with deniable authentication messages with respect to $\mathcal{P}_0$ (resp. $\mathcal{P}_1$) to $\mathcal{P}_1$ (resp $\mathcal{P}_0$).
- Forging: Eventually, the adversary outputs a valid forgery $\tilde{m}$ between $\mathcal{P}_0$ and $\mathcal{P}_1$. If the valid forgery $\tilde{m}$ was not the output of a Send query made during the game, we say the adversary wins the game.

Definition 5. (Unforgeability). *Let A denote an adversary that plays the game above. If the quantity $Adv^{UF}_{IBDAP}[A] = Pr[Awins]$ is negligible we say that the ID-based deniable authentication protocol in question is existentially unforgeable against adaptive chosen-message attacks.*

To capture the property of deniablity of deniable authentication protocol, we consider the following game run by a challenger. **[Game 2]**

- Initial: Let $\mathcal{P}_0$ and $\mathcal{P}_1$ be two honest players that follow the deniable authentication protocol, and let $\mathcal{D}$ be the distinguisher that is involved in the game with $\mathcal{P}_0$ and $\mathcal{P}_0$.
- Challenging: The distinguisher $\mathcal{D}$ submits a message $m \in \{0,1\}^*$ to the challenger. The challenger first randomly chooses a bit $b' \in \{0,1\}^*$, then invokes the player P_b to make a deniable authentication message $\tilde{m}$ on m between $\mathcal{P}_0$ and $\mathcal{P}_1$. In the end, the challenger returns $\tilde{m}$ to the distinguisher $\mathcal{D}$.
- Guessing: The distinguisher $\mathcal{D}$ returns a bit $b \in \{0,1\}^*$. We say that the distinguisher $\mathcal{D}$ wins the game if $b = b'$.

Definition 6. (Deniablity). *Let D denote the distinguisher that is involved the game above. If the quantity $Adv^{DN}_{IBDAP}[D] = |Pr[b = b'] - \frac{1}{2}|$ is negligible we say that the ID-based deniable authentication protocol in question is deniable.*

VI. Proposed Protocol

Our proposed protocol involves two entities : a sender S and a intended receiver R. It is described as follows.

- **Setup** Let $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ be a 5-tuple generated by polynomial time algorithm $\mathcal{G}(k)$ and let $\mathcal{H} : \{0,1\}^* \to \{0,1\}^l$ be a secure cryptographic hash function which is of collision free. The certificate CEA chooses $Q \in \mathbb{G}$ as one public parameter of the protocol. Let $P \in \mathbb{G}$ be the generator of the group $\mathbb{G}$, so $\exists t \in \mathbb{Z}_q^*$ such that $Q = t \cdot P$. Let $E_{\Pi_{prv}}()$ a public key digital signature algorithm over Elliptic Curves using pairings technique. The private key Π_{prv} is only known by the sender S and Π_{pub} is a public key. S has a certificate $crt = crt(\Pi_{pub}; \sigma)$ issued by the CEA. The certificate contains the public key Π_{pub} for $E()$, and the signature of CEA for the signed certificate. The receiver can also obtain Π_{pub} from the CEA and verify the validity of it.
- **Extract** Assume that a sender S having $ID_s \in \{0,1\}^*$ who holds the public key and private key pair (Q_s, a_s), where the private key $a_s = \mathcal{H}(ID_s) \oplus t_s, t_s \in \mathbb{Z}_q^*$ and public key $Q_s = a_s \cdot P$. Similarly the receiver has the public key and private key are (Q_r, a_r), where $Q_r = a_r P$, $a_r = \mathcal{H}(ID_r) \oplus t_r, t_r \in \mathbb{Z}_q^*$.
- **Send**
  1) **Step 1**: The sender S use his own private key and computes $\alpha = e(Q_r, TQ)^{a_s}$, where $T \in \mathbb{Z}_q^*$ is the timestamp.
  2) **Step 2**: When Sender S authenticates the deniable message $m \in \{0,1\}^l$, computes the session key $K = \mathcal{H}(\alpha, m)$ and cipher $C = E_{\Pi_{pub}}(K, m)$.
  3) **Step 3**: The resulting deniable authenticated message is the 4 tuples $\psi = (ID_s, T, MAC, C)$
  4) **Step 4**: Finally S sends ψ to the recipient R.
- **Receive**
  1) **Step 1**:After receiving $\psi = (ID_s, T, MAC, C)$, the recipient R computes the session key $\tilde{K} = \mathcal{H}(\tilde{\alpha}, m)$, where $\tilde{\alpha} = e(TQ, Q_s)^{a_r}$
  2) **Step 2**: If the timestamp T is valid ,Sender decrypts the encrypted message (cipher text) C to obtain the message $\tilde{m}$ and then computes $M\tilde{A}C = \mathcal{H}(\tilde{K}, m)$, where $\tilde{K} = \mathcal{H}(\tilde{\alpha}, m)$.
  3) **Step 3**: The recipient R verifies $M\tilde{A}C = MAC$, if the equation hold R accepts otherwise reject it.

The protocol is illustrated in the following fig.

Sender S	Receiver R
Select random number	Select random number
$t_s \in \mathbb{Z}_q^*$	$t_r \in \mathbb{Z}_q^*$
Compute $a_s = \mathcal{H}(ID_s) \oplus t_s$	Compute $a_r = \mathcal{H}(ID_r) \oplus t_r$
Compute $Q_s = a_s \cdot P$	Compute $Q_r = a_r \cdot P$
Compute $K = \mathcal{H}(\alpha, m)$	
Where $\alpha = e(Q_r, QT)^{a_s}$	
Compute $C = E_{\Pi_{pub}}(K, m)$	
$\psi = (ID_s, T, MAC, C)$	
$\longrightarrow$	
	Compute $\tilde{K} = \mathcal{H}(\tilde{\alpha}, m)$
	$\tilde{\alpha} = e(TQ, Q_s)^{a_r}$
	If the timestamp T
	is valid
	Decrypts C obtain m
	Compute $M\tilde{A}C = \mathcal{H}(\tilde{K}, m)$
	where $\tilde{K} = \mathcal{H}(\tilde{\alpha}, m)$
	Verify $MAC \overset{?}{=} M\tilde{A}C$

VII. Correctness

Theorem 1. *If $\psi = (ID_s, T, MAC, C)$ is a authentication message produced by the Sender S honestly, the recipient R will always accept it.*

Proof: The property of correctness is satisfied. In effect, if the deniable authetication message ψ is correctly generated, then we have

$$\alpha = e(Q_r, TQ)^{a_s} = e(a_rP, TtP)^{a_s} = e(P,P)^{Tta_ra_s}$$
$$\text{Similarly } \tilde{\alpha} = e(TQ, Q_s)^{a_r} = e(TtP, a_sP)^{a_r} = e(P,P)^{Tta_ra_s}$$
$$\text{So } K = \mathcal{H}(\alpha, m) = \mathcal{H}(\tilde{\alpha}, m) = \tilde{K}$$
$$\tilde{MAC} = \mathcal{H}(\tilde{K}, m) = \mathcal{H}(K,, m) = MAC$$

VIII. Security Analysis

The security of our protocol is based on Computational Diffie-Hellman (CDH), Decisional Diffie-Hellman (DDH) and the Hashed Diffie-Hellman (HDDH) Problems.In this section, we analyze the security of our proposed deniable authentication protocol. Subsequently also prove the securities requirement.

A. Security Model for the protocol

The protocol is defined by the following game between an adversary A and a challenge C

- **Setup** : On input of security parameters, C runs the algorithm to generate the system parameters and public key and private key pairs $(pk_i, sk_i), 1 \leq i \leq n$, of n users $\{U = U_1, U_2, \ldots U_n\}$, and sends the system parameters and all public keys $pk_1, pk_2 \ldots pk_n$ to A.
- **Corrupt Queries**: A can corrupt some users in U and obtain their private keys.
- **User Authentication Queries**: A also can make several user authentication queries on some uncorrupted users in U.
- **Impersonate** : In the end, A impersonates an uncorrupted user in U by outputting a valid login authentication message.

The success probability of A to win the game is defined by $\mathbf{Succ(A)}$.

Definition 7. *A user authentication scheme is secure if the probability of success of any polynomial bounded adversary A in the above game is negligible.*

Theorem 2. *Assume that $\mathcal{H}$ behaves as a random oracle. Then the proposed authentication scheme is secure provided that the BDH assumption holds in $\mathbb{G}_T$.*

Proof: Assume that A is an adversary, who can with non-negligible probability,break the proposed authentication scheme. Then, we can use A to construct another algorithm $\tilde{A}$, which is parameters $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ and $\mathcal{H}$), where $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^l$ be a secure cryptographic hash function, behaves a random oracle [**?**], and a BDH instance (P, aP, bP, cP), where $a, b, c \in \mathbb{Z}_q^*$ as her challenge, and her task here is to compute $e(P,P)^{abc}$. Let $U = U_1, U_2 \ldots U_n$ be a set of n users who may participate in the system. $\tilde{A}$ first picks a random number j from $\{1, 2 \ldots n\}$, and sets the user U_j's public key $Q_j = t_j \cdot P$. Then, $\tilde{A}$ chooses another $n-1$ random numbers $t_i \in \mathbb{Z}_q^*$ as user U_i's secret key, where $1 \leq i \leq n$ and $i \neq j$, and computes the corresponding public key $Q_i = t_i \cdot P$. Finally, $\tilde{A}$ sends all pubic key $Q_1, Q_2 \ldots Q_n$ to the adversary A.

Corrupt Queries: When A wants to corrupt the user U_i's secret key, $\tilde{A}$ will process as follows:

- If $i = j$, $\tilde{A}$ has to terminate the game and reports failure, since she has no knowledge on user U_j 's secret key.
- If $i \neq j$, $\tilde{A}$ returns the corresponding t_i to A.

Clearly, after q_c times corrupting queries, this game doesn't terminate with probability

$$1 - \tfrac{q_c}{n}, \text{ where } q_c < n.$$

Theorem 3. *The proposed Protocol achieves the authentication between the sender and the intended receiver.*

Proof : In our proposed protocol, if the receiver accepts the authentication message ψ, the receiver can always identify the source of the message. If an adversary wants impersonate the sender S, he can obtain a timestamp $T \in \mathbb{Z}_q^*$, a message M. But, he could not construct the parameter MAC without known α. If the adversary tries to compute α he has to know the senders private key a_s, recipients private key a_r or master-key t.

Definition 8. *Informally, a deniable authentication protocol is said to achieve the property of confidentiality, if there is no polynomial time algorithm that can distinguish the transcripts of two distinct messages.*

Theorem 4. *The proposed protocol achieves the property of confidentiality provided that the HDDH problem is hard in $\mathbb{G}$.*

Proof : $C = E_{\Pi_{pub}}(K, m)$ is actually a hashed ElGamal cipher text [13]. Hashed ElGamal encryption is semantically secure in the random oracle model under the Computational Diffie-Hellman (CBDH) assumption. This is the assumption that given P, aP, bP, cP, it is hard to compute $e(P,P)^{abc}$ in $\mathbb{G}_T$, where a, b, c are random elements of $\mathbb{Z}_q^*$. The CBDH assumption is more precisely formulated as follows.
Let $\mathbb{A}$ be an algorithm that takes as input a pair of group elements, and outputs a group element. We define the CBDH-advantage of $\mathcal{A}$ to be $[a, b, c \leftarrow \mathbb{Z}_q^* : \mathcal{A}(aP, bP, cP) = e(P,P)^{a,b,c}]$.
The CBDH assumption $(\mathbb{G})$ is the assumption that any efficient algorithms CBDH advantage is negligible.As a result, our proposed protocol can achieves the confidentiality.

Theorem 5. *Our proposed protocol also achieves the property of deniability.*

Proof :To prove that our proposed protocol has the property of deniability, we should prove that all transcripts transmitted between the sender S and the receiver R could be simulated by the receiver R himself in polynomial time algorithm. We first construct a simulator. Then we use this simulator to simulate the communication transcripts. Thus, the deniable property can be proved via the simulation process of the simulator.
Transcript Simulation
To simulate the transcripts on message M,the simulator follow the following steps

- **Step 1** The simulator chooses a random number $u \in \mathbb{Z}_q^*$ and calculates $Q_s = uP \in \mathbb{G}$ and then sends to R.
- **Step 2** Recipient R chooses a random number $v \in \mathbb{Z}_q^*$ and calculates $Q_r = vP \in \mathbb{G}$, and then send to the simulator.
- **Step 3** R calculates $\alpha = e(TQ, Q_s)^{a_r} \in \mathbb{G}_T$. The simulator calculates.Therefore, the simulator and R have a shared common key $K = \tilde{K}$
- **Step 4** The receiver could send messages to the simulator. That is, she sends a message m and the corresponding authentication message $MAC = \mathcal{H}(\tilde{K}, m)$ to the simulator.

The communication transcripts could be simulated by a probabilistic polynomial time algorithm. Based on the construction of the simulator, the hash code is indistinguishable to the third party. Thus the protocol has the deniable property. Clearly, the transcripts (ID_s, T, MAC, C) in simulation are indistinguishable from those of the sender S. As a result, the receiver R is not able to prove to a third party that the transcripts were produced by the sender S. According to the receiver's indistinguishable transcript simulation above, our proposed protocol also achieves the property of deniability.

Also we can prove considering the security model describe in section-5. Let us consider a distinguisher $\mathcal{D}$ and two honest players $\mathcal{P}_0$ and $\mathcal{P}_1$ involved in **Game** 2. The distinguisher $\mathcal{D}$ first submits a message $m \in \{0,1\}^*$ to the challenger. Then, the challenger chooses a bit $b \in \{0,1\}$ uniformly at random, and invokes the player $\mathcal{P}_b$ to make a deniable authentication message $\psi = (ID_b, T_b, MAC_b, C)$ on m between $\mathcal{P}_0$ and $\mathcal{P}_1$. In the end, the challenger returns $\psi = (ID_b, T_b, MAC_b, C)$ to the distinguisher $\mathcal{D}$. Since both $\mathcal{P}_0$ and $\mathcal{P}_1$ can generate a valid deniable authentication message $\psi = (ID_b, T_b, MAC_b, C)$, which can pass the verification equation, in an indistinguishable way, when $\mathcal{D}$ returns the guessed value b, we can sure that the probability $\Pr[b = b']$ is $\frac{1}{2}$, and the quantity $Adv_{IBDAP}^{DN}[D] = |Pr[b = b'] - \frac{1}{2}| = |\frac{1}{2} - \frac{1}{2}| = 0$ Based upon the analysis above, we can conclude that our proposed protocol can achieve the deniable authentication.

Definition 9. Secure against Man-in-the-middle *An authentication protocol is secure against an Man-in-the-middle, if Man-in-the-middle can not establish any session key with either the sender or the receiver.*

Theorem 6. *The proposed protocol is secure with respect to the man-in-the-middle (MIA) attack. provided that the ECDLP and BDHP is hard in $\mathbb{G}$ and $\mathbb{G}_T$ respectively.*

Proof: MIA pretends to be the sender to cheat the receiver, he needs to produce the key Q_r of the receiver in the protocol for which he has to find out secret key a_r for computing $Q_r = a_r P$. So he has to solve Elliptic Curve Discrete Logarithm Problem (ECDLP) in the group $\mathbb{G}$ which take fully exponential time. Further to produce $\alpha = e(Q_r, TQ)^{a_s}$, is to solve BDHP in the group $\mathbb{G}_T$. Similarly, MIA can't pretend to be R. Therefore, MIA and R (or S) can not share a common key K in any case. Hence proposed protocol is a secure deniable authentication protocol, since it simultaneously provides deniable property, authenticable property, as well as the property secure against MIA.

IX. Efficiency Analysis

The computation cost for the performance of this new protocol is as follows: the sender needs to compute a point multiplication, a pairing evaluation, an encryption, as well as a hash evaluation. In addition, the most expensive work for the sender is the use of a public-key digital signature algorithm.Since the receiver and the sender stand in the symmetric position, so the receiver shares the same computation costs. The communication cost of the proposed protocol is that the sender and the receiver carry out two rounds for communications in order for the receiver to obtain a message from the sender.

In practical implementation, we can use some existing tools for these computations including point multiplication, bilinear pairing evaluation, and hash function evaluation over elliptic curves.

X. Conclusion

The security of the proposed protocol is based on the Diffie-Hellman algorithm on pairing. The archives deniable authentication as well as confidentiality. Also it is resistant against Man-in-Middle attack. The protocol is also easy to implement for mobile devices.

References

[1] Yonatan Aumann, Michael O. Rabin *Efficient Deniable Authentication of Long Messages, Int. Conf. on Theoretical Computer Science in honour of Professor Manuel Blums 60th birthday, 1998. (http://www.cs.cityu.edu.hk/dept/video.html)*

[2] C. Boyd, W. Mao, K. G. Paterson *Deniable authenticated key establishment for Internet protocols, 11th International Workshop on Security Protocols, Cambridge (UK), April 2003.*

[3] J. S. Chou, Y. L. Chen, J. C. Huang *A ID-Based Deniable Authentication Protocol on pairings, Cryptology ePrint Archive: Report, (335)(2006).*

[4] X. Deng, Lee, C. H. Lee, and H. Zhu *Deniable authentication protocols, IEEE Proc. Comput. Digit. Tech., Vol. 148 (2), March 2001, pp. 101104.*

[5] L. Fan, C. X. Xu, J. H. Li *Deniable authentication protocol based on Diffie-Hellman algorithm, Electronics Letters 38. (4) (2002) 705706.*

[6] E. J. Yoon, E. K. Ryu, K. Y. Yoo *Improvement of Fan et al.'s Deniable Authentication Protocol based on Diffie-Hellman Algorithm, Applied Mathematics and Computation, Vol. 167 (1), August 2005, pp. 274-280*

[7] Robert W. Zhu, Duncan S. Wong, and Chan H. Lee *Cryptanalysis of a Suite of Deniable Authentication Protocols, IEEE COMMUNICATIONS LETTERS, VOL. 10, NO. 6, JUNE 2006, pp. 504-506.*

[8] N. Koblitz. *A course in Number Theory and Cryptography ,2nd edition* Springer-Verlag-1994

[9] A. Menezes, P. C Van Oorschot and S. A Vanstone *Handbook of applied cryptography. CRC Press, 1997.*

[10] D. Hankerson, A .Menezes and S.Vanstone. *Guide to Elliptic Curve Cryptography, Springer Verlag, 2004.*

[11] Dwork C., Naor M and Sahai A *Concurrent zero-knowledge, in Proceedings of 30th ACM STOC'98, 409-418,1998*

[12] Aumann, Y. and Rabin, M. *Authentication, enhanced security and error correcting codes, in Advances in Cryptology - Crypto'98, LNCS, 1462, 299-303.*

[13] Shoup V Sequences of games: a tool for taming complexity in security proofs, in Cryptology ePrint Archive: Report 2004/332, available at: http://eprint.iacr.org/2004/332