A non-interactive deniable authentication scheme based on designated verifier proofs

Bin Wang

Information Engineering College of Yangzhou University

Yangzhou City, Jiangsu Province, P.R.China

E-mail:xiaobinw@yahoo.com

Abstract: A deniable authentication protocol enables a receiver to identify the source of the messages but unable to prove to a third party the identity of the sender. In recent years, several non-interactive deniable authentication schemes have been proposed in order to enhance efficiency. In this paper, we propose a security model for non-interactive deniable authentication schemes. Then a non-interactive deniable authentication scheme is presented based on designated verifier proofs. Furthermore, we prove the security of our scheme under the DDH assumption.

Keywords: Deniable authentication, Authentication, Designated verifier proofs, DDH assumption, Provable security;

1. Introduction

In an open network environment, an authentication protocol enables a receiver to verify the legitimacy of a particular sender. A deniable authentication protocol is an authentication protocol which enables the receiver to identify the source of the messages but unable to prove to a third party the identity of the sender.

The notion of deniable authentication was introduced by Dwork et al. [4], which is based on concurrent zero-knowledge proof. Another scheme was developed by Aumann and Rabin [1] independently. Later, Deng et al.[3] proposed two schemes based on Aumann and Rabin's scheme. In 2002, Fan et al. [5] proposed a simple deniable authentication protocol based on the Diffie-Hellman key distribution protocol. Since then deniable authentication schemes have enjoyed a considerable amount of interest from the cryptographic research community. However, security of these schemes is argued by presenting attacks that fail, which provides very weak security guarantees. Several proposed schemes were broken or improved [12,13].

In addition, the above-mentioned schemes are interactive and inefficient. Consequently, several non-interactive deniable authentication schemes have been proposed in order to enhance efficiency [9,10,11].

For the notion of security with respect to authentication schemes the reader is referred to [2, 6]. In 1996, Jacobsson et al. introduced the concept of designated-verifier proofs [8]. Such proofs enable a prover Alice to convince a designated verifier Bob that a statement is true. However, Bob cannot transfer the proofs to a third party. The reason is that Bob himself can simulate such proofs. Combining designated-verifier proofs with the security model for authentication schemes, we develop a security model for non-interactive deniable authentication schemes in this paper. Then we propose a non-interactive deniable authentication scheme. Finally, we prove the security of our scheme under the DDH assumption.

2. Preliminaries

2.1 DDH Problem

DDH problem: Let G be a multiplicative cyclic group of prime order q generated by g. Given $< g^x, g^y, g^z >$, $x, y, z \in Z_q$, decide whether $z = (xy) \bmod q$.

A distinguishing algorithm D is said to (t, ε)-solve the DDH problem in group G if D runs in time at most t and

$$|\Pr[x, y, z \leftarrow Z_q : D(g^x, g^y, g^z) = 1] - \Pr[x, y \leftarrow Z_q : D(g^x, g^y, g^{xy}) = 1]| \geq \varepsilon .$$

We say that G is a (t, ε)-DDH group if there is no polynomial time algorithm D (t, ε)-solves the DDH problem in group G.

3. Deniable authentication schemes

3.1 Syntax of deniable authentication schemes

A non-interactive deniable authentication scheme is a tuple $(\mathrm{Kg,P,V,Sim})$. On input a security parameter $k \in \mathrm{N}$, the randomized polynomial time algorithm **Kg** generates a public key pk and a matching secret key sk. The public/secret key pairs for the prover and the verifier are (pk_P, sk_P), (pk_V, sk_V) respectively.

P and **V** are polynomial time algorithms called the prover and the verifier algorithm respectively.

On input prover's public key pk_P, receiver's secret key sk_V, and a message M, the simulation algorithm Sim generates an authenticator $Authen$ for the message M.

In an initialization step, the prover picks a message M. Then a non-interactive deniable identification protocol can be described as follows:

$(C,d) \leftarrow \begin{bmatrix} (M, Authen) \leftarrow P(M, sk_P, pk_V) \\ (M, Authen) \rightarrow V(pk_P, sk_V) \end{bmatrix}$, where $Authen$ is the authenticator for M. The decision bit $d \in \{0,1\}$. The transcript of the conversation is $C = M \,||\, Authen$.

Correctness: For all $k \in \mathrm{N}$, $(pk, sk) \leftarrow \mathrm{Kg}(1^k)$, we require perfect consistency, meaning that $\Pr\left[d = 1 : (C,d) \leftarrow \begin{bmatrix} (M, Authen) \leftarrow P(M, sk_P, pk_V) \\ (M, Authen) \rightarrow V(pk_P, sk_V) \end{bmatrix} \right] = 1$.

3.2 Security model for deniable authentication schemes

3.2.1 Unforgeability of deniable authentication schemes

Let $\mathrm{NDI} = (\mathrm{Kg,P,V,Sim})$ be a non-interactive deniable authentication scheme. Consider the following game $\mathrm{Exp}^{\mathrm{imp}}_{\mathrm{NDI,A}}(k)$ between an active adversary A and a game challenger S:

Stage1: The challenger S runs the algorithm $\mathrm{Kg}(1^k)$ to obtain key pairs (pk_P, sk_P), (pk_V, sk_V). An empty set Res is created. Then the adversary A is provided with pk_P, pk_V.

Stage2: The adversary A makes the following queries:

(1) Conv queries: A adaptively picks a message M. On input the message M, the initial state St_P of the prover algorithm is set to (sk_P, pk_V, ρ), where ρ denotes fresh random coins chosen by the challenger S. Then the challenger S executes

$(C,d) \leftarrow \begin{bmatrix} (M, Authen) \leftarrow P(M, St_P) \\ (M, Authen) \rightarrow V(pk_P, sk_V) \end{bmatrix}$. S provides A with C (the transcript of the conversation) and sets $\mathrm{Res} \leftarrow \mathrm{Res} \bigcup \{\mathrm{C}\}$.

Output: Eventually, A outputs St_A. Then A picks a message M^*. The adversary A wins if :

(1) $(C^*, d^*) \leftarrow \begin{bmatrix} (M^*, Authen^*) \leftarrow A(M^*, St_A) \\ (M^*, Authen^*) \rightarrow V(pk_P, sk_V) \end{bmatrix}$; and

(2) $d^* = 1$, $C^* \notin \mathrm{Res}$.

The advantage of A in this game is $\mathrm{Adv}_{\mathrm{NDI,A}}^{\mathrm{imp}}(k) = \Pr[\mathrm{Exp}_{\mathrm{NDI,A}}^{\mathrm{imp}}(k) = 1]$. We say that NDI is secure against impersonation attack if $\mathrm{Adv}_{\mathrm{NDI,A}}^{\mathrm{imp}}(k)$ is negligible for every polynomial time active adversary A.

3.2.2 Deniability

Consider the following game $\mathrm{Exp}_{\mathrm{NDI,D}}^{\mathrm{Den}}(k)$ between a distinguisher D and a game challenger S:

Stage1: The challenger runs the algorithm **Kg** to obtain key pairs (pk_P, sk_P), (pk_V, sk_V). Two empty sets Res and $\overline{\mathrm{Res}}$ are created. Then the distinguisher D is provided with pk_P, pk_V.

Stage2: The distinguisher D makes the following queries:

(1) Conv queries: On input a message M chosen by D, the initial state St_P of the prover algorithm is set to (sk_P, pk_V, ρ), where ρ denotes fresh random coins chosen by the challenger. Then the challenger executes $(C,d) \leftarrow \begin{bmatrix} (M, Authen) \leftarrow P(M, St_P) \\ (M, Authen) \rightarrow V(pk_P, sk_V) \end{bmatrix}$. S provides D with C (the transcript of the conversation) and sets $\mathrm{Res} \leftarrow \mathrm{Res} \bigcup \{\mathrm{C}\}$.

(2) $\overline{\mathrm{Conv}}$ queries: On input a message M chosen by D, the initial state St of the simulation algorithm Sim is set to (sk_V, pk_P, ρ), where ρ denotes fresh random coins

chosen by the challenger. Then the challenger runs the simulation algorithm by executing $M \| \overline{Authen} \leftarrow Sim(M, St)$. S sets $\overline{C} = \overline{Authen} \| M$, $\overline{\mathrm{Res}} \leftarrow \overline{\mathrm{Res}} \bigcup \{\overline{C}\}$ and provides D with $\overline{C}$.

Challenge: Once D decides that Stage2 is over, D picks a message M^* such that M^* has not been submitted as one of the Conv queries, $\overline{\mathrm{Conv}}$ queries. Then the challenger picks a random bit $b \in \{0,1\}$. If $b = 0$, S runs $[P(M^*,*) \rightarrow V(*)]$ and returns the transcript of the conversation to D. If $b = 1$, S runs $M^* \| \overline{Authen} \leftarrow Sim(M^*,*)$ and returns $\overline{C} = M^* \| \overline{Authen}$ to D.

Guess: Finally, D outputs a bit $b^/$. D wins the game if $b^/ = b$.

The advantage of D is $\mathrm{Adv}_{\mathrm{NDI,D}}^{\mathrm{Den}}(k) = |\Pr[b^/ = b] - \frac{1}{2}|$. We say that NDI is deniable if $\mathrm{Adv}_{\mathrm{NDI,D}}^{\mathrm{Den}}(k)$ is negligible for every polynomial time D.

4. Our scheme

Our scheme is based on the techniques used in [7, 8]. Let G be a multiplicative cyclic group of prime order q generated by g. G^* denotes $G \setminus \{1\}$. Then we choose a secure hash function H, $H : \{0,1\}^* \rightarrow Z_q$. The system parameters are **params=**$< G, q, g, H >$.

Kg(Key Generation)**:** Chooses a random $h \in G$ and a random $x \in Z_q$. Computes $y_1 = g^x$ and $y_2 = h^x$. The public key is $pk = (h, y_1, y_2)$ and the secret key is x. The public/secret key pairs for the prover and the verifier are (pk_P, x_P), (pk_V, x_V) respectively, where $pk_P = (h, y_{1P}, y_{2P})$, $pk_V = (h, y_{1V}, y_{2V})$.

P(**Prover**)**:** The prover picks a message M and performs as follows:

(1) Picks random $w \in Z_q, r \in Z_q^*$ and computes $c^/ = g^w (y_{1V})^r$.

(2) Picks a random $k \in Z_q^*$ and computes $A = g^k, B = h^k$.

(3) Computes $c = H(M, c^/, A, B)$. If $(c + w) = 0 \bmod q$, then goto step (2); otherwise computes $s = x_P(c + w) + k \bmod q$.

(4) Sends $C = (w, g^r, c, s, M)$ to the verifier.

V(**Verifier**)**:** Given a tuple $C = (w, g^r, c, s, M)$, the verifier performs as follows:

(1) Computes $c^{/} = g^w (g^r)^{x_V}$.

(2) Computes $A = (g^s (y_{1P})^{-(c+w)}), B = (h^s (y_{2P})^{-(c+w)})$. Returns 1 if and only if $c = H(M, c^{/}, A, B)$.

Sim(Simulation algorithm)**:** On input a message M , prover's public key pk_P and verifier's secret key x_V , the algorithm performs as follows:

(1) Picks random $\alpha, \beta \in Z_q^*$, $s \in Z_q$ and computes $c^{/} = g^{\alpha}$.

(2) Computes $A = (g^s (y_{1P})^{-\beta}), B = (h^s (y_{2P})^{-\beta})$.

(3) Computes $c = H(M, c^{/}, A, B)$.

(4) Computes $w = \beta - c \bmod q$. If $w = \alpha \bmod q$, then goto step (1); otherwise computes $r = (\alpha - w)/x_V$.

(5) Produces $\overline{C} = (w, g^r, c, s, M)$.

Lemma 1: Given the public/secret key pairs (pk_P, x_P) , (pk_V, x_V) , the following distributions are the same:

$$\delta = \left\{ (w, g^r, c, s) \left| \begin{array}{l} w \in_R Z_q, r \in_R Z_q^* \\ k \in_R Z_q^*, c \in_R Z_q \\ s = x_P (c + w) + k \bmod q \end{array} \right. \right\}$$

$$\delta^{/} = \left\{ (w, g^r, c, s) \left| \begin{array}{l} \alpha, \beta \in_R Z_q^* \\ s, c \in_R Z_q \\ w = \beta - c \bmod q, \alpha \neq w \\ r = (\alpha - w)/x_V \bmod q \end{array} \right. \right\}$$

Proof: We use the notation $x \in_R S$ to mean "the element x is chosen with uniform probability from the set S ". At first, we choose a valid tuple $(\overline{w}, R, \overline{c}, \overline{s})$, $\overline{w}, \overline{c}, \overline{s} \in Z_q, R \in G^*$ such that

(1) $\overline{c'} = g^{\overline{w}}(R)^{x_V}, \overline{A} = (g^{\overline{s}}(y_{1P})^{-(\overline{c}+\overline{w})}), \overline{B} = (h^{\overline{s}}(y_{2P})^{-(\overline{c}+\overline{w})})$.

(2) $\overline{c} = H(M, \overline{c'}, \overline{A}, \overline{B})$.

We then compute the probability of appearance of this tuple following each distribution of probabilities.

$$\Pr_{\delta}[(w, g^r, c, s) = (\overline{w}, R, \overline{c}, \overline{s})] = \Pr_{\delta}[\begin{Bmatrix} w = \overline{w}, g^r = R \\ c = \overline{c}, s = \overline{s} \end{Bmatrix}] = 1/(q^3(q-1))$$

$$\Pr_{\delta'}[(w, g^r, c, s) = (\overline{w}, R, \overline{c}, \overline{s})] = \Pr_{\delta'}[\begin{Bmatrix} s = \overline{s}, c = \overline{c} \\ w = \overline{w}, g^r = R \end{Bmatrix}] = 1/(q^3(q-1))$$

Hence, both distributions of probabilities are the same.

5. Security Analysis

Theorem 1: Let G be a multiplicative cyclic group of prime order q generated by g and assume G is a (t', ε')-DDH group such that the exponentiation in G takes time t_1. Assume there is a polynomial-time adversary A can break our non-interactive deniable authentication scheme with success probability ε in time at most t. Suppose A makes at most q_h random oracle queries, and q_c **Conv** queries. Then there is an algorithm B that solves the DDH problem in G_1 with

$$\varepsilon' \approx (\varepsilon - (q_h + 2)/q)$$

$$t' \approx t + O(q_c \cdot t_1)$$

Proof: Algorithm B is given as input a tuple (g, g^x, g^y, g^z). The system parameters are **params=**$< G, q, g, H >$, where H is a random oracle controlled by B.

B sets the public key of the prover $pk_P = (h, y_{1P}, y_{2P})$, where $h = g^y$, $y_{1P} = g^x, y_{2P} = g^z$. Next, B picks a random $x' \in Z_q$ and sets the public key of the verifier $pk_V = (h, y_{1V} = g^{x^/}, y_{2V} = h^{x^/})$. Then B works by interacting with the adversary A.

Stage1: An empty set Res is created by B. Then the adversary A is provided with pk_P, pk_V.

Stage2: B should answer A's queries as follows:

H queries: A picks $k \in Z_q^*, c^/ \in G$, and computes $A = g^k, B = h^k$. In response to a query $H(M, c^/, A, B)$ issued by A, B first checks if the output of H on this input has been previously defined. If so, B returns the previously assigned value. Otherwise, B responds with a value chosen uniformly at random from Z_q and stores $(M, H(M, c^/, A, B))$.

Conv queries: A picks a message M and issues a Conv query. In response to a Conv query， B performs as follows:

(1) Picks random $\alpha, \beta \in Z_q^*$, $s \in Z_q$ and computes $c^/ = g^\alpha$.

(2) Computes $A = (g^s (y_{1P})^{-\beta}), B = (h^s (y_{2P})^{-\beta})$.

(3) Computes $c = H(M, c^/, A, B)$.

(4) Computes $w = \beta - c \bmod q$. If $w = \alpha \bmod q$, then goto step (1); otherwise computes $r = (\alpha - w)/x_V$.

(5) Sets $C = (w, g^r, c, s, M)$. Then B provides A with C and sets $\mathrm{Res} \leftarrow \mathrm{Res} \bigcup \{C\}$.

Eventually, A picks a message M^* and outputs $C^* = M^* \| Authen^*$, $C^* \notin \mathrm{Res}$. On input C^*, if the verifier algorithm returns 1, then B outputs 1; otherwise B outputs 0.

Lemma 2: Assume $y_{2P} \in G$, but there is no $x \in Z_q$ with $g^x = y_{1P}$, $h^x = y_{2P}$. Given $w \in Z_q, r \in Z_q^*$ and a message M, a cheating prover with the public key

$pk_P = (h, y_{1P}, y_{2P})$ can pick at most one value of c over Z_q for which the honest verifier will accept (w, g^r, c, s, M).

Proof: Given w, r, assume that the cheating prover can pick two $c_1, c_2, c_1 \neq c_2 \bmod q$, such that the honest verifier will accept (w, g^r, c_i, s_i, M), $i \in \{1,2\}$. Then we have

$$c^{/} = g^w (g^r)^{x_V}\ ,\ \ A_i = (g^{s_i} (y_{1P})^{-(c_i+w)}), B_i = (h^{s_i} (y_{2P})^{-(c_i+w)}),$$

$$c_i = H(M, c^{/}, A_i, B_i), i \in \{1,2\}$$

The probability that at least one of $(M, c^{/}, A_i, B_i)$, $i \in \{1,2\}$ not having been queried by A is at most $2/q$ (i.e., A can guess B's uniform random answer). As $2/q$ is negligible, we always assume that $(M, c^{/}, A_i, B_i)$, $i \in \{1,2\}$ have been queried by A. Hence we know that $A_i = g^{k_i}, B_i = h^{k_i}$, $i \in \{1,2\}$, where both k_i are chosen by A.

Since $c_1 \neq c_2 \bmod q$, we have

$g^{((s_1-s_2)-(k_1-k_2))(c_1-c_2)^{-1} \bmod q} = y_{1P}$, $h^{((s_1-s_2)-(k_1-k_2))(c_1-c_2)^{-1} \bmod q} = y_{2P}$, which is contrary to the assumption of the lemma.

We first analyze the probability that B outputs 1 when (g, g^x, g^y, g^z) is a Diffie-Hellman tuple. According to Lemma 1, B provides a perfect simulation for A. The view of A is indistinguishable from that of A during the real interaction. Hence B outputs 1 with probability at least ε in this case.

Secondly, if (g, g^x, g^y, g^z) is a random tuple, then it is not a Diffie-Hellman tuple with probability $1 - 1/q$. In this case, for any w, r, it follows from Lemma 2 that there is at most one possible value of c for which there exists a s satisfying

$$c^{/} = g^w (g^r)^{x_V}\ ,\ \ A = (g^s (y_{1P})^{-(c+w)}), B = (h^s (y_{2P})^{-(c+w)})$$

$$c = H(M, c^{/}, A, B)$$

Thus A outputs a forgery (and hence B outputs 1) with probability $\varepsilon/q + ((q_h+1)/q)(1-1/q) \leq 1/q + ((q_h+1)/q)$ in this case (the additive factor of 1 occurs in case A did not make the relevant H-query for its forgery). Putting everything together, we have

$$|\Pr[x,y,z \leftarrow Z_q : D(g^x, g^y, g^z)=1] - \Pr[x,y \leftarrow Z_q : D(g^x, g^y, g^{xy})=1]|$$

$$\geq (\varepsilon - (q_h + 2)/q)$$

Theorem 2: Let G be a multiplicative cyclic group of prime order q generated by g. Suppose A makes at most q_h random oracle queries, q_c **Conv** queries **,** and $q_{\bar{c}}$ $\overline{\text{Conv}}$ queries. Our non-interactive deniable authentication scheme is deniable against A.

Proof: The system parameters are **params=**$<G,q,g,H>$, where H is a random oracle controlled by B.

B first picks a random $h \in G$ and random $x_P, x_V \in Z_q$. The public key of the prover is $pk_P = (h, y_{1P}, y_{2P})$, where $y_{1P} = g^{x_P}, y_{2P} = h^{x_P}$. Similarly, the public key of the verifier is $pk_V = (h, y_{1V}, y_{2V})$, where $y_{1V} = g^{x_V}, y_{2V} = h^{x_V}$. Obviously, B generates key pairs with the same distribution as in the real interaction. Then B works by interacting with the adversary A.

Stage1: Two empty sets Res and $\overline{\text{Res}}$ are created. Then the adversary A is provided with pk_P, pk_V.

Stage2: Since B knows the secret keys of the prover and the verifier, Conv queries and $\overline{\text{Conv}}$ queries issued by A can be answered correctly.

Challenge: At the end of Stage2, A picks a message M^* such that M^* has not been submitted as one of the Conv queries, $\overline{\text{Conv}}$ queries. Then B picks a random bit $b \in \{0,1\}$. If $b=0$, S runs $[P(M^*,*) \rightarrow V(*)]$ and returns C (the transcript of the conversation) to A. If $b=1$, S runs $M^* \,||\, \overline{Authen} \leftarrow Sim(M^*,*)$ and returns $\overline{C} = M^* \,||\, \overline{Authen}$ to A.

It follows from Lemma 1 that the advantage of A is negligible.

6.Conclusion

In this paper, we present a security model for non-interactive deniable authentication schemes. Then we construct a non-interactive deniable authentication scheme based on the concept of designated-verifier proofs. Finally, we show that our scheme satisfies the deniable property and is unforgeable against an active adversary. The security of our scheme is proved under the DDH assumption.

References

[1] Y. Aumann, M.O. Rabin, “Authentication enhanced security and error correcting codes” , in Proceedings of CRYPTO 1998. Springer, 1998, LNCS 1462, 299–303.

[2] M. Bellare , C. Namprempre , G. Neven, “Security Proofs for Identity-Based Identification and Signature Schemes”, in EUROCRYPT 2004, LNCS 3027,2004, 268-286.

[3] X. Deng, C. Lee, H. Lee, H. Zhu, “Deniable Authentication Protocols”, IEE Proc.Comput. Digit. Tech. ,148(2)(2001), 101–104.

[4] C. Dwork, M. Naor, A. Sahai, “Concurrent zero-knowledge”, in: Proceedings of 30th ACM STOC’98, 1998, 409–418.

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

[6] U. Feige, A. Fiat, and A. Shamir, “Zero knowledge proofs of identity”, Journal of Cryptology, 1(2):77–94, 1988.

[7] E. J. Goh, S. Jarecki, J. Katz, N. Wan, "Efficient Signature Schemes with Tight Reductions to the Diffie-Hellman Problems", Journal of Cryptology, 20(4)(2007), 493–514.

[8] M. Jacobsson, K. Sako, R, Impagliazzo, "Designated verifier proofs and their applications", in EUROCRYPT 1996, LNCS 1070,1996, 143-154.

[9] W.B. Lee, C.C. Wu, W.J. Tsaur, “A novel deniable authentication protocol using generalized ElGamal signature scheme”, Information Sciences, 177 (2007), 1376–1381.

[10] R. Lu, Z.F. Cao, “A new deniable authentication protocol from bilinear pairings”, Applied Mathematics and Computation, 168 (2005), 954–961.

[11] Z.Shao, “Efficient deniable authentication protocol based on generalized ElGamal signature scheme”, Computer Standards & Interfaces, 26 (2004), 449–454.

[12] 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, 167 (2005), 274–280.

[13] R.W. Zhu, D.S. Wong, and C.H. Lee, “Cryptanalysis of a suite of deniable authentication protocols”, IEEE Communications Letters, 10(6) (2006)504-506.