Non-interactive Deniable Authentication Protocol using generalized ECDSA Signature Scheme

Jayaprakash Kar

Faculty of Computing & Information Technology
Department of Information Systems
King Abdul Aziz University, Kingdom of Saudi Arabia
jayaprakashkar@yahoo.com

Abstract. 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. This paper describes a secure non-interactive deniable authentication protocol using ECDSA signature scheme. The security of the protocol is based on difficulty of breaking Elliptic Curve Discrete Logarithm Problem. It 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.

Key words: deniable authentication, ECDLP, non-interactive, ECDSA.

1 Introduction

Nowadays, authentication had emerged to be an essential communication process. 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 signer's 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 sender's 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. This protocol is known as deniable authentication protocol.

2 Background

In this section we brief overview of prime field, Elliptic Curve over that field and Elliptic Curve Discrete Logarithm Problem.

2.1 The finite field $\mathbb{F}_p$

Let p be a prime number. The finite field F_p is comprised of the set of integers $0, 1, 2, \ldots p-1$ with the following arithmetic operations [12] [13] [14]:

- Addition: If $a, b \in \mathbb{F}_p$, then $a + b = r$, where r is the remainder when $a + b$ is divided by p and $0 \leq r \leq p-1$. This is known as addition modulo p.
- Multiplication: If $a, b \in \mathbb{F}_p$, then $a.b = s$, where s is the remainder when $a.b$ is divided by p and $0 \leq s \leq p-1$. This is known as multiplication modulo p.
- Inversion: If a is a non-zero element in $\mathbb{F}_p$, the inverse of a modulo p, denoted a^{-1}, is the unique integer $c \in \mathbb{F}_p$ for which $a.c = 1$.

2.2 Elliptic Curve over $\mathbb{F}_p$

Let $p \geq 3$ be a prime number. Let $a, b \in F_p$ be such that $4a^3 + 27b^2 \neq 0$ in $\mathbb{F}_p$. An elliptic curve E over $\mathbb{F}_p$ defined by the parameters a and b is the set of all solutions $(x, y), x, y \in \mathbb{F}_p$, to the equation $y^2 = x^3 + ax + b$, together with an extra point $\mathcal{O}$, the point at infinity. The set of points $E(\mathbb{F}_p)$ forms a Abelian group with the following addition rules [16]:

1. Identity : $P + \mathcal{O} = \mathcal{O} + \mathcal{P} = \mathcal{P}$, for all $P \in E(\mathbb{F}_p)$
2. Negative : if $P(x, y) \in E(\mathbb{F}_p)$ then $(x, y) + (x, -y) = \mathcal{O}$, The point $(x, -y)$ is dented as $-P$ called negative of P.
3. Point addition: Let $P((x_1, y_1), Q(x_2, y_2)) \in E(\mathbb{F}_p)$,then $P + Q = R \in E(\mathbb{F}_p)$ and coordinate $(x_3, y_3)$of R is given by $x_3 = \lambda^2 - x_1 - x_2$ and $y_3 = \lambda(x_1 - x_3) - y_1$ where $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$
4. Point doubling : Let $P(x_1, y_1) \in E(\mathbb{F}_p)$ where $P \neq -P$ then $2P = (x_3, y_3)$ where $x_3 = (\frac{3x_1^2 + a}{2y_1})^2 - 2x_1$ and $y_3 = (\frac{3x_1^2 + a}{2y_1})(x_1 - x_3)$- y_1

2.3 Elliptic Curve Discrete Logarithm Problem (ECDLP)

In 1985, Neal Koblitz and Victor Miller independently proposed the concepts of ECC. It is based on the Discrete Logarithm Problem (DLP) in a group defined by points on Elliptic Curve over a finite field.

Definition 1. *Given an elliptic curve E defined over a finite field $\mathbb{F}_p$,a point $P \in E(\mathbb{F}_p)$ of order n, and a point $Q \in < P >$,find the integer $l \in [0, n-1]$such that $Q = lP$. The integer l is called discrete logarithm of Q to base P,denoted $l = log_p Q$.*

3 Elliptic Curve Digital Signature Algorithm

The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve analogue of the Digital Signature Algorithm (DSA), and is under consideration for standardization by the ANSI X9 committee. Unlike the normal discrete logarithm problem and the integer factorization problem, the elliptic curve discrete logarithm problem has no sub-exponential time algorithm. For this reason, the strength-per key-bit is substantially greater in an algorithm that uses elliptic curves.

3.1 ECDSA Signature Generation and Verification

This section describes the procedure for generating and verifying signature using the ECDSA. To sign a message m, an entity A having the key pair (d, Q) executes the following steps.

ECDSA Signature Generation

1. Select a random or pseudorandom integer $k \in [1, n-1]$.
2. Compute $k \cdot P = (x_1, y_1)$ and convert x_1 to an integer $\bar{x}_1$.
3. Compute $r = x_1 \pmod{n}$. If $r = 0$ then go to step 1.
4. Compute $k^{-1} \pmod{n}$.
5. Compute SHA-1(m) and convert to the bit string e.
6. Compute $s = k^{-1}(e + dr) \pmod{n}$. If $s = 0$ then go to step 1.
7. A's signature for the message m is the pair (r, s).

ECDSA Signature Verification

1. Verify r and s are integers in the interval $[1, n-1]$.
2. Compute SHA-1(m) and convert the bit string to an integer e.
3. Compute $\beta = s^{-1} \pmod{n}$.
4. Compute $u_1 = e\beta \pmod{n}$ and $u_2 = r\beta \pmod{n}$.
5. Compute $R = u_1 \cdot P + u_2 \cdot Q$.
6. If $R = \mathcal{O}$, then reject the signature. Otherwise, convert the x-coordinate x_1 of R to an integer $\bar{x}_1$, and compute $v = \bar{x}_1 \pmod{n}$
7. Accept the signature if and only if $v = r$.

4 Preliminaries

4.1 Notations

We first introduce common notations used in this paper as follows.

- p is the order of underlying finite field;
- $\mathbb{F}_p$ is the underlying finite field of order p
- E is an an elliptic curve defined on finite field $\mathbb{F}_p$ with large order.
- G is the group of elliptic curve points on E.

- P is a point in $E(\mathbb{F}_p)$ with order n , where n is a large prime number.
- $\mathcal{H}()$ is a secure one-way hash function which is collision resistant.
- $\|$ denotes concatenation operation between two bit stings.
- Let S denotes be the Sender.
- Let R denotes be the Receiver.
- Public and Private key pair of the Sender S is (d_s, Q_s), where $Q_s = d_s \cdot P$.
- Public and Private key pair of Receiver R is (d_r, Q_r), where $Q_r = d_r \cdot P$.

5 Relatated Works

In 1998, Dwork et al. [8] 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 [17] proposed some other deniable authentication protocols based on the factoring problem. In 2001, Deng et al. [9] also proposed two deniable authentication protocols based on the factoring and the discrete logarithm problem respectively.

In the past several years, numerous deniable authentication protocols have been proposed but many of them have also been proven to be vulnerable to various cryptanalytic attacks [5] [10] [11] . The concept of deniable authentication protocol was initially introduced by Dwork et al. [8], which is based on the concurrent zero knowledge proof. However, this scheme requires a timing constraint. Not only that, the proof of knowledge is also time-consuming [7]. Another notable scheme which was developed by Aumann and Rabin [1] is based on the intractability of the factoring problem, in which a set of public data is needed to authenticate one bit of a given message. Few years later, Deng et al. [7] have proposed two deniable authentication schemes based on Aumaan and Rabins scheme. The proposed schemes are based on the intractability of the factoring problem and the logarithm problem. However, in 2006, Zhu et al. [11] have successfully demonstrated the Man-in-the-Middle attack against Aumann and Rabins scheme and this indirectly results in an insecure implementation of Deng et al.s schemes. In 2003, Boyd and Mao [2]have proposed another two deniable authenticated key establishment for Internet protocols based on elliptic curve cryptography. These schemes are believed to be able to solute the complexity of computation and appear to be more efficient than others but their vulnerability to Key Compromise Impersonation (KCI) attack has been exploited by Chou et al. [4] in 2005. Besides that, Fan et al.have proposed a simple deniable authentication protocol based on Diffie-Hellman key distribution protocol in 2002. Unfortunately, in 2005, Yoon et al. [10] have pointed out that their protocol suffers from the intruder masquerading attack and subsequently proposed their enhanced deniable authentication protocol based on Fan et al.s scheme.

6 Model of Deniable Authentication Protocol

A deniable authentication protocol (DAP) consists of the following four algorithms: **Setup**, **Extract**, **Send** and **Receive**. We describe the functions of each as follows [6].

- **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

$$\tilde{m} \leftarrow \textbf{Send}(m, sk_S, pk_R)$$

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

7 Security model

Security Notions In this subsection describes about security notions of 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 2. (Unforgeability). *Let A denote an adversary that plays the game above. If the quantity* $Adv^{UF}_{DAP}[A] = Pr[Awins]$ *is negligible we say that the 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 3. (Deniablity). *Let D denote the distinguisher that is involved the game above. If the quantity* $Adv^{DN}_{DAP}[D] = |Pr[b = b'] - \frac{1}{2}|$ *is negligible we say that deniable authentication protocol in question is deniable.*

8 Proposed Protocol

Security of the proposed is based on the difficulty of breaking of ECDLP problem. This will be achieving the following security 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.
- **Confidentiality**: Any outside adversary has no ability to gain the deniable authentication message from the transmitted transcripts.

– **Completeness**: If a sender and a receiver follows the protocol to negotiate with each other, the receiver can identify the source of message.

The protocol involves two entities : a sender S and a intended receiver R. It follows the followings steps.

– **Setup** Let $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^l$ be a secure cryptographic hash function which is of collision free. Let E be the elliptic curve define over the prime field $\mathbb{F}_p$. The key pair of both Sender S and receiver are (d_s, Q_s) and (d_r, Q_r) respectively.
– **Extract** Here S executes the the following steps

- Choose a random integer $k \in [1, n-1]$
- Computes

$$U = k \cdot P \tag{1}$$

$$r = x_1 \bmod n \tag{2}$$

where $x_1 = (U)_x$, x- coordinate of the point $U \in E(\mathbb{F}_p)$

$$\gamma = \mathcal{H}(M)d_s + rk \tag{3}$$

$$\alpha_1 = \gamma \cdot Q_r \tag{4}$$

$$MAC = \mathcal{H}(\alpha_1 \| M) \tag{5}$$

Authentication message is $\psi = (U, MAC, M)$

– **Send** $\psi = (U, MAC, M)$ to R
– **Receive** in this phase R executes the following steps.

- Computes

$$r = x_1 \bmod n, \text{ Where } x_1 = (U)_x$$

$$\alpha_2 = \{\mathcal{H}(M) \cdot Q_s + r \cdot U\} \cdot d_r \tag{6}$$

- Verify whether $\mathcal{H}(\alpha_2 \| M) = MAC$. If valid, accepts M otherwise reject.

The protocol is illustrated in the following fig.

Sender S		Receiver R
Select random number $d_s \in [1, n-1]$ Computes $Q_s = d_s \cdot P$ Select $k \in [1, n-1]$ Computes the following $U = k \cdot P$ $r = x_1 \bmod n$ $\gamma = \mathcal{H}(M)d_s + rk$ $\alpha_1 = \gamma \cdot Q_r$ $MAC = \mathcal{H}(\alpha_1 \| M)$	ψ $\longrightarrow$	Select random number $d_r \in [1, n-1]$ $Q_r = d_r \cdot P$ Compute $\alpha_2 = \{\mathcal{H}(M) \cdot Q_s + r \cdot U\} \cdot d_r$ Verify whether $\mathcal{H}(\alpha_2 \| M) = MAC$ accept M otherwise reject

9 Correctness

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

Proof: The proposed protocol satisfies the property of correctness. In effect, if the deniable authentication message ψ is correctly generated, then $\alpha_1 = \alpha_2$
$\alpha_2 = \{\mathcal{H}(M) \cdot Q_s + r \cdot U\} \cdot d_r$
$= \{\mathcal{H}(M)d_s \cdot P + rk \cdot P\}d_r$
$= \mathcal{H}(M)d_s \cdot Pd_r + rd_rk \cdot P$
$= \mathcal{H}(M)d_s \cdot Q_r + rkQ_r$
$= \{\mathcal{H}(M)d_s + rk\} \cdot Q_r$
$= \gamma \cdot Q_r = \alpha_1$

10 Security Analysis

In this section, we analyze the security of our proposed deniable authentication protocol . The security of our protocol is based on the difficulty of breaking of Elliptic Curve Discrete Logarithms. The security of the proposed protocol is analyzed and illustrated a model for the protocol.

Theorem 2 *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, receiver R can always identify the source of the message. If an adversary wants impersonate the sender S, he has to construct the α_2. If the adversary tries to compute α_2 he has to know the sender's private key d_s for that it needs to solve ECDLP. □

Definition 4. *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 3 *The proposed protocol achieves the property of confidentiality provided that the ECDLP is hard in $E(\mathbb{F}_p)$.*

Proof : $MAC = (\alpha_1 \| M)$ is actually a hashed cipher text [18]. Hashed based encryption is semantically secure in the random oracle model provided ECDLPis hard. As a result, the proposed protocol can achieves the confidentiality. □

Theorem 4 *The proposed protocol also achieves the property of deniability.*

Proof : To prove that the proposed protocol has deniable property, first we should prove that it enables an intended receiver R to identify the source of the given message M and can not able to prove the source of message to the third party.

Relationship between U and MAC for a given message M can be verified only by knowing α_1. When M and R are given, α_1 can be derived from Eq.(4) or (6). Therefore, both the sender with the knowledge of d_s and the receiver with knowledge of d_r have the same ability to generate (U, MAC) for a given message M. Obviously, it is difficult to verify whether the message was send by the sender or forged by the receiver, so the receiver can only identify the source of message but can not prove the source of message to the third party. □

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 (U_b, MAC_b) on M between $\mathcal{P}_0$ and $\mathcal{P}_1$. In the end, the challenger returns $\psi = (U_b, MAC_b, M)$ to the distinguisher $\mathcal{D}$. Since both $\mathcal{P}_0$ and $\mathcal{P}_1$ can generate a valid deniable authentication message $\psi = (U, MAC, M)$, 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 $Adv^{DN}_{DAP}[D] = |Pr[b = b'] - \frac{1}{2}| = |\frac{1}{2} - \frac{1}{2}| = 0$ Based upon the analysis above, we can conclude that the proposed protocol can achieve the deniable authentication. □

Theorem 5 *The Protocol authenticates the source of the message.*

Proof: If someone proves MAC to R, he must be S. Since from Eq.(5), to computes MAC, he has to calculate $\alpha_1 = \gamma \cdot Q_r$, for that he needs to find γ *i.e*

nothing but solving of ECDLP problem. If an adversary gets all the information Q_r in **Extract** phase, he can not compute the session key α_1. Hence the protocol authenticates the sources of message. □

Definition 5. 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. This is also called forgery attack.*

Theorem 6 *The proposed protocol is secure with respect to the man-in-the-middle (MIA) attack.*

Proof: Since the session key $\alpha_2 = \{\mathcal{H}(M) \cdot Q_s + r \cdot U\} \cdot d_r = \gamma \cdot Q_r = \alpha_1$, only an attacker who has the ability to create γ can forge valid deniable authentication message. However γ can be computed by Eq.(3). No one can forge γ without knowing the private key of S *i.e* d_s. Therefore it is resistant against forgery attack. □

Theorem 7 *(Completeness).If a sender and a receiver follows the protocol to negotiate with each other, the receiver can identify the source of message.*

Proof : From Theorem 1, it can be seen that the sender and the receiver share the same session secret key $\alpha_1 = \alpha_2$. Hence the receiver can identify the source of message M according to $\mathcal{H}(\alpha_1 \| M) = \mathcal{H}(\alpha_2 \| M)$.□

Theorem 8 *A compromised session secret does not affect the security of the proposed protocol.*

Proof: The session secret can be derived from Since the session key $\alpha_2 = \{\mathcal{H}(M) \cdot Q_s + r \cdot U\} \cdot d_r = \gamma \cdot Q_r = \alpha_1$, where a random number k is chosen independently for each session. If an attacker wants to forge the deniable information with the forged message $\tilde{M}$ by using the compromised session secret α_1, the receiver will derive a different session secret from the forged information. This is because the message and its corresponding session secret are interdependent. To solve this problem, the session secret for each round must be independent. This has been realized in our protocol which as well guarantees the underlying signature scheme as shown in Eq. (3). Therefore, a compromised session secret does not affect the security of other sessions. □

11 Conclusion

The proposed protocol is an non-interactive protocol where ECDSA signature scheme has been used. The security of the proposed protocol is based on difficulty of breaking the Elliptic Curve Discrete Logarithm Problem. It archives deniable authentication confidentiality and completeness. Also it is resistant against Man-in-Middle attack. It can be easy to implemented in mobile devices such as PDA, smart card etc. Since the protocol is based on the elliptic curve cryptography (ECC) and thus it has high security complexity with short key size.

References

1. Yonatan Aumann, Michael O Rabin Efficient Deniable Authentication of Long Messages,*Int. Conf. on Theoretical Computer Science in honour of Professor Manuel Blum's 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. C.Dwork, M.Naor, A. Sahai Concurrent zero-knowledge,*Proc. 30th ACM STOC 98,Dallas TX, USA, 1998, pp. 409418.*
4. J.S.Chou, Y.L.Chen, J.C.Huang A ID-Based Deniable Authentication Protocol on pairings, *Cryptology ePrint Archive: Report, (335)(2006).*
5. J.S.Chou, Y.L.Chen, M.D.Yang, "Weaknesses of the Boyd-Mao Deniable Authenticated key Establishment for Internet Protocols", *Cryptology ePrint Archive: Report, (451)(2005).*
6. J.P.Kar & B.Majhi "A Novel Deniable Authentication Protocol based on Diffie-Hellman Algorithm using Pairing techniques", *ACM International Conference on Communication, Computing and Security (ICCCS 2011) India. pp-493-498*
7. X. Deng, Lee, C. H. Lee, and H. Zhu Deniable authentication protocols,*IEEE Proc. Comput. Digit. Tech., Vol. 148 (2), March 2001, pp. 101-104.*
8. C. Dwork, M. Naor, A. Sahai Concurrent zero-knowledge,*Proc. 30th ACM STOC '98,Dallas TX, USA, 1998, pp. 409-418.*
9. Deng, X., Lee, C.H. & Zhu. H Deniable authentication protocols,*IEE Proceedings. Computers and Digital Techniques, 148,101-104, 2001*
10. 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.*
11. Robert W.Zhu, Duncan S.Wong, $ Chan H. Lee Cryptanalysis of a Suite of Deniable Authentication Protocols",*IEEE Communication Letter, VOL. 10, NO. 6, JUNE 2006, pp. 504-506.*
12. N. Koblitz. *A course in Number Theory and Cryptography ,2nd edition* Springer-Verlag-1994
13. K. H Rosen *"Elementary Number Theory in Science and Communication"*, 2nd ed., Springer-Verlag, Berlin, 1986.
14. A. Menezes, P. C Van Oorschot and S. A Vanstone *Handbook of applied cryptography. CRC Press, 1997.*
15. D. Hankerson, A .Menezes and S.Vanstone. *Guide to Elliptic Curve Cryptography, Springer Verlag, 2004.*
16. *"Certicom ECC Challenge and The Elliptic Curve Cryptosystem"* available :http://www.certicom.com/index.php.
17. Aumann,Y.and Rabin M. Authentication, enhanced security and error correcting codes,*in Advances in Cryptology - Crypto'98, LNCS, 1462, 299-303.*
18. 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*