A Non-interactive Deniable and Negative Authentication Scheme in Random Oracle Model

Hongfeng Zhu, Tianhua Liu, Shuai Geng, Yuanle Zhang, Liwei Wang

Software College, Shenyang Normal University, China
zhuhongfeng1978@163.com, liutianhua@sina.com, 1036103490@qq.com,
412792619@qq.com, 1696751943@qq.com

Abstract

Deniable authentication is an essential cryptography paradigm, which enables a receiver to identify the source of a given message, but the receiver cannot prove the source of the message to any third party over an insecure network. Based on negative database (NDB) generate algorithm, this paper presents a negative deniable authentication protocol, named NDAP, aiming to require one ciphertext for achieve mutual authentication, deniability and the message transmission secretly without a central node. Moreover, our scheme owns non-interactive attribute which make it more efficient. Compared with the related literatures recently, our proposed scheme can not only own high efficiency and unique functionality, but is also robust to various attacks. Finally, we give the security proof and the comparison with the related works.

Keywords: Non-interactive, Deniability, Privacy protection, Negative authentication

1 Introduction

Deniable authentication protocol is a special authentication protocol with deniable attributes. Deniable attribute refers to the receiver can confirm the source of the message, but cannot prove the origin of the message to a third party, because the whole process can be done by the receiver himself/herself. Electronic voting systems and secure negotiation over the Internet are always used deniable authentication scheme, because only the specified receiver can know the real identity of the sender, in this way can protect the friendship between voters.

The concept of deniable authentication was first proposed by Dwork et al. [1] in 1998 based on concurrent zero-knowledge proof, but this scheme required timing constrain and the proof of knowledge is subject to a time delay in the authentication process. Another deniable authentication protocol was developed independently by Aumann and Rabin [2] under the factoring assumption, but the shortage of this protocol is it needs a public directory between the sender and the receiver. Later, Deng [3] proposed two deniable authentication schemes based on the factoring problem and the discrete logarithm problem respectively, but these schemes also need a trusted directory. Fan et al. [4] proposed a simple deniable authentication protocol based on the Diffie-Hellman key exchange protocol in 2002. But the schemes [4] did not provide formal analysis and were broken or improved in [6-7]. Raimondo et al. [8] provided a security proof for his new approaches about deniable authentication. They [8] extended the ideas from authenticated key exchange protocols to the setting of deniable authentication protocols. However, the main disadvantage of the interactive deniable authentication scheme is the cost of communication between the sender and the receiver.

Though these schemes have improved on the safe side, they also cannot to reduce the cost and time. Non-interactive models are more efficient then interactive models because the information is transmitted in a single exchange. Later, to reduce the communication cost, several non-interactive deniable authentication schemes have been proposed in [12-20]. In 2004, Shao et al. [12] proposed the first non-interactive deniable authentication (NIDA) protocol based on generalized ElGamal signature scheme. But these non-interactive schemes did not give a security model. Hwang and Chao [14] present a non-interactive deniable authentication protocol with anonymous sender protection in 2010. In 2014, Li et al. [15] proposed an efficient identity-based deniable authentication protocol using bilinear pairings for ad hoc networks and provides a formal security proof. In 2015, YH Chuang et al. [16] proposed a secure non-interactive deniable authentication protocol with certificates based on elliptic curve cryptography, and WM Shi et al. [17] also proposed a novel quantum deniable authentication protocol without entanglement. Then S Mandal [18] proposed an ID-based non-interactive deniable authentication protocol based on ECC in 2017, we also proposed an efficient chaotic maps-based deniable

*Corresponding Author: Hongfeng Zhu; E-mail: zhuhongfeng1978@163.com
DOI: 10.3966/160792642020052103003

authentication group key agreement protocol [19] in the same year. In 2018, we also proposed [20] based on chaotic maps.

In our proposed scheme, we also used the negative database (NDB). Negative representation [21] is a new way of representing data, which was first proposed by Esponda et al. [21-23] in 2004. Negative database is a form of negative representation. Enhance the security of data by storing the compressed form of the complement of the positive database. Data can be hidden in this way, on the one hand to make the data more secure during transmission, on the other hand, to carry out security authentications [5, 9, 13]. In [10, 21], Esponda et al. proved that it is NP-hard to recover the original data from negative database. At present, some negative database generation algorithms have been proposed.

At present, the negative database has been applied to some fields, but the characteristics of the negative database have not been fully exploited and utilized, and it has a good application prospect. First of all, it is difficult to recover from the negative database to the original data. This forms the security foundation of the negative database, and also makes the negative database available for various privacy protection and data security applications. Traditional security technologies, such as the classic encryption and decryption algorithm, encrypt data and then decrypt the data before it can be operated or calculated. Repeated encryption and decryption consumes more computing resources and brings certain security risks. This is difficult to apply in big data environments or devices with weak computing power, such as mobile terminals. The negative database can support some operations and calculations without repeated "encryption and decryption", and has higher efficiency, suitable for big data environments and devices with weak computing power. In some applications that require high security, you need to change the ciphertext frequently to improve security. All of the above are the nature of the negative database that has been explored at present. The research of negative database is still in its infancy, and more features will be discovered in the future.

The purpose of introducing a negative database in this paper is to encrypt the transmitted information, thereby improving the security of data transmission. It is also an application of the negative database encryption property. The negative database generation algorithm generates a negative database that is difficult to recover. Even if the adversary gets the information, it cannot recover the original data information.

The rest of the paper is organized as follows: Some related works are given in Section 2. Next, a non-interactive deniable authentication scheme based on the negative database is described in Section 3. In Section 4, we give the security of our proposed scheme. The efficiency analysis of our proposed scheme is given in Section 5. This paper is finally concluded in Section 6.

2 Related Work

2.1 Negative Database

Our proposed scheme is based on the negative database (NDB) [11]. Negative database is a new type of privacy protection technology, the so-called negative database refers to the positive database will be negative representation, that is, according to certain algorithms the positive database will be change into its complement, and then compressed this complement. As [23], assuming the original data is a database containing nx records, that means $DB = \{x_1, x_2, ..., x_{nx}\}$. Each record in the DB is a binary string of length m. The complete set expressed as $U = \{0,1\}^m$, the complement set of DB expressed as $U - DB$, NDB only stores the record of $U - DB$, $U - DB$ is often much larger than DB. Usually, NDB need to cover a large number of binary strings, it is difficult to accurately represent and store one by one. "*" is defined as a "do not care" notation to compress the NDB. The symbol "*" can represent "1" and "0". Given a string defined upon the alphabet $\{0,1,*\}$, the specified positions are the positions with value "0" or "1", and the unspecified positions are the positions with "*". From the table we can see that the size of NDB can be much smaller than that of $U - DB$, even smaller than that of DB, and that one $U - DB$ can generate many different NDBs. If an NDB can be reversed to obtain the DB in polynomial time, the NDB is said to be easy-to-reverse, otherwise, it is hard-to-reverse. The relationship between DB, NDB and U-DB-NDB is shown in Figure 1.

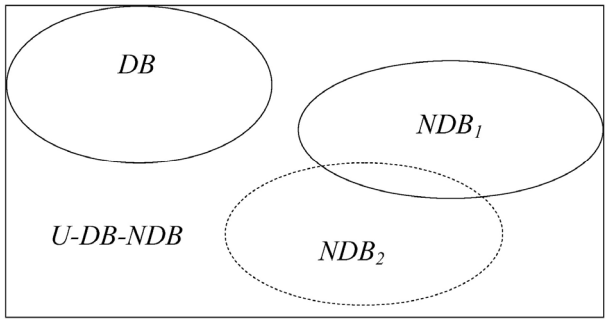


Figure 1. The relationship between DB, NDB and U-DB-NDB

2.2 Generation Algorithm of the Negative Database

At present, some negative database generation algorithms have been proposed. Prefix-algorithms [23] is the first algorithm used to generate a binary negative database, the feature of this algorithm is simple and efficient. However, the negative database generated by

the prefix-algorithm is complete and easy-to-reverse. In order to generate a hard-to-reverse negative database, a RNDB algorithm also be proposed in [23]. RNDB algorithm uses some random numbers, in order to increase the uncertainty of the negative database, make it more difficult to reverse. Esponda et al. used q-hidden algorithm in [5], this kind of NDB called K-NDB. Although the negative database generated by the q-hidden algorithm may not be complete, it is hard-to-reverse. Liu et al. [13] proposed hybrid-NDB by combine prefix-algorithm and q-hidden algorithm, this algorithm not only complete but also hard-to-reverse. Then Liu et al. also proposed another algorithm called p-hidden algorithm later in [9], which is an improved version of the q-hidden algorithm. It was demonstrated that the p-hidden algorithm could generate more hard-to-reverse NDBs than the q-hidden algorithm. Zhao et al. [10] proposed K-hidden algorithm, which is more fine-grained than the hybrid-NDB algorithm. It could generate more hard-to-reverse NDBs than the hybrid-NDB algorithm. Besides, there are also 0-hidden algorithm [10] and 1-hidden algorithm.

3 Our Proposed Scheme

In this section, we outline our proposed a non-interactive deniable authentication scheme, our proposed protocol in this paper is based on the negative database (NDB). This scheme comprises three phases: setup phase, encrypt phase and decrypt phase. The notations used throughout this paper are defined in Table 1.

Setup phase. In this phase, without loss of generality, we choose Alice as one of the *N*-1 voters, and her *ID* is denoted as ID_A , her private key is *a*, and the corresponding public key is g^a . For the random chosen vote count node/person, we choose Bob, and her ID is denoted ID_B , his private key is *b*, and the corresponding public key is g^b .

Encrypt phase. When Alice wants to send the message M to the receiver Bob, she chooses a random number r from $(1, p\text{-}1)$, and then according to Bob's public key g^b and the random number r computes $c_1 = (g^b)^r ID_A$, $c_2 = (g^b)^a M$. Next, Alice computes NDB by the negative database generation algorithm G , we also use timestamp T to compute the NDB, $NDB = G(ID_B \| c_1 \| c_2 \| T)$. Finally, Alice sends $\{g^r, c_1, c_2, NDB, T\}$ to Bob.

Table 1. Notations

Symbol	Description
ID_A	Alice's ID
ID_B	Bob's ID
a	Private key of Alice
g^a	Public key of Alice
b	Private key of Bob
g^b	Public key of Bob
M	The message Alice wants to transmit to Bob
r	The random number Alice selects for Bob
G	Negative database generation algorithm
NDB	Negative database generated by G
$X \to Y : M$	Mes Message M is sent by X to Y

Decrypt phase.

(1) On receiving the message $\{g^r, c_1, c_2, NDB, T\}$ from Alice, Bob first check whether the timestamp T outdated, terminate the session if the check fails. Otherwise, Bob can recover the identity of the sender by using his private key b by computing $ID_A = c_1 / (g^r)^b$.

(2) Bob also can know the message M sent by the sender by computing $M = c_2 / (g^a)^b$. This step is also authenticating the sender, if the sender is the "Alice", the messages M will be the valid information, and if not, the recovered messages M will be as the invalid information.

(3) Next, Bob also need to compute $NDB' = G(ID_B \| c_1 \| c_2 \| T)$ to checking the integrity of the message, if $NDB = NDB'$, M is valid. Otherwise, the messages M are invalid or have been damaged during transmission.

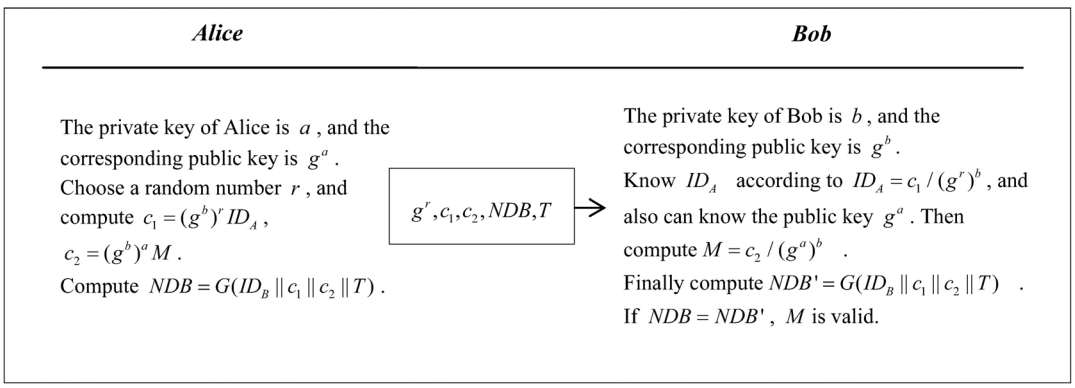


Figure 2. Our proposed protocol

4 Security Analysis

4.1 Security Model

The model that we use is as follows in Table 2.

Table 2. Security model of our scheme

Participants.	In a network runs a deniable authentication protocol Π with a number of participants, where each participant is either a sender or a receiver. We assumed that there is only one receiver. Each participant may have serveral instances called oracles involved in distinct executions of the protocol Π. We use Π_S^i (resp. Π_R^j) to represent sender's *i*th protocol instance.
Queries.	We define a series of games between a challenger and the attacker A, in order to prove our scheme is security. The attacker A, interacts with the participants and tries to break the authentication of the participants. The attacker A is allowed to issue the following queries in any order.
$Excute(\Pi_S^i, \Pi_R^j)$	This query models passive attacks, where the attacker eavesdrops on the scheme's honest executions between Π_S^i and Π_R^j.
$\mathrm{Re}veal(\Pi_S^i)$	This query models the security on transmission. The attacker makes this query to obtain the information on transmission of the instance Π_S^i.
$SendSender(\Pi_S^i, m)$	This query models when A send message to instance Π_S^i, A obtain the message m which the response Π_S^i generates in processing according to the scheme. The attacker is also allowed to use $SendSender(\Pi_S^i, Start)$ initiate the scheme.
$Send\,\mathrm{Re}ceiver(\Pi_R^i, m)$	This query models attacker obtain the message that the server instance Π_R^i would generation receipt of the message m.
$Corrupt(U)$	This query returns to the attacker A the long-lived key pw_U for participant sender.
$Test(\Pi_U^i)$	The attacker only can query this form once time, but can execute this query at any time. To respond to this query by $T_e + O(l) + O(N)$, a random bit $b \in \{0,1\}$ is selected. If $b = 1$, the authentication result is returned. Else, a random value is returned. Eventually, A outputs $b' \in \{0,1\}$ as its guess result, according to this result decide what A will get. If $b = b'$, we say that the attacker A wins the game.

Freshness of oracle. An oracle Π_S^i is called fresh if and only if the following conditions hold: (1) Π_S^i has accepted, (2) Π_S^i or its partner (if exists) has not been asked a Reveal query after their acceptance.

Protocol security. The security of our scheme is modeled by a game $Game(\Pi, A)$. During the game, the attacker can do many queries mentioned above between Π_U^i and Π_S^j. If A ask the query $Test(\Pi_S^i)$, A outputs $b' \in \{0,1\}$ as its guess result, because the attacker wants to know the bit b correctly. More precisely, we define the advantage of A as follows:

$$Adv_{\Pi,D}(A) = |2\Pr[b' = b] - 1|$$

If $Adv_{\Pi,D}(A)$ is negligible, we call the scheme Π is secure.

4.2 Computational Assumption

We use the decisional Diffie-Hellman (DDH) assumption in our scheme to do security proof.

Definition 1. The decisional Diffie-Hellman (DDH) assumption can be defined by $Exp_{P,p}^{ddh\text{-}real}(W)$ and $Exp_{P,p}^{ddh\text{-}rand}(W)$ precisely. An attacker W is provided by uP, vP and uvP in the $Exp_{P,p}^{ddh\text{-}real}(W)$, and uP, vP, wP in the $Exp_{P,p}^{ddh\text{-}rand}(W)$, and u, v, w are drawn at random from Z_p^*. More precisely, we define the advantage of W as follows:

$$Adv_{P,p}^{ddh}(W) = \max\{|\Pr[Exp_{P,p}^{ddh\text{-}real}(W) = 1] - \Pr[Exp_{P,p}^{ddh\text{-}rand}(W) = 1]|\}$$

4.3 Security Analysis for Security Requirements

We discuss our scheme in three main security attributes: deniability, ciphertext with authentication and privacy protection. And the security proof of correctness will show in Appendix A.

(1) The Deniability of Our Scheme

Theorem 4.1. Our proposed scheme owns deniability.

Proof. Figure 3 describes the simulation processes of our proposed scheme. To prove the proposed scheme is deniable, we should prove that Bob can simulate all the process between Alice and Bob by Bob himself. Although Bob (the receiver) cannot know Alice's (the sender) private key, Bob (the receiver) still can simulate the whole transcript process because the public key and the identity can be easily achieved. Although Bob cannot get the private key of Alice, he still can compute $c_1 = (g^b)^r ID_A$ based on the public

key of Bob and the identity of Alice. To simulate the transcripts on message, Bob selects a random number r, then Bob computes $c_1 = (g^b)^r ID_A$, and $c_2 = (g^a)^b M$. Bob computes c_2 by Alice's public key, and Bob's private key. The transcripts $\{g^r, c_1, c_2, NDB, T\}$ in simulation are indistinguishable from those of the sender Alice. Therefore, the receiver Bob cannot prove to a third party that the transcripts were produced by Alice. The core reason is that Bob can use his own private key and the voter's public key to simulate all the processes. Furthermore, our proposed scheme has also achieved the strong deniability (Strong deniability means that the sender can deny to have ever authenticated anything to receiver after execution of the protocol).

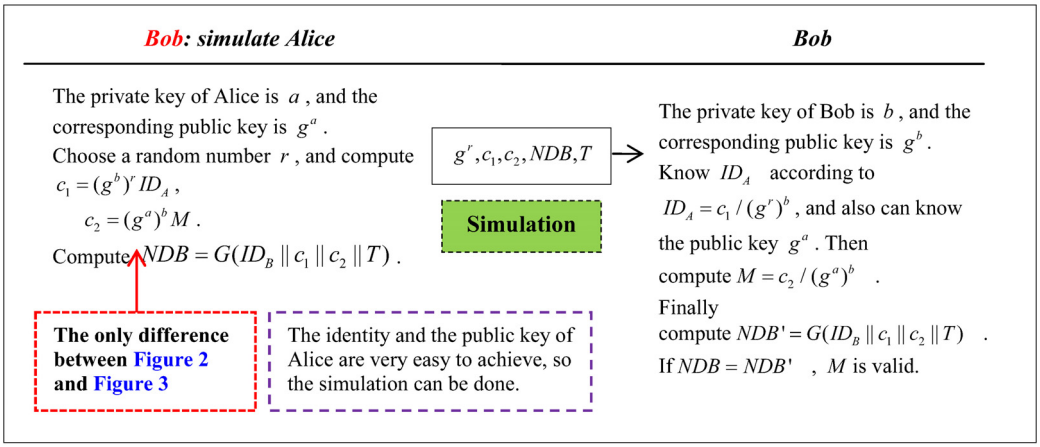


Figure 3. The simulated processes of proposed scheme

In addition, the deniable of the cryptographic scheme can also be demonstrated by game. As long as it can be proved in the game that the verifier D does not have any advantage to win the game, it means that the scheme is deniable. As defined in game, the game consists of three parts: the initialization phase, the challenge phase, and the verify phase.

Initialization phase. Set C to be the challenger in the game, D is the verifier in the game, and assumes that C has the ability to make any legitimate user in the system generate a legitimate deniable authentication encryption (DAE) ciphertext, P0 and P1 play two honesty participants in the game.

Challenge phase. The verifier D submits a plaintext message m to the challenger. After receiving the plaintext information of the verifier, the challenger C first selects a random bit $\gamma \in \{0,1\}$, and then, through interaction with P_γ, causes the user P_γ to generate m. DAE ciphertext δ. Finally, the challenger C returns the ciphertext δ of m to D as its challenge.

Verify phase. The verifier D returns to the challenger C one bit $\gamma' \in \{0,1\}$. If $\gamma' = \gamma$, D won the game.

Both P0 and P1 in the game have the ability to generate the legal DAE ciphertext δ of m, and the ciphertext they produce has the same probability distribution, which is indistinguishable from the third party D. In other words, the advantage of D winning in the game is $Adv(D) = |\Pr[\gamma' = \gamma] - 1/2| \approx 0$. Based on the above analysis process, it can be concluded that the deniable authentication encryption scheme obtains deniable.

(2) The Security of Ciphertext with Mutual Authentication

Theorem 4.2. Our proposed scheme is ciphertext with authentication

Proof. Our proposed scheme is based on PKC(Public Key Cryptosystem), so there are two key points should be taken into account: the first one is the transcripts must mix with a large random number, and the second one is the secret message cannot be encrypted by any public key directly. Therefore, we construct $c_2 = (g^b)^a M$ to covered the secret message M and others' necessary information. And for assuring integrity, we construct $NDB = G(ID_B \| c_1 \| c_2)$ by negative database generate algorithm. Only Bob can decrypt c_2 using his own secret key. and furthermore authenticate the integrity by comparing with the $NDB = G(ID_B \| c_1 \| c_2)$. Additionally, since the random number r is different in every time, attackers cannot guess the number easily. Therefore, the proposed scheme provides ciphertext with mutual authentication security.

(3) The Security of Privacy Protection

Theorem 4.3. Our proposed scheme is privacy protection

Proof. in our security proof, we divide the participants into three characters: the sender, the receiver and the outsiders (including attacker, any curious nodes and so on). The sender's identity is anonymity, because the identity ID_A is covered by $c_1 = (g^b)^r ID_A$, and only the legal Bob can recover the real identity ID_A by Bob's private key. In our scheme, because our scheme is

PKC-based, so only the real identity ID_A known by the legal receiver, the sender's public key can be known. The sender must know the receiver's identity because our scheme is adopted PKC.

We construct $c_1 = (g^b)^r ID_A$ to cover the sender's identity. The encrypted message c_1 is generated from r which is different in each session and is only known by the sender Alice. The receiver can decrypt c_1 by his own secret key. Additionally, since the value r of the random element is different in every time, attackers cannot guess the number easily. Therefore, the proposed scheme provides privacy protection.

(4) Anti-replay Attack

The important parameter T added in the proposed protocol, under the assumption that the sender and the receiver are time synchronized, the timestamp T can identify the expiration time of the message validity period, and the validity period of different messages is different, and the receiver cannot determine if the message is valid, and then make a decision to accept or reject the message.

5 Performance Analysis

5.1 Space and Time Complexity

In our proposed scheme, our scheme is PKC-based, and we also used negative database in our scheme. If we assume that l is the length of the key, and there are m voters have registered successfully, the space complexity is $O(l*m)$.

In our proposed scheme, we used public key encryption system to achieve identity anonymity, which involves the exponentiation. Suppose the time complexity for generating an NDB is T_{NDB}, the time complexity of the public key encryption system is T_e. So the time complexity of the our proposed scheme is $T_e + T_{NDB}$. If SHA-512 is selected as the one-way hash function, then $T_h = O(l)$, if the NDB generated by the K-hidden algorithm, the $T_{NDB} = O(l*r*K)$, so if K and r are small, the time complexity of T_{NDB} will be $O(l)$. Consequently, if SHA-512 and the K-hidden algorithm are adopted, then the time complexity of the login and authentication phase will reduce to $T_e + O(l)$. Besides, in the scheme, the server also need to compare the NDB' computed by the server and the NDB received from the client. If we assume that N is the entries NDB contains, so the time complexity of the comparison is $O(N)$. Therefore, the total time complexity of the authentication phase is $T_e + O(l) + O(N)$.

According to above analyses, the efficiency of the proposed protocol is very competitive compare with other non – interactive deniable authentication schemes. We used negative database algorithm in our scheme, in this way, our protocol can be more security, because this is an algorithm rarely used in other schemes. The computational cost of our scheme shows in Table 3. The efficiency comparison shows in Table 4.

Table 3. Computational cost of our scheme

	Total
Sender	$2T_e + 2T_m + T_{NDB}$
Receiver	$2T_e + 2T_m + T_{NDB}$
Total	$4T_e + 4T_m + 2T_{NDB}$

Table 4. Efficiency comparison

	Protocols		Susmita Mandal [18]	Bin Wang [11]	Ours
Efficiency	Computation	The sender	$4T_{pm} + 4T_h$	$5T_e$	$2T_e + 2T_m + T_{NDB}$
		The receiver	$2T_{pm} + 2T_h$	$3T_e$	$2T_e + 2T_m + T_{NDB}$
		Total	$6T_{pm} + 6T_h$	$8T_e$	$4T_e + 4T_m + 2T_{NDB}$
	Communication	Messages	1	1	1
		Rounds	1	1	1
		FS	No	No	Yes
		RA	No	No	Yes
		ST	Yes	Yes	Yes
		RGA	Yes	Yes	Yes
		SNDB	No	No	Yes
	Design	Non-interactive	Yes	Yes	Yes

Note. T_e: the time required to calculate exponentiation; T_m: the time required to calculate multiplication; T_h: the time required to calculate a one-way hash function operation; T_{pm}: the time required to compute elliptic curve scalar point multiplication; T_{NDB}: the time required to compute a negative database. UA: User Anonymous; FA: Forward Anonymous; FS: Forward secrecy; RA: Replay attack; ST: Security on transmission; RGA: Resist guessing attack; SNDB: Security of the NDB.

6 Conclusion

In this paper, we introduce some contents about the deniable authentication scheme, negative database generate algorithm and authentication scheme base on negative database, then base on the past protocols we proposed a new non-interactive deniable and negative authentication scheme. The addition of negative database technology enhances security during information transfer. Then we analyzed our protocol, and proved our protocol can be simulated by the receiver itself, our scheme is really deniable. We compared with other protocols in computational cost and security at last. The security level of the proposed scheme is not less than other non-interactive deniable authentication protocols based on both the encryption algorithm and one-way hash function. The proposed protocol is more efficient than other deniable authentication schemes.

Acknowledgement

This work was supported by the Liaoning Provincial Natural Science Foundation of China (Grant No. 2019-MS-286), and Shenyang Science & Technology Innovation Talents Program for Young and Middle-aged Scientists (2019).

References

[1] C. Dwork, A. Sahai, Concurrent Zero-knowledge: Reducing the Need for Timing Constraints, in: H. Krawczyk (Ed.), *Advances in Cryptology — CRYPTO '98. CRYPTO 1998, Lecture Notes in Computer Science*, Vol. 1462, Springer, 1998, pp. 442-457.

[2] Y. Aumann, M. Rabin, Efficient Deniable Authentication of Long Messages, *International Conference on Theoretical Computer Science in Honor of Professor Manuel Blum's 60th birthday*, Hong Kong, China, 1998, pp. 20-24.

[3] X. Deng, C. H. Lee, H. Zhu, Deniable Authentication Protocols, *IEE Proceedings- Computers and Digital Techniques*, Vol. 148, No. 2, pp. 101-104, March, 2001.

[4] L. Fan, C. X. Xu, J. H. Li, Deniable Authentication Protocol Based on Deffie-Hellman Algorithm, *Electronics Letters*, Vol. 38, No. 14, pp. 705-706, July, 2002.

[5] F. Esponda, E. S. Ackley, P. Helman, H. Jia, S. Forrest, Protecting Data Privacy through Hard-to-Reverse Negative Databases, *International Journal of Information Security*, Vol. 6, No. 6, pp. 403-415, October, 2007.

[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, No. 1, pp. 274-280, August, 2005.

[7] R.-W. Zhu, D.-S. Wong, C. H. Lee, Cryptanalysis of a Suite of Deniable Authentication Protocols, *IEEE Communications Letters*, Vol. 10, No. 6, pp. 504-506, June, 2006.

[8] M. D. Raimondo, R. Gennaro, New Approaches for Deniable Authentication, *Journal of Cryptology*, Vol. 22, No. 4, pp. 572-615, October, 2009.

[9] R. Liu, W. Luo, L. Yue, The p-hidden Algorithm: Hiding Single Databases More Deeply, *International Journal of Immune Computation*, Vol. 2, No. 1, pp. 43-55, March, 2014.

[10] D. Zhao, W. Luo, One-time Password Authentication Scheme Based on the Negative Database, *Engineering Applications of Artificial Intelligence*, Vol. 62, pp. 396-404, June, 2017

[11] B. Wang, Q. Zhao, K. Dai, A Non-interactive Deniable Authentication Scheme in the Standard Model, *Iacr Cryptology Eprint Archive*, Report 2011/693, December, 2011.

[12] Z. Shao, Efficient Deniable Authentication Protocol Based on Generalized ElGamal Signature Scheme, *Computer Standards & Interfaces*, Vol. 26, No. 5, pp. 449-454, September, 2004.

[13] R. Liu, W. Luo, X. Wang, A Hybrid of the Prefix Algorithm and the q-hidden Algorithm for Generating Single Negative Databases, *Proceedings of the 2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS 2011)*, Paris, France, 2011, pp. 31-38.

[14] S.-J. Hwang, C.-H. Chao, An Efficient Non-interactive Deniable Authentication Protocol with Anonymous Sender Protection, *Journal of Discrete Mathematical Sciences and Cryptography*, Vol. 13, No. 3, pp. 219-231, June, 2010.

[15] F. Li, P. Xiong, C. Jin, Identity-based Deniable Authentication for Ad Hoc Networks, *Computing*, Vol. 96, No. 9, pp. 843-853, September, 2014.

[16] Y.-H. Chuang, C.-L. Hsu, W. Shu, K.-C. Hsu, M.-W. Liao, A Secure Non-interactive Deniable Authentication Protocol with Certificates Based on Elliptic Curve Cryptography, in: D. Barbucha, N. Nguyen, J. Batubara (Eds.), *New Trends in Intelligent Information and Database Systems. Studies in Computational Intelligence*, Vol. 598, Springer, 2015, pp. 183-190.

[17] W.-M. Shi, J.-B. Zhang, Y.-H. Zhou, Y.-G. Yang, A Novel Quantum Deniable Authentication Protocol without Entanglement, *Quantum Information Processing*, Vol. 14, No. 6, pp. 2183-2193, June, 2015.

[18] S. Mandal, S. Mohanty, B. Majhi, An ID-based Non-Interactive Deniable Authentication Protocol Based on ECC, *Proceedings of the 2017 the 7th International Conference on Communication and Network Security*, Tokyo, Japan, 2017, pp. 48-52.

[19] H.-F. Zhu, Y. Zhang, An Efficient Chaotic Maps-Based Deniable Authentication Group Key Agreement Protocol, *Wireless Personal Communications*, Vol. 96, No. 1, pp. 217-229, September, 2017.

[20] H.-F. Zhu, Y. Zhang, A Secure Non-interactive Chaotic Maps-based Deniable Authentication Scheme with Privacy Protection in Standard Model, *Journal of Computers (Taiwan)*, Vol. 29, No. 3, pp. 109-120, June, 2018.

[21] F. Esponda, E. S. Ackley, S. Forrest, P. Helman, Online Negative Databases, *Proceedings of the Third International Conference on Artificial Immune Systems*, Catania, Sicily, Italy, 2004, pp. 175-188.

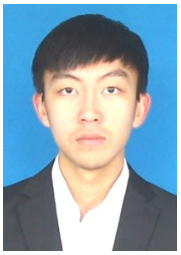
[22] Y. Bao, W. Luo, Y. Lu, On the Dependable Level of the Negative Survey, *Statistics & Probability Letters*, Vol. 89, pp. 31-40, June, 2014.

[23] Y. Lu, W. Luo, D. Zhao, Fast Searching Optimal Negative Surveys, *in Proceeding of the 2014 International Conference on Information and Network Security* (ICINS), Beijing, China, 2014, pp. 82-90.

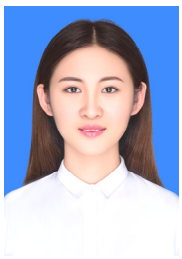
Biographies

Hongfeng Zhu obtained his Ph.D. degree in Information Science and Engineering from Northeastern University. Hongfeng Zhu is a full professor of the software college at Shenyang Normal University. He is also a master's supervisor. He has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. Dr. Zhu had published more than 70 international journal and international conference papers on the above research fields.

TianHua Liu obtained his Ph.D. degree in Information Science and Engineering from Northeastern University. Tianhua Liu is the director of the Academic Affairs Department. TianHua Liu is a full professor of the software college at Shenyang Normal University. He is also a master's supervisor. He has research interests in wireless networks, cloud computing and quantum cryptography. Dr. Liu had published more than 20 international journal and international conference papers on the above research fields.



Shuai Geng, an undergraduate from Shenyang Normal University. He has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. In the four years of college, after completing her studies, he enjoys reading the book related to this major. Under the guidance of the teacher, he has published 3 international journal papers on the above research fields.



Yuanle Zhang, an graduate student of Shenyang Normal University. She has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. After completing her studies, she enjoys reading the book related to this major. nder the guidance of the teacher, she has published one article in EI journals.

Liwei Wang, 24 years old, a postgraduate studying at Shenyang Normal University. She has research interests in wireless networks, mobile computing, cloud computing, social networks and quantum cryptography. After completing her studies, she enjoys reading the book related to this major. Under the guidance of the teacher, she has published one article in EI journals.

Appendix: Security Proof

Theorem 1: Parameters in our scheme as we can see in the next, D is a private key dictionary of length $|D|$. Let Π describes the authentication phase in Figure 2. Suppose that DDH assumption holds, then, $Adv_{\Pi,D}(A) \le \frac{q_G^2}{2^{k+1}} + \frac{(q_s+q_e)^2}{p^2} + 2q_e \cdot Adv_{P,p}^{DDH}(A)$ $+2\max\{\frac{q_s}{|D|}\}$, where q_s denotes the number of *Send* queries; q_e denotes the number of *Execute* queries; q_G denotes the number of NDB generate algorithm to G.

Proof. We prove this theorem through a sequence of games beginning with the real scheme and ending up with a game where attacker A's advantage is zero. For each game $G_i (0 \le i \le 4)$, if A correctly guesses the bit b in the test session, we will use $Succ_i$ to represent.

***GameG*$_0$.** In the random oracle model, this game corresponds to a real attack. In this game, all the instance of sender and the receiver, are modeled as the real execution in the random oracle. $Succ_i$ means that the attacker successfully guesses *Test* query in the pre-use of the bit b, you can get:

$$Adv_{\Pi,D}(A) = 2\,|\Pr(Succ_0) - \frac{1}{2}|$$

***GameG*$_1$.** This game is as the same as the game G_0, G as G_{List}, On each query for which there exists a record (Inp, Outp) in the list, return Outp. Otherwise, randomly choose $Output \in \{0,1\}^k$, send it to A and store the new tuple (Inp, Outp) into the list. For the attacker, this game is almost as the same as the real attack. Hence,

$$\Pr[Succ_1] = \Pr[Succ_0].$$

***GameG*$_2$.** In this game, we simulate all the oracles in game G_1, but not same to G_1. The probability of collisions in output of NDB are at most $q_G^2 / 2^{k+1}$. So we also can compute the probability of collisions in the transcripts is at most $(q_s+q_e)^2/(2p)^2$, where q_s represents the number of queries to the *SendSender* and *SendReceiver* oracles, and q_e represents the number of queries to the *Excute* oracle. So we can get:

$$|\Pr[Succ_2] - \Pr[Succ_1]| \le \frac{q_G^2}{2^{k+1}} + \frac{(q_s+q_e)^2}{(2p)^2}.$$

***GameG*$_3$.** In this game, we change the simulation of queries to the *SendSender* oracle. First, we randomly select a session executed by partner instances Π_U^i and Π_S^j. If $Send\,\mathrm{Re}veiver(\Pi_U^i, M)$ is asked, we choose a random value r from [1, $p-1$], and compute $c_1 = (g^b)^r ID_A$, $c_2 = (g^b)^a M$, $NDB = G(ID_B \| c_1 \| c_2 \| T)$, where g^b is the public key of the receiver. And then return $\{g^r, c_1, c_2, NDB, T\}$ to the attacker. From above, it can be easily seen that this game is perfectly indistinguishable from the previous game G_2. Hence,

$$\Pr[Succ_3] = \Pr[Succ_2]$$

***GameG*$_4$.** In this game, we also change the simulation of queries to the *SendSender* oracle for the selected session in game G_3. This time, when $Send\,\mathrm{Re}ceiver(\Pi_S^j, \{g^r, c_1, c_2, NDB, T\})$ is asked, the server then computes $ID_A = c_1/(g^r)^b$, $M = c_2/(g^a)^b$, computes $NDB' = G(ID_B \| c_1 \| c_2 \| T)$, and check whether $NDB' = NDB$. So we can know the difference between G_4 and G_3 is:

$$|\Pr[Succ_4] - \Pr[Succ_3]| \le q_e \times Adv_{P,p}^{DDH}(W)$$

We construct a DDH solver W, which is a successful attacker to distinguish G_3 and G_4. In the game G_4, the NDB is random independent with the sender's identity. So if the attacker queries $\{g^r, c_1, c_2, NDB, T\}$ to G. If the attacker asks *SendSender* and impersonates sender to receiver successfully, the probability that this event occurs is $q_G / 2^k$. The attacker cannot get the identity because the k is large enough. So the probability of the attacker obtain the information of identity is $q_S / |D|$.

So we can get:

$$\Pr[Succ_4] = \frac{1}{2} + \max\{\frac{q_G}{2^k}, \frac{q_S}{|D|}\}$$

From above, we can get the result:

$$\begin{aligned} Adv_{\Pi,D}(A) &= 2\,|\Pr[Succ_0] - \frac{1}{2}| \\ &= 2\,|\Pr[Succ_0] - \Pr[Succ_4] + \max\{\frac{q_G}{2^k}, \frac{q_S}{|D|}\}| \\ &\le 2(|\Pr[Succ_0] - \Pr[Succ_4] + \max\{\frac{q_G}{2^k}, \frac{q_S}{|D|}\}|) \\ &\le 2(|\Pr[Succ_1] - \Pr[Succ_2] + \Pr[Succ_3] \\ &\quad - \Pr[Succ_4] + \max\{\frac{q_G}{2^k}, \frac{q_S}{|D|}\}|) \\ &\le \frac{q_G^2}{2^{k+1}} + \frac{(q_s+q_e)^2}{p^2} + 2q_e \cdot Adv_{P,p}^{DDH}(W) \\ &\quad + 2\max\{\frac{q_s}{|D|}\} \end{aligned}$$