INFORMATICA, 2007, Vol. 18, No. 2, 239–252

Identity Based Strong Designated Verifier Signature Scheme

Phani Kumar KANCHARLA, Shailaja GUMMADIDALA

Secure Technology Lab., Institute for Development and Research in Banking Technology
Castle Hills, Masab Tank, Hyderabad 500057, India
e-mail: {kpkumar,gshailaja}@mtech.idrbt.ac.in

Ashutosh SAXENA [*]

Application Security and Privacy, SETLabs, Infosys Technologies Limited
Survey No.210, Lingampally, Hyderabad 500019, India
e-mail: ashutosh_saxena01@infosys.com

Received: March 2006

Abstract. We propose an Identity Based Strong Designated Verifier Signature (IBSDVS) scheme using bilinear pairings. Designated Verifier Signature finds application in e-voting, auctions and call for tenders. We prove that the scheme is secure against existential forgery under adaptively chosen message and identity attack in random oracle model. We also show that the problem of delegatability does not exist in our scheme.

Key words: designated verifier signatures, deligatability, random oracle, bilinear pairings.

1. Introduction

Designated verifier signature (DVS), first proposed at Eurocrypt'96 by Jakobsson *et al.* (1996) is special type of digital signature which provides message authentication without non-repudiation. These signatures have several applications such as E-voting, call for tenders, software licensing etc. Suppose Alice has sent a DVS to Bob. Unlike the conventional digital signatures, Bob cannot prove to a third party that Alice has created the signature. This is possible, as Bob also posses the capability of creating the signature designated to himself which is indistinguishable from Alice's signature. So, there is no reason for a third party to believe that the signature has been created by Alice. However, Bob has two reasons to accept the DVS as he knows that (i) only he and Alice are capable of creating it and (ii) he has not created it. Thus, DVS provides signer ambiguity between Alice and Bob to the rest of the world. Even though signer ambiguity exists in DVS, they do not prevent a third party to check the correctness of the signature. In a scenario, where Bob can prove to a third party that he has not yet received the signature, the third party

[*] Part of work was carried out when the author was affiliated to Secure Technology Lab., IDRBT, Hyderabad, India.

believes with high probability that Alice has created it. Strong Designated Verifier Signatures (SDVS), introduced in (Saeednia *et al.*, 2003), overcomes this problem by forcing the Designated verifier (DV) to use his secret key at the time of verification. Thus, no one else other than the DV can verify SDVS.

Lipmaa *et al.* (2005) pointed out an attack called delegatability on DVS and SDVS schemes, where Alice can delegate her signing ability, with respect to a fixed designated verifier, to a third party without disclosing her secret. In the scenario of library system, the librarian expects a SDVS designated to him, by the members to authenticate and issue the material. Suppose that a member Alice has delegated her designated verifier signing ability, with respect to librarian, to a non member Cindy, then Cindy can also borrow the material in the account of Alice. Though this is not a severe attack, it is undesirable in many such applications.

The first identity based SDVS scheme has been proposed by Susilo *et al.* (2004). Identity based signatures were first introduced by Shamir (1984). In identity based cryptosystems (IBC), user's public key is derived from the identity and there is a trusted third party called Key Generation Center(KGC) which generates the secret keys of the users. Shamir also conveyed that IBC has the advantages as it does not require the public key directories and key revocation is simplified.

Related Work

Chaum and Van (1989) proposed undeniable signatures, where the verifier needs to interact with signer for verifying the signature. Jakobsson *et al.* (1996) and Chaum (1996) introduced designated verifier signatures and private signatures independently, which can also be treated as non-interactive undeniable signatures. In (Rivest *et al.*, 2001), Rivest *et al.* introduced the ring signatures, which have signer ambiguity. By setting the ring size to two, ring signatures lead to DVS, but these schemes may not be strong DVS. Later on, several DVS and SDVS schemes (Saeednia *et al.*, 2003; Steinfeld *et al.*, 2003; Steinfeld *et al.*, 2004; Laguillaumie and Vergnaud, 2004a; Huang *et al.*, 2005) were proposed. Susilo *et al.* (2004) proposed a generic construction of strong designated verifier signatures. However, the resulting schemes are not efficient, since they require an additional identity based encryption scheme. In the same paper authors also presented a IBSDVS scheme. Unfortunately, all the schemes mentioned above suffer from the delegatability attack (Lipmaa *et al.*, 2005), including (Susilo *et al.*, 2004). Laguillaumie and Vergnaud (2004b) proposed a strong bi-designated verifier signature scheme, where the signer can designate the signature to two members.

In this paper, first we review the Susilo *et al.*'s (2004) IBSDVS scheme and show that the scheme is vulnerable to non deligatability. We then propose an Identity Based Strong Designated Verifier Signature (IBSDVS) scheme using bilinear pairings. We show that the problem of delegatability does not exist in our scheme. Security of our scheme is based on Bilinear Diffie-Hellman Problem (BDHP). We prove that our scheme is secure against existential forgery under adaptively chosen message and identity attack in random oracle model.

The rest of the paper is organized as follows. In Section 2, we briefly describe background concepts on bilinear pairings and some related mathematical problems. Review of (Susilo *et al.*, 2004) is presented in Section 3. Section 4 presents the model for our IBSDVS scheme and its security notion. In Section 5, we describe the proposed identity based strong designated verifier signature (IBSDVS) scheme. We give the security proofs of the scheme in the random oracle model in Section 6. Finally, we conclude the paper in Section 7.

2. Background Concepts

In this section, we briefly review the basic concepts on bilinear pairings and some related mathematical problems.

2.1. *Bilinear Pairings*

Let G_1 be an additive cyclic group of large prime order q, G_2 be a multiplicative cyclic group of the same order and P be a generator of G_1. A cryptographic bilinear map e is defined as e: $G_1 \times G_1 \rightarrow G_2$ with the following properties:

Bilinear: $e(aR, bS) = e(R, S)^{ab}\ \forall R, S \in G_1$ and $a, b \in Z_q^*$.

Non-degeneracy: For each $R \in G_1$ there exists $S \in G_1$ such that $e(R, S) \neq 1$

Computable: There exists an efficient algorithm to compute $e(R, S)\ \forall R, S \in G_1$.

In general implementation, G_1 is the group of points on an elliptic curve and G_2 denotes a multiplicative subgroup of a finite field. Typically, the mapping e is derived from either the Weil or the Tate pairing on an elliptic curve over a finite field. We refer to (Boneh and Franklin, 2001) for more comprehensive description on how these groups, pairings and other parameters are defined.

2.2. *Computational Problems*

We present some computational hard problems here, which will form the basis of security of our IBSDVS scheme.

Computational Diffie-Hellman Problem (CDHP). For any a, $b \in Z_q^*$, given $<P$, aP, $bP>$, compute abP.

Decisional Diffie-Hellman Problem(DDHP). For any $a, b, c \in Z_q^*$, given $<P$, aP, bP, $cP>$, decide whether $c \equiv ab$ mod q.

Gap Diffie-Hellman Problem(GDHP). A class of problems where DDHP can be solved in polynomial time but no probabilistic polynomial time algorithm exists which can solve CDHP.

Bilinear Diffie-Hellman Problem (BDHP). For any $a, b, c \in Z_q^*$, given $<P$, aP, bP, $cP>$, compute $e(P, P)^{abc}$.

For the BDH problem to be hard, G_1 and G_2 must be chosen such that there is no known algorithm for solving DHP in either of the groups.

GDH Parameter Generator. A polynomial time algorithm $\mathcal{IG}_{GDH}$ is called *GDH parameter generator* if for a given positive integer k, security parameter, it outputs a cyclic group G of prime order and a polynomial time algorithm $\mathcal{D}$ which solves DDHP in G. In our scheme we consider G as an additive group.

BDH Assumption. If $\mathcal{IG}$ is a GDH parameter generator, the advantage $Adv_{\mathcal{IG}}(\mathcal{A})$ that an algorithm $\mathcal{A}$ has in solving the BDH problem is defined to be the probability that the algorithm $\mathcal{A}$ outputs $e(P,P)^{abc}$ on inputs $G_1, G_2, e, P, aP, bP, cP$ where G_1 is output of $\mathcal{IG}$ for sufficiently large security parameter k and $a, b, c \in Z_q$. The BDH assumption is that $Adv_{\mathcal{IG}}(\mathcal{A})$ is negligible for all efficient algorithms $\mathcal{A}$.

3. Review of Susilo *et al.*'s Scheme

In this section, we first give a brief review of Susilo *et al.*'s (2004) scheme. Authors has claimed that the scheme is Strong UDVS, however We show that the scheme dose not satisfy the strongness property and also it suffers from delegatability attack.

3.1. *Review of Scheme (Susilo et al., 2004)*

The scheme consists of four algorithms namely Setup, Signature Generation, Signature Verification and Transcript Simulation.

I. Setup. In this phase the trusted third party TA generates public parameters $(G_1, G_2, e, q, P, P_{pub}, H_0, H_1)$, where G_1, G_2 are two groups of prime order q, e: $G_1 \times G_1 \to G_2$ is bilinear map, H_0: $\{0,1\}^* \to G_1$ and H_1: $\{0,1\}^* \to Z_q$ are hash functions, P is the generator of G_1 and $P_{pub} = sP$ for some randomly chosen $s \in Z_q$. For any user with identity ID, public key $Q_{ID} = H_0(ID)$ and the corresponding secret key is $S_{ID} = sQ_{ID}$.

II. Signature Generation. To sign a message m for B, A chooses two random numbers $k, t \in Z_{q^*}$, computes

$$c = e(Q_{ID_B}, P)^k; \quad r = H_1(m,\, c); \quad T = t^{-1}kP - rS_{ID_A}$$

and sends the signature (T, r, t) on message m to B.

III. Signature Verification. On receiving the signature B verifies the its validity by testing whether

$$H_1\Big(m, \big(e(T, Q_{ID_B})e(Q_{ID_A}, S_{ID_B})^r\big)^t\Big) == r.$$

IV. Transcript Simulation. Simulation of the signature is constructed as follows: B chooses random point $R \in G_1$ and a random number $a \in Z_q^*$ and generates the signature (T', r', t') on message $m$by computing

$$c' = e(R, Q_{ID_B})e(Q_{ID_A}, S_{ID_B})^a; \quad r' = H_1(m, c');$$
$$t' = (r')^{-1}a\ (mod\, p); \text{ and } T' = (t')^{-1}R.$$

3.2. *Our Attacks*

Suppose either A or B has given $e(Q_{ID_A}, S_{ID_B})$ (both A, B can compute) to an other person C, then the following two attacks are possible:

I. Delegatability. Now C can also produce signature designated to B (or A) such that it has been created by A (or B), on any message using the **Transcript Simulation** phase described above. No one, including A and B, can distinguish this signature from the signature produced by A *or* B.

II. Not Strong. Suppose A has sent a designated verifier signature constructed using the scheme to B. Any one who possess $e(Q_{ID_A}, S_{ID_B})$, in the above case C, can verify the validity of the signature, even though he does not have the secret key (S_{ID_B}) of B. Thus, the signature scheme is not strong.

4. Model for Proposed IBSDVS

In this section, we state the definition of identity based SDVS and its security notion. Entities involved in the proposed protocol are key generation center (KGC), signer (S) and designated verifier (DV). We observe that IBSDVS must satisfy the following properties:

Let $(A \rightarrow B)_{DVS}$ denote the signature generated by A and designated to B.

Correctness. A properly formed IBSDVS must be accepted by the verifying algorithm.

Unforgeability. Given entities A and B, it is infeasible, without the knowledge of the secret key of either A or B, to construct $(A \rightarrow B)_{DVS}$ or $(B \rightarrow A)_{DVS}$ a IBSDVS designated to B as it is generated by A and vice versa.

Source Hiding. Given an IBSDVS $(A \rightarrow B)_{DVS}$, it is infeasible to determine who formed it either the original signer (A) or the designated verifier (B).

Non Deligatability. Given any indirect form of secret key of the signer, it is infeasible to construct IBSDVS to any designated verifier.

4.1. *Phases of the Proposed Scheme*

The proposed identity based strong designated verifier signature (IBSDVS) scheme has five phases namely, *Setup, KeyGen, DeSign, DeVerify* and *Simulation*. These phases are described as follows:

`IBSDVS-Setup:` Given security parameter *k*, this phase generates the public parameters `params` and the master secret key `msk`.

`IBSDVS-KeyGen:` Given a user identity ID, this phase computes users public key Q_{ID} and the secret key S_{ID}.

`IBSDVS-DeSign:` On receiving the message m, the secret key of the signer and the public key of the DV, this phase computes the designated signature σ.

`IBSDVS-DeVerify:` On receiving the message-signature pair (m, σ) and the secret key of the DV, this phase checks whether σ is valid or not.

`IBSDVS-Simulation:` On receiving secret key of the DV and the public key of the signer, this phase simulates the signature designated to DV such that it satisfies verification process.

4.2. *Security Model for IBSDVS*

Let ID_{sign}, ID_{ver} are the identities of the signer and the verifier respectively. Let $\mathcal{A}$ be an adversary and $(ID_{sign} \rightarrow ID_{ver})_{SDVS}$ is the strong designated verifier signature generated by ID_{sign} and designated to ID_{ver}.

DEFINITION 1. The *adaptively chosen-message and identity attack*, having the knowledge of the public keys (identities) of the signer and verifier, $\mathcal{A}$ can ask the challenger to sign any message that he wants. He can then adapt his queries according to previous message-signature pairs. Finally, $\mathcal{A}$ has to produce a tuple $(\sigma, M, ID_{sign}, ID_{ver})$ where M, ID_{sign}, ID_{ver} are of his own choice and σ is a valid $(ID_{sign} \rightarrow ID_{ver})_{SDVS}$.

DEFINITION 2. The *adaptively chosen-message and* ***given*** *identities attack*. $\mathcal{A}$ is given two identities ID_{sign}, ID_{ver}. $\mathcal{A}$ can ask the challenger to sign any message that he wants. He can then adapt his queries according to previous message-signature pairs. Finally, $\mathcal{A}$ has to produce a tuple $(\sigma, M, ID_{sign}, ID_{ver})$ where σ is a valid $(ID_{sign} \rightarrow ID_{ver})_{SDVS}$ and M is of his won choice.

Note. In both the above two attacks, adversary should not ask the sign query for this message M and private key queries for ID_{sign}, ID_{ver}.

For identity based signatures, the known security notion is to be secure against *existential forgery under adaptively chosen message and identity attack* (Cha and Cheon, 2003). We present a slightly modified version of (Cha and Cheon, 2003) and use to prove the security of the IBSDVS scheme in random oracle model. In this model, the adversary $\mathcal{A}$ wins if it produces a message, pair of identities (signer and verifier) and a valid IBSDVS. The adversary is allowed to query the hash oracles, secret key generation oracle and signature oracle. The adversary can adaptively choose messages and identities, to query the oracles, except for the following two queries: (i) sign query for the message that it finally produces. (ii) secret key generation (IBSDVS-KeyGen) query for either one of the identities that it finally produces.

We can visualize the security model by the following game:

- Challenger $\mathcal{C}$ runs IBSDVS-Setup of the scheme and sends the resulting public parameters to the adversary $\mathcal{A}$.
- Adversary $\mathcal{A}$ issues the following queries adaptively to the oracles:
  - hash function query: $\mathcal{C}$ computes the hash value of the requested input and sends it to $\mathcal{A}$;
  - IBSDVS-KeyGen query: on receiving the ID, $\mathcal{C}$ computes the corresponding secret key and sends it to $\mathcal{A}$;

- IBSDVS-DeSign query: on receiving the message, senders and receivers identities, $\mathcal{C}$ computes the designated signature and sends it to $\mathcal{A}$.

- Finally adversary $\mathcal{A}$ outputs $(ID_{sign},\, ID_{ver},\, M,\, \sigma)$, where ID_{sign} is the signer identity, ID_{ver} is the designated verifier identity, σ is the signature on message M such that ID_{sign} and ID_{ver} have not been queried to IBSDVS-KeyGen and $(M,\, ID_{sign},\, ID_{ver})$ have not been queried to IBSDVS-DeSign.
- $\mathcal{C}$ verifies the validity of the signature σ. If it is valid, $\mathcal{A}$ wins the game.

DEFINITION 3. We say that IBSDVS is $(\epsilon,\, t)$-secure if there is no adversary $\mathcal{A}$, capable of existential forging IBSDVS under adaptively chosen message and identities attack with advantage $>= \epsilon$ and running time $<= t$.

5. Identity Based Strong Designated Verifier Signature Scheme

In this section, we propose an ID-based SDVS scheme that can be built upon a Gap Diffie-Hellman group described in the Section 2. The scheme consists of five phases: IBSDVS-Setup, IBSDVS-KeyGen, IBSDVS-DeSign, IBSDVS-DeVerify and IBSDVS-Simulation. The first two phases are carried out at KGC.

Let G_1 be a GDH group of order q, a large prime number and G_2 be a multiplicative sub group of a finite field F of same order.

`IBSDVS-Setup`: In this phase, KGC chooses a generator $P \in G_1$, a random number $s \in Z_q^*$ and computes $P_{Pub} = sP$. KGC also chooses two cryptographic hash functions H_1: $\{0,1\}^* \rightarrow G_1$ and H_2: $\{0,1\}^* \times G_2 \rightarrow G_1$ and a bilinear pairing e: $G_1 \times G_1 \rightarrow G_2$. The system parameters $(G_1,\, G_2,\, P,\, P_{pub},\, H_1,\, H_2,\, e)$ are published and s is kept as the master secret.

`IBSDVS-KeyGen`: Given an identity ID, this phase generates secret key $S_{ID} = sH_1(ID)$. We remark that $Q_{ID} = H_1(ID)$ is the public key of the user ID .

`IBSDVS-DeSign`: Given the secret key S_{ID_A} of the signer A, the public key Q_{ID_B} of the designated verifier B and message M, this phase computes the signature σ as follows: chooses three random numbers $r_1, r_2, r_3 \in Z_q^*$, and computes

$$\begin{aligned}
&U_1 = r_1 Q_{ID_B},\\
&U_2 = r_2 Q_{ID_A},\\
&U_3 = r_1 r_3 Q_{ID_B},\\
&V = r_3 H + r_1^{-1} S_{ID_A}, \quad \text{where } \; H = H_2\bigl(M, e(r_2 Q_{ID_B}, S_{ID_A})\bigr).
\end{aligned}$$

Signer A sends the signature $\sigma = (U_1,\, U_2,\, U_3,\, V)$ along with the message M to the designated verifier B.

`IBSDVS-DeVerify`: On receiving (M, σ), the designated verifier computes $H = H_2(M, e(U_2, S_{IDB}))$ and accepts the signature as valid if the following equality holds: $e(U_1, V) == e(U_3, H)\, e(S_{ID_B}, Q_{ID_A})$.

The following equations give the correctness of the verification:

$$\begin{aligned} e(U_1, V) &= e(r_1 Q_{ID_B}, r_3 H + r_1^{-1} S_{ID_A}) \\ &= e(r_1 Q_{ID_B}, r_3 H) e(r_1 Q_{ID_B}, r_1^{-1} S_{ID_A}) \\ &= e(r_3 r_1 Q_{ID_B}, H) e(Q_{ID_B}, s Q_{ID_A}) \\ &= e(U_3, H) e(s Q_{ID_B}, Q_{ID_A}) \\ &= e(U_3, H) e(S_{ID_B}, Q_{ID_A}). \end{aligned}$$

`IBSDVS-Simulation`: The designated verifier (B) cannot prove to a third party that the signature σ has been produced by the signer A, as B can also create an indistinguishable signature σ' on the same message M. The third party can be conveyed that the user B can produce the signature σ' in the following way:

The user B can choose three random numbers $r_1', r_2', r_3' \in Z_q^*$ and computes

$$\begin{aligned} &U_1' = r_1' Q_{ID_A}, \\ &U_2' = r_2' Q_{ID_B}, \\ &U_3' = r_1' r_3' Q_{ID_A}, \\ &V' = r_3' H' + {r_1'}^{-1} S_{ID_B}, \quad \text{where } H' = H_2\big(M, e(U_2', S_{ID_B})\big). \end{aligned}$$

Clearly the signature $\sigma' = (U_1', U_2', U_3', V')$ satisfies the verification described earlier. With this we complete the description of our scheme.

It may be noted, that one can not delegate the signing capability of IBSDVS, as the signing phase requires the secret key of the signer explicitly.

6. Security Analysis

In this section, we prove that our signature scheme is secure and that its security is based on hardness of BDHP problem. We can prove this by using contrapositive method i.e. if there is an adversary algorithm A_0, which is capable of existential forging IBSDVS under adaptively chosen message and identity attack, then we show that we can construct an adversary algorithm A_2 which can solve BDHP in polynomial time and with non-negligible probability.

The following lemma reduces the *adaptively chosen message and identity attack* to *adaptively chosen message and given identities attack*. Note that $\binom{x}{y}$ denote "x choose y" from here onwards.

Lemma 1. *If there is an algorithm $\mathcal{A}_0$ for an adaptively chosen message and ID attack to our scheme with running time t_0 and advantage ϵ_0, then there is an algorithm $\mathcal{A}_1$ for an adaptively chosen message and given IDs attack which has running time $t_1 <= t_0$ and advantage $\epsilon_1 >= \epsilon_0\big(1 - \frac{1}{\binom{q}{2}}\big)\frac{1}{\binom{q_{H_1}}{2}}$, where q_{H_1} is the maximum number of queries to H_1 asked by $\mathcal{A}_0$. In addition, the number of queries to hash functions, `IBSDVS-KeyGen` and `IBSDVS-DeSign` asked by $\mathcal{A}_1$ are the same as those of $\mathcal{A}_0$.*

Proof. $\mathcal{A}_1$ is given two identities ID_{sign} and ID_{ver}. As explained in the security model presented in Section 2, adversary $\mathcal{A}_0$ is capable of querying the random oracles, *KeyGen* oracle and *DeSign* oracle. Without loss of generality we can assume that these queries are distinct. Algorithm $\mathcal{A}_1$ simulates the Challenger as follows:

Step 1. $\mathcal{A}_1$ chooses two random numbers $m, n \in \{1, 2, ...q_{H_1}\}$ randomly. ID_i denotes the i^{th} query made by $\mathcal{A}_0$ to H_1. Let

$$ID'_i = \begin{cases} ID_{sign} & \text{if } i = m, \\ ID_{ver} & \text{if } i = n, \\ ID_i & \text{otherwise.} \end{cases}$$

Define the functions $H'_1(ID_i) = H_1(ID'_i)$, $KeyGen'(ID_i) = KeyGen(ID'_i)$ and $DeSign'(ID_i, ID_j, M) = DeSign(ID'_i, ID'_j, M)$. Algorithm $\mathcal{A}_1$ uses these functions to answer the $\mathcal{A}_0$ queries. Hash oracle H_2 is common for both the $\mathcal{A}_1$ as well as $\mathcal{A}_0$.

Step 2. By running $\mathcal{A}_0$ with the given system parameters, we will get an output (ID_1, ID_2, M, σ) at the end.

Step 3. If $ID_1 == ID_m$, $ID_2 == ID_n$ and the signature is valid, $\mathcal{A}_1$ outputs $(ID_{sign}, ID_{ver}, M, \sigma)$; otherwise it fails.

Since the distributions produced by H'_1, $KeyGen'$ and $DeSign'$ are indistinguishable from $H_1, KeyGen$ and $DeSign$, adversary $\mathcal{A}_0$ does not gain any information about $\mathcal{A}_1$.

We have that

$$Pr[(ID_1, ID_2, M, \sigma)] >= \epsilon_0.$$

Since H_1 is a random oracle, the probability that the output (ID_1, ID_2, M, σ) of $\mathcal{A}_0$ is valid, without querying $H'_1(ID_1), H'_1(ID_2)$ is negligible and is less than the value $\frac{1}{q_{C_2}}$. Let

$$Pr_1 = Pr\big[ID_1 = ID_i \text{ and } ID_2 = ID_j, \text{ for some } i,\ j/(ID_1, ID_2, M, \sigma) \text{ is valid}\big].$$

So, $Pr_1 >= 1 - \frac{1}{\binom{q}{2}}$.

Put

$$Pr_2 = Pr\big[ID_1 = ID_m \text{ and } ID_2 = ID_n / ID_1 = ID_i \text{ and } ID_2 = ID_j \text{ for some } i,\ j\big].$$

Since m, n are randomly chosen, $Pr_2 >= \frac{1}{\binom{q_{H_1}}{2}}$.

Put

$$\epsilon_1 = Pr\big[ID_1 = ID_{sign}, ID_2 = ID_{ver} \text{ and } (ID_1, ID_2, M, \sigma) \text{ is valid}\big].$$

Combining these values we get the total probability

$$\epsilon_1 >= \epsilon_0\big(1 - \frac{1}{\binom{q}{2}}\big)\frac{1}{\binom{q_{H_1}}{2}}.$$

Thus we have proven that if the algorithm $\mathcal{A}_0$ exists, then we can construct an algorithm $\mathcal{A}_1$ which can forge the IBSDVS with the given identities.

The following lemma shows that BDHP can be solved with non negligible advantage and in finite time, provided that there exists an adversary $\mathcal{A}_1$ as described in Lemma 1.

Lemma 2. *If there is an adversary algorithm $\mathcal{A}_1$ capable of existentially forging IBSDVS under adaptively chosen message and given IDs attack with the running time t_1 and advantage ϵ_1, which queries H_1, H_2, `DeSign` and `KeyGen` at most q_{H_1}, q_{H_2}, q_S and q_K times respectively, then there is an algorithm $\mathcal{A}_2$ which solves the BDHP with the running time $t_2 <= t_1$ and advantage $\epsilon_2 >= \epsilon_1(1 - \frac{1}{q})$. q is the size of the output of H_2 hash function.*

Proof. We assume that all the queries made by $\mathcal{A}_1$ are distinct and $\mathcal{A}_1$ queries $H_1(ID)$ before ID is used as an input of any query to H_2, `KeyGen` and `DeSign`. Finally, it outputs an IBSDVS for the identities $ID_{sign}\ and\ ID_{ver}$. Here, we construct an algorithm $\mathcal{A}_2$ which solves the BDHP i.e. given $P, aP, bP, cP \in G_1$, $\mathcal{A}_2$ has to compute $e(P,P)^{abc}$.

Step 1. Fix identities ID_{sign} and ID_{ver}. Put $P_{pub} = aP$ and choose randomly $x_i \in Z_q$ for $i = 1, 2, ..., q_{H_1}$, $y_i \in Z_q$ for $i = 1, 2, ..., q_S$, and $h_i \in Z_q$ for $i = 1, 2, ..., q_{H_2}$. Denote by $ID_i,\ ID_{i_k}$ and $(ID_{i_j}, ID_{i'_j}, M_j)$ the inputs of the i^{th} H_1 query, the k^{th} `KeyGen` query and j^{th} `DeSign` query asked by $\mathcal{A}_1$ respectively. Define

$$\mathcal{H}_1''(ID_i) = \begin{cases} bP & \text{if } ID_i = ID_{sign}, \\ cP & \text{if } ID_i = ID_{ver}, \\ x_iP & \text{otherwise.} \end{cases}$$

$$\texttt{KeyGen}''(ID_{i_k}) = x_{i_k}(aP),$$
$$\texttt{DeSign}''(ID_{i_j}, ID_{i'_j}, M_j) = (ID_{i_j}, ID_{i'_j}, M_j, U_{1_j}, U_{2_j}, U_{3_j}, H_j, V_j),$$

where $U_{1_j} = r_{1_j}x_{i'_j}P$, $U_{2_j} = r_{2_j}x_{i_j}P$, $U_{3_j} = r_{1_j}r_{3_j}x_{i'_j}P$, $V = r_{3_j}H_j + r_{1_j}^{-1}x_{i_j}aP$, where $H_j = \mathcal{H}_2''(M_j, e(r_{2_j}x_{i'_j}P,\ x_{i_j}aP))$. Here, $\mathcal{H}_2''$ is identical to H_2 except for the queries $(?, e(Q_{ID_{ver}}, S_{ID_{sign}})^{r_2})$, and for these queries $\mathcal{H}_2''$ gives h_iP. $\mathcal{A}_2$ responds to $\mathcal{A}_1$'s queries to H_1, H_2, `DeSign` and `KeyGen` by evaluating $\mathcal{H}_1''$, $\mathcal{H}_2''$, `DeSign`$''$ and `KeyGen`$''$ respectively.

It can be observed from the above equations, that the key pair of the signer with identity ID_{sign} is (bP, abP) and the verifier with identity ID_{ver} is (cP, acP).

Step 2. Finally, $\mathcal{A}_1$ produces a valid signature $\sigma = (U_1, U_2, U_3, V)$ on message M with signer identity ID_{sign} and designated verifier identity ID_{ver} with advantage ϵ_1.

Since H_2 is a random oracle, the probability that the output $(ID_{sign}, ID_{ver}, M, \sigma)$ of $\mathcal{A}_1$ is valid, without querying $\mathcal{H}_2''(M, e(Q_{ID_{ver}}, S_{ID_{sign}})^{r_2})$ is negligible and is less than the value $\frac{1}{q}$. Hence, we have

$$Pr\Big[\big(M, e(Q_{ID_{ver}}, S_{ID_{sign}})^{r_2}\big) \text{ queried to } \mathcal{H}_2''/(ID_{sign}, ID_{ver}, M, \sigma) \text{ valid}\Big] >= 1 - \frac{1}{q}.$$

Step 3. In this step $\mathcal{A}_2$ solves BDHP. Since the signature σ is valid, it satisfies the verification process as given by

$$e(U_1, V) \,=\, e(U_3, H)\, e(S_{ID_{ver}}, Q_{ID_{sign}}).$$

From this $\mathcal{A}_2$ can arrive at

$$\begin{aligned} & e(U_1, V) = e(U_3, H)\, e(acP, bP) \\ & \quad \Rightarrow e(acP, bP) = e(U_1, V)\, e(U_3, H)^{-1} \\ & \quad \Rightarrow e(P, P)^{abc} = e(U_1, V)\, e(U_3, H)^{-1}. \end{aligned}$$

$\mathcal{A}_2$ can compute the right hand side of the equation, since H is queried by $\mathcal{A}_1$ with high probability and $U_1,\ U_3,\ V$ are public. Thus, $\mathcal{A}_2$ has solved BDHP.

Step 4. Clearly the advantage (ϵ_2) for solving BDHP is the product of the advantage of $\mathcal{A}_1$ and the probability that $\mathcal{A}_1$ asks the query H to $\mathcal{A}_2$, and hence the advantage $\epsilon_2 >= \epsilon_1(1 - \frac{1}{q})$.

Theorem 1. *If there is an algorithm $\mathcal{A}_0$ for an adaptively chosen message and identities attack to our scheme with running time t_0 and advantage ϵ_0 which queries H_1,H_2, `KeyGen` and `DeSign` at most $q_{H_1},\ q_{H_2},\ q_K$ and q_S time respectively, then BDHP can be solved with the probability $\epsilon_2 >= \epsilon_0(1 - \frac{1}{q_{C_2}})(1 - \frac{1}{q})$.*

Proof. Proof of this theorem directly follows from the above two lemmas.

7. Conclusions

Strong designated verifier signatures are applicable in e-voting, auctions and call for tenders, where the designated verifier only can verify and convince himself the authenticity of the signature. We reviewed the Susilo *et al.*'s (2004) IBSDVS scheme and shown that the scheme is vulnerable to deligatability. We proposed an identity based strong designated verifier signature scheme whose security is based on the hardness of the BDHP. The deligatability attack (Lipmaa *et al.*, 2005) does not exist on our scheme, since the signer has to use his secret key explicitly while signing. The security of the proposed scheme has been proven in the random oracle model against existential forgery under adaptively chosen message and identity attack.

References

Boneh, D., and M. Franklin (2001). Identity based encryption from the Weil pairing. *SIAM Journal of Computing*, **32**(3), 586–615.

Cha, J., and J.H. Cheon (2003). An identity-based signature from Gap Diffie-Hellman groups. In *PKC'03*, *LNCS*, vol. 2567. Springer-Verlag. pp. 18–30.

Chaum, D., and H. Van (1989). Undeniable signatures. In *Crypto'1989*, *LNCS*, vol. 435. Springer-Verlag. pp. 212–216.

Chaum, D. (1996). *Private Signature and Proof Systems*. United States Patent 5,493,614.

Huang, X., Y. Mu, W. Susilo and F. Zhan (2005). Short designated verifier proxy signature from pairings. In *Security in Ubiquitous Computing Systems – SecUbiq 2005*, *LNCS*, vol. 3823. pp. 835–844.

Jakobsson, M., K. Sako and R. Impagliazzo (1996). Designated verifier proofs and their applications. In *Eurocrypt'1996*, *LNCS*, vol. 1070. Springer-Verlag. pp. 142–154.

Laguillaumie, F., and D. Vergnaud (2004a). Designated verifier signatures: anonymity and efficient construction from any bilinear map. In *Security in Communication Networks*, *SCN 2004*, *LNCS*, vol. 3352. pp. 105–119.

Laguillaumie, F., and D. Vergnaud (2004b). Multi-designated verifiers signatures. In *Information and Communications Security – ICICS 2004*, *LNCS*, vol. 3269. pp. 495–507.

Lipmaa, H., G. Wang and F. Bao (2005). Designated verifier signature schemes: attacks, new security notions and a new construction. In *32nd International Colloquium on Automata, Languages and Programming*, *ICALP 2005*, *LNCS*, vol. 3580. pp. 459–471.

Rivest, R., A. Shamir and Y. Tauman (2001). How to leak a secret. In *ASIACRYPT 2001*, *LNCS*, vol. 2248. pp. 552–565.

Saeednia, S., S. Kremer and O. Markowitch (2003). An efficient strong designated verifier signature scheme. In *Information Security and Cryptology – ICISC 2003*, *Lecture Notes in Computer Science*, vol. 2971. Springer-Verlag. pp. 40–54.

Shamir, A. (1985). ID-based cryptosystems and signature schemes. In *Crypto 84*, *LNCS*, vol. 196. Springer. pp. 47–53.

Steinfeld, R., L. Bull, H. Wang and J. Pieprzyk (2003). Universal designated-verifier signatures. In *ASIACRYPT 2003*, *LNCS*, vol. 2894. pp. 523–542.

Steinfeld, R., H. Wang and J. Pieprzyk (2004). Efficient extension of standard Schnorr/RSA signatures into universal designated-verifier signatures. In *PKC 2004*, *LNCS*, vol. 2947. pp. 86–100.

Susilo, W., F. Zhang and Y. Mu (2004). Identity-based strong designated verifier signature schemes. In *Proceedings of the Information Security and Privacy*, *ACISP 2004*, *LNCS*, vol. 3108. Springer. pp. 313–324.

P.K. Kancharla completed his master's degree in IT with specialization in banking technology and information security from University of Hyderabad. Currently he is working as software engineer and his research interests include cryptography, systems and network security.

S. Gummadidala completed her master's degree in IT with specialization in banking technology and information security from University of Hyderabad. Currently she is working as software engineer and her research interests include cryptography, network security and operating systems.

A. Saxena completed his PhD computer science in 1996 and post doctoral work in 2002 from ISRC, QUT, Brisbane. He has authored more than 70 research articles and also a book on "PKI: Concepts, Design and Deployment". Currently heading Application Security and Privacy Group in SETLabs of Infosys Technologies Limited. For eight years he worked as a professor with Institute for Development and Research in Banking Technology, Hyderabad, India. He is member IEEE and life member of Cryptology Research Society of India and Computer Society of India. He served as program committee member in many International Conferences. He is also on Board of Editors for International Journal on Information and Management, Elsevier Publication. His research interests include authentication technologies, smart cards, key management and privacy.

Identiškumu grindžiama stipraus priskyrimo parašo schema

Phani Kumar KANCHARLA, Shailaja GUMMADIDALA, Ashutosh SAXENA

Straipsnyje siūloma identiškumu grindžiama stipraus priskyrimo parašo schema, naudojanti bitiesinius poravimus. Priskyrimo parašo schema taikoma elektroniniame balsavime, aukcionuose ir kviečiant į tenderius. Įrodoma, kad ši schema yra saugi prieš egzistencinę klastotę, naudojančią adaptyviai parinktą pranešimą, ir prieš identišką ataką atsitiktiniuose oraklo modeliuose. Taip pat parodoma, kad šioje schemoje neegzistuoja įgaliojamumo problema.