Anonymous yet Traceable Strong Designated Verifier Signature

Veronika Kuchta[1], Rajeev Anand Sahu[2(✉)], Vishal Saraswat[3], Gaurav Sharma[2], Neetu Sharma[4], and Olivier Markowitch[2]

[1] Monash University, Clayton, Australia
[2] Université Libre de Bruxelles, Brussels, Belgium
rajeev.sahu@ulb.ac.be
[3] Indian Institute of Technology Jammu, Jammu, India
[4] PRS University, Raipur, Raipur, India

Abstract. In many privacy-preserving protocols, protection of the user's identity, called anonymity, is a desirable feature. Another issue is that, if a signed document is leaked then anyone can be convinced of the authenticated data, which is strictly not allowed for sensitive data, instead the authentication only by a designated receiver is recommended. There are many scenarios in real life, for example e-auction, where both the functionalities– anonymity and designated verification are required simultaneously. For such an objective, in this paper we introduce a compact scheme of identity-based strong designated verifier group signature (ID-SDVGS) by combining the good features of strong designated verifier signature and group signature in ID-based setting. This scheme provides anonymity to the signer of a designated verifier signature with the feature of the revocation of signer's identity in case of misuse or dispute. Moreover, our scheme fulfils all the security properties of the individual components. We have obtained an ID-based instantiation of the generic group signature given by Bellare et al. in Eurocrypt 2003, and have proposed our scheme on that framework. To the best of our knowledge, this is the first construction of ID-SDVGS.

1 Introduction

There are numerous cryptographic protocols made up by combination of different primitives to achieve desired features for required applications. But it is important that a scheme designed for an optimal solution should not render additional security breaches. This work attempts to construct a cryptographic solution for a situation where authentication is desired only by the authorized receiver while protecting identity of the sender. In cryptography, anonymous signatures offer the anonymity of signer, while the designated verifier signatures (DVS) offer authorised verification with the property of *non-transferability* of the verification. Achieving signer's anonymity in a DVS is crucial in many application. Unfortunately, this issue has not been widely addressed and this functionality (anonymity) has not been achieved yet for such a signature (DVS) on identity

L. Chen et al. (Eds.): ISC 2018, LNCS 11060, pp. 403–421, 2018.
https://doi.org/10.1007/978-3-319-99136-8_22

(ID)-based setting with achieving all the security properties. In this paper, we come up with a secure cryptographic construction of ID-based strong designated verifier signature which provides anonymity to the signer.

Designated verifier signatures [15] belong to special class of digital signature, which enables the signer to sign a document for an authorised recipient. In such signature, the verifier can validate the authenticity of the signature without being able to transfer the conviction to a third party. Group signature [7] is a candidate tool for construction of an anonymous signature with the property of revocation of signer's identity when required. In a group signature scheme any member of the group can sign on behalf of the group and the signature can be verified by a common public key of the group. There is a group manager who holds an opening key by which he can *open* the identity of the signer, for a given signature. However, there are situations where both the properties i.e. authorised verification and signer's anonymity are required together. In this paper we present an ID-based strong designated verifier group signature (ID-SDVGS) scheme, which fulfils both the above properties in a single compact construction.

1.1 Related Work

***Strong* Designated Verifier Signature.** The idea of designated verifier signature (DVS) was first introduced in 1996 by Jakobsson et al. [15]. In the DVS schemes the property of strongness was first achieved by Saeednia et al. [23]. The first ID-based strong designated verifier signatures (ID-SDVS) scheme is due to Susilo et al. [24] which can be viewed as an ID-based variant of [23]. Lipmaa et al. [19] discussed an issue of *delegatablity* towards the security of SDVS. To address the issue, Zhang et al. [25] proposed a non-delegatable ID-SDVS scheme. But Kang et al. [16] observed a security flaw in the strongness property of [25] and proposed another scheme with security guarantees, however, it was observed in [18] that the signature in [16] is universally forgeable. Another variant of ID-SDVS [17] was presented to address the issue in the short ID-SDVS of [14], but they did not address various security properties of an SDVS, moreover their scheme has been examined to be universally forgeable in [10].

Group Signature. The idea of group signature is due to Chaum and van Heyst [7]. In 1997, Park et al. [22] presented the first ID-based group signature scheme. In 2000, Ateniese et al. [1] proposed first practical and provably secure coalition-resistant construction for group signature scheme based on knowledge proof signature. In 2003, Bellare et al. [2] analysed security properties of group signature and proposed a generic construction of group signature. They captured important properties including anonymity and traceability in their BMW model of security in this paper. Bellare et al. [3] later addressed the securities also for the dynamic group structure. Motivated by the BMW model [2] and its variant [5], Boyen and Waters [6] proposed compact group signature in the standard model in bilinear groups by combining provably secure hierarchical signature and the

Non-Interactive Zero Knowledge (NIZK) proof [13]. In the existing literature, the schemes [8,9,11] are similar in many contexts to the presented work. However, in [8] the anonymity has been realised by the ring signature which has no provision of identity revocation of the actual signer, also we have addressed more security properties like strongness with compare to [8]. In contrast to the schemes [9,11], ours is on the ID-based setting which meets all the requirements for an effective enterprise key management system. Moreover our scheme qualifies more security properties like unverifiability, non-transferability, strongness etc. Though the property 'non-transferability' is defined in [11], we concretely show that our scheme actually achieves it.

1.2 Contribution

Though there are signature schemes in isolation like ID-based signature, group signature, strong designated verifier signature etc., yet for many applications, there is need of a compact single signature which can address properties of all these signatures in one algorithm. We provide a rigorous construction of *ID-based strong designated verifier group signature* to achieve it. Security of our construction relies on the standard assumptions, the decisional bilinear Diffie-Hellman (DBDH) assumption and the decisional linear (DLIN) assumption. The proposed scheme is suitable for the cloud-based electronic health record (EHR) where patients require access to all their medical records in a fine-grained secure way, as discussed in [9]. Our scheme also enjoys application in biometric authentication and identity-management as mentioned in [11].

1.3 Outline of the Paper

In Sect. 2, we introduce some related mathematical definitions, problems and assumptions. In Sect. 3, we present the formal definition of an ID-based strong designated verifier group signature scheme and a formal security model for it. Our proposed scheme is presented in Sect. 4. Lastly in Sect. 6 we briefly conclude the outcomes of the paper presented in Sect. 4. In Sect. 5 we analyze the security of the proposed scheme.

2 Preliminaries

In this section, we introduce the notations used in the paper, some relevant definitions, computational problems and hardness assumptions.

A probabilistic polynomial time (PPT) algorithm is a probabilistic/randomized algorithm that runs in time polynomial in the length of input. We denote by $y \leftarrow A(x)$ the operation of running a randomized or deterministic algorithm A with input x and storing the output to the variable y. If X is a set, then $v \xleftarrow{\$} X$ denotes the operation of choosing an element v of X according to the uniform random distribution on X. We say that a given function $f : N \rightarrow [0, 1]$ is *negligible in* n if $f(n) < 1/p(n)$ for any polynomial p for sufficiently large n [20]. For a group G and $g \in G$, we write $G = \langle g \rangle$ if g is a generator of G.

Definition 1 (Bilinear Map). Let G_1 be an additive cyclic group with generator P and G_2 be a multiplicative cyclic group. Let both the groups are of the same prime order q. Then a map $e : G_1 \times G_1 \rightarrow G_2$ is called a *cryptographic bilinear map* if it satisfies the following properties.

Bilinearity: For all $a, b \in \mathbb{Z}_q^*$, $e(aP, bP) = e(P, P)^{ab}$, or equivalently, for all $Q, R, S \in G_1$, $e(Q+R, S) = e(Q, S)e(R, S)$ and $e(Q, R+S) = e(Q, R)e(Q, S)$.
Non-Degeneracy: There exists $Q, R \in G_1$ such that $e(Q, R) \neq 1$. Note that since G_1 and G_2 are groups of prime order, this condition is equivalent to the condition $g := e(P, P) \neq 1$, which again is equivalent to the condition that $g := e(P, P)$ is a generator of G_2.
Computability: There exists an efficient algorithm (viz. Miller's algorithm [21]) to compute $e(Q, R) \in G_2$ for all $Q, R \in G_1$.

Definition 2 (Bilinear Diffie-Hellman Problem). Given a security parameter λ, let $\langle q, e : G_1 \times G_1 \rightarrow G_2, P, g\rangle \leftarrow \mathfrak{B}(\lambda)$. Let $BDH : G_1 \times G_1 \times G_1 \rightarrow G_2$ be a map defined by $BDH(X, Y, Z) = \omega$ where $X = xP,\ Y = yP,\ Z = zP$ and $\omega = e(P, P)^{xyz}$.

The bilinear Diffie-Hellman problem (BDHP) is to evaluate $BDH(X, Y, Z)$ given $X, Y, Z \xleftarrow{\$} G_1$. (Without the knowledge of $x, y, z \in \mathbb{Z}_q$ – obtaining $x \in \mathbb{Z}_q$, given $P, X \in G_1$ is solving the discrete logarithm problem (DLP)).

Definition 3 (Decisional Bilinear Diffie-Hellman Problem). Given a security parameter λ, let $\langle q, e : G_1 \times G_1 \rightarrow G_2, P, g, X, Y, Z\rangle \leftarrow \mathfrak{C}(\lambda)$. Let $\omega \xleftarrow{\$} G_2$. *The decisional bilinear Diffie-Hellman problem* (DBDHP) is to decide if

$$\omega = BDH(X, Y, Z).$$

That is, if $X = xP, Y = yP, Z = zP$, for some $x, y, z \in \mathbb{Z}_q$, then the DBDHP is to decide if $\omega = e(P, P)^{xyz}$.

Definition 4 (Decisional Bilinear Diffie-Hellman Assumption). Given the parameters mentioned in the above Definition 3 of DBDHP, the *decisional bilinear Diffie-Hellman assumption* (DBDHA) states that, for any PPT algorithm $\mathcal{A}$ which attempts to solve DBDHP, its *advantage* $\mathrm{Adv}_{\mathfrak{D}}(\mathcal{A})$, defined as

$$\begin{aligned}|\Pr[\mathcal{A}(q, e : G_1 \times G_1 \rightarrow G_2, P, g, X, Y, Z, BDH(X, Y, Z)) = 1] - \\ \Pr[\mathcal{A}(q, e : G_1 \times G_1 \rightarrow G_2, P, g, X, Y, Z, \omega) = 1]|\end{aligned}$$

is negligible in λ.

Definition 5 (Decisional Linear Problem). Given a security parameter λ, let the instance $\rho = (q, G_1, G_2, P, aP, bP, arP, bsP, Y_\beta) \leftarrow \mathfrak{D}_\beta^{\mathrm{DLIN}}(\lambda)$. Where $a, b, r, s \in \mathbb{F}_q$. The *decisional linear problem* (DLINP) is to decide whether $\beta = 0$ or 1, where $Y_0 := (r + s)P$, and $Y_1 \leftarrow G_1$. Thus the DLINP is to decide if

$$Y_\beta = Y_0 \ \text{ or } \ Y_\beta = Y_1.$$

Definition 6 (Decisional Linear Assumption). Given the parameters mentioned in the above Definition 5 of DLINP, the *decisional linear assumption* (DLINA) states that, for any PPT algorithm $\mathcal{A}$ which attempts to solve DLINP, its *advantage* $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{DLIN}}(\lambda)$, defined as

$$|\Pr[\mathcal{A}(Y_0, \rho) = 1 | \rho \leftarrow \mathfrak{D}_0^{\mathrm{DLIN}}(\lambda)] - \Pr[\mathcal{A}(Y_1, \rho) = 1 | \rho \leftarrow \mathfrak{D}_1^{\mathrm{DLIN}}(\lambda)]|$$

is negligible in λ.

2.1 Non-Interactive Zero-Knowledge Proof [12]

A Non-Interactive Zero-Knowledge (NIZK) proof [12] is a well studied system in public key cryptography. Due to page constraint we omit the details here. In the full version of this paper, a brief discussion on it has been mentioned.

3 Identity-Based Strong Designated Verifier Group Signature (ID-SDVGS) Scheme

We present here the formal definition of an ID-Based Strong Designated Verifier Group Signature (ID-SDVGS) scheme and formalise a security model for it. We rely on the strong one-time signature (SOTS) scheme [12]. Our scheme achieves the CMA-unforgeability i.e. secure against one-time chosen message attack, following the security notion of [12].

3.1 ID-SDVGS Scheme

In our ID-SDVGS scheme there is a group of $n+2$ members, where $i = 1, \ldots, n$ are the users with identity ID_i who can generate signatures for a fixed designated verifier $\mathcal{V}$, there is a certificate issuing authority (CIA) who issues certificates for the n users and assist them in joining the group. Additionally, there is a group manager (GM) who holds a secret key and can revoke the identity of the signer in case of dispute, without learning private keys of the users. Lastly, in our ID-based setting there is a private key generator (PKG) who issues keys for all the $n+2$ members of the group and for the designated verifier. The structure of our ID-SDVGS scheme is as follows:

1. $params \leftarrow \mathsf{DVGSetup}(\lambda)$: On input security parameter λ, this algorithm generates the system's public parameters *params* and a common reference string (CRS) Σ. In all the algorithms from here onward, *params* will be considered as an implicit input.
2. $(Q_{ID}, S_{ID}) \leftarrow \mathsf{DVGGen}(\mathsf{ID})$: This is the *Key Extraction* algorithm run by the PKG. On input user's identity ID, the algorithm generates user's (public key, private key) pair $(Q_{\mathsf{ID}}, S_{\mathsf{ID}})$.

3. $(Cert_i) \leftarrow \mathsf{DVGJoin}((\mathsf{i}, \mathsf{ID_i}), \mathsf{S_{ID_K}})$: This algorithm is an interactive protocol between the user, and the CIA. On input credentials (i, ID_i) of the user i and private key S_{ID_K} of the CIA and S_{ID_K} by S_{ID_C} and also everywhere in the paper, this algorithm generates user's membership certificate $Cert_i$ with respect to its credentials.
4. $\tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S_{ID_i}}, \mathsf{ID}_\mathcal{V}, (\mathsf{i}, \mathsf{ID_i}), \mathsf{Cert_i}, \mathsf{ID_{GM}}, \Sigma, \mathsf{M})$: This is the *Signature* algorithm run by the signer. On input the SOTS [12], signer's secret key S_{ID_i}, designated verifier's identity $\mathsf{ID}_\mathcal{V}$, signer's credentials (i, ID_i), signer's membership certificate $Cert_i$, group manager's identity $\mathsf{ID_{GM}}$, CRS Σ and the message M, this probabilistic algorithm finally generates an ID-SDVGS $\tilde{\sigma}$ on message M.
5. $b \leftarrow \mathsf{DVGVer}(\mathsf{SOTS}, \mathsf{S_{ID_\mathcal{V}}}, \mathsf{Q_{ID_i}}, \mathsf{M}, \tilde{\sigma})$: This is the *Verification* algorithm run by the designated verifier. On input the SOTS [12], verifier's secret key $S_{\mathsf{ID}_\mathcal{V}}$ the signer's public key Q_{ID_i}, the message M and the ID-SDVGS $\tilde{\sigma}$, this deterministic algorithm confirms whether the signature $\tilde{\sigma}$ is valid or invalid.
6. $\sigma' \leftarrow \mathsf{DVGTran}(\mathsf{SOTS}, \mathsf{S_{ID_\mathcal{V}}}, \mathsf{Q_{ID_i}}, \mathsf{M})$: This is the *Transcript Simulation* algorithm run by the designated verifier. On input the SOTS [12], verifier's secret key $S_{\mathsf{ID}_\mathcal{V}}$, signer's public keys Q_{ID_i} and the message M this deterministic algorithm outputs a valid ID-SDVGS $\tilde{\sigma}'$.
7. $(i, \mathsf{ID}_i) \leftarrow \mathsf{DVGOpen}(\mathsf{SOTS}, \mathsf{S_{ID_{GM}}}, \tilde{\sigma})$: This is the *Open* algorithm run by the group manager GM. On input SOTS, group manager's secret key $gmsk = S_{\mathsf{ID_{GM}}}$, and signature $\tilde{\sigma}$ this deterministic algorithm outputs the actual signer's credential (i, ID_i).

3.2 Security Model for ID-SDVGS Scheme

An ID-SDVGS scheme must satisfy the following security properties.

1. ***Correctness:*** The verification algorithm takes place properly for the correctly generated signature, i.e. if a signature on a message M is correctly computed by a signer, then the designated verifier must be able to verify the correctness of the signature, on the given message.
2. ***Unforgeability:*** It is computationally infeasible to construct a valid ID-SDVGS signature without the knowledge of the private key of either the signer or the designated verifier.
3. ***Unverifiability:*** It is computationally infeasible to verify the validity of an ID-SDVGS signature without the knowledge of the private key of either the signer or the designated verifier. We define below *existential designated unverifiability against an adaptive chosen message and adaptive chosen identities attack.*

Definition 7 (Unverifiability). An ID-SDVGS scheme is said to be *existential designated unverifiable against adaptive chosen message and adaptive chosen identities attack* if for any security parameter λ, no PPT adversary

$\mathcal{A}(q_{H_i}, q_J, q_E, q_i, q_\mathcal{V}, \varepsilon_\mathcal{A}(\lambda), t)$ which runs in time t has a non-negligible advantage

$$\mathrm{Adv}^{\text{EDV-CID2-CMA2}}_{\text{ID-SDVGS},\mathcal{A}}(\lambda) := \varepsilon_\mathcal{A}(\lambda) := |\Pr[\mathcal{A}(Q_{\mathsf{ID}^*_i}, Q_{\mathsf{ID}^*_\mathcal{V}}, m^*, \mathsf{DVGSign}(\mathsf{S}_{\mathsf{ID}^*_i}, \mathsf{D}_{\mathsf{ID}^*_i}, \mathsf{Q}_{\mathsf{ID}^*_\mathcal{V}}, \mathsf{m}^*)) = 1] - \Pr[\mathcal{A}(Q_{\mathsf{ID}^*_i}, Q_{\mathsf{ID}^*_\mathcal{V}}, m^*, \sigma^*) = 1] \quad (1)$$

against the challenger $\mathcal{B}$ in the below security experiment:

1. *Setup*: $\mathcal{B}$ generates *params* for security parameter λ.
2. *Oracle Queries*: $\mathcal{A}$ may request: up to (a) $q_{H_i}, i \in \mathbb{N}$ hash queries on its adaptively chosen identities and messages (b) q_E key extraction queries on its adaptively chosen identities (c) q_J join queries on its adaptively chosen identities (d) q_i signature queries on its adaptively chosen messages and adaptively chosen identities (e) $q_\mathcal{V}$ verification queries on signatures on its adaptively chosen messages m and adaptively chosen identities; and obtain responses for each of its query from $\mathcal{B}$ who acts as a random oracle.
3. *Challenge*: At some point, $\mathcal{A}$ outputs a message m^* and identities ID^*_i of the signer and $\mathsf{ID}^*_\mathcal{V}$ of the designated verifier on which it wishes to be challenged such that $\mathcal{A}$ has never submitted ID^*_i or $\mathsf{ID}^*_\mathcal{V}$ during the key extraction queries. The challenger $\mathcal{B}$ responds with a "signature" σ^* and challenges $\mathcal{A}$ to verify if it is valid or not.
4. *Query Phase 2*: $\mathcal{A}$ continues its queries as in Query Phase 1 with an additional restriction that now it cannot submit a verification query on σ^*.
5. *Output*: Finally, $\mathcal{A}$ outputs a bit b^* which is 1 if the signature is valid and 0 if invalid.

4. ***Non-transferability:*** Given a signature σ on message m, it is infeasible for any PPT adversary $\mathcal{A}$ to decide whether σ was produced by the signer or by the designated verifier, even if $\mathcal{A}$ is also given the private keys of the signer and the designated verifier. In other words, it is impossible for the designated verifier to prove (i.e. to convince) to an outsider that the signature is actually generated by the signer.

Definition 8 (Non-transferability). An ID-SDVGS scheme is said to achieve non-transferability if the signature generated by the signer is computationally indistinguishable from that generated by the designated verifier, that is,

$$\tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID}_\mathsf{i}}, \mathsf{ID}_\mathcal{V}, (\mathsf{i}, \mathsf{ID}_\mathsf{i}), \mathsf{Cert}_\mathsf{i}, \mathsf{ID}_{\mathrm{GM}}, \Sigma, \mathsf{M}) \approx \sigma' \leftarrow \mathsf{DVGTran}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID}_\mathcal{V}}, \mathsf{Q}_{\mathsf{ID}_\mathsf{i}}, \mathsf{M}).$$

5. ***Strongness***: Let $\tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID}_\mathsf{i}}, \mathsf{ID}_\mathcal{V}, (\mathsf{i}, \mathsf{ID}_\mathsf{i}), \mathsf{Cert}_\mathsf{i}, \mathsf{ID}_{\mathrm{GM}}, \Sigma, \mathsf{M})$ be a signature on a message M from a signer i to a designated verifier $\mathcal{V}$. *Strongness* requires that $\tilde{\sigma}$ could have been produced by any other third party i^* for some designated verifier $\mathcal{V}^*$ other than $\mathcal{V}$.

Definition 9 (Strongness). An ID-SDVGS scheme is said to be *strong designated* if given $\tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID_i}}, \mathsf{ID}_{\mathcal{V}}, (\mathsf{i}, \mathsf{ID_i}), \mathsf{Cert_i}, \mathsf{ID}_{\mathrm{GM}}, \Sigma, \mathsf{M})$, anyone, say $\mathcal{V}^*$, other than the designated verifier $\mathcal{V}$ can produce identically distributed transcripts that are indistinguishable from those of $\tilde{\sigma}$ from someone, say i^*, except the signer i. That is,

$$\tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID_i}}, \mathsf{ID}_{\mathcal{V}}, (\mathsf{i}, \mathsf{ID_i}), \mathsf{Cert_i}, \mathsf{ID}_{\mathrm{GM}}, \Sigma, \mathsf{M}) \approx \tilde{\sigma} \leftarrow \mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{S}_{\mathsf{ID}_\mathsf{i}^*}, \mathsf{ID}_{\mathcal{V}}^*, (\mathsf{i}^*, \mathsf{ID}_\mathsf{i}^*), \mathsf{Cert}_\mathsf{i}^*, \mathsf{ID}_{\mathrm{GM}}, \Sigma, \mathsf{M}).$$

6. ***Anonymity:*** By *anonymity* we mean that no one except the group manager should be able to determine the identity of the original signer from the dynamic group. The formal definition is provided as follows:

Definition 10 (Anonymity). Let $\mathcal{A}_{ano}$ be an adversary against the anonymity of our ID-based strong designated verifier group signature scheme (ID-SDVGS). An ID-SDVGS scheme is said to be *anonymous* if for any security parameter λ, no probabilistic polynomial time adversary $\mathcal{A}_{ano}(q_{H_i}, q_J, q_E, q_O, \varepsilon_{\mathcal{A}}(\lambda), t)$ which runs in time t has a non-negligible advantage

$$\mathrm{Adv}^{\mathrm{ANO}}_{\text{ID-SDVGS},\mathcal{A}_{ano}}(\lambda) = |\Pr[\mathrm{Expt}^{ANO-1}_{\text{ID-SDVGS},\mathcal{A}_{ano}}(\lambda) = 1] - \Pr[\mathrm{Expt}^{ANO-0}_{\text{ID-SDVGS},\mathcal{A}_{ano}}(\lambda) = 1] \quad (2)$$

in the security below security experiment:

1. *Setup:* On input security parameter λ, the challenger $\mathcal{B}$ generates the group public key *gpk*, the issuing key *ik* and the opening key *ok*.
2. *Oracle Queries:* The adversary $\mathcal{A}$ may request up to (a) $q_{H_i}, i \in \mathbb{N}$ hash queries q_E key extraction queries q_J join queries, as described in the security experiment of the unverifiability property. Here, $\mathcal{A}$ is also allowed to pose up to q_{Ch_b} queries to the challenge oracle on input $\mathsf{ID}_{i_0}, \mathsf{ID}_{i_1}$ of identities and a message M to obtain a signature of the message under the signing key of ID_{i_b} for $b \in \{0, 1\}$, and up to q_O open queries to the opening oracle on input a message M and a signature $\tilde{\sigma}$ in order to obtain the output of the open algorithm.
3. *Output:* At some point, $\mathcal{A}$ outputs a credential (i^*, ID_i^*) of the signer.
4. *Solution to DLINP*: Challenger outputs a solution of DLINP.

7. ***Traceability:*** *Traceability* is an underlying property of group signature schemes. The property requires that in case of malicious signature, signer's identity should be recoverable by the group manager. In other words, it means that no subgroup of members, even the whole group should be able to generate a valid signature which cannot be opened by the manager in case of misuse and cannot be traced back to the malicious signer or to a member of the coalition.
8. ***Non-frameability:*** By *Non-frameability* we denote the property which implies that an honest user cannot form a valid signature which can be opened by the group manager and can be traced back to another user which has not generated the signature.

4 Proposed Scheme

In this section, we present our proposed ID-SDVGS. As described in Sect. 3, the proposed scheme consists of the following seven algorithms: DVGSetup, DVGGen, DVGJoin, DVGSign, DVGVer, DVGTran and DVGOpen.

DVGSetup: On input security parameter λ, this algorithm generates the system's public parameters $params = (\lambda, G_1, P, G_2, q, e, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8)$ where G_1 is an additive cyclic group of prime order q with generator P, G_2 is a multiplicative cyclic group of prime order q, $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear map a defined in Sect. 2 and $H_1 : \{0,1\}^* \rightarrow G_1$, $H_2 : \{0,1\}^* \times G_1 \rightarrow \mathbb{Z}_q^*$, $H_3 : G_1 \times G_1 \times G_1 \rightarrow \mathbb{Z}_q^*$, $H_4 : \{0,1\}^* \times G_1 \times G_1 \times G_2 \rightarrow \mathbb{Z}_q^*$, $H_5 : G_1 \times G_1 \rightarrow \mathbb{Z}_q^*$, $H_6 : \{0,1\}^* \times G_1 \times G_1 \times G_2 \rightarrow \{0,1\}^\lambda$, $H_7 : G_2 \rightarrow \{0,1\}^\lambda$ and $H_8 : \mathbb{Z}_q^* \rightarrow \{0,1\}^\lambda$ are secure cryptographic hash functions. This algorithm also generates a CRS following the NIZK construction [12] as:

- choose $x_e, y_e \xleftarrow{\$} \mathbb{Z}_q^*$;
- compute $f_e = x_e P, h_e = y_e P$;
- choose $r_c, s_c \xleftarrow{\$} \mathbb{Z}_q$;
- compute $c_1 = E_{f_e,h_e}(P; r_c, s_c) = (r_c f_e, s_c h_e, (r_c + s_c)P)$;
- run the key generation algorithm of [12] and output commitment key $ck \leftarrow K(q, G_1, G_2, e, P)$.

The CRS is $\Sigma = (pk, f_e, h_e, c_1, ck)$. The simulated ciphertext is computed by the simulator as $c_1 = E_{f_e,h_e}(1; r_c, s_c)$. The simulator outputs (Σ, τ, ξ), where $\tau = (sk, r_c, s_c)$ is a trapdoor key and $\xi = (x_e, y_e)$ is extraction key.

DVGGen: Everyone – the group manager (GM), the CIA, the group members $\{i = 1, 2, \ldots, n\}$ and the designated verifier $\mathcal{V}$ submit their respective identities $\mathsf{ID}_{\mathrm{GM}}$, ID_C, ID_i and $\mathsf{ID}_{\mathcal{V}}$ to the PKG. The PKG chooses a random $s \xleftarrow{\$} Z_q$, sets $P_{pub} = sP$ as system's public value and keeps the master secret s confidential. Further for a user with identity $ID \in \{0,1\}^*$, the PKG computes

- public key as $Q_{ID} = H_1(ID) \in G_1$ and
- corresponding private key as $S_{ID} = sQ_{ID} \in G_1$.

Finally, following the above extraction PKG shares the private keys to the corresponding users via a secure channel.

DVGJoin: To join the group, a user provide its credential (i, ID_i) to the KIA. The KIA outputs a certificate $Cert_i$, for the user i, by signing its credential (i, ID_i) as following:

- chooses $t \xleftarrow{\$} \mathbb{Z}_q^*$; and computes
- $T_1 = tP \in G_1$; $\quad h_2 = H_2(i||\mathsf{ID}_i, T_1)$; $\quad T_2 = tP_{pub} + h_2 S_{\mathsf{ID}_K}$.

Finally, sends the $Cert_i = (T_1, T_2)$ to the user i as its membership certificate of the group. On receiving the certificate, the member can check its authenticity by checking the equality $e(T_2, P) = e(T_1 + h_2 Q_{\mathsf{ID}_K}, P_{pub})$.

DVGSign: Prior to sign any message for the designated verifier, the group member runs the key generation algorithm of SOTS [12] to generate the verification and private key pair $(vk_{\text{sots}}, sk_{\text{sots}})$, namely $((f_s, h_s), (x_s, y_s)) \leftarrow KeyGen_{\text{sots}}(q, G_1, G_2, e, P)$ where $vk_{\text{sots}} = (f_s, h_s) = (x_s P, y_s P)$ and $sk_{\text{sots}} = (x_s, y_s)$, for $x_s, y_s \xleftarrow{\$} \mathbb{Z}_q^*$.

Further, using the secret key S_{ID_i}, the user with identity ID_i generates a signature on verification key vk_{sots} as follows:

- chooses $x \xleftarrow{\$} \mathbb{Z}_q^*$ and computes
- $V_1 = xP \in G_1$;
- $h_{(3,vk_{\text{sots}})} = H_3(vk_{\text{sots}}, V_1) = H_3(f_s, h_s, V_1)$;
- $V_2 = xP_{pub} + h_{3,vk_{\text{sots}}} S_{\mathsf{ID}_i} \in G_1$;
- $V = e(V_2, Q_{\mathsf{ID}_{\nu}}) \in G_2$.

The signature on vk_{sots} is $\sigma = (V_1, V_2, V)$. The member encrypts the above signed verification key σ along with its credential (i, ID_i) and membership certificate $cert_i = (T_1, T_2)$ motivated by the Boneh-Franklin CCA-secure encryption scheme [4], using group manager's public key $Q_{\mathsf{ID}_{\text{GM}}}$ as follows:

- chooses $\gamma \xleftarrow{\$} \{0,1\}^{\lambda}$,
- computes $r_1 = H_4(\gamma||i||\mathsf{ID}_i, \sigma)$ and $r_2 = H_5(T_1, T_2)$, $h_\sigma = H_6(i||\mathsf{ID}_i, \sigma)$.
- Computes the ciphertext Ct as the following tuple:

$$\begin{aligned} Ct &= \langle A, B, C, D, E \rangle \\ &= \left\langle r_1 P, r_2 P, h_\sigma \oplus H_7\left(g_{\mathsf{ID}_{\text{GM}}}^{r_1}\right), H_8(r_2) \oplus \gamma, \gamma \oplus H_7\left(g_{\mathsf{ID}_{\text{GM}}}^{r_2}\right) \right\rangle \end{aligned}$$

where $g_{\mathsf{ID}_{\text{GM}}} = e\left(Q_{\mathsf{ID}_{\text{GM}}}, P_{pub}\right) \in G_2$. Furthermore, the user provides a proof of satisfiability of pairing product equations and the statement that $(i||\mathsf{ID}_i, cert_i, \sigma)$ is a plaintext of Ct corresponding to the technique in [12], such that

$$\pi \leftarrow P(\Sigma, S_{gs}(gpk, vk_{\text{sots}}, Ct), W_{gs}(gpk, vk_{\text{sots}}, pk_i, cert_i, \sigma, R))$$

where S_{gs} is a set of pairing equations which are used to verify the group signature (as used in the DVGVer below), W_{gs} is the witness of the NIZK proof and $R = (r_1, r_2)$ is randomness used in the encryption.

Finally the member forms a strong one-time signature (SOTS) σ_{sots} on message M, ciphertext Ct, proof π and the key vk_{sots} using SOTS' signing key sk_{sots}. According to the signing algorithm of SOTS from [12], the signature is formed as follows: Choose $r \leftarrow \mathbb{Z}_q^*$ and compute

$$\sigma_{\text{sots}} = (r, s) = (r, (x_s(r_s - r) + y_s s_s - H(M||vk_{\text{sots}}||Ct||\pi))/y_s).$$

where 'H' is some suitable hash function. The final signature is $\tilde{\sigma} = (Ct, \pi, \sigma_{\text{sots}}, vk_{\text{sots}})$.

DVGVer: The verification proceeds in two steps.

1. Firstly, receiver of the signature $\tilde{\sigma}$ runs the verification algorithm of SOTS-scheme from [12] such that

$$c_s = H\left(M||vk_{\text{sots}}||Ct||\pi\right)P + rf_s + sh_s$$

2. Secondly, he runs the verification part of the NIZK proof

$$V\left(\Sigma, S_{gs}(gpk, vk_{\text{sots}}, Ct)\right).$$

That is, he checks the satisfiability of the following bilinear equations:

$$e(T_2, P) = e(T_1 + h_2 Q_{\mathsf{ID}_K}, P_{pub}),$$
$$V = e(V_1 + h_{(3,vk_{\text{sots}})} Q_{\mathsf{ID}_i}, S_{\mathsf{ID}_{\mathcal{V}}})$$

where the first equation satisfies the witness represented by the certificate and the second equation validates the signature $\sigma = (V_1, V_2, V)$ of the i-th member of the group on the verification key vk_{sots}, for designated verifier $\mathcal{V}$. The correctness of the equations are described in Sect. 5. Finally, satisfying all the above proofs and equalities, the designated verifier validates truthfulness of the signature $\tilde{\sigma}$.

DVGTran: It can be evidenced that upon receiving a signature from the group member i possessing identity ID_i, the designated verifier $\mathcal{V}$ can simulate the signature using its secret key by following the signature algorithm of the scheme. It is sufficient here to show that a designated verifier can generate an identical signature on vk_{sots} as follows:

- chooses $x' \xleftarrow{\$} \mathbb{Z}_q^*$ and computes;
- $V_1' = x'P \in G_1$;
- $h'_{(3,vk_{\text{sots}})} = H_3(vk_{\text{sots}}, V_1') = H_3(f_s, h_s, V_1')$;
- $V_2' = x'P_{pub} + h'_{3,vk_{\text{sots}}} S_{\mathsf{ID}_{\mathcal{V}}} \in G_1$;
- $V' = e(V_2', Q_{\mathsf{ID}_i}) \in G_2$;
- the signature tuple is (V_1', V_2', V').

It follows from the correctness of the scheme that the above simulation is identical to the signature generated by the user i for the verifier $\mathcal{V}$

$$V' = e(V_2', Q_{\mathsf{ID}_i}) = e(V_1' + h'_{3,vk_{\text{sots}}} Q_{\mathsf{ID}_{\mathcal{V}}}, S_{\mathsf{ID}_i})$$

DVGOpen: On input group manager's secret key $gmsk$, group public key gpk, message M, signature $\tilde{\sigma} = (Ct, \pi, \sigma_{\text{sots}}, vk_{\text{sots}})$, the group manager verifies the signature and returns 0 if $V(\Sigma, S_{gs}(gpk, vk_{\text{sots}}, Ct)) = 0$. Otherwise it decrypts Ct using his $gmsk = S_{\mathsf{ID}_{\text{GM}}}$ as follows:

- set

$$Ct = \langle A, B, C, D, E\rangle$$
$$= \left\langle r_1 P, r_2 P, h_\sigma \oplus H_7\left(g_{\mathsf{ID}_{\text{GM}}}^{r_1}\right), H_8(r_2) \oplus \gamma, \gamma \oplus H_7\left(g_{\mathsf{ID}_{\text{GM}}}^{r_2}\right)\right\rangle$$

- compute $C \oplus H_7\big(e(S_{\mathsf{ID}_{\mathrm{GM}}}, A)\big) = h_\sigma = H_6(i||\mathsf{ID}_i||, \sigma)$,
- compute $E \oplus H_7\big(e(S_{\mathsf{ID}_{\mathrm{GM}}}, B)\big) = \gamma$,
- compute $D \oplus \gamma = H_8(H_5(T_1, T_2))$,
- set $r_1 = H_4(\gamma||i||\mathsf{ID}_i, \sigma)$, $r_2 = H_5(T_1^*, T_2^*)$,
- check $A = r_1 P$ and $B = r_2 P$. If not, reject the ciphertext.

Note that, GM can compute r_1 for those identities from the list of users, which satisfy $h_\sigma = H_6(i||\mathsf{ID}_i||, \sigma)$, and select (T_1^*, T_2^*) from the list of certificates which satisfies $D \oplus \gamma = H_8(H_5(T_1^*, T_2^*))$. Further, the GM perform the decryption and output $(i||\mathsf{ID}_i, \sigma, cert_i, \sigma_{\mathrm{sots}})$. Finally, he runs the verification algorithm of SOTS-scheme and output i hence the ID_i.

5 Analysis of the Proposed Scheme

5.1 Correctness of the Proposed Scheme

The correctness of the scheme follows since: if (T_1, T_2) is a correctly generated certificate on user's public key, (V_1, V_2, V) is a valid signature on the verification key vk_{sots} of the underlying strong one-time signature scheme and σ_{sots} is a valid signature on a message M from a signer with identity ID_i for a designated verifier with identity $\mathsf{ID}_{\mathcal{V}}$, it follows from the following equalities and proof:

$$\begin{aligned} e(T_2, P) &= e(tP_{pub} + h_2 S_{\mathsf{ID}_K}, P) = e(T_1 + h_2 Q_{\mathsf{ID}_K}, P_{pub}); \\ V &= e(V_2, Q_{\mathsf{ID}_{\mathcal{V}}}) = e(xP_{pub} + h_{(3,vk_{\mathrm{sots}})} S_{\mathsf{ID}_i}, Q_{\mathsf{ID}_{\mathcal{V}}}) \\ &= e(xP + h_{(3,vk_{\mathrm{sots}})} Q_{\mathsf{ID}_i}, S_{\mathsf{ID}_{\mathcal{V}}}) = e(V_1 + h_{(3,vk_{\mathrm{sots}})} Q_{\mathsf{ID}_i}, S_{\mathsf{ID}_{\mathcal{V}}}); \end{aligned}$$

as by the definition of g_{ID} in the signature protocol.

Further, the correctness of the SOTS signature follows from [12]. Furthermore, to achieve perfect correctness we provide completeness of our NIZK proof below, which together with the correctness of the signature scheme completes the security property of perfect correctness of our ID-SDVGS scheme.

Theorem 1. *The non-interactive protocol of the underlying signature schemes is a perfectly complete non-interactive zero-knowledge proof of the statement that a member certificate is a signature on user's public key.*

Proof. According to the proof in [12], perfect completeness of our NIZK proof follows from the NIZK proof for commitment to zero. We remember the values defined in $\mathsf{DVGSetup}$ algorithm: for randomly chosen $x_e, y_e \leftarrow \mathbb{Z}_q^*$, set $f_e = x_e P, h_e = y_e P$. For random values r_c, s_c a relation describing commitments to 1 is given by $R_1 := \{c_1, r_c, s_c | c_1 = Com(1, r_c, s_c) = (r_c f_e, s_c h_e, (r_c + s_c)P = (c_{11}, c_{12}, c_{13})\}$. The proof is given by $\pi_1 = r_c P$. The verification of the proof follows on input a commitment c_1 if and only if $e(P, r_c f_e) = e(\pi_1, f_e)$ and $e(c_{12}, P) = e(h_e, c_{13} - \pi)$. The latter equation follows, since the left side of the equation is equal to $e(c_1 2, P) = e(s_c h_e, P) = e(s_c y_e P, P)$ and the right side of the equation is equal to

$$e(h_e, c_{13} - \pi) = e(y_e P, (r_c + s_c)P - r_c P) = e(y_e P, s_c P) = e(s_c y_e P, P).$$

5.2 Unforgeability

Unforgeability of our scheme relies on the one-time CMA security of the underlying SOTS scheme. Thus, in the following we provide security proof of the remained security properties, namely - unverifiability, non-transferability, strongness, anonymity, traceability and non-frameability.

5.3 Unverifiability

We now prove that the proposed ID-SDVGS is strongly designated. That is, any third party other than the signer and the designated verifier, cannot verify the validity of a signature from a signer for a designated verifier with non-negligible probability. We show that if there exists a PPT adaptive chosen message and adaptive chosen identity algorithm which can verify the proposed ID-SDVGS, then there exists another PPT algorithm which can use the earlier algorithm to solve the DBDHP. In particular, we prove the following theorem:

Theorem 2. *Given a security parameter λ, if there exists a PPT adversary $\mathcal{A}(q_{H_1}, \ldots, q_{H_8}, q_E, q_J, q_i, q_{\mathcal{V}}, \varepsilon_{\mathcal{A}}(\lambda), t)$ which breaks the designated unverifiability of the proposed ID-SDVGS scheme in time t with success probability $\varepsilon_{\mathcal{A}}(\lambda)$, then there exists a PPT adversary $\mathcal{B}(t', \varepsilon_{\mathcal{B}}(\lambda))$ which solves DBDHP with success probability at least*

$$\varepsilon_{\mathcal{B}}(\lambda) \geq \Big(1 - \frac{1}{q^2}\Big)\Big(1 - \frac{1}{q^4}\Big)\Big(1 - \frac{2}{q_{H_1}}\Big)^{q_E+q_{\mathcal{V}}+q_i}\Big(1 - \frac{2}{q_{H_2}q_{H_1}q_{H_3}q_{H_4}q_{H_5}}\Big)^{q_i} \Big(1 - \frac{2}{q_{H_3}}\Big)^{q_i+q_{\mathcal{V}}}\Big(\frac{2}{q_{H_2}q_{H_3}(1 - q_{H_3}q_{H_2})}\Big)\Big(\frac{2}{q_{H_1}(q_{H_1} - 1)}\Big)\varepsilon_{\mathcal{A}}(\lambda)$$

in time at most

$$t' \leq (q_{H_1} + q_{H_2} + q_{H_3} + q_{H_4} + q_{H_5} + q_{H_6} + q_{H_7} + q_{H_8} + q_E + q_J + 3q_i + q_{\mathcal{V}})S_{G_1} + (q_i + q_{\mathcal{V}})P_e + q_i O_{G_1} + S_{G_1} + S_{G_2} + P_e + t$$

where S_{G_1}, S_{G_2}, O_{G_1}, O_{G_2} is the time taken for one scalar multiplication in G_1 (resp. G_2) and O_{G_1} (resp. O_{G_2}) is the time taken for one group operation in G_1 (resp. G_2), and P_e is the time taken for one pairing computation.

Proof. Let for a security parameter λ, $\mathcal{B}$ is challenged to solve the DBDHP for $\langle q, e : G_1 \times G_1 \to G_2, P, aP, bP, cP, \omega\rangle$ where G_1 is an additive cyclic group of prime order q with generator P, G_2 is a multiplicative cyclic group of prime order q with generator $e(P, P)$, and $e : G_1 \times G_1 \to G_2$ is a cryptographic bilinear map as described in Sect. 2 and $\omega \xleftarrow{\$} G_2$. $a, b, c \xleftarrow{\$} \mathbb{Z}_q^*$ are unknown to $\mathcal{B}$. The goal of $\mathcal{B}$ is to solve DBDHP by verifying if $e(P, P)^{abc} = \omega$ using $\mathcal{A}$, the adversary who claims to forge our proposed ID-SDVGS scheme. In order to simulate public parameters of our ID-SDVGS scheme, $\mathcal{B}$ sets $Q_{\mathsf{ID}_j} = aP$ for $j \in \{i, \mathcal{V}, GM\}$ denoting the identity either of the group member, verifier or group manager. $\mathcal{B}$

simulates the security game for unverifiability with $\mathcal{A}$ by running the *Setup* and by responding all hash queries H_i (i = 1, 2, ..., 8), join queries, key extraction queries, signature queries and verification queries appropriately.

Output: After $\mathcal{A}$ has made its queries, it finally outputs a message M^*, an identity $\mathsf{ID}_{\mathcal{S}}^*$ of a signer and an identity $\mathsf{ID}_{\mathcal{V}}^*$ of a designated verifier on which it wishes to be challenged.

If $\mathcal{A}$ did not make H_1-query for the identities $\mathsf{ID}_{\mathcal{S}}^*$ and $\mathsf{ID}_{\mathcal{V}}^*$, then the probability that verification equality holds is less than $1/q^2$. Thus, with probability greater than $1 - 1/q^2$, both the public keys were computed using H_1-oracle and there exist indices $i, j \in [1, q_{H_1}]$ such that $\mathsf{ID}_{\mathcal{S}}^* = \mathsf{ID}_i$ and $\mathsf{ID}_{\mathcal{V}}^* = \mathsf{ID}_j$. If $\{i, j\} \in RegList$ of registered entities, then abort.

Solution to DBDHP: Otherwise, $\mathcal{B}$ chooses a random $r \xleftarrow{\$} \mathbb{Z}_q^*$ and $T \xleftarrow{\$} G_1$; sets $V_1 = xP$; sets $h_{(3,vk_{\text{sots}})} = H_3(vk_{\text{sots}}, V_1) = H_3(f_s, h_s, V_1)$; $V_2 = xP_{pub} + h_{(3,vk_{\text{sots}})} S_{\mathsf{ID}_i} \in G_1$; sets $V = e(V_2, Q_{\mathsf{ID}_{\mathcal{V}}}) = e(bP, cP)^r \omega^h$; where he sets $V_1 = cP$ and $s = b$, such that $S_{\mathsf{ID}_i} = abP$ and challenges $\mathcal{A}$ to verify the validity of the signature (V_1, V_2, V). Then, the verification holds if and only if each of the following equations holds

$$e(T_2, P) = e(T_1 + h_{(3,i)} Q_{\mathsf{ID}_K}, P_{pub}), \quad V = e(V_2, Q_{\mathsf{ID}_i}) = e(V_1 + h_{(3,vk_{\text{sots}})} Q_{\mathsf{ID}_{\mathcal{V}}}, S_{\mathsf{ID}_i}),$$
$$g_{\mathsf{ID}_{\text{GM}}}^{r_1} = e(r_1 P, S_{\mathsf{ID}_{\text{GM}}}), \quad g_{\mathsf{ID}_{\text{GM}}}^{r_2} = e(r_2 P, S_{\mathsf{ID}_{\text{GM}}}).$$

where

$$\begin{aligned}
\sigma &= e(V_1 + h_{(3,vk_{\text{sots}})} Q_{\mathsf{ID}_{\mathcal{V}}}, S_{\mathsf{ID}_i}) = e(cP + h_{3,vk_{\text{sots}}} abP, bP_{pub}) \\
&= e(rP, bP_{pub}) e(haP, bP_{pub}) = e(P, bP_{pub})^r e(aP, bP_{pub})^h \\
&= e(bP, P_{pub})^r e(aP, bP_{pub})^h = e(bP, cP)^r e(aP, bcP)^h \\
&= e(bP, cP)^r (e(P, P)^{abc})^h = e(bP, cP)^r \omega^h \\
&\Rightarrow \omega^h = (e(P, P)^{abc})^h \Longleftrightarrow \omega = e(P, P)^{abc}
\end{aligned}$$

Then, from the above equation, $\mathcal{B}$ solves the DBDHP by simply returning the response of $\mathcal{A}$ to the strongness challenge.

Probability Calculation: If $\mathcal{B}$ does not abort during the simulation then $\mathcal{A}$'s view is identical to its view in the real attack. The responses to H_1-, H_2-, H_3- and H_4-queries are as in the real attack, since each response is uniformly and independently distributed in G_1 or in G_2 and $\mathbb{Z}_q^*$ respectively. The key extraction, signature and verification queries are answered as in the real attack.

The probability that $\mathcal{B}$ does not abort during the simulation is

$$\left(1 - \frac{2}{q_{H_1}}\right)^{q_E + q_{\mathcal{V}} + q_i} \left(1 - \frac{2}{q_{H_3}}\right)^{q_i + q_{\mathcal{V}}} \left(1 - \frac{2}{q_{H_1} q_{H_2} q_{H_3} q_{H_4} q_{H_5}}\right)^{q_i}.$$

The probability that $\mathcal{A}$ did H_1-query for the identities $\mathsf{ID}_{\mathcal{S}}^*$ and $\mathsf{ID}_{\mathcal{V}}^*$ is $\left(1 - \frac{1}{q^2}\right)$ $\left(\frac{2}{q_{H_1}(q_{H_1} - 1)}\right)$.

The probability that $\mathcal{A}$ issued H_2-query and H_3-query for the identities $\mathsf{ID}_i^*, \mathsf{ID}_K^*$ is

$$\Big(1-\frac{1}{q^4}\Big)\Big(\frac{2}{q_{H_2}q_{H_3}(1-q_{H_3}q_{H_2})}\Big).$$

Clearly $\mathcal{B}$'s advantage $\varepsilon_{\mathcal{B}}(\lambda)$ for solving the BDHP, that is, the total probability that $\mathcal{B}$ succeeds to solve BDHP, is the product of $\mathcal{A}$'s advantage $\varepsilon_{\mathcal{A}}(\lambda)$ of forging the proposed ID-SDVGS and the above three probabilities. Hence

$$\varepsilon_{\mathcal{B}}(\lambda) \geq \Big(1-\frac{1}{q^2}\Big)\Big(1-\frac{1}{q^4}\Big)\Big(1-\frac{2}{q_{H_1}}\Big)^{q_E+q_{\mathcal{V}}+q_i}\Big(1-\frac{2}{q_{H_2}q_{H_1}q_{H_3}q_{H_4}q_{H_5}}\Big)^{q_i}$$
$$\Big(1-\frac{2}{q_{H_3}}\Big)^{q_i+q_{\mathcal{V}}}\Big(\frac{2}{q_{H_2}q_{H_3}(1-q_{H_3}q_{H_2})}\Big)\Big(\frac{2}{q_{H_1}(q_{H_1}-1)}\Big)\varepsilon_{\mathcal{A}}(\lambda)$$

Time Calculation: It can be observed that running time of the algorithm $\mathcal{B}$ is same as that of $\mathcal{A}$ plus time taken to respond to the hash queries, key extraction queries, join queries, signature queries and verification queries, $q_{H_1}+q_{H_2}+q_{H_3}+q_{H_4}+q_{H_5}+q_{H_6}+q_{H_7}+q_{H_8}+q_E+q_J+q_S+q_V$. Hence the maximum running time required by $\mathcal{B}$ to solve the BDHP is

$$t' \leq (q_{H_1}+q_{H_2}+q_{H_3}+q_{H_4}+q_{H_5}+q_{H_6}+q_{H_7}+q_{H_8}+q_E+q_J+3q_i+q_{\mathcal{V}})S_{G_1}$$
$$+(q_i+q_{\mathcal{V}})P_e+q_iO_{G_1}+O_{G_2}+S_{G_2}+t$$

as $\mathcal{B}$ requires to compute one scalar multiplication in G_1 to respond to H_1, H_2, H_3 and H_4 hash query, one scalar multiplication in G_1 to respond to key extraction query and join query, three scalar multiplications in G_1 to respond to signature query, one scalar multiplication in G_1 to respond to verification query; one pairing computation to respond to signature query, one pairing computation to respond to verification query, one group operation in G_1 to respond to signature query, and, one scalar multiplication in G_1, one scalar multiplication in G_2 and one pairing computation to output a solution of DBDHP.

5.4 Non-transferability

As described in Sect. 3, the property of non-transferability implies that the signatures simulated by the designated verifier are indistinguishable from those that he receives from the signer. In $\mathsf{DVGTran}$ of Sect. 4 (proposed scheme) it has been already shown that we achieve this property in our scheme.

5.5 Strongness

To prove this property, we will show that if the i-th signer, with identity ID_i, of the group outputs $V = \mathsf{Sig}(S_{\mathsf{ID}_i}, Q_{\mathsf{ID}_{\mathcal{V}}}, vk_{\text{sots}})$ using his private key S_{ID_i} and the public key $Q_{\mathsf{ID}_{\mathcal{V}}}$ of the designated verifier $\mathcal{V}$ with identity $\mathsf{ID}_{\mathcal{V}}$ as its signature on the verification key vk_{sots} during the $\mathsf{DVGSign}(\mathsf{SOTS}, \mathsf{Q}_{\mathsf{ID}_{\mathcal{V}}}, \mathsf{S}_{\mathsf{ID}_i}, \mathsf{m})$, then the

same value V can be generated using the private key $S_{\mathsf{ID}_i^*}$ of a signer with identity ID_i^* (other than the i-th signer) and the public key $Q_{\mathsf{ID}_\nu^*}$ of a designated verifier with identity ID_ν^* (other than the verifier $\mathcal{V}$). That is, we show that $V = \mathsf{Sig}(S_{\mathsf{ID}_i^*}, Q_{\mathsf{ID}_\nu^*}, vk_{\mathrm{sots}})$ (where $Q_{\mathsf{ID}_\nu^*}$ and $S_{\mathsf{ID}_i^*}$ are defined as in the following) since

$$\begin{aligned}
V &= e(V_2, Q_{\mathsf{ID}_\nu}) \\
&= e(xP_{pub} + h_{(3,vk_{\mathrm{sots}})} S_{\mathsf{ID}_i}, tQ_{\mathsf{ID}_\nu^*}) \quad \text{where } Q_{\mathsf{ID}_\nu} = tQ_{\mathsf{ID}_\nu^*} \\
&= e(xtP_{pub} + h_{(3,vk_{\mathrm{sots}})} tS_{\mathsf{ID}_i}, Q_{\mathsf{ID}_\nu^*}) \\
&= e(xP_{pub} + x(t\text{–}1)P_{pub} + h_{(3,vk_{\mathrm{sots}})} tS_{\mathsf{ID}_i}, Q_{\mathsf{ID}_\nu^*}) \\
&= e(xP_{pub} + x(t\text{–}1)h_{(3,vk_{\mathrm{sots}})} P'_{pub} + h_{(3,vk_{\mathrm{sots}})} tS_{\mathsf{ID}_i}, Q_{\mathsf{ID}_\nu^*}) \\
&\qquad (\text{where } P'_{pub} = h^{-1}_{(3,vk_{\mathrm{sots}})} P_{pub}) \\
&= e(xP_{pub} + h_{(3,vk_{\mathrm{sots}})} (x(t\text{–}1)P'_{pub} + tS_{\mathsf{ID}_i}), Q_{\mathsf{ID}_\nu^*}) \\
&= e(xP_{pub} + h_{(3,vk_{\mathrm{sots}})} S_{\mathsf{ID}_i^*}, Q_{\mathsf{ID}_\nu^*}) \\
&= e(xP + h_{(3,vk_{\mathrm{sots}})} Q_{\mathsf{ID}_i^*}, S_{\mathsf{ID}_\nu^*}) \\
&= e(V_1 + h_{(3,vk_{\mathrm{sots}})} Q_{\mathsf{ID}_i^*}, S_{\mathsf{ID}_\nu^*}).
\end{aligned}$$

where $S_{\mathsf{ID}_i^*} = x(t\text{–}1)P'_{pub} + tS_{\mathsf{ID}_i}$

5.6 Anonymity

Theorem 3 (Anonymity). *Our ID-SDVGS described above is anonymous if DLIN assumption holds for G_1.*

Proof. According to the anonymity experiment in Definition 10 an adversary against anonymity of our ID-SDVGS scheme has access to the following oracles: key extraction oracle, join oracle, challenge oracle and opening oracle. Furthermore, $\mathcal{A}_{ano}$ can issue hash queries to the four hash oracles. According to [12] we are applying a hybrid argument to our proof technique. The consequence of that is that it is sufficient to consider only the challenge oracle queries in order to prove anonymity of the scheme.

Let $b \in \{0, 1\}$ be a bit and $\mathsf{ID}_{i_0}, \mathsf{ID}_{i_1}, M$ is the input for the challenge oracle query which produces a challenge signature as $\sigma^* \leftarrow \mathsf{DVGSign}(\mathsf{gpk}, \mathsf{gsk}[\mathsf{i_b}], \mathsf{M})$. Furthermore, we assume that $\mathcal{A}_{ano}$ has access to all the opening outputs except the one on input (M, σ^*). It also knows all the secret keys of the group members as well as the issuing key for the certificate generation.

We simulate first the challenge oracle on input ID_{i_0}. Following the proof technique from [12], we amend the oracle that we receive the simulation-extractor which we run in the setup algorithm in order to receive the common reference string Σ. The output by the simulation-extractor is $\Sigma = (pk, f_e, h_e, c_1, \sigma)$ where $c_1 = E_{f_e,h_e}(1; r_c, s_c)$ is the ciphertext which encrypts 1. In the challenge phase, we have to simulate the NIZK proof π for the CCA-ciphertext Ct which encrypts $(i_0 || \mathsf{ID}_{i_0}, cert_{i_0}, \sigma_{i_0})$.

Furthermore we note that it is impossible to reuse the verification key vk_{sots} of the strong one-time signature scheme in any of the issued opening queries, since the signature can be used only once. This leads to the conclusion that the tuple $(gpk, vk_{\text{sots}}, \tilde{\sigma})$, where $\tilde{\sigma} = (Ct, \pi, \sigma_{\text{sots}})$ is different from the challenge tuple during the challenge signature formation. Therefore for the extraction of the plaintext $(pk_i, cert_i, \sigma)$ we use the knowledge extractor of the NIZK proof instead of the group master secret key $gmsk$. The extraction works as follows: given a proof π, the extraction algorithm uses extraction key $\xi = (x_e, y_e)$ to decrypt ciphertext Ct. From the perfect soundness of the used NIZK proof we know that Ct encrypts either a satisfying assignment $(pk_i, cert_i, \sigma)$ or it encrypts 1. Since Ct does not encrypt 1 it follows that it encrypts $(pk_i, cert_i, \sigma)$ which implies that S_{gs} is a satisfiable set and we get opening possible with satisfying probability.

Since we are using a CCA-secure encryption procedure motivated by the scheme in [4], we note that adversary's advantage in winning the anonymity experiment in Definition 10 is the same as if the ciphertext Ct would be encrypting 1 instead of $(pk_i, cert_i, \sigma)$.

The same argumentation as above we can apply to the scenario where $\mathcal{A}_{ano}$ has access to the challenge oracle on input ID_{i_1}, where the challenge consists of the ciphertext encrypting 1 and the corresponding simulated proof π.

5.7 Traceability

Let $\tilde{\sigma} = (Ct, \pi, \sigma_{\text{sots}}, vk_{\text{sots}})$ be a valid group signature on a message M. From the perfect-soundness of the NIZK proof we know that Ct encrypts either a satisfying assignment $(pk^*, cert^*, \sigma^*_{\text{sots}})$, where the designated verification algorithm on input $(cert^*, \sigma^*_{\text{sots}})$ outputs 1, or it encrypts 1. After running the opening algorithm the output i corresponds to the registered challenge verification key vk^*, or it aborts if no registration of the key took place. Consequently, the non-registered verification key means that no honest signature took place on that key vk^* and the existing signature σ^* must be a forgery. However this is not possible due to the CMA-security of the underlying signature scheme.

5.8 Non-frameability

Let $\mathcal{A}_{\text{n-fra}}$ be an adversary against non-frameability property of our ID-SDVGS scheme. The adversary outputs a valid signature $\tilde{\sigma} = (C, \pi, \sigma_{\text{sots}}, vk_{\text{sots}})$ on a message M. Furthermore he succeeds in opening procedure and outputs $(i||\mathsf{ID}_i||\sigma, cert^*, \sigma^*_{\text{sots}})$ which associates with the user i. Let $\tilde{\sigma}'$ be the signatures generated by the group member with this revealed identity. It means that the user formed a signature σ'_{sots} on the verification key vk'_{sots} using sk'_i. Assuming the strong unforgeability of the underlying strong one-time signature scheme, $\mathcal{A}_{\text{n-fra}}$ cannot reuse the key vk'_{sots}. That means $\mathcal{A}_{\text{n-fra}}$ needed a new vk_{sots} which was never signed by the member with identity ID_i. Consequently, σ^*_{sots} is a forged signature on vk_{sots}, which contradicts to the CMA-unforgeability of the underlying one-time signature scheme [12].

6 Conclusion

To realise a compact secure cryptographic construction for the situations where signer's anonymity is desired with the designated verification, in this paper we have proposed an ID-based strong designated verifier group signature (ID-SDVGS) scheme by combining the good features of ID-based strong designated verifier signature and the group signature. The scheme is proved secure under standard security notions. More particularly, we have considered all the security properties of the ingredient signatures of the proposed compact signature. More particularly, the *unverifiability* and the *strongness* are essential security properties of a SDVS, however they have not been addressed properly in the literature. We have provided proofs for both the properties of our scheme along with other security proofs. We have realized the proposed construction by obtaining an ID-based instantiation of the generic group signature frame, given by Bellare et al. in Eurocrypt 2003. To the best of our knowledge this is the first construction of ID-based *strong* designated verifier *group* signature.

References

1. Ateniese, G., Camenisch, J., Joye, M., Tsudik, G.: A practical and provably secure coalition-resistant group signature scheme. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 255–270. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44598-6_16
2. Bellare, M., Micciancio, D., Warinschi, B.: Foundations of group signatures: formal definitions, simplified requirements, and a construction based on general assumptions. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 614–629. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39200-9_38
3. Bellare, M., Shi, H., Zhang, C.: Foundations of group signatures: the case of dynamic groups. In: Menezes, A. (ed.) CT-RSA 2005. LNCS, vol. 3376, pp. 136–153. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-30574-3_11
4. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44647-8_13
5. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28628-8_3
6. Boyen, X., Waters, B.: Compact group signatures without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 427–444. Springer, Heidelberg (2006). https://doi.org/10.1007/11761679_26
7. Chaum, D., van Heyst, E.: Group signatures. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 257–265. Springer, Heidelberg (1991). https://doi.org/10.1007/3-540-46416-6_22
8. Chen, Y., Susilo, W., Mu, Y.: Identity-based anonymous designated ring signatures. In: International Conference on Wireless Communications and Mobile Computing, pp. 189–194. ACM (2006)
9. Derler, D., Krenn, S., Slamanig, D.: Signer-anonymous designated-verifier redactable signatures for cloud-based data sharing. In: Foresti, S., Persiano, G. (eds.) CANS 2016. LNCS, vol. 10052, pp. 211–227. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48965-0_13

10. Du, H., Wen, Q.: Attack on Kang et al.'s identity-based strong designated verifier signature scheme. IACR Cryptology ePrint Archive, 2008:297 (2008)
11. Emura, K., Miyaji, A., Omote, K.: An anonymous designated verifier signature scheme with revocation: how to protect a company's reputation. In: Heng, S.-H., Kurosawa, K. (eds.) ProvSec 2010. LNCS, vol. 6402, pp. 184–198. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16280-0_12
12. Groth, J.: Simulation-sound NIZK proofs for a practical language and constant size group signatures. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 444–459. Springer, Heidelberg (2006). https://doi.org/10.1007/11935230_29
13. Groth, J., Ostrovsky, R., Sahai, A.: Perfect non-interactive zero knowledge for NP. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 339–358. Springer, Heidelberg (2006). https://doi.org/10.1007/11761679_21
14. Huang, X., Susilo, W., Yi, M., Zhang, F.: Short designated verifier signature scheme and its identity-based variant. Int. J. Netw. Secur. **6**(1), 82–93 (2008)
15. Jakobsson, M., Sako, K., Impagliazzo, R.: Designated verifier proofs and their applications. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 143–154. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_13
16. Kang, B., Boyd, C., Dawson, E.: Identity-based strong designated verifier signature schemes: attacks and new construction. Comput. Electr. Eng. **35**(1), 49–53 (2009)
17. Kang, B., Boyd, C., Dawson, E.D.: A novel identity-based strong designated verifier signature scheme. J. Syst. Softw. **82**(2), 270–273 (2009)
18. Lee, J.-S., Chang, J.K., Lee, D.H.: Forgery attacks on Kang et al.'s identity-based strong designated verifier signature scheme and its improvement with security proof. Comput. Electr. Eng. **36**(5), 948–954 (2010)
19. Lipmaa, H., Wang, G., Bao, F.: Designated verifier signature schemes: attacks, new security notions and a new construction. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 459–471. Springer, Heidelberg (2005). https://doi.org/10.1007/11523468_38
20. Mao, W.: Modern Cryptography: Theory and Practice. Prentice Hall Professional Technical Reference (2003)
21. Miller, V.S.: Use of elliptic curves in cryptography. In: Williams, H.C. (ed.) CRYPTO 1985. LNCS, vol. 218, pp. 417–426. Springer, Heidelberg (1986). https://doi.org/10.1007/3-540-39799-X_31
22. Park, S., Kim, S., Won, D.: Id-based group signature. Electron. Lett. **33**(19), 1616–1617 (1997)
23. Saeednia, S., Kremer, S., Markowitch, O.: An efficient strong designated verifier signature scheme. In: Lim, J.-I., Lee, D.-H. (eds.) ICISC 2003. LNCS, vol. 2971, pp. 40–54. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24691-6_4
24. Susilo, W., Zhang, F., Mu, Y.: Identity-based strong designated verifier signature schemes. In: Wang, H., Pieprzyk, J., Varadharajan, V. (eds.) ACISP 2004. LNCS, vol. 3108, pp. 313–324. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27800-9_27
25. Zhang, J., Mao, J.: A novel id-based designated verifier signature scheme. Inf. Sci. **178**(3), 766–773 (2008)