Identity Based Strong Designated Verifier Signature Scheme

K. Phani Kumar, G. Shailaja, Ashutosh Saxena

Secure Technology Lab.,
Institute for Development and Research in Banking Technology
Castle Hills, Masab Tank, Hyderabad 500057, INDIA.
kpkumar@mtech.idrbt.ac.in, gshailaja@mtech.idrbt.ac.in, asaxena@idrbt.ac.in

Abstract. Identity based cryptosystem simplifies the key management and revocation problem. Here we propose an Identity Based Strong Designated Verifier Signature (IBSDVS) scheme using bilinear pairings. The Designated Verifier Signature scheme described in [10] is identity based but it suffers from the deligatability as pointed out in [4]. We analyse the security of the scheme and show that the problem of delegatability does not exist in our scheme.

1 Introduction

Designated verifier signature (DVS), first proposed at Eurocrypt'96 by Jakobsson et al [1], 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 and software licensing. 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 accomplished by the Bob's 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, it does not prevent a third party from checking 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 believes with high probability that Alice has created it. Strong Designated Verifier Signature (SDVS), introduced in [3], overcomes this problem by forcing the Designated verifier (DV) to use his private key at the time of verification. Thus, no one else except the DV can verify SDVS.

In [4], Lipmaa et al. 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 with out disclosing her secret. In the scenario of library system, the librarian expects a SDVS designated to him from a member, to 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.

Recently Huang et al[10] proposed an identity based DVS. Identity based signatures were first introduced by Shamir [13] in 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 private keys of the users. IBC has several advantages such as it does not require the public key directories and key revocation is simplified.

Related Work: In 1989, Chaum et al. [2] proposed undeniable signatures, where the verifier needs to interact with signer for verifying the signature. In 1996, Jakobsson et al. [1] and Chaum [5] introduced designated verifier signatures and private signatures independently, which can also be treated as non-interactive undeniable signatures. In [6], 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 [3] [7] [8] [9] [10] were proposed. Unfortunately, all the schemes mentioned above suffer from the delegatability attack [4]. In 2004, Laguillaumie et al [11] proposed a strong bi-designated verifier signature scheme, where the signer can designate the signature to two members.

In this paper we propose an Identity Based Strong Designated Verifier Signature (IBSDVS) scheme using bilinear pairings. Security of our scheme is based on Bilinear Diffie-Hellman Problem (BDHP). We show that the problem of delegatability does not exist in our scheme.

Rest of the paper is organized as follows: in section 2, we describe background concepts of bilinear pairings and some related mathematical problems. Section 3 presents the model for our IBSDVS scheme. In section 4, we describe the proposed identity based strong designated verifier signature (IBSDVS) scheme. We give the security analysis of the scheme in section 5. Finally, we conclude the paper in section 6.

2 Background Concepts

In this section, we briefly review the basic concepts of 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 $O \neq R \in G_1$, there exists $S \in G_1$ such that $e(R, S) \neq$

1, where O is the identity element in G_1 and 1 is the identity element in G_2.
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 [12] for more comprehensive description on how these groups, pairing 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.

Discrete Logarithm Problem (DLP): Given two elements $R, S \in G_1$, find an integer $a \in Z_q^*$, such that $S = aR$ whenever such an integer exists.

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 $c \equiv ab \bmod q$.

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

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.

3 Model for Proposed IBSDVS

In this section, we state the definition of identity based SDVS. 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:

1. **Correctness:** A properly formed IBSDVS must be accepted by the verifying algorithm.
2. **Unforgeability:** It is computationally infeasible to construct a valid IBSDVS without the knowledge of the secret key of either the signer or the designated verifier.
3. **Source Hiding:** Given a message M and an IBSDVS on M, it is infeasible to determine who from the original signer or the designated verifier performed the signature, even if one knows all the secret keys.

4. **Non-Deligatability:** Given any derivative of the secret key of the signer, it is infeasible to construct a valid IBSDVS.

3.1 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 algorithm outputs the public parameters.
- **IBSDVS-KeyGen:** Given a user identity and the public parameters, this algorithm computes private key of the user.
- **IBSDVS-DeSign:** On receiving the message m, the private key of the signer and the public key of the DV, this algorithm computes the designated signature σ on message m.
- **IBSDVS-DeVerify:** On receiving the message-signature pair (m, σ) and the private key of the DV, this algorithm tests whether σ is valid or not.
- **IBSDVS-Simulation:** On receiving private key of the DV and the public key of the signer, this algorithm simulates the signature designated to DV such that it satisfies verification process.

4 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 a large prime number q and G_2 be a multiplicative subgroup of a finite field F of same order.

1. **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.
2. **IBSDVS-KeyGen:** Given an identity ID, this phase generates $S_{ID} = sH_1(ID)$ and sends it to the user ID. We remark that $Q_{ID} = H_1(ID)$ is the public key of the user ID.
3. **IBSDVS-DeSign:** Given a private key S_{ID_A} of the signer A, the public keys Q_{ID_A}, Q_{ID_B} of the signer A and the designated verifier B respectively and message M, this phase computes the signature σ as follows:
   By choosing random numbers $r_1, r_2 \in Z_q^*$, it computes and outputs

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

Signer sends $(M,\ \sigma)$ to the designated verifier, where $\sigma = (U_1,\ U_2,\ U_3,\ V)$.

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

5. **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 user B can produce the signature σ' in the following way: The user B chooses two random numbers $r'_1, r'_2 \in Z_q^*$ and computes

$$
\begin{aligned}
U'_1 &= r'_1 Q_{ID_A} \\
U'_2 &= r'_1 Q_{ID_B} \\
U'_3 &= r'_1 r'_2 Q_{ID_A} \\
H' &= H_2(M, e(U'_1, S_{ID_B})) \\
V' &= r'_2 H' + {r'_1}^{-1} S_{ID_B}.
\end{aligned}
$$

Clearly the signature $\sigma' = (U'_1, U'_2, U'_3, V')$ satisfies the verification.
This completes the description of our scheme.

5 security analysis

In this section, we analyze the security of the proposed scheme.

5.1 Correctness

The following equations gives the correctness of the verification:

$$
\begin{aligned}
e(U_1, V) &= e(r_1 Q_{ID_B}, r_2 H + r_1^{-1} S_{ID_A}) \\
&= e(r_1 Q_{ID_B}, r_2 H) e(r_1 Q_{ID_B}, r_1^{-1} S_{ID_A}) \\
&= e(r_2 r_1 Q_{ID_B}, H) e(Q_{ID_B}, s Q_{ID_A}); \\
&= e(U_3, H) e(s Q_{ID_B}, Q_{ID_A}) \\
&= e(U_3, H) e(S_{ID_B}, Q_{ID_A})
\end{aligned}
$$

5.2 Strongness

The Designated verifier has to use his secret key S_{ID_B} during the verification process while computing the hash H. Therefore, no one else except the designated verifier can perform the signature verification. Thus, our scheme is a strong designated verifier scheme.

5.3 Unforgeability

It is not possible to construct the terms H and V without the knowledge of either the signer secret key S_{ID_A} or the verifier secret key S_{ID_B}. Thus, the signature is unforgeable.

5.4 Non-transferability privacy

The designated verifier cannot prove to a third party that the designated signature on message M was produced by the signer, using the designated signature σ and message M, since he is also capable of producing an indistinguishable signature on the same message.

5.5 Source hiding

Even if the signer secret key S_{ID_A} and the verifier secret key S_{ID_B} are known to a third party, he cannot identify whether S_{ID_A} or S_{ID_B} has been used in the construction of the term V, as he does not have the knowledge of the random numbers used during the signing process.

5.6 Non-Delegatability

The construction of the term V requires the signer secret key S_{ID_A}. So, it is not possible for the signer to delegate his signing capability to any third party without disclosing his secret. Thus, the problem of delegatability does not exist in our scheme.

6 Conclusion

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 proposed an identity based strong designated verifier signature scheme whose security is based on the hardness of the BDHP.The deligatability attack [4] does not exist on our scheme as presented in Section XXX since the signer has to use his private key explicitly while signing. Further work is on the way to have a formal security analysis in random oracle model.

References

1. Jakobsson, M., Sako, K., Impagliazzo, R.(1996). Designated Verifier Proofs and their Applications. *Eurocrypt 1996* Lecture Notes in Computer Science, Vol. 1070, Springer-Verlag, 142154.
2. Chaum, D., Van, H.(1990). Undeniable Signatures. in *Crypto 1989*, Lecture Notes in Computer Science, vol. 435,Springer-Verlag, 212 216.
3. Saeednia, S., Kremer, S., Markowitch, O.(2003). An Efficient Strong Designated Verifier Signature Scheme. In *Information Security and Cryptology - ICISC 2003*, Lecture Notes in Computer Science, vol. 2971, Springer-Verlag, 4054.
4. Lipmaa, H., Wang, G., Bao, f.(2005) Designated Verifier Signature Schemes: Attacks, New Security Notions and A New Construction, in *32nd International Colloquium on Automata, Languages and Programming, ICALP 2005*, Lecture Notes in Computer Science, vol. 3580, 459-471.
5. Chaum, D.(1996).Private signature and proof systems. United States Patent 5,493,614.
6. Rivest, R., Shamir, A., Tauman, Y.(2001) How to leak a secret. In *ASIACRYPT 2001*, Lecture Notes in Computer Science, vol. 2248, 552-565.
7. Steinfeld, R., Bull, L., Wang, H., Pieprzyk, J.(2003). Universal Designated-Verifier Signatures. In *ASIACRYPT 2003*, Lecture Notes in Computer Science, vol. 2894, 523542.
8. Steinfeld, R.,Wang, H., Pieprzyk, J.(2004). Efficient Extension of Standard Schnorr/RSA Signatures into Universal Designated-Verifier Signatures. In *PKC 2004*, Lecture Notes in Computer Science, vol. 2947 , 86100.
9. Laguillaumie, F., Vergnaud, D.(2004). Designated Verifier Signatures: Anonymity and Efficient Construction from Any Bilinear Map. In *Security in Communication Networks, 4th International Conference, SCN 2004*, Lecture Notes in Computer Science, vol 3352, 105119.
10. Huang, X., Mu, Y., Susilo, W., Zhan, F.(2005). Short Designated Verifier Proxy Signature from Pairings. in*The First International Workshop on Security in Ubiquitous Computing Systems - SecUbiq 2005*, Lecture Notes in Computer Science, vol. 3823, 835 844.
11. Laguillaumie, F., Vergnaud, D.(2004) Multi-Designated Verifiers Signatures. in *Information and Communications Security - ICICS 2004*, Lecture Notes in Computer Science, vol. 3269, 495 507.
12. Boneh, D., Franklin, M.(2001). Identity Based Encryption from the Weil Pairing. *SIAM Journal of Computing*, **32**(3), (2003) 586-615.
13. Shamir, A. (1985). ID-based Cryptosystems and Signature Schemes. In Proceedings of *Crypto 84*, Lecture Notes in Computer Science, Vol. 196, Springer, 47-53.
14. Cha, J., Cheon, J.H. (2003). An Identity-Based Signature from Gap Diffie-Hellman Groups. *In PKC'03*, Lecture Notes in Computer Science, Vol.2567, Springer-Verlag. 18-30