An Authentication Protocol Combining Deniability and Forward Secrecy for Resisting Adaptive Attacks

Hung-Min Sun[1], King-Hang Wang[1], Shih-Ying Chang[1] and Li Wan[2]
[1]*Dept. of Computer Science, National Tsing Hua University, Taiwan*
[2]*Networks & Multimedia Institute, Institute for Information Industry, Taiwan*
hmsun@cs.nthu.edu.tw , { khwang0 ,godspeed}@is.cs.nthu.edu.tw, liwan@nmi.iii.org.tw

ABSTRACT

For some applications of message authentication, a sender may not want to be proven by a third party that he has sent some messages to a receiver. Deniable authentication protocols are designed to adopt this situation. A paper proposed by Chang et al. suggested a way to cryptanalyze previous works and names a new deniable authentication protocol. In this paper, we point out their work can be broken by a new proposed adaptive attacks. We further provide an adaptively secure protocol which can achieve forward secrecy. This is currently the only adaptive deniable authentication scheme which can achieve forward secrecy. In addition, we classify deniable protocols into four categories. A comparison of our protocol and other methods will be given at the end of this paper.

1: INTRODUCTIONS

Message authentication is of paramount importance for communication of computer network. When a sender sends a message to a receiver, the receiver is allowed to verify that the message is indeed send by the sender. By using digital signature, not only the receiver can verify the source of the message, any third party can also ascertain it. This is an essential property of cryptography called non-repudiation. However, non-repudiation of messages is not always desirable if the sender wants to prove the source of messages to the receiver only for the sake of privacy. That is, the sender would not like to expose the communication with the receiver to any third party. So, deniable authentication protocols are designed to provide message authentication and deniability. A deniable authentication protocol can be used in many specialized applications. It can avoid coercion in electronic voting and secure negotiation over internet [10]. For example, if the involved entities are performing a sealed auction, participants would not like the dealer to prove their bids to other participants.

Forward secrecy is another important feature in message authentication. Generally speaking, an authentication protocol is said to have the property of forward secrecy if the leakage of the long term key (the private key or shared password) of the involved parties does not lead to disclosure of any previous session key. Moreover, applications like electronic voting and secure negotiation over internet also require forward secrecy. Therefore, authentication protocol with forward secrecy and deniability is practically important. In this paper, we try to develop such a protocol.

A deniable authentication protocol has two main characteristics. First, a receiver can identify the sources of messages like traditional authentication processes. Second, a receiver cannot prove the source of a message to any third party with the data in the message and the data already stored in receiver concerning the sender. To maintain the deniability, the receiver can simulate the communication process by itself after communicating with the sender. Therefore, a zero-knowledge authentication protocol is always deniable (but not adaptive deniable; we will explain the term adaptive deniable later). For some reason, if we want to keep the secrecy of the message, we may want to agree a session key which does not only provide data integrity, but also preserves secrecy from the third party. In this case, zero-knowledge authentication protocols are not desirable.

We can see that some protocols can be deniable only within some classes of attacks. This motivates us to further classify attacks related to deniable protocol. We classify these attacks into four categories, namely: Plausible deniable, Non-adaptive deniable, Adaptive deniable, and Secret-leaking deniable. We will explain more detail about these classes and categorize the previous literatures into these classes.

We discover a very interesting generalization - an adaptive deniable protocol usually cannot achieve forward secrecy. That is, to suppose a receiver is allowed to interact with a verifier while a protocol is executed. For example, a deniable scheme generates a forward secure key by Diffie-Hellman key exchange and involves a key flow that the receiver sends $Sig_R(g^y \bmod p)$ (signature of g^y using receiver private key) to the sender. The receiver can ask the verifier to pick y and calculate g^y. Then the receiver signs this value and sends it to the sender. We will see that the receiver will not know the DH agreed key, and thus, the protocol is undeniable because the receiver can not compute the message authentication code with this key. Indeed, currently there does not exist any adaptive deniable protocol that achieves forward secrecy. In this paper, we propose an adaptive deniable protocol, which achieves forward secrecy.

The rest of the paper is organized as follows. In Section 2 we describe related research work on deniability. Section 3 categorizes deniable authentication protocols according to their properties. In Section 4 and 5, we give an overview and cryptanalysis on Chang *et al.* [7] scheme. Section 6 points out the problem that how difficult is putting forward secrecy and adaptively deniable together. In Section 7 and 8, we proposed a new adaptive deniable protocol and then provide a security analysis on it. Section 9 compares popular deniable schemes with our new proposed scheme. Finally, we conclude this paper in Section 10.

2: RELATED WORK

In 1998, Dwork *et al.* [11] raised the concern about deniability and applied concurrent zero-knowledge in their authentication protocol. However, their scheme cost timing constraints and too many rounds of communication which makes the protocol infeasible.

Aumann and Rabin [1] proposed a scheme based on factoring problem. Deng *et al.* [10] proposed two schemes based on discrete logarithm. These schemes need to publish some verified data on the public directories and need many communication steps. Regardless their inefficiency, these protocols are not adaptive deniable protocol.

Fan *et al.* [12] proposed new method to improve the efficiency of deniable authentication protocol by employing Diffie-Hellman key exchange protocol. This protocol also provides forward secrecy. Although this protocol is more efficient than the others, it easily suffered attacks named in Chang *et al.* [7] and Hsieh *et al.* [14], whose later improved it to prevent their proposed attacks. The two modified protocols are sounds under the sense non-adaptive deniable, but are not adaptive deniable.

Shao [19] proposed one-step protocol by using ElGamal signature. This protocol like traditional key transport protocol and achieves deniability in the same time. Unfortunately, this protocol does not include forward secrecy. Shi *et al.* [6] and Cao *et al.* [20] followed Shao's [19] steps to propose new one-step protocol that based on pairing and ID based signature. These protocol are light and sound, but neither of them can achieve forward secrecy.

Moreover, some new applications about deniability have been developed, such as deniable ring authentication [16] and deniable multicasting [4]. On the other hand, some papers such as deniable encryption [5] and undeniable signature [8, 9] also employ the word "deniable". However, they are illustrating different concepts and are not related to our research.

Our protocol is the only adaptive deniable authentication protocol, which provides forward secrecy at the same time. To clarify forward secrecy, which is adopted from [15] and will be discussed later in this paper, is different from forward deniability mentioned in [18].

3: DEFINING DENIABLE AUTHENTICATION PROTOCOL

The semantic meaning of "deniable" was stated in the previous literatures. However, since that the behaviors of the receiver had not been formalized, disputes arose in these literatures. To resolve these disputes, we aggregate their ideas and group them into several classes. Terms and notations will be briefly explained as listed in Table 1.

Symbol	Descriptions
S	Sender
R	Receiver
V	Verifier
$P_k()$	public key encryption using the public key of entity *k*
$Sig_k()$	signature using the private key of entity *k*
||	Concatenation
$H(x)$	cryptographic hash function
M	Message
X_k	DH private key of entity *k*
Y_k	DH public key of entity k $Y_k = g^{X_k} \bmod p$
$Cert(k)$	certificate of k, proving the public key of *k*
p	a large prime
g	a generator of GF(*p*)
=?	Check whether this equation existed.

Table 1. Notations

Three roles are involved in a deniable authentication protocol. They are sender *S*, receiver *R*, and verifier *V*. A sender *S* wishes to send a message *M* to a receiver *R* in which *R* is convinced that the message comes from *S*. At the same time, *R* may disclose some information, either on his will or not, to a verifier *V* in order to prove that the message *M* is sent from *S*. We assume that every party has a public key which is either publicly published in a trust third party domain or signed with a trust third party's signature. They also have the corresponding private keys where we assume that these keys are kept secretly unless they want to disclose it.

The four types of deniable authentication are categorized according to the behaviors of *R* and are listed as follows:

1. Plausible deniable authentication [3]:

   A receiver *R* executes the protocol faithfully except that he will keep all the temporary secret value in records. During the execution of the protocol, *R* would not interact with *V*. After the execution of the protocol, if *R* wants to betray a sender *S* or *R* has been hacked in by a verifier *V*, *R* can disclose the temporary secrets and all messages from the communication with *S* to convince *V*. However, *V* is unable to obtain a proof that *S* did send a message *M* to *R* and we call this kind of protocol plausible deniable. Protocol of [12] is one of the examples of plausible deniable authentication.

2. Non-adaptive deniable authentication:

   A receiver *R* is willing to give out all the information, including session key, temporary secret, and a message *M*, except his secret key to a verifier *V*. These are the same as plausible deniable, but here *V* can involve in the communication between *S* and *R* non-adaptively. That is, *R* can modify or forge some messages according to the information given by *V* in advance. For instance, picking the identity or public key of *V* as a random value in the protocol, but *R* cannot interact with *V* during the execution of the protocol. If *R* is unable to convince *V* that *S* indeed sent him a message, we call this kind of protocol non-adaptive deniable. The cryptanalysis part in [14] obeys this rule and his new proposed protocol belongs to this category.

3. Adaptive deniable authentication:

   The requirements of adaptive deniable are the same as non-adaptive deniable except that *V* can involve in the communication between *S* and *R* adaptively. That is, *R* colludes with a verifier *V* in advance. They are agree to run a protocol in order to show a sender *S* send a message to *R* with the only restriction that they will not leak the secret key to each other. This assumption is stronger than the non-adaptive and is employed in [7] to cryptanalyze the protocol proposed in [12]. We will further show that the scheme proposed in [7] would neither achieve adaptive deniable authentication.

4. Secret-leaking authentication:

   In order to prove a sender *S* did send a message *M* to a receiver *R*, *R* not only agree to run an adaptive protocol with a verifier *V*, but also is willing to show his private key to *V*. The research [19] belong this protocol class. This assumption is ultimately strong and somehow unrealistic.

These definitions have implicative relations. A secret-leaking authentication scheme must be an adaptive deniable authentication scheme. An adaptive deniable authentication scheme must be a non-adaptive one, a non adaptive scheme must be a plausible deniable scheme.

4: REVIEW OF CHANG ET AL.'S PROTOCOL

In [7], the authors first cryptanalyzed the scheme of Fan *et al.* [12] with an adaptive deniable attacker and proposed a protocol that claimed to be "deniable". They did not well define the terms deniable there, thus we assume the word "deniable" was also under the sense of "adaptive deniable". The protocol is described as follows:

Sender *S* picks a random variable t_S and calculate the followings

$$I=(Y_R)^{t_S X_S} \bmod p = g^{t_S X_S X_R} \bmod p\ ,\quad r_S = Y_S^{t_S} \bmod p,$$

$$S_S = t_S + X_S{}^{-1} \cdot H(I) \bmod p-1$$

and sends them together with *Cert*(*S*) to the receiver *R*.

After obtaining the message from *S*, *R* obtains the *S*'s public key from *Cert*(S) and verifies if

$$Y_S^{S_S} \overset{?}{=} r_S \cdot g^{H(I)} \bmod p\ .$$

Then *R* randomly picks a number be t_R and evaluates the followings

$$J=(Y_S)^{t_R X_R} \bmod p = g^{t_R X_S X_R} \bmod p\ ,\quad r_R = Y_R{}^{t_R} \bmod p$$

$$k = I^{t_R} \bmod p = g^{t_S t_R X_S X_R} \bmod p\ ,$$

$$S_R = t_R + X_R^{-1} \cdot H(J \,\|\, k) \bmod (p-1)$$

and sends *J*, r_R, *Cert*(*R*) and S_R to *S*.

Once *S* confirm the public key of *R* with *Cert*(*R*), he will compute the session key by

$$k = J^{t_S} \bmod p = g^{t_S t_R X_S X_R} \bmod p$$

and verify if

$$Y_R^{S_R} \overset{?}{=} r_R \cdot g^{H(J\|k)} \bmod p$$

Then he will obtain *D* by operating

$$D = H(k \,\|\, M)$$

and send *D*, and *M* to *R*. *R* is able to confirm this message is indeed from *S*. Later *R* can compute *D* by itself and this protocol is believed to be a non-adaptive deniable authentication.

5: CRYPTANALYSIS OF CHANG ET AL.'S PROTOCOL

We will employ adaptive attack to break the deniable property of the protocol in [7] in this section. For a sender *S* sends a message to a receiver *R* using this protocol, *R* is able to prove to a verifier *V* that the involvement of *S* during the protocol. The attacking protocol is illustrated in Fig. 1.

After receiving the message sent from *S*, *R* sends all the information he received to *V* except *I*.

V will randomly pick a nonce t_V and calculates the followings:

$$J = Y_S^{t_V}\ , k = r_S{}^{t_V}\ , a = H(J \,\|\, k) + t_v, r_R = g^{tv}$$

and replies *R* with *J*, *a*, r_R.

Upon receiving the message, *R* performs the following equations:

$$S_R = X_R^{-1} \cdot a \bmod (p-1)$$

J, S_R, r_R and *Cert*(*R*) will be sent to *S*.

S will be convinced that the message is really come from *R* by verifying following equation.

$$Y_R^{S_R} = ?\, r_R \cdot g^{H(J\|k')}$$

Then he will calculate *D* with *k*

$$k = J^{t_S} = g^{t_S t_V X_S}$$

We claim that the message pair *M*, *D* cannot be forged by *R*, therefore *V* is convinced that *S* sent the message *M* to *R*.

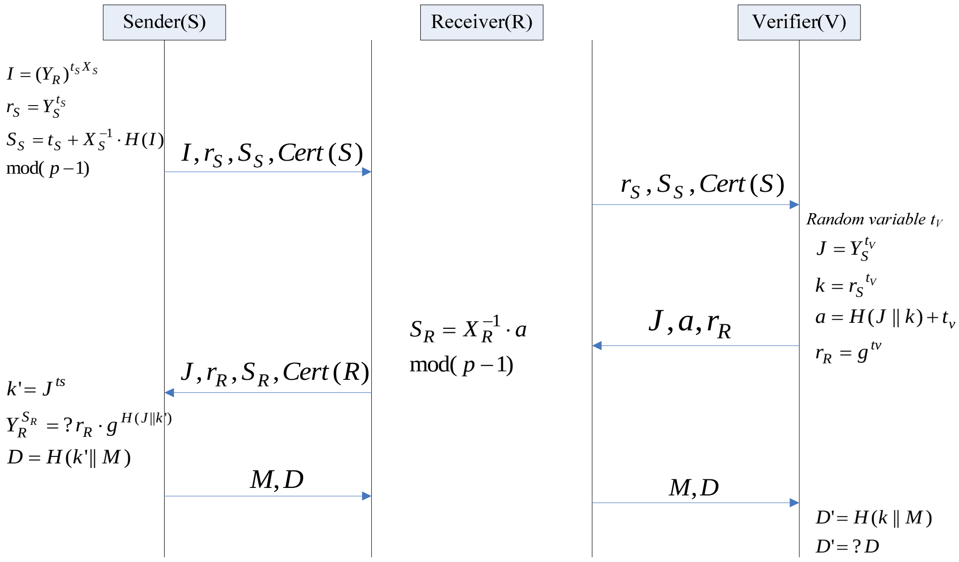

Fig. 1. Adaptive attack on Chang *et al.*'s protocol

We remark that the original paper does not check the validation of r_R. Before sending the last message, the sender should further verify that

$$ {r_R}^{X_S}=?J $$

In our case, the r_R also fits this equation. So, *S* and *V* will generate the same *k* which *R* can not obtain and *V* can convince that *S* indeed send message *M* to *R* because *R* can not simulate this value.

6. DIFFICULTY OF PUTTING FORWARD SECRECY AND ADAPTIVELY DENIABLE TOGETHER

In the introduction part, we have shown an example of trying to put forward secrecy and adaptive deniable scheme together. Before that, we formally define the word "forward secrecy" here. Forward secrecy is adopted from [15] which means that if the long term secrets of the involved parties is divulged to adversary. For the term "long term secrets", can be referred to private key of an asymmetric cryptosystem.

Some previous works [12][7][14] achieved forward secrecy but not adaptive deniable, while some of those [19][3] are adaptive deniable but not forward secure. We will propose a scheme in the next section which can provide both at the same time without making strong and unrealizable assumption. Before going deep into our protocol, we present why it is not easy to do the both.

No matter how the protocol is designed, the sender and the receiver must agree or distribute a session key at the end of the protocol. A forward secure protocol should not allow an intruder to know the previous session key after divulging the secret key of the sender or the receiver. In order to address this problem, the session key must be produced by agreement instead of distribution.

To agreement a session key, both parties have to contribute a cipher or a non-invertible message like $g^x \bmod p$. The receiver can always ask a verifier to choose this cipher or message for him. For instance, the receiver asks a verifier to compute $g^y \bmod p$ while keeping *y* secretly. Therefore, even if the protocol requires the receiver to sign the message $g^y \bmod p$, it is still impossible that the receiver can compute the session key *k*. Since the receiver do not know the session key *k*, the verifier can be convinced that the sender sends some messages to the receiver.

From the above logic, if we can ensure the receiver knows the unencrypted text used in agree session key, then the receiver must be able to compute the session key. The easiest way is asking the receiver to sign that unencrypted text. There left one more problem, if intruder discloses both party secret key, we hope that the unencrypted text will not be divulged. This can be achieved by employing temporary public key technique. And this forms the skeleton of our protocol.

7: PROPOSING A NEW PROTOCOL

We present a new adaptive deniable authentication protocol, which is able to provide perfect forward secrecy. Our protocol is founded on the following assumptions:

1. The public key encryption is secure against adaptive chosen ciphertext attack [17].
2. The signature scheme is existential unforgeable against adaptive adversary [13].
3. Random oracle exists [2].

The protocol is described as follows:

S generates a temporary public key *y* and the corresponding private key *x*, and calculates the followings:

$$ I=P_R(y) $$

and sends *I* and *Cert*(*S*) to *R*.

Upon receiving the message, *R* first verifies the certificate of *S* and picks a random variable t_R to evaluate *J* by:

$$ J=P_Y(P_S(Sig_R(t_R),t_R)) $$

and sends *J* and *Cert*(*R*) to *S*.

After verifying the certificate of R, S will calculate a session key *k* and the message pair (*M*, *D*) by the following equations:

$$k = h(y \| t_R)\,, \quad D = h(k \| M)$$

S will send (*M*, *D*) to *R*. *R* can calculates the session key as well as shown in Fig. 2..

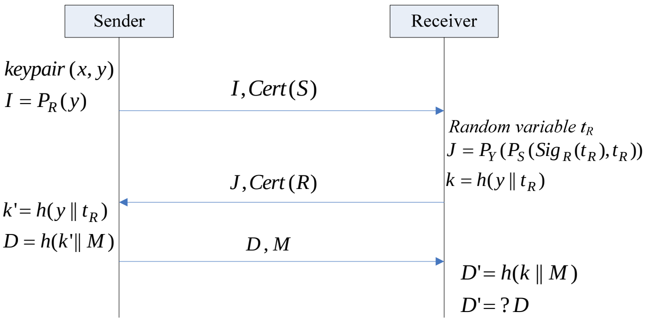

Fig. 2. Proposed protocol

8: SECURITY ANALYSIS

We analysis our protocol in the following aspects: Completeness, Soundness, Adaptive deniable, and Forward Secure.

1. Completeness: It directly follows the flow of the protocol. If the entities honestly execute the protocol without the intervention of the adversary, they will be able to calculate the same key *k*.
2. Soundness: Any passive or active adversary is unable to obtain the session key *k* or send a message pair (*M*, *D*) which is accepted by the receiver. The reason is that the key *k* is calculated by the value *y* and t_R. However, the value *y* is encrypted with the receiver's public key. And more important, the value t_R is encrypted with the sender long term public key and short term public key and is also signed by the receiver. This guarantees that only the receiver and the sender are able to calculate the key *k*. Also, without the key *k*, no one may output a valid message pair (*M*, *D*) which is accepted by the receiver, since *D* asserts the integrity of *M*.
3. Adaptive deniable: If a scheme is not adaptive deniable, we notice that there must exist a protocol that the sender will "accept" at the end but the receiver is unable to calculate the key *k*. We argue that the receiver here must be able to calculate the key. Since the key *k* comes from two value *y* and t_R. The receiver must know the value *y*, for the reason that *y* is encrypted with the receiver public key. The value t_R is signed by the receiver, which also implicate the receiver must know the value t_R. Therefore, this protocol is adaptive deniable.

   Such a protocol is vulnerable against adaptive attack with blind signatures. Any attacker can compromise with the receiver that t_R is selected by the attacker. This t_R will be blinded using any suitable blind signature and signed by the message receiver. Then, the attack will break the protocol and forbidding the receiver *R* to know the key *k*.

   There are several ways to solve this problem. Firstly, we may set a fixed padding system for the signature of t_R. For such a padding scheme, it may help the message receiver to recover the blinded message. Secondly, to help the message receiver to recover the value t_R, we may select the public key encryption scheme in 4th flow as those insecure against chosen message attack. In this way, after seeing the encrypted message in the 4th flow, the receiver can deduce the value t_R, as well as the session key. Take a look of the problem in another way, blind signature is a strong assumption in the attack. Attacks involving blind signatures require the receiver to sign something he does not know. It is an unfair and dangerous requirement for the receiver. A dishonest attacker can blind a contract, and ask the receiver to sign it.

   Our protocol is plausibly weak against attacks involving blind signature, which is believed to be a too strong assumption and can be possibly solved in the above methods. It is also out of our scope owing to it is too strong assumed.
4. Forward Secrecy: Although in deniable protocol, we send the message in the form (*M*, *D*), if we use the *k* to encrypt the message instead, then the importance of forward secrecy is more significant.

Key *k* can only be calculated by the value *y* and t_R. Once the long term secret (private key) of the receiver is known to the adversary, *y* is immediately exposed. That's why Boyd *et al.* [3] and Shao *et al.* [19] do not provide forward secrecy. But we can see that the value t_R is not sent publicly and is explicitly delete after each session key is calculated. We not only encrypt t_R with the sender's public key, but also packaging it with the temporary public key *y*. Thus, even if the long term secret (private key) of the sender is disclosed later, the temporary private key *x* is erased and therefore no one, including the sender, can decrypt the message and retrieve the value t_R.

9: COMPARISONS

We compare the previous literature with our scheme from security to performance aspects as shown in Table 2. We can see that our scheme is the only one achieves adaptive deniable and forward secure simultaneously. Also, we use only 3 rounds of message and do not require a public directory as a setup. The public directory not only stores the public key of each user, it also records some verifier of the message. When the sender is accused for sending a particular message, he may use those verifiers to defense his actions.

Protocols	Aumann *et al.* [1]	Boyd *et al.* [3]	Fan *et al.* [12]	HSIEH *et al.* [14]	Chang *et al.* [7]	Shao *et al.* [19]	Our protocol
Security Analysis							
Non-adaptive attack.	Secure	Secure	Insecure	Secure	Secure	Secure	Secure
Adaptive attack	Secure	Secure	Insecure	Insecure	Insecure	Secure	Secure
Forward Secrecy	No	No	Yes	Yes	Yes	No	Yes
Mutual Authentication	No discussion	No	No	No	Yes	No	Yes
Cost analysis							
Public Directory	Yes	No	No	No	No	No	No
Communication rounds							
Senders	2	1	2	2	2	1	2
Receivers	2	2	1	1	1	0	1
Total	4	3	3	3	3	1	3
Exponentiations or public key operations							
Senders	1	1	3	3	4	1	4
Receivers	1	1	3	3	4	3	4
Total	2	2	6	6	8	4	8
Hash							
Senders	0	2	1	2	2	2	2
Receivers	0	2	1	2	3	2	2
Total	0	4	2	4	5	4	4

Table 2. Comparisons of deniable authentication protocol

10: CONCLUSIONS

In this paper, we first categorize the possible types of deniable authentication protocols according to the security strength. After the categorization, we may resolve disputes over the term “deniable”. Next we show Chang et al.’s scheme is not secure under adaptive attack. Then we propose a new scheme, which can remain secure under the adaptive attack and at the same time provide forward secrecy. Finally, we compare our scheme with those deniable authentication protocols and show the characteristics of our protocol. In the future research we will provide a formal proof for the security of our protocol. It is not difficult to show authentication property of our scheme, but it is not easy to prove the deniable property, especially when the scheme is adaptive deniable.

References

1. Y. Aumann, M. Rabin, “Authentication enhanced security and error correcting codes,” Crypto 98, Santa Barbara, CA, USA, LNCS 1462, Springer-Verlag, Berlin, 1998, pp. 299–303.

2. M. Bellare and P. Rogaway, “Random oracles are practical: a paradigm for designing efficient protocols,” In First ACM Conference on Computer and Communications Security, pp. 62-73, 1993.

3. C. Boyd, W. Mao and K. G. Paterson, “Deniable Authenticated Key Establishment for Internet Protocols.” Security Protocols, 11th International Workshop on Security Protocols, Cambridge University, April 2003, Lecture Notes in Computer Science.

4. D. R. L. Brown, “Deniable Authentication with RSA and Multicasting,” from Cryptology ePrint Archive: Report 2005/056, 24 Feb 2005.

5. R. Canetti, C. Dwork, M. Naor and R. Ostrovsky, “Deniable encryption,” Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology ACM 1997, pp. 90-102.

6. T. Cao, D. Lin, and R. Xue, “An Efficient ID-based Deniable Authentication Protocol from Pairings,” Proceeding of the 19th international Conference on Advanced Information Networking and Applications (AINA’05), IEEE 2005, pp. 388-391.

7. Y.F. Chang, C.C. Chang, and C.L. Kao, “An Improvement on a Deniable Authentication Protocol,” ACM SIGOPS Operating Systems Review Volume 38 , Issue 3 (July 2004) pp. 65-74.

8. D. Chaum and H. van Antwerpen, “Undeniable signature,” Advances in Cryptology – CRYPTO’89, LNCS 435, Springer 1991, pp. 212-216.

9. D. Chaum and E. van Heyst and B. Pfitzmann, “Cryptographically Strong Undeniable Signature, Unconditionally Secure for the Signer, Advances in Cryptology – CRYPTO’91, LNCS 576, Springer 1992, pp. 470-484.

10. X. Deng, C.H. Lee, H. Zhu, “Deniable authentication protocols,” IEE Proceedings Computers and Digital Techniques 148 (2) (2001) pp. 101–104.

11. C. Dwork, M. Naor, A. Sahai, “Concurrent zero-knowledge,” Proc. 30th ACM STOC ’98, Dallas TX, USA, 1998, pp. 409– 418.

12. L. Fan, C.X. Xu, J.H. Li, “Deniable authentication protocol based on Diffie – Hellman algorithm,” Electronics Letters 38 (4) (2002) pp. 705– 706.

13. S. Goldwasser, S. Micali, R. L. Rivest, “A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks,” SIAM Journal on Computing, pp. 281-308, 1988

14. B.T. Hsieh and H.M. Sun, “An Improvement of a Deniable Authentication Protocol,” IEICE TRANS. COMMU., VOL.E87-B, NO.10 OCTOBER 2004.

15. D. Jablon, “Strong password-only authenticated key exchange. ACM Computer Communication Review,” ACM SIGCOMM, 26(5): pp. 5-20, 1996.

16. M. Naor, “Deniable Ring Authentication,” Proceedings of the 22nd Annual International Cryptology Conference on Advances in Cryptology ACM 2002, pp.481-498.

17. C. Rackoff and D. Simon, “Noninteractive zero-knowledge proof of knowledge and chosen ciphertext attack,” In Advacnces in Cryptology-Crypto’91, pp. 433-444, 1991.

18. M. Di Raimondo and R. Gennaro, “New approaches for deniable authentication,” In Workshop on Provable Security, 2004.

19. Z.H. Shao, “Efficient deniable authentication protocol based on generalized ElGamal signature scheme,” Compute. Stand. Interfaces, 2004, 26, pp. 449–454

20. Y. Shi and J. Li, “Identity-based deniable authentication protocol,” Electronics Letters Vol.41 No.05, 3rd March 2005, pp. 241-242.