Conditionally Verifiable Signatures

Ian F. Blake[1] and Aldar C-F. Chan[2]

[1] University of Toronto
[2] INRIA Rhône-Alpes

Abstract. We introduce a new digital signature model, called conditionally verifiable signature (CVS), which allows a signer to specify and convince a recipient under what conditions his signature would become valid and verifiable; the resulting signature is not publicly verifiable immediately but can be converted back into an ordinary one (verifiable by anyone) after the recipient has obtained proofs, in the form of signatures/endorsements from a number of third party witnesses, that all the specified conditions have been fulfilled. A fairly wide set of conditions could be specified in CVS. The only job of the witnesses is to certify the fulfillment of a condition and none of them need to be actively involved in the actual signature conversion, thus protecting user privacy. It is guarantee that the recipient cannot cheat as long as at least one of the specified witnesses does not collude. We formalize the concept of CVS and define the related security notions. We also derive the relations between these notions. We give a generic CVS construction based on any CPA-secure identity based encryption scheme. Theoretically, we show that the existence of IBE with indistinguishability under a chosen plaintext attack (a weaker notion than the standard one) is necessary and sufficient for the construction of a secure CVS.

1 Introduction

Balancing between the accountability and privacy of a signer is an important but largely unanswered issue of digital signatures. A digital signature scheme usually consists of two parties, a signer and a recipient, with the former giving his signature on a message/document to the latter as his commitment or endorsement on the message. To ensure that the signer is held accountable for his commitment, his signature needs to be publicly verifiable (by anyone) or, at least, verifiable by a mutually trusted third party; otherwise, the signer could deny having signed the document as nobody can prove he really did, and the non-repudiation property cannot be achieved. However, public verifiability of a digital signature would put the signer's privacy at risk as a digital signature could be replicated and spread so easily, compared to its handwritten counterpart. More importantly, if the message presents valuable information about the signer, then the signed message itself is a certified piece of that information. Hence, the interests of the signer and the recipient are in conflict.

Of course, ensuring signer privacy and accountability simultaneously seems to be impossible. But, fortunately, in most real world scenarios, we usually wish to maintain privacy or limited verifiability of a digital signature up to a certain instant and restore non-repudiation afterwards. This could be better illustrated by the example of future/option trading. In a future trade, the seller signs a contract with the buyer specifying the price and quantity he has agreed with the buyer but the contract is not effective before a future execution date. For reasons like preventing other sellers from manipulating the price or avoiding any adverse effects which affect further negotiation with other buyers, ideally, before the execution date, the seller does not want anyone to be able to associate him with the contract, at least ensuring that the buyer is unable to convince others of the validity of their agreement. Whereas, on or after the execution date, an honest seller usually does not worry about his signature being publicly verifiable. In fact, to protect the interest of the recipient, the seller's signature has to be verifiable by others. Hence, we could view reaching the execution date as a certain condition to be fulfilled before the signature of the signer (seller) could be revealed to the recipient, and before such fulfillment, we wish to achieve signer privacy. We notice that many business activities involving digital signatures have similar situations. The essence is how the signer could ensure non-verifiability of his signature before certain conditions are fulfilled (in the future trading case, the condition is the execution date has passed) but still can convince the recipient that he will be obligated to exercise his commitment; in other words, he

needs to give the recipient some guarantee that his commitment or his signature will become effective or publicly verifiable once all the conditions are fulfilled.

On the other hand, the non-repudiation property of a digital signature could also have a serious repercussion to the signer if there is no way to allow him to control when and how a recipient could obtain his signature when sending it out. In the worst case, a signer may get into a fraud trap. In the online world, the lack of physical proximity could render a careful signer hesitant in giving his signature (say for a payment authorization) to another party because he is not given any guarantee that he will obtain what he is supposed to as an exchange of his signature. From the recipient's perspective, if the signer does not send out his signature, the recipient will not give what the signer needs. For instance, if the signer makes an online purchase, he may not receive any guarantee that his order will be delivered but the seller (recipient) will not send it out unless the signer gives out his signature on a payment authorization. This kind of deadlock due to mistrusting parties is not easily solved but could be partially solved if the signer could ensure that the recipient can never obtain a valid signature of his unless some conditions (specified by the signer) are fulfilled, namely, the recipient sends out the signer's order in the online purchase example.

To provide a flexible solution to this problem of controllably passing signatures from one party to another without actively involving a trusted third party, we introduce a new signature concept called conditionally verifiable signatures (CVS). In a CVS scheme, the signer gives the recipient some seemingly random number, what we call a *partial signature*, and specifies a set of conditions the fulfillment of which will allow the recipient to extract the signer's signature from the partial signature. The partial signature is not immediately verifiable; fulfilling the specified conditions is necessary to retrieve a valid ordinary signature from it. To convince the recipient that his ordinary signature could be extracted form a partial signature, the signer runs a confirmation protocol with the recipient to prove that his signature could be retrieved once all the specified conditions are fulfilled. Before the ordinary signature becomes effective (that is, extracted), the partial signature is no more convincing than any random number, namely, nobody could link the partial signature to its alleged signer. We formulate this property by the notion of *simulatability* in this paper, that is, anyone could use just public information of the signer to simulate a given partial signature while others cannot judge whether it is genuine. In other words, nobody could distinguish between a genuine partial signature and a simulated one. In fact, in our model, even given the signer's private key, nobody could tell the validity of a given partial signature if the random coins used to generate it are not available. That is, one can determine the genuineness of a partial signature only if he has both the signer's private key and the random coins used to generate it. In order to enforce the verification of condition fulfillment, we need a number of third party witnesses mutually trusted by both the signer and recipient. In our model, the only job of these witnesses is to verify whether the given conditions are fulfilled and they are unaware of the conversion or even the existence of the partial signature. That is, the witnesses do not participate in the actual signature conversion. Details of the model are given in the next section.

1.1 Conditionally Verifiable Signature

In the CVS model, a signer is allowed to embed a set of verifiability conditions C into his ordinary signature σ to create a partial signature δ that is solely verifiable by the recipient, who cannot immediately convince others of the validity of δ but can convert it back to the universally verifiable one σ (i.e. verifiable by everyone) after obtaining from a number of witnesses (appointed by the signer) the proofs that all the specified verifiability conditions have been fulfilled.[3] These proofs are in the form of signatures on condition statements, signed by the witnesses, about how the specified conditions are considered as fulfilled. In order to convince the recipient to accept a given partial signature δ on a message M (whose

[3] Throughout the rest of this paper, we will denote the ordinary (universally verifiable) signature and the CVS partial signature by σ and δ respectively, unless otherwise specified.

validity could not be verified), the signer runs a proof/confirmation protocol, which could be interactive or non-interactive, with the recipient to convince the latter that δ is indeed his partial signature on M, from which the corresponding ordinary signature could be recovered using the specified witnesses' signatures on the specified verifiability condition statements in C.

Given that $\mathcal{W}$ is the set of all possible witnesses, an instance set of verifiability conditions C is of the form $\{(c_i, W_i) : c_i \in \{0,1\}^*, W_i \in \mathcal{W}\}$ where each condition statement c_i is a string of arbitrary length describing a condition to be fulfilled. Examples of c_i include "A reservation has been made for Alice on flight CX829, 14 Jul 2005.", "A parcel of XXX has been received for delivery to Bob.", "It is now 02:00AM 18 Jan 2003 GMT.", "An emergency has happened." and so on. The recipient needs to request each one of the specified witnesses, say W_i, to verify whether the condition stated in c_i is fulfilled and in case it is, to sign on c_i to give him a witness signature σ_i. These witness signatures σ_i's would allow the recipient to recover the publicly verifiable, ordinary signature σ from the partial signature δ.

It is not necessary for a recipient to present the partial signature or the message itself to the witnesses in order to get their endorsements on the statement about the fulfillment of a condition. Even so, the witness signatures could still recover the ordinary signature from the collected witness signatures. The only trust we place on the witnesses is that they only give out their signatures on a condition statement when the specified conditions are indeed fulfilled. In fact, it is not difficult to imagine that the existence of such witnesses is abundant in any business transaction; in most cases, any party involved in processing an order would inherently be trusted by both the signer and recipient, a good candidate as a witness. A typical example is the postal office which is involved in delivering the order the signer placed on the recipient of a signature for his payment authorization. In addition, we could achieve a fairly high level of privacy in that the witnesses are unaware of the message or the partial signature when verifying the fulfillment of a given condition, namely, he does not learn the deal between the signer and the recipient.

We could view the partial signature as a blinded version of the ordinary signature, that is, nobody could verify its validity. In our CVS formulation, this non-verifiability property is expressed by the notion of *simulatability* — there exists a polynomial time simulator which is computable using only public information of the signer and outputs a fake signature computationally indistinguishable from the partial signature; that is, even given a genuine partial signature, nobody with bounded computation power could assure that it is not a fake one generated by the simulator. As a result, when the recipient presents a partial signature to others to convince them of its validity, nobody could tell whether the signer has really created it or the recipient has generated it himself using the simulator. Of course, it is natural to worry about whether the confirmation protocol would leak out useful information to help distinguishing between a genuine and a fake partial signature. We show in Section 2 that if the confirmation protocol is zero knowledge, then it would leak no useful information for such a purpose and the CVS scheme is said to be *non-transferable*.

As usual, unforgeability is a basic requirement for a secure CVS scheme. More specifically, we require that even colluding with all the witnesses and allowed to query ordinary and partial signatures of his choice, nobody could present a message signature pair not previously queried such that the signature is valid for the message. This is often called existentially unforgeability against a chosen message attack.

As mentioned earlier, beside protecting the signer's privacy, CVS is also aimed at protecting the signer from fraud trap. It offers the signer the guarantee that the recipient would not get his signature on a document if he could not get what the recipient are committed to. In other words, if the specified conditions are not fulfilled, that is, the corresponding witness signatures are not available, the ordinary signature could never be retrieved from a given partial signature. This is the *cheat-immunity* property of a CVS scheme. We assume that the witnesses would not collude with the recipient; this is reasonable because the signer can choose the witnesses at his wish. We show that this property is implicitly achieved in an unforgeable and simulatable CVS scheme if its confirmation protocol is also zero knowledge.

1.2 Comparison with Related Work

Related work on controlling the verifiability of a digital signature includes designated verifier signatures [26, 34], undeniable signatures [4, 6, 8, 9, 12, 19, 16, 27, 28], designated confirmer signatures [7, 5, 15, 24, 29, 32], fair exchanges [1], and timed release of signatures [17, 18, **?**]. Despite the considerable amount of work in limiting the verifiability of a digital signature, the conditions that could be incorporated into a digital signature scheme are still very restrictive; the existing protocols merely ensure that only a designated recipient can verify but cannot convince anybody else of the validity of a signature (in designated verifier signatures) and/or collaboration of the signer (in undeniable signatures) or a third party designated by the signer (in designated confirmer signatures, fair exchange) is needed in verifying the signature. Implementing more complex policies or specifying more varied conditions in these schemes has to resort to appending the condition/policy description inside the message and rely on a third party to enforce them in signature verification and conversion. Hence, there is almost no protection of the privacy of the signer and the recipient with respect to any third party which, if present, is involved in the actual signature conversion and sees the message. In contrast, the only information a third party needs to know in CVS is the condition to be fulfilled. In fact, roughly speaking, these signature schemes could possibly be considered as special instantiations of CVS.

Fair exchange of digital signatures [1] has drawn much attention mainly due to its potential application in electronic commerce. In essence, it is an instantiation of a convertible designated confirmer signature. However, beside contract signing, the applications of fair exchange are still limited to trading regenerable (digital) goods. CVS could provide a seemingly better solution for trading non-regenerable items. Concurrent signatures [10] are another similar proposal for solving the contract signing problem but CVS cannot give a construction for concurrent signatures.

Timed release of signatures are usually implemented by the time-lock puzzle [17, 18] requiring the recipient to go through a series of computation tasks in order to control when he could recover the signature; the main advantage is no third party is needed but it requires intensive computation resources and the only condition specifiable is relative time. More importantly, resuming verifiability of a signature has a rough timing and may not be spontaneous. CVS does not have these problems but needs a trusted time server periodically broadcasting a single endorsement about the current time to all users. When used for timed release of digital signatures, CVS can be considered as a generalization of time capsule signatures [**?**].

Verifiable signature sharing [14] is a well studied technique to limit the verifiability of a signature in which a signature is divided in such a way that a certain minimum number of parties, each holding a share of the signature, need to pool out their shares in order to recover the signature. When receiving a share, each party could verify its validity. However, it is not trivial to incorporate verifiability conditions in such a scheme and finding such a number of trusted parties in a trading activity is not easy either. The verifiability of a share also implies that one could link a signature share to its alleged signer even though it is not a complete signature with binding power. As a result, the privacy of the signer as required in scenarios like the future trading example could not be achieved.

1.3 Our Contributions

The main contribution of this paper is two-fold: First, a new signature model with controllable verifiability, particularly useful in electronic commerce and digital business, is introduced. Second, the equivalence between CVS and CPA-secure IBE in terms of existence is shown.

Through the new model of conditionally verifiable signatures, a signer can incorporate a wide range of verifiability conditions into an ordinary signature scheme to control its verifiability and validity while minimizing the requirement or trust on third-parties. To the best of our knowledge, it is the first scheme of its kind in the literature. Before this work, it is fair to say that the problem of seamlessly incorporating verifiability conditions into a signature scheme to control its validity and allowing spontaneous signature

recovery upon the fulfillment of the specified conditions remains largely open. In fact, we could possibly view CVS as a more general, unified concept incorporating the ideas of existing work (including undeniable signatures, designated confirmer signature, fair exchange and timed release of signatures), but provides more effective and flexible solutions to the scenarios these existing schemes could not solve satisfactorily, particularly those in digital business or electronic commerce. A typical example of these would be the deadlock scenario mentioned earlier about the online purchase between mistrusting parties; using the post office as a witness, CVS would reasonably solve this problem.

We also demonstrate the feasibility of CVS through giving a generic construction based on any existential unforgeable signature scheme and any semantic secure identity based encryption (IBE) scheme. We also show that a secure CVS scheme is equivalent to an IBE scheme which is indistinguishable under a chosen plaintext attack (IND-ID-CPA), a weaker notion than the commonly accepted security notion against an adaptive chosen ciphertext attack (IND-ID-CCA) in IBE. Hence, we believe that CVS could be constructed based on a weaker assumption than IBE. Furthermore, to the best of our knowledge, it has only been shown that the existence of CCA-secure public key encryption is necessary for the existence of CPA-secure IBE [**?**]. In this paper, we establish the result that the existence of a new theoretical construct of CVS is necessary for the existence of CPA-secure IBE.

In addition, we give a detailed treatment on modeling the security goals and the adversary capabilities of CVS. We show the relationships and implications between these notions. In particular, we show cheat-immunity is implied by unforgeability and simulatability if the confirmation protocol is zero knowledge.

The rest of this paper is organized as follows. We give the definition of a conditionally verifiable signature scheme and its notions of security and derive relationships between these notions in the next section. In Section 4, we give a generic CVS construction and show the equivalent between CVS and IBE. Finally, our conclusions are given in Section 5.

2 Definitions and Security Notions

The players in a conditionally verifiable signature scheme include a signer S, a recipient or verifier V, and a number of witnesses $\{W_i\} \subseteq \mathcal{W}$ (let $|\{W_i\}| = L$). A CVS scheme consists of the following algorithms and a confirmation protocol.

Key Generation (CVKGS, CVKGW). Given a security parameter λ, let $\mathsf{CVKGS}(1^\lambda) \rightarrow (PK_S, sk_S)$ and $\mathsf{CVKGW}(1^\lambda) \rightarrow (PK_W, sk_W)$ be two probabilistic algorithms. Then, (PK_S, sk_S) is the public/private key pair for a signer S and (PK_W, sk_W) is the public/private key pair for a witness W.

Signing and Verification (Ordinary Signatures) ($\mathsf{SigS}, \mathsf{VerS}$)/($\mathsf{SigW}, \mathsf{VerW}$). $\mathsf{SigS}(m, sk_S) \rightarrow \sigma_S$ is an algorithm generating an ordinary (universally verifiable) signature σ_S of the signer S for a message $m \in \mathcal{M}$. $\mathsf{VerS}(m, \sigma_S, PK_S) \rightarrow \{0, 1\}$ is the corresponding signature verification algorithm, which outputs 1 if σ_S is a true signature of S on the message m and outputs 0 otherwise. As usual, for all $(PK_S, sk_S) \in \mathsf{CVKGS}(1^\lambda)$ and all $m \in \mathcal{M}$, we require the following: $\mathsf{VerS}(m, \mathsf{SigS}(m, sk_S), PK_S) = 1$. Similarly, $\mathsf{SigW}(m, sk_W) \rightarrow \sigma_W$ and $\mathsf{VerW}(m, \sigma_W, PK_W) \rightarrow \{0, 1\}$ are the signature generation and verification algorithms of the witness W. Sometimes, we may write SigW as CVEndW to reflect the fact that it is actually an endorsement of W.

Partial Signature Generation (CVSig). Given a set of verifiability conditions $C \subseteq \mathcal{C} \times \mathcal{W}$ and the corresponding set of witness public keys PK_C, $\mathsf{CVSig}(m, C, sk_S, PK_S, PK_C) \rightarrow \delta$ is a probabilistic algorithm for generating the partial signature δ on message $m \in \mathcal{M}$ under the set of verifiability conditions C. *Note that δ is not universally verifiable.*

Ordinary Signature Extraction (CVExtract). $\mathsf{CVExtract}(m, C, \delta, PK_S, \sigma_C) \rightarrow \sigma / \perp$ is an algorithm which extracts the corresponding ordinary signature σ from a partial signature δ for a message m under the verifiability condition specified by C and a signing public key PK_S when given the set of witness signatures or endorsements σ_C. The extracted signature σ is a universally verifiable one. In case the extraction fails, it outputs $\perp$. *Note that* $\sigma_C = \{\mathsf{SigW}(sk_{W_i}, c_i) : (c_i, W_i) \in C\}$.

CVS Confirmation/Verification. $\mathsf{CVCon}_{(S,V)} = \langle \mathsf{CVConS}, \mathsf{CVConV} \rangle$ is the signature confirmation protocol between the signer and recipient, which could be interactive or non-interactive:

$$\mathsf{CVCon}_{(S,V)}(m, C, \delta) = \langle \mathsf{CVConS}(\sigma, sk_S, r), \mathsf{CVConV}() \rangle (m, C, \delta, PK_S, PK_C) \rightarrow v = \begin{cases} 0 \\ 1 \end{cases}$$

The common input consists of the message m, the set of verifiability conditions C, the partial signature δ, and the public keys of the signer PK_S and the involved witnesses public keys PK_C. The private input of the signer S is σ, sk_S, and r where σ is the corresponding ordinary signature (on the message m) embedded in δ, and r represents the random coins S used in generating δ. The output is either 1 ("true") or 0 ("false"). In essence, this protocol allows the signer S to prove to the recipient V that δ is indeed his partial signature on m, which can be converted back into a publicly verifiable signature σ (i.e. $\mathsf{VerS}(m, \sigma, PK_S) = 1$), once V has obtained all the witness signatures/endorsements on the condition statements as specified in C. Ideally, we want this protocol to be zero-knowledge; the interactive version is considered in this paper.

In general, a CVS scheme should satisfy both completeness and perfect convertibility property. Completeness ensures that a valid ordinary signature can be retrieved from a valid partial signature. A CVS scheme is perfectly convertible if nobody could distinguish whether a given ordinary signature is extracted from a partial signature or generated directly. Regarding security, a secure CVS scheme should also satisfy unforgeability, simulatability, cheat-immunity, and have a zero knowledge confirmation protocol.

Oracle Queries — Allowed Adversary Interaction. In our security model, two types of adversary interaction are allowed:

1. **Signing Oracle** $O_S(m, C)$**.** For fixed keys PK_S, sk_S, $\{PK_{W_l}\}$, $\{sk_{W_l}\}$, on input a signing query $\langle m, C \rangle$ (where $m \in \mathcal{M}$ and $C = \{(c_i, W_i) : c_i \in \mathcal{C}, W_i \in \mathcal{W}\}$ is a set of verifiability conditions), O_S responds by running CVSig to generate the corresponding partial signature δ. After sending δ to the querying party, O_S runs the confirmation protocol $\mathsf{CVCon}_{(S,V)}$ with the querying party to confirm the validity of δ. *Note that a malicious verifier is allowed to put in any random number in place of δ when running the confirmation protocol.*
2. **Endorsement Oracle** $O_E(c, W)$**.** For fixed keys $\{PK_{W_l}\}$, $\{sk_{W_l}\}$, on input an endorsement query $\langle c, W \rangle$, Q_E responds by retrieving the needed witness private key sk_W and then running the witness endorsement/signing algorithm SigW (or CVEndW) to create a witness signature $\sigma_W(c)$ on the condition statement c.

As we consider adaptive attacks in our model, these oracle queries may be asked adaptively, that is, each query may depend on the replies of the previous queries.

2.1 Unforgeability

Unforgeability ensures that there is a negligible probability to forge an ordinary signature even though all the witnesses collude and are given access to other ordinary and partial signatures of their choice. The details are given by the following game between a challenger and an adversary:

In the setup, the challenger takes a security parameter λ, runs the key generation algorithms for the signer and all witnesses, that is, $(PK_S, sk_S) \leftarrow \{\mathsf{CVKGS}(1^\lambda)\}$ and $(PK_{W_l}, sk_{W_l}) \leftarrow \{\mathsf{CVKGW}(1^\lambda)\}$. The challenger gives the adversary all the public keys, PK_S and $\{PK_{W_l}\}$ and all the witness private keys $\{sk_{W_l}\}$. The challenger keeps the signer's private key sk_S. Then, the adversary is allowed to make queries to O_S to request a partial signature δ_j for $\langle m_j, C_j \rangle$. Note that the adversary has the witness private keys so no O_E query is necessary. Finally, the adversary has to output a message-signature pair (m, σ) where $m \neq m_j$ for all j. The adversary $\mathcal{A}$ is said to win this game if $\mathsf{VerS}_S(m, \sigma) = 1$.

Definition 1 *A CVS scheme is unforgeable if the probability of winning the above game, $p_{\mathcal{A}}^{UF}$, is negligible in the security parameter λ for all PPT (Probabilistic Polynomial Time) adversaries $\mathcal{A}$.*

2.2 Simulatability

In order to ensure the protection of signer privacy or to void out the non-repudiation property of the ordinary signature before all the verifiability conditions of a given partial signature are fulfilled, the partial signature should be (computationally) indistinguishable from the output of a certain **public** PPT simulator: $\mathsf{Fake}(m, C, PK_S, PK_C) \rightarrow \delta'$.

As can be seen, the simulator only uses public information of the signer; hence, a partial signature is not linkable to its alleged signer, thus protecting his privacy. The notion about the indistinguishability between a genuine partial signature and a simulator output is best described by the following game between a challenger and an adversary:

In the setup, the challenger takes a security parameter λ, runs the key generation algorithms for the signer and all witnesses, that is, $(PK_S, sk_S) \leftarrow \{\mathsf{CVKGS}(1^\lambda)\}$ and $(PK_{W_l}, sk_{W_l}) \leftarrow \{\mathsf{CVKGW}(1^\lambda)\}$. The challenger gives the adversary all the public keys, PK_S and $\{PK_{W_l}\}$. The challenger keeps the private keys $\{sk_{W_l}\}$. We consider the strongest security model in this paper — the signer's private key sk_S is also given to the adversary.[4] Then, the adversary is allowed to make queries to obtain the signer's partial signatures and witness signatures of messages of his choice until it is ready to receive a challenged partial signature. It can make two types of oracle queries: (1) Signing Query $\langle m_j, C_j \rangle$ to O_S; (2) Endorsement Query $\langle c_j, W_j \rangle$ to O_E. As the simulator Fake is publicly known, the adversary could also freely get a simulator output for any message and condition of his choice. Once the adversary decides it is ready for a challenge, it outputs a message $m \in \mathcal{M}$ and a set of conditions $C \subset \mathcal{C} \times \mathcal{W}$ on which it wishes to be challenged. Let C_E^1 denote the set of all endorsement queries sent to O_E previously. The only constraint is that $C \backslash C_E^1 \neq \phi$ (the empty set). The challenger flips a coin $b \in \{0, 1\}$ and outputs the following challenge to the adversary:

$$\delta_b = \begin{cases} \mathsf{CVSig}(m, C, sk_S, PK_S, PK_C), & b = 0 \\ \mathsf{Fake}(m, C, PK_S, PK_C), & b = 1 \end{cases}$$

The adversary is allowed to run until it outputs a guess. Let C_E^2 be the set of queries that have been made to O_E so far after the challenge is issued. The adversary can issue more (but polynomially many) queries, both signing and endorsement, but for any endorsement query (c_j, W_j), the following must hold: $C \backslash (C_E^1 \cup C_E^2 \cup \{(c_j, W_j)\}) \neq \phi$. Finally, the adversary halts and outputs a guess b' for the hidden coin b. The adversary is said to win this game if $b' = b$. The advantage of the adversary $\mathcal{A}$ is defined as: $Adv_{\mathcal{A}}^{Sim}(\lambda) = \left| Pr[b' = b] - \frac{1}{2} \right|$.

Definition 2 *If there exists a PPT simulator Fake such that the advantage of winning the above game is negligible in the security parameter λ for all PPT adversaries, then the given CVS scheme is simulatable (with respect to Fake).*

2.3 Zero Knowledge Confirmation Protocol and Non-transferability

In this paper, we use the notion of simulatability of the communication transcript as a formulation for the zero knowledge property of the confirmation protocol. In details, any communication transcript recorded in carrying out the confirmation protocol could be simulated by a PPT simulator SimT (using only public information) whose output is indistinguishable from a genuine transcript.

The definition of simulatability of CVS ensures that nobody could associate a partial signature to its signer or tell its validity given just the partial signature. If given also the communication transcript of the confirmation protocol for the partial signature, nobody could still tell its validity, then the CVS scheme is said to be **non-transferable**. The formulation of **non-transferability** is very similar to that

[4] In addition to O_S queries, the adversary can generate partial signatures of arbitrary messages and conditions on its own. But even on identical input, these signatures may not be the same as those from the challenger since the random coins used are likely to be different.

of simulatability described previously except that it includes an additional simulator for the transcript of the confirmation protocol, and in the challenge phase, the adversary receives either a genuine partial signature and its confirmation protocol transcript or a fake (simulated) partial signature and its simulated transcript. Note that while the confirmation protocol is carried out in all oracle queries, no confirmation protocol would be carried out in the challenge phase; otherwise, it is meaningless to give the adversary a challenge as the validity of a partial signature could simply be asserted through the interaction in carrying out the protocol. We show later that a CVS scheme is non-transferable if it is simulatable and its confirmation protocol is zero knowledge, and the transcript simulator SimT for the zero knowledge proof could be used as a transcript simulator for the fake partial signature.

2.4 Cheat-immunity

Cheat-immunity guarantees that the recipient of a partial signature cannot recover the ordinary signature without collecting all the needed witness signatures. Details are described by the following game:

In the setup, the challenger takes a security parameter λ, runs the key generation algorithms for the signer and all witnesses, that is, $(PK_S, sk_S) \leftarrow \{\mathsf{CVKGS}(1^\lambda)\}$ and $(PK_{W_l}, sk_{W_l}) \leftarrow \{\mathsf{CVKGW}(1^\lambda)\}$. The challenger gives the adversary all the public keys, PK_S and $\{PK_{W_l}\}$. The challenger keeps all the private keys sk_S and $\{sk_{W_l}\}$. The adversary makes queries to obtain the signer's partial signatures and witness signatures of messages of his choice until it is ready to receive a challenge partial signature. It can make two types of queries: (1) Signing Query $\langle m_j, C_j\rangle$ to O_S; (2) Endorsement Query $\langle c_j, W_j\rangle$ to O_E. *With these two types of queries, the adversary can obtain the ordinary signature of the signer on any message of his choice.* Once the adversary decides it is ready for a challenge, it outputs a message $m \in \mathcal{M}$ not queried before and a set of verifiability conditions $C = \{(c_i, W_i)\} \subset \mathcal{C} \times \mathcal{W}$ on which it wishes to be challenged. Let C_E^1 denote the set of all the endorsement queries made to O_E before the challenge. The only constraint is that $C \backslash C_E^1 \neq \phi$. The challenger uses CVSig to generate a partial signature δ on a message m under the conditions in C. It sends δ as the challenge to the adversary and runs the confirmation protocol $\mathsf{CVCon}_{(S,V)}$ with it. Let C_E^2 denote the set of all the endorsement queries made to O_E so far after the challenge is issued. The adversary can issue more queries, both signing and endorsement, but for any endorsement query (c_j, W_j), the following must hold: $C \backslash (C_E^1 \cup C_E^2 \cup \{(c_j, W_j)\}) \neq \phi$, and for any signing query, the queried message is not the challenged message. Finally, the adversary halts and outputs an ordinary signature σ for message m. The adversary $\mathcal{A}$ is said to win this game if $\mathsf{VerS}_S(m, \sigma) = 1$.

Definition 3 *A CVS scheme is cheat-immune if the probability of winning the above game is negligible in the security parameter λ for all PPT adversaries.*

We show later that unforgeability and simulatability imply cheat immunity if the confirmation protocol is zero knowledge. Hence, proving that a CVS scheme is secure reduces to showing that it is unforgeable and simulatable and its confirmation protocol is zero knowledge.[5]

3 Relations between Security Notions

We discuss the relations between the security notions of a CVS scheme; the purpose is to find out whether one notion is implicitly implied in the other or they are exclusive, and under what conditions such an implication exists. With this knowledge, one could simply focus on a smaller set of security properties.

[5] This model is reasonable as the only restriction in practice is the signer should not give to the same party multiple partial signatures on the same message but with different verifiability conditions. First, the event in question is rare; otherwise, the restriction can be easily achieved by adding a serial number if the same message is signed multiple times.

3.1 Simulatability and Zero Knowledge Confirmation Protocol imply Non-transferability

Just like the simulatability property whose fulfillment hinges on the existence of a PPT simulator Fake, the fulfillment of the non-transferability property depends on the existence of a PPT transcript simulator FakeT. If we recall that in the zero knowledge definition, a zero knowledge confirmation protocol implies the existence of a PPT transcript simulator SimT which, on input a partial signature δ_t, outputs a transcript indistinguishable from a true one recorded during a run of the confirmation protocol on δ_t, one may be tempted to use SimT as an implementation for FakeT. At first glance, it seems to be a reasonable action. However, the indistinguishability between the real transcript and the simulated transcript in any zero-knowledge proof is based on the assumption that they come up from the same input and the claim to prove is true. If we use SimT to implement FakeT, the input to the simulator is no longer a genuine partial signature, thus violating this basic assumption. In fact, the distribution of the resulting transcript could be very different. Nevertheless, if the CVS scheme is simulatable, non-transferability could be achieved.

Theorem 1 *Given that a CVS scheme is simulatable with respect to a PPT partial signature simulator Fake, if its confirmation protocol $\mathsf{CVCon}_{(S,V)}$ is zero knowledge with respect to a PPT transcript simulator SimT, then it is non-transferable in the same attack model with adaptive queries as in the simulatability definition and SimT can be used as the transcript simulator FakeT for the output of Fake. That is, the following two distributions are indistinguishable for all S, m, C even with adaptive endorsement queries:*

$$\{\mathsf{CVSig}_S(m,C),\ \pi_{S,V}^{\mathsf{CVCon}}(m,C,\mathsf{CVSig}_S(m,C))\},\ \{\mathsf{Fake}_S(m,C),\ \pi^{\mathsf{FakeT}}(m,C,\mathsf{Fake}_S(m,C))\}$$

where $\pi_{S,V}^{\mathsf{CVCon}}(\cdot)$ and $\pi^{\mathsf{FakeT}}(\cdot)$ are transcript outputs of a real confirmation protocol run and FakeT respectively. (Proof in Appendix B.)

The practical significance of Theorem 1 is that it allows one to separate the designs of the CVS signing algorithm from that of the confirmation protocol, thus breaking down the design problem.

3.2 Ensuring Cheat-immunity

The following theorem allows one to ignore the cheat-immunity requirement when designing a CVS scheme.

Theorem 2 *An unforgeable and simulatable CVS scheme is also cheat-immune given its confirmation protocol is zero knowledge. (Proof in Appendix B.)*

4 The Existence of a Secure CVS Scheme

In this section, we give a generic CVS construction from IBE and show the equivalence between CVS and IBE.

4.1 A Generic Construction of CVS from IBE

We show how to construct a secure CVS scheme based on the following components whose details could be found in the appendix: (1) A secure signature $SIG = (SKG, Sig, Ver)$ which is existentially unforgeable against an adaptive chosen message attack [23]; (2) An IBE scheme $IBE = (Setup, Extract, Enc, Dec)$ with semantic security, that is, IND-ID-CPA [2]; (3) A computationally hiding commitment scheme $COM = (Com)$ [30, 13]; (4) A pseudorandom generator (PRG) [20, 25]. Let the plaintext and ciphertext spaces of IBE be $\mathcal{P}_{IBE}$ and $\mathcal{C}_{IBE}$ respectively. Let the message and

signature spaces of SIG be $\mathcal{M}$ (same as the message space of CVS) and $\mathcal{S}_\sigma$ (same as the ordinary signature space of CVS) respectively. Let $h : \{0,1\}^{l_p} \rightarrow \{0,1\}^{l_s}$ be a PRG where l_p and l_s are the length of an IBE plaintext and a SIG signature respectively. Let $\mathcal{C}_{COM}$ be the output space of the commitment scheme COM and $Com : \mathcal{P}_{IBE} \times \mathcal{S}_\sigma \rightarrow \mathcal{C}_{COM}$ be its committing function.

Depending on the number of witnesses, the IBE scheme is used multiple times with each witness W_i being a private key generator (PKG) for its IBE scheme. Assume there are N witnesses and the partial signature is: $\delta \in \mathcal{S}_\sigma \times \mathcal{C}_{IBE}^N \times \mathcal{C}_{COM}$. The generic CVS construction is as follows.

Key Generation. $\mathsf{CVKGS} \stackrel{\text{def}}{=} SKG$ for generating (Pk_S, sk_S) for the signer S. $\mathsf{CVKGW} \stackrel{\text{def}}{=} Setup$ for generating (PK_{W_i}, sk_{W_i}) for the witnesses W_i.

Partial Signature Generation. Given an input message $m \in \mathcal{M}$, a condition set $C = \{(c_i, W_i) : 1 \leq i \leq N\}$, a signing key sk_S, a signer's public key PK_S and the set of witness public keys $PK_C = \{PK_{W_i} : 1 \leq i \leq N\}$,

1. Generate an ordinary signature using the signing algorithm of SIG: $\sigma = Sig(m, sk_S)$
2. For each $(c_i, W_i) \in C$, pick a random $a_i \in \mathcal{P}_{IBE}$, $1 \leq i \leq N$ and the CVS signature is: $\delta = \left\langle \sigma \oplus h\left(\bigoplus_i^N a_i\right),\ \{Enc(PK_{W_i}, c_i, a_i) : 1 \leq i \leq N\},\ Com\left(\sigma, h\left(\bigoplus_i^N a_i\right)\right)\right\rangle$ where $Enc(PK_{W_i}, c_i, a_i)$ is the IBE ciphertext on message a_i using W_i (witness) as the PKG and c_i (condition statement) as the identity.[6]

Witness Signature Generation. $\mathsf{SigW}(c, sk_W) \stackrel{\text{def}}{=} Extract(c, sk_W)$. Taking the condition statement c as an identity, the witness W could extract the private key d_c^W corresponding to c. The private key d_c^W could be considered as a kind of signature on c as in [3].

Signature Extraction. Given a partial signature $\delta = \langle \alpha, \{\beta_i : 1 \leq i \leq N\}, \gamma\rangle$ and $\sigma_i = d_{c_i}^{W_i}$, $1 \leq i \leq N$,

1. For $1 \leq i \leq N$, get $a_i' = Dec(PK_{W_i}, \beta_i, \sigma_i)$.
2. Recover $\sigma' = \alpha \oplus h(\bigoplus_i^N a_i')$.
3. Check if $Com(\sigma', h(\bigoplus_i^N a_i')) \stackrel{?}{=} \gamma$. If not, output "fail", otherwise, σ' is the ordinary signature.

Signature Verification. $\mathsf{VerS} \stackrel{\text{def}}{=} Ver$.

Confirmation Protocol. Using general interactive zero-knowledge proofs [21] or concurrent zero-knowledge proofs [11], the signer with private input $a_1, \ldots, a_i, \ldots, a_N$ and σ and all the random coins used to generate β_i could convince the verifier that there exists $(\sigma, a_1, \ldots, a_i, \ldots, a_N)$ satisfying the following equations: $\delta = \langle \alpha, \{\beta_1, \beta_2, \ldots, \beta_i, \ldots, \beta_N\}, \gamma\rangle$; $\alpha = \sigma \oplus h\left(\bigoplus_i^N a_i\right)$; $\beta_i = Enc(PK_{W_i}, c_i, a_i), 1 \leq i \leq N$; $\gamma = Com\left(\sigma, h\left(\bigoplus_i^N a_i\right)\right)$; $Ver(m, \sigma, PK_S) = 1$. The common input to the confirmation protocol is PK_S, PK_{W_i} $(1 \leq i \leq N)$, m, $C = \{(c_i, W_i) : 1 \leq i \leq N\}$ and δ. Since verifying whether a given tuple $(\sigma, a_1, a_2, \ldots, a_i, \ldots a_N)$ satisfies the above equations is a poly-time predicate, a general zero-knowledge proof for it should exist.

Fake Signature Simulator — $\mathsf{Fake}(C) : C = \{(c_i, W_i) : 1 \leq i \leq N\}$

1. Randomly (uniformly) pick $\sigma_f \in \mathcal{S}_\sigma$.
2. Randomly pick $b_i \in \mathcal{P}_{IBE}$, for $1 \leq i \leq N$ and output the fake partial signature:

$$\delta_f = \left\langle \sigma_f \oplus h\left(\bigoplus_i^N b_i\right),\ \{Enc(PK_{W_i}, c_i, b_i) : 1 \leq i \leq N\},\ Com\left(\sigma_f, h\left(\bigoplus_i^N b_i\right)\right)\right\rangle$$

Obviously, this simulator is PPT.

The generic CVS construction from IBE is slight over-designed: The commitment scheme is generally not needed; it is mainly used to allow detection of failure in ordinary signature extraction which may occur when invalid witness signatures are used in ordinary signature extraction .

[6] For short, we may denote $Enc(PK_{W_i}, c_i, a_i)$ as $Enc_{W_i}(c_i, a_i)$ in the following discussion.

Security of the Generic CVS Construction The completeness of the above CVS construction is guaranteed by the correctness of the underlying IBE scheme. Besides, it is also perfectly convertible. The security of this CVS construction is best summarized with the following lemmas and theorem.

Lemma 3 *If SIG is existentially unforgeable under an adaptive chosen message attack, then the generic CVS construction is unforgeable. (Proof in Appendix C.)*

Lemma 4 *If IBE is IND-ID-CPA secure, COM is a computationally hiding commitment scheme, and h is a PRG, then the generic CVS construction is simulatable with respect to the simulator Fake. (Proof in Appendix C.)*

Theorem 5 *Given any semantically secure IBE scheme (under a chosen plaintext attack) and any existentially unforgeable signature scheme, together with a PRG and a computationally hiding commitment scheme, a secure CVS scheme can be constructed.*

4.2 A Generic Construction of IBE from CVS

We show how to construct a 1-bit IBE scheme with semantic security (i.e. IND-ID-CPA) using a CVS scheme. We assume the CVS scheme is simulatable with respect to a fake partial signature simulator Fake. Our construction is similar to that in the seminal work of probabilistic encryption by Goldwasser and Micali [22]. While they used the indistinguishability between the quadratic residues and non-residues in $\mathbb{Z}_n^*$ for some composite n (Quadratic Residuosity Problem) to encrypt a single bit, we leverage the indistinguishability between a true and a simulated (fake) partial signature of CVS to create a ciphertext. By repeating the operation of the 1-bit scheme k times as in [22], we could construct an IBE scheme for k-bit long messages. Now, we just need to focus on a 1-bit IBE scheme. We consider a CVS scheme with just a single witness $G \in \mathcal{W}$ which is used as the PKG for the IBE scheme. Suppose Fake is a PPT simulator for the CVS scheme. The IBE scheme works as follows.

Key Setup. The public and private keys of the witness G in the CVS scheme are used as the public and private keys of the PRG in the IBE scheme. We set $Setup \stackrel{\text{def}}{=} \mathsf{CVKGW}$ to generate the public/private key pair of the PRG: $\mathsf{CVKGW}(1^\lambda) \rightarrow (PK_G, sk_G)$.

Private Key Extraction. The identity ID_i of any user could be treated as a condition statement in the CVS scheme as they are both a bit string of arbitrary length. We set $Extract \stackrel{\text{def}}{=} \mathsf{SigW/CVEndW}$, then extracting the private key d_i for ID_i is the same as requesting an endorsement or signature on the statement ID_i: $\mathsf{SigW}(ID_i, sk_G) \rightarrow d_i$.

Encryption. The identity of a user i is the bit string ID_i (treated as a condition statement in the underlying CVS scheme) and its private key is the witness endorsement d_i obtained from G. We consider a 1-bit plaintext $b \in \{0, 1\}$. To encrypt, randomly pick a message $m \in \mathcal{M}$, run $\mathsf{CVKGS}(1^\lambda)$ to generate the public/private key pair (PK_S, sk_S) of the signer, the encryption function is then: $Enc(PK_G, ID_i, b) \rightarrow (m, \delta_b, PK_S)$, where

$$\delta_b = \begin{cases} \mathsf{CVSig}(m, ID_i, sk_S, PK_S, PK_G), & b = 0 \\ \mathsf{Fake}(m, ID_i, PK_S, PK_G), & b = 1 \end{cases}$$

That is, when $b = 0$, δ_b is a valid partial signature on m, whereas, when $b = 1$, δ_b is a fake one.

Decryption. Given an identity ID_i, a PKG public key PK_G and the user private key d_i, to decrypt a given ciphertext $C = (m', \delta', PK'_S)$, the decryption function $Dec(PK_G, C, d_i) \rightarrow b$ is implemented as follows: extract the ordinary signature from δ' using $\mathsf{CVExtract}(m', ID_i, \delta', PK'_S, d_i) \rightarrow \sigma'$, and the plaintext b' is given by the following[7]: $b' = 0$ if $\mathsf{VerS}(m', \sigma', PK'_S) = 1$ and 1 otherwise.

[7] The case in which CVExtract returns $\perp$ is covered by the "otherwise" part.

Correctness of the CVS-based IBE. The completeness of the CVS scheme guarantees the correctness of decryption in the above IBE scheme. The completeness property of the CVS scheme ensures that, if $\delta = \mathsf{CVSig}(m, ID_i, sk_S, PK_S, PK_G)$ and $d_i = \mathsf{CVEndW}(ID_i, sk_G)$, then the verification must return 1, that is, $\mathsf{VerS}(m, \mathsf{CVExtract}(m, ID_i, \delta, PK_S, d_i), PK_S) = 1$. The CVS scheme also guarantees that with negligible probability a valid ordinary signature on message m could be extracted form $\mathsf{Fake}(m, ID_i, PK_S, PK_G)$, otherwise, the CVS scheme would be forgeable. These together ensure that $Dec(PK_G, Enc(PK_G, ID_i, b), d_i) = b$ with probability almost 1.

Security of the CVS-based IBE. The security of above IBE construction is contained in the following:

Theorem 6 *The above IBE construction from CVS is semantically secure against a chosen plaintext attack (IND-ID-CPA). (Proof in Appendix D.)*

4.3 The Equivalence between CVS and IBE

A secure CVS scheme is equivalent to a secure IBE scheme in terms of existence, summarized below.

Theorem 7 *A secure conditionally verifiable signature (CVS) scheme (unforgeable, simulatable, with zero knowledge confirmation protocol) exists if and only if an IND-ID-CPA secure IBE scheme exists.*

Proof. The **only if part** follows directly from the CVS-based IBE construction given above. For the **if part**, we assume the existence of a IND-ID-CPA secure IBE. Then a one-way function exists (We could use $Setup$ of the IBE scheme to construct a one-way function.), which implies the existence of an ordinary signature scheme existentially unforgeable under an adaptive chosen message attack[33, 31]. Besides, a PRG exists as Impagliazzo et. al. [25] showed how to construct a PRG from any one-way function. The existence of a PRG further implies the existence of a computationally hiding multi-bit commitment function[30]. Finally, the existence of a one-way function also implies the existence of zero-knowledge proofs. By Theorem 5, we could use the generic construction to build a secure CVS scheme which is unforgeable and simulatable and a zero knowledge proof for its confirmation protocol exists. Hence, the existence of a secure IBE scheme implies the existence of a secure CVS scheme.

We should mention that we showed in Theorem 7 that a weaker notion of IBE, namely, one with IND-ID-CPA security, is necessary and sufficient for the construction of a secure CVS scheme. It is thus fair to say that CVS could be constructed based on weaker assumptions than IBE with the standard IND-ID-CCA security [2].

5 Conclusions

In this paper, we introduce a new signature concept called CVS which could provide effective solutions in many digital business scenarios, in particular, those involving mutually distrusting parties. We demonstrate its feasibility by giving a generic construction using IBE and show that it is equivalent to CPA-secure IBE. The result showing the equivalence between CPA-secure IBE and CVS could imply that CVS can be constructed based on weaker computational assumptions compared with IBE which should usually be CCA-secure. One open problem is whether CVS can be constructed from primitives other than IBE.

References

1. N. Asokan, V. Shoup, and M. Waidner. Optimistic fair exchange of digital signatures. *IEEE Journal on Selected Areas in Communication*, 18(4):591–610, April 2000.
2. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In *Advances in Cryptology — CRYPTO 2001, Springer-Verlag LNCS vol. 2139*, pages 213–229, 2001.

3. D. Boneh, B. Lynn, and H. Shacham. Short signatures from Weil pairing. In *Advances in Cryptology — Asiacrypt 2001, Springer-Verlag LNCS vol. 2248*, pages 514–532, 2001.
4. J. Boyar, D. Chaum, I. Damgård, and T. Pedersen. Convertible undeniable signatures. In *Advances in Cryptology — CRYPTO 1990, Springer-Verlag LNCS vol. 537*, pages 189–205, 1991.
5. J. Camenisch and M. Michels. Confirmer signature schemes secure against adaptive adversaries. In *Advances in Cryptology — EUROCRYPT 2000, Springer-Verlag LNCS vol. 1870*, pages 243–258, 2000.
6. D. Chaum. Zero-knowledge undeniable signatures. In *Advances in Cryptology — EUROCRYPT 90, Springer-Verlag LNCS vol. 473*, pages 458–464, 1990.
7. D. Chaum. Designated confirmer signatures. In *Advances in Cryptology — EUROCRYPT 1994, Springer-Verlag LNCS vol. 950*, pages 86–91, 1995.
8. D. Chaum and H. van Antwerpen. Undeniable signatures. In *Advances in Cryptology — CRYPTO 1989, Springer-Verlag LNCS vol. 435*, pages 212–216, 1989.
9. D. Chaum, H. van Antwerpen, and B. Pfitzmann. Cryptographically strong undeniable signatures, unconditionally secure for the signer. In *Advances in Cryptology — CRYPTO 1991, Springer-Verlag LNCS vol. 576*, pages 470–484, 1992.
10. L. Chen, C Kudla, and K. G. Paterson. Concurrent signatures. In *Advances in Cryptology — EUROCRYPT 2004, Springer-Verlag LNCS vol. 3027*, pages 287–305, 2004.
11. I. Damgård. Efficient concurrent zero-knowledge in the auxiliary string model. In *Advances in Cryptology — EUROCRYPT 2000, Springer-Verlag LNCS vol. 1807*, pages 418–430, 2000.
12. I. Damgård and T. Pedersen. New convertible undeniable signature schemes. In *Advances in Cryptology — EUROCRYPT 1996, Springer-Verlag LNCS vol. 1070*, pages 372–386, 1996.
13. I. Damgård, B. Piftzmann, and T. Pedersen. Statistical secrecy and multi-bit commitments. *IEEE Transaction on Information Theory*, 44:1143–1151, 1998.
14. M. Franklin and M. Reiter. Verifiable signature sharing. In *Advances in Cryptology — EUROCRYPT 1995, Springer-Verlag LNCS vol. 921*, pages 50–63, 1995.
15. S. Galbraith and W. Mao. Invisibility and anonymity of undeniable and confirmer signatures. In *Cryptographers' Track RSA Conference (CT-RSA 2003), Springer-Verlag LNCS vol. 2612*, pages 80–97, 2003.
16. S. D. Galbraith, W. Mao, and K. G. Paterson. RSA-based undeniable signatures for general moduli. In *Cryptographers' Track RSA Conference (CT-RSA 2002), Springer-Verlag LNCS vol. 2271*, pages 200–217, 2002.
17. J. Garay and M. Jakobsson. Timed release of standard digital signatures. In *Financial Cryptography (FC 2002), Springer-Verlag LNCS vol. 2357*, pages 168–182, 2002.
18. J. Garay and C. Pomerance. Timed fair exchange of standard signatures. In *Financial Cryptography (FC 2003), Springer-Verlag LNCS vol. 2742*, pages 190–203, 2003.
19. R. Gennaro, H. Krawczyk, and T. Rabin. RSA-based undeniable signatures. In *Advances in Cryptology — CRYPTO 1997, Springer-Verlag LNCS vol. 1294*, pages 397–416, 1997.
20. O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *Journal of ACM*, 33(4):792–807, 1986.
21. O. Goldreich, S. Micali, and A. Wigderson. How to prove all NP-statements in zero-knowledge, and a methodolgy of cryptographic protocol design. In *Advances in Cryptology — CRYPTO 1986, Springer-Verlag LNCS vol. 263*, pages 171–185, 1986.
22. S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.
23. S. Goldwasser, S. Micali, and R. Rivest. A secure signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.
24. S. Goldwasser and E. Waisbard. Transformation of digital signature schemes into designated confirmer signature schemes. In *Theory of Cryptography Conference (TCC 2004), Springer-Verlag LNCS vol. 2951*, pages 77–100, 2004.
25. I. Impagliazzo, L. Levin, and M. Luby. Pseudo-random generation from one-way functions. In *ACM Symposium on Theory of Computing (STOC 1989)*, 1989.
26. M. Jakobsson, K. Sako, and R. Impagliazzo. Designated verifier proofs and their applications. In *Advances in Cryptology — EUROCRYPT 1996, Springer-Verlag LNCS vol. 1070*, pages 143–154, 1996.
27. B. Libert and J. J. Quisquater. Identity based undeniable signatures. In *Cryptographers' Track RSA Conference (CT-RSA 2004), Springer-Verlag LNCS vol. 2964*, pages 112–125, 2004.
28. M. Michels and M. Stadler. Efficient convertible undeniable signature schemes. In *Proceedings International Workshop on Selected Area of Cryptography (SAC'97)*, pages 231–244, 1997.
29. M. Michels and M. Stadler. Generic constructions for secure and efficient confirmer signature schemes. In *Advances in Cryptology — EUROCRYPT 1998, Springer-Verlag LNCS vol. 1403*, pages 402–421, 1998.
30. M. Naor. Bit commitment using pseudo-randomness. In *Advances in Cryptology — CRYPTO 1989, Springer-Verlag LNCS vol. 435*, pages 128–136, 1990.
31. M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *ACM Symposium on Theory of Computing (STOC 1989)*, pages 33–43, 1989.
32. T. Okamoto. Designated confirmer signatures and public-key encryption are equivalent. In *Advances in Cryptology — CRYPTO 1994, Springer-Verlag LNCS vol. 839*, pages 61–74, 1994.
33. J. Rompel. One-way functions are necessary and sufficient for secure signature. In *Proceedings 22ndACM Symposium on Theory of Computing (STOC 1990)*, pages 387–394, 1990.

34. W. Susilo, F. Zhang, and Y. Mu. Identity-based strong designated verifier signature schemes. In *ACISP2004, Springer-Verlag LNCS vol. 3108*, pages 313–324, 2004.

Appendix A: Basic Primitives for the Generic CVS Construction

Identity Based Encryption

We use similar notations as in [2] for identity based encryption. A standard IBE scheme $IBE = \{Setup, Extract, Enc, Dec\}$ consists of a private key generator (PKG) and a number of users, and is made up of four algorithms:

$Setup(1^\lambda) \rightarrow (PK_G, sk_G)$**:** the key setup algorithm which outputs a public/private key pair (PK_G, sk_G) for the PKG.

$Extract(ID, sk_G) \rightarrow d_{ID}$**:** the private key extraction algorithm run by PKG which outputs a private key d_{ID} for the identity ID.

$Enc(PK_G, ID, M) \rightarrow C$**:** the encryption algorithm taking an identity ID and a message m to output the ciphertext C.

$Dec(PK_G, C, d_{ID}) \rightarrow M$**:** the decryption algorithm taking a ciphertext C and a private key d_{ID} to output the plaintext M.

Note that, unlike the description in [2], we incorporate all the public parameters in the PKG public key PK_G, and this public key is needed in all encryption and decryption.

Security of IBE. In [2], Boneh and Franklin considered the strongest security notion for IBE, namely semantic security or indistinguishability against an adaptive chosen ciphertext attack (IND-ID-CCA). Although chosen-ciphertext security is the standard acceptable notion for encryption schemes, we only consider a weaker notion — semantic security against a chosen plaintext attack (IND-ID-CPA) or semantic security for short — which is sufficient for our generic construction of CVS. An IBE is semantically secure if no PPT adversary $\mathcal{A}$ could win the following game with a non-negligible advantage: The challenger runs $Setup$ to generate a PKG public/private key pair (PK_G, sk_G), and gives the public key PK_G to the adversary but keeps the private/master key sk_G. The adversary could issue to the challenger one type of queries: (1) Extraction Query $\langle ID_j \rangle$. The challenger responds by running $Extract$ on ID_j to generate the corresponding private key $d_j = d_{ID_j}$ and gives it to the adversary. Once the adversary decides that the first query phase is over it outputs two plaintexts M_0, M_1 and an identity ID to be challenged. The only constraint is that ID did not appear in any of the previous extraction queries, that is, $ID \neq ID_j, \forall j$. The challenger flips a coin $b \in \{0, 1\}$, set $C = Enc(PK_G, ID, M_b)$ and sends C to the adversary. The adversary is allowed to make more queries as previously done but no query can be made on the challenged ID. Finally, the adversary outputs a guess $b' \in \{0, 1\}$ for b. The adversary wins the game if $b' = b$. The advantage of the adversary is defined as:$Adv_{\mathcal{A}}^{IBE} = \left|Pr[b' = b] - \frac{1}{2}\right|$

Signatures

A signature scheme $SIG = \{SKG, Sig, Ver\}$ consists of three algorithms:

$SKG(1^\lambda) \rightarrow (PK_S, sk_S)$**:** the key generator which generates the public/private key pair (PK_S, sk_S) for a signer S.

$Sig(m, sk_S) \rightarrow \sigma$**:** the signing algorithm taking a message m and a private key sk_S to output a signature σ on m.

$Ver(m, \sigma, PK_S) \rightarrow v \in \{0, 1\}$**:** the signature verification algorithm taking a message m a signature σ and a public key PK_S to check whether σ is a valid signature of S on m. If it is, Ver outputs 1, otherwise, 0.

Security of SIG. A signature scheme is considered secure if the probability of successful existential forgery is negligible even under chosen message attacks. In detail, this means the following: An adversary $\mathcal{A}$ is allowed to make oracle access adaptively to obtain signatures of a targeted signer S on any message

m_j of his choice; he could make a query based on the results of the previous queries. Finally, $\mathcal{A}$ has to output a message-signature pair (m, σ). The probability that the signature is a valid one for the message (i.e. $Ver(m, \sigma, PK_S) = 1$) and the message has not be queried before (i.e $m \neq m_j$, $\forall j$) should be negligible for all PPT $\mathcal{A}$.

Pseudorandom Generators

Assume $l(n) > n$. Let $x \leftarrow X$ denote that x is uniformly sampled from X. $h : \{0,1\}^n \rightarrow \{0,1\}^{l(n)}$ is a PRG [20] if the following is negligible in n for all PPT distinguisher D:
$\big|Pr[y \leftarrow \{0,1\}^{l(n)} : D(y) = 1] - Pr[s \leftarrow \{0,1\}^n : D(h(s)) = 1]\big|$. This in essence means that h take a seed s to generate a string $h(s)$ of longer length $l(n)$ and nobody could distinguish $h(s)$ from a uniformly sampled string from $\{0,1\}^{l(n)}$.

Commitments

We adopt the multi-bit commitment definitions [30, 13] instead of the common single-bit commitment [30]. The core of a cryptographic commitment scheme is the committing algorithm $Com(s, m) \rightarrow c$ on input a message m and a randomly chosen salt s outputting a commitment c. By revealing s and m, one can check whether a commitment c is properly formed. A commitment scheme should satisfy the following properties:

Binding. Let λ be the security parameter, then the following is negligible (computationally binding) or zero (perfectly binding) for all PPT algorithm A:

$$Pr[(s, m, s', m') \leftarrow \{A(1^\lambda)\} : Com(s, m) = Com(s', m')]$$

Hiding. For all $m, m' \in \{0,1\}^*, m \neq m'$, the following is negligible (computationally hiding) or zero (perfectly hiding) for all PPT distinguisher D:

$$\big|Pr[s \leftarrow \{0,1\}^*; c \leftarrow Com(s, m) : D(c) = 1] - Pr[s' \leftarrow \{0,1\}^*; c \leftarrow Com(s', m') : D(c') = 1]\big|$$

The binding property essentially means that, once a message m is committed in c, nobody could change its value without being detected. In perfectly hiding schemes, the distribution of the commitments for different messages should be identical. Note that we use a different definition for the hiding property than that of the multi-bit scheme in [30] which states that, for a message $m = b_1 b_2 \dots b_n$ ($b_i \in \{0,1\}, 1 \leq i \leq n$), given a commitment on m, nobody could guess any bit b_i correctly with a probability greater than $\frac{1}{2} + \varepsilon(\lambda)$ (where $\epsilon(\lambda)$ is a negligible function in λ), even when told $b_1, b_2, \dots, b_{i-1}, b_{i+1}, \dots, b_n$. However, it could be shown that the two definitions are equivalent.

Appendix B: Proofs — Relations between Security Notions

Proof: Simulatability implies Non-transferability

Proof of Theorem 1.

We prove by contradiction. We assume that the CVS scheme is simulatable with respect to a PPT simulator Fake, that is, the advantage $Adv_{\mathcal{D}}^{Sim}$ for breaking the simulatability with respect to Fake is negligible for all PPT adversaries $\mathcal{D}$. Assume we use the transcript simulator SimT of the zero knowledge proof for the confirmation protocol as the transcript simulator FakeT for the fake signature. Suppose there is a PPT distinguisher D which could break the non-transferability property with respect to Fake and FakeT with non-negligible advantage Adv_D^{NT}, then we can construct D' to break the simulatability property as follows:

$D'(\delta_b)$ where δ_b is a genuine/fake partial signature when $b = 0/1$

Setup.
Ask its challenger for the public keys of the signer and the witnesses
Run D on the same set of public keys.
Get the signer's private key from its challenger and pass it to D.
Query.
Answer all the signing queries itself.
Pass all the endorsement queries from D to its oracle and relay the results back to D.
Challenge.
D outputs (m, C) it wish to be challenged.
Output (m, C) as its challenge request and receive a challenge δ_b.
Compute $\pi^{\mathsf{SimT}}(\delta_b)$ and pass $(\delta_b, \pi^{\mathsf{SimT}}(\delta_b))$ as a challenge to D.
Guess.
D outputs b' as a guess for b. Output b'.

The query responses are perfectly simulated; the view of D in the simulated environment is identical to its view in a real attack. Let $\delta_t = \mathsf{CVSig}_S(m, C)$ and $\delta_f = \mathsf{Fake}_S(m, C)$. When $b = 1$, the challenge is a fake partial signature δ_f and $\pi^{\mathsf{SimT}}(\delta_b) = \pi^{\mathsf{FakeT}}(\delta_f)$, and the input to D is $(\delta_f, \pi^{\mathsf{FakeT}}(\delta_f))$. Whereas, when $b = 0$, the challenge is a true partial signature δ_t and $\pi^{\mathsf{SimT}}(\delta_b) = \pi^{\mathsf{SimT}}(\delta_t)$, and the input to D is $(\delta_t, \pi^{\mathsf{SimT}}(\delta_t))$. Due to the zero knowledge property of the confirmation protocol, $(\delta_t, \pi^{\mathsf{SimT}}(\delta_t))$ could perfectly simulate $(\delta_t, \pi^{\mathsf{CVCon}}_{S,V}(\delta_t))$, a valid challenge to D. As a result, the challenge to D is perfectly simulated no matter $b = 0$ or $b = 1$. It could be seen that $Adv^{Sim}_{D'} = Adv^{NT}_{D}$ which is non-negligible if D can break the non-transferability property. This concludes the reduction.

Instead of stating the zero knowledge property informally as above, a more rigorous treatment is possible by evaluating the probability of success of D and D' respectively.
The probability of success of D' with respect to the simulatability game is given by:

$$\begin{aligned} Pr^{Sim}_{D'}[Success] &= \tfrac{1}{2}\left(Pr[D'(\delta_t) = 0] + Pr[D'(\delta_f) = 1]\right) \\ &= \tfrac{1}{2}(Pr[D(\delta_t, \pi^{\mathsf{SimT}}(\delta_t)) = 0] + Pr[D(\delta_f, \pi^{\mathsf{FakeT}}(\delta_f)) = 1]) \end{aligned}$$

The probability of success of D with respect to the non-transferability game is given by:

$$\begin{aligned} Pr^{NT}_{D}[Success] &= \tfrac{1}{2}(Pr[D(\delta_t, \pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 0] + Pr[D(\delta_f, \pi^{\mathsf{FakeT}}(\delta_f)) = 1]) \\ &= \tfrac{1}{2}(Pr[D(\delta_t, \pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 0] - Pr[D(\delta_t, \pi^{\mathsf{SimT}}(\delta_t)) = 0] \\ &\quad + Pr[D(\delta_t, \pi^{\mathsf{SimT}}(\delta_t)) = 0] + Pr[D(\delta_f, \pi^{\mathsf{FakeT}}(\delta_f)) = 1]) \\ &= Pr^{Sim}_{D'}[Success] + \tfrac{1}{2}(Pr[D(\delta_t, \pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 0] - Pr[D(\delta_t, \pi^{\mathsf{SimT}}(\delta_t)) = 0]) \end{aligned}$$

Taking absolute values on both sides,

$$\begin{aligned} Adv^{NT}_{D} &\leq Adv^{Sim}_{D'} + \tfrac{1}{2}|Pr[D(\delta_t, \pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 0] - Pr[D(\delta_t, \pi^{\mathsf{SimT}}(\delta_t)) = 0]| \\ &= Adv^{Sim}_{D'} + \tfrac{1}{2}|Pr[D(\pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 1] - Pr[D(\pi^{\mathsf{SimT}}(\delta_t)) = 1]| \end{aligned}$$

Due to the zero knowledge property, that is, $\{\pi^{\mathsf{CVCon}}_{S,V}(\delta_t)\} \cong \{\pi^{\mathsf{SimT}}(\delta_t)\}$, which actually means that $|Pr[\mathcal{D}(\pi^{\mathsf{CVCon}}_{S,V}(\delta_t)) = 1] - Pr[\mathcal{D}(\pi^{\mathsf{SimT}}(\delta_t)) = 1]|$ is negligible in the security parameter λ for all PPT $\mathcal{D}$. As a result, $Adv^{NT}_{D} \leq Adv^{Sim}_{D'}$ up to a negligible term (in λ). If Adv^{NT}_{D} is non-negligible, then $Adv^{Sim}_{D'}$ must also be non-negligible, which is a contradiction as we assume $Adv^{Sim}_{\mathcal{D}}$ is negligible in λ for all PPT $\mathcal{D}$ (the simulatability property). In other words, simulatability implies non-transferability if the confirmation protocol is zero knowledge.

Proof: Simulatability and Unforgeability imply Cheat-immunity

Theorem 2. An unforgeable and simulatable CVS scheme is also cheat-immune given its confirmation protocol is zero knowledge.

Proof of Theorem 2.

Assume the given CVS scheme is unforgeable and simulatable with respect to a PPT simulator $\mathsf{Fake}_S(m, C)$. Let $\mathsf{SimT}(\delta)$ be the transcript simulator of the zero knowledge proof used for the confirmation protocol where δ is a partial signature.

In the cheat-immunity game defined in the paper, an adversary is always given a valid partial signature as a challenge. In the following proof, we force an adversary, capable to win the cheat-immunity game with non-negligible probability, to run on a challenge which is not a valid partial signature but a fake one from the simulator Fake. Since the adversary is just an algorithm, it is thus definitely possible to run it on a deviated input. Of course, it is likely that the adversary would not output the desired result on the deviated input, but this is what we want to show.

In order to run the adversary on a deviated input, we modify the definition of the cheat-immunity game slightly, namely, in the challenge phase, no confirmation protocol would be run between the challenger and the adversary, but instead the adversary is given a challenged partial signature and a transcript of a confirmation protocol run on that partial signature. Note that a run of the interactive confirmation protocol is replaced by a transcript without any interaction. We argue that the proof obtained in this amended model also applies to the original model of cheat-immunity if the confirmation protocol is zero knowledge. The justification is as follows:

> If the confirmation protocol of a CVS scheme is zero-knowledge, the only information obtainable from running the confirmation protocol is whether a given partial signature is true/valid. Hence, the only difference between the information obtainable from a given partial signature and the transcript recorded during the confirmation protocol run on it and the information obtainable from a given partial signature and a simulated transcript of the confirmation protocol is the validity of the given partial signature and nothing else. In other words, if an adversary can extract the ordinary signature from a valid partial signature after running the confirmation protocol on it, it should also be able to do so with almost the same computational effort even without running the confirmation protocol. Consequently, we would neglect running the confirmation protocol in the challenge phase to force as adversary to run on an invalid partial signature. In fact, if we insist on running the confirmation protocol between the adversary and the challenger in the challenge phase, it is still possible (even though inefficient) using the rewinding technique commonly found in the transcript simulator of any zero knowledge proof, as it is used in [24]. In order to make an adversary accept a partial signature input and run on it, in each round of iteration of the confirmation protocol, we prepare the answer of some of all the possible challenged questions. If the challenge question comes out to be what has been prepared, then this round is successful; otherwise, we reset the adversary to the start of the current iteration round and restart this round again. As mentioned before, this rewinding is possible because the adversary is just another algorithm or Turing machine we use as a subroutine. Of course, we have to take more computations to complete an iteration round now but in most zero knowledge proofs, the overall computation would still remain polynomial time.

Now we can describe the proof. Suppose there exists a PPT adversary A which can win the cheat-immunity game with non-negligible probability p_A^{CI}. We show how to construct a distinguisher D from A for the simulatability game, which can distinguish a true partial signature (CVSig) from a fake one generated by Fake.

$D(\delta_b)$: δ_b is a true/fake partial signature when $b = 0/1$

Setup.
Get from its challenger the public keys of the signer and witnesses, and pass them to A.
Run A on the same set of public keys.
Keep the signer private key if given one.
Query.
Pass all signing and endorsement queries from A to its oracles and return the results to A.
For signing queries, run the confirmation protocol as an agent in between A and the challenger.
Challenge.
A outputs (m, C), $m \in \mathcal{M}$, $C \subset \mathcal{C} \times \mathcal{W}$, to be challenged.
Pass (m, C) to its challenger and receive the challenge δ_b.
Compute the confirmation transcript $\pi(\delta_b) = \mathsf{SimT}(\delta_b)$ for δ_b.
Pass $(\delta_b, \pi(\delta_b))$ as a challenge to A.
Guess.
A outputs σ. Output guess b' where:

$$b' = \begin{cases} 0, \mathsf{VerS}(m, \sigma) = 1 \\ 1, otherwise \end{cases}$$

First, it can be seen that D is PPT if A and VerS are both PPT.
The probability of success of D with respect to the simulatability game is:[8]

$$\begin{aligned} Pr_D^{Sim}[success] &= Pr[b' = b|\delta_b] \\ &= \tfrac{1}{2}Pr[b' = 0|\delta_0] + \tfrac{1}{2}Pr[b' = 1|\delta_1] \\ &= \tfrac{1}{2}Pr[\delta_0 \leftarrow \{\mathsf{CVSig}_S(m, C)\}; \sigma \leftarrow \{A(\delta_0)\} : \mathsf{VerS}(m, \sigma) = 1] \\ &\quad + \tfrac{1}{2}Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 0] \\ &= \tfrac{1}{2}p_A^{CI} + \tfrac{1}{2} - \tfrac{1}{2}Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 1]. \end{aligned}$$

Note we use the fact: $p_A^{CI} = Pr[\delta_0 \leftarrow \{\mathsf{CVSig}_S(m, C)\}; \sigma \leftarrow \{A(\delta_0)\} : \mathsf{VerS}(m, \sigma) = 1]$. Rearranging terms, we have:

$$\tfrac{1}{2}p_A^{CI} = \left(Pr_D^{Sim}[success] - \tfrac{1}{2}\right) + \tfrac{1}{2}Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 1]$$

Taking absolute values on both sides and denoting $Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 1]$ by ε_f, we have:

$$\begin{aligned} \tfrac{1}{2}p_A^{CI} &\leq \left|Pr_D^{Sim}[success] - \tfrac{1}{2}\right| + \left|\tfrac{1}{2}Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 1]\right| \\ &= Adv_D^{Sim} + \tfrac{1}{2}Pr[\delta_1 \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta_1)\} : \mathsf{VerS}(m, \sigma) = 1] \end{aligned}$$

$$p_A^{CI} \leq 2Adv_D^{Sim} + \varepsilon_f. \tag{1}$$

If p_A^{CI} is non-negligible, then either Adv_D^{Sim} or ε_f is non-negligible. We consider the following two cases:

[8] For the sake of simple notations, we tend to use short notations for the probability in question. For example, we just write $Pr[b' = b|\delta_b]$ to denote the probability that the guess of D, that is, b' is the same as the challenged bit b given δ_b which could be generated from CVSig (if $b = 0$) or Fake (if $b = 1$). We also neglect the preamble like public key generation. Formally, this probability should be written as:

$$Pr\left[\begin{array}{l} (PK_S, sk_S) \leftarrow \{CVKGS(1^\lambda)\}; (PK_W, sk_W) \leftarrow \{CVKGW(1^\lambda), \forall W; \\ m \leftarrow \mathcal{M}; C \leftarrow 2^{\mathcal{C}\times\mathcal{W}}; b \leftarrow \{0,1\}; \delta_b \leftarrow \begin{cases} \{\mathsf{CVSig}_S(m, C)\}, & b = 0 \\ \{\mathsf{Fake}_S(m, C)\}, & b = 1 \end{cases}; \\ \sigma \leftarrow \{A(\delta_b)\}; b' = \neg(\mathsf{VerS}(m, \sigma) = 1\}) \end{array} : b' = b\right]$$

Case 1 — Adv_D^{Sim} is non-negligible. Obviously, the existence of such a PPT algorithm D would break the simulatability property, which is a contradiction as we assume the CVS scheme is simulatable.

Case 2 — ε_f is non-negligible. We argue that if ε_f is non-negligible, then we could use A to create an existential forgery as follows.

F
Setup.
Get all the public keys of the signer and witnesses.
Run A on the same set of public keys.
Keep the witness private keys.[9]
Query.
Pass all signing queries to its oracle and relay the results back to A.
Run the confirmation protocol as an agent in between A and the challenger.
Answer all endorsement queries itself using the witness private keys.
Challenge.
A outputs (m, C), $m \in \mathcal{M}$, $C \subset \mathcal{C} \times \mathcal{W}$, to be challenged.
Create $\delta = \mathsf{Fake}_S(m, C)$, and compute the confirmation transcript $\pi(\delta) = \mathsf{SimT}(\delta)$ for δ.
Pass $(\delta_b, \pi(\delta_b))$ as a challenge to A.
Guess.
Output the final output σ of A as a forgery output.

Obviously, if A is PPT, then F is also PPT as Fake is PPT. As m is chosen to be not queried before, the probability of successful existential forgery by F is then given by:

$$p_F^{UF} = Pr[\delta \leftarrow \{\mathsf{Fake}_S(m, C)\}; \sigma \leftarrow \{A(\delta)\} : \mathsf{VerS}(m, \sigma) = 1]$$

Note that p_F^{UF} should be equal to ε_f which is non-negligible. This concludes that the given CVS scheme is existentially forgeable if ε_f is non-negligible, which is a contradiction as we assume the CVS scheme is unforgeable.

In conclusion, if the given CVS scheme is simulatable (i.e. $Adv_{\mathcal{D}}^{Sim}$ is negligible for all PPT $\mathcal{D}$) and unforgeable (i.e. $p_{\mathcal{F}}^{UF}$ is negligible for all PPT $\mathcal{F}$), then it is also cheat-immune with negligible $p_{\mathcal{A}}^{CI}$ for all PPT $\mathcal{A}$.

Appendix C: Proofs — The Security of the Generic CVS Construction

Security of the Generic CVS Construction from IBE

Lemma 3. If the underlying ordinary signature scheme SIG is existentially unforgeable under a chosen message attack, then the generic CVS construction is unforgeable.

Proof of Lemma 3.

We prove the unforgeability property of the generic construction by contradiction. Assume SIG is existentially unforgeable under chosen message attacks. Suppose there is a PPT forging algorithm $\mathcal{F}$ which can forge a CVS partial signature with probability of success $p_{\mathcal{F}}^{CVS}$. We show how to construct another forging algorithm $\mathcal{F}'$ from $\mathcal{F}$ to forge a signature for SIG.

$\mathcal{F}'$

Setup.
Ask its challenger for the signer public key PK_S.
Run $Setup$ to get all the witness public/private key pairs $(PK_{W_i}, sk_{W_i}),\ 1 \leq i \leq N$.
Run $\mathcal{F}$ on PK_S and (PK_{W_i}, sk_{W_i}).
Query.
When $\mathcal{F}$ issues a O_S query for $\langle m_j, C_j \rangle$ where $C_j = \{(c_{ji}, W_{ji}) : 1 \leq i \leq N\}$,
ask its signing orale for an ordinary signature $\sigma_j = Sig(sk_s, m_j)$.
Randomly choose a_{ji} ($1 \leq i \leq N$) to create a partial signature:

$$\delta_j = \left\langle \sigma_j \oplus h\left(\bigoplus_i^N a_{ji}\right), \left\{Enc(PK_{W_{ji}}, c_{ji}, a_{ji})\right\}, Com\left(\sigma_j, h\left(\bigoplus_i^N a_{ji}\right)\right)\right\rangle$$

With a_{ji}'s, σ_j, and all random coins used, run the confirmation protocol with $\mathcal{F}$.
Guess.
$\mathcal{F}$ outputs a guess (m, σ). Output (m, σ).

Obviously, if F is PPT, then F' is also PPT (as Enc and Com are also PPT). Note that $\mathcal{F}$ should output $m \neq m_j,\ \forall j$ The probability of success of $\mathcal{F}'$ is:

$$p_{\mathcal{F}'}^{SIG} = Pr[Ver(m, \sigma, PK_S) = 1] = p_{\mathcal{F}}^{CVS}$$

If the CVS scheme is forgeable, that is, $p_{\mathcal{F}}^{CVS}$ is non-negligible, then $p_{\mathcal{F}'}^{SIG}$ is also non-negligible (a contradiction). Hence, if SIG is unforgeable in the sense that p_A^{SIG} is negligible for all PPT A, then so is the CVS scheme given by the generic construction.

Lemma 4. Given a pseudorandom generator and a computationally hiding commitment scheme, if the underlying IBE scheme is semantic secure, then the generic CVS construction is simulatable with respect to the given simulator Fake.

Proof of Lemma 4.
It is easy to show that the given CVS scheme with one witness is secure, then a CVS scheme with many witnesses is also secure. Hence, we will consider a single witness case. The details of the multiple-witness case could be found at the end of this section.

Assume IBE is IND-ID-CPA secure, h is a pseudorandom generator, and COM is computationally hiding. Suppose $\mathcal{D}$ is a PPT distinguisher which has non-negligible advantage $Adv_{\mathcal{D}}^{Sim}$ in winning the simulatability game associated with Definition 2. We can base on $\mathcal{D}$ to construct another distinguisher $\mathcal{D}'$ to break the semantic security of IBE.

To avoid confusion, we should clarify that in the following discussion, we denote the challenge ciphertext of the IBE game by $C_b, b \in \{0, 1\}$ and the queried verifiability condition set by C_j.

$\mathcal{D}'(C_b), \quad b \in \{0,1\}$

Setup.
Ask its challenger for the public key PK_G of the PKG. Use it as the witness public key for W.
Run CVKGS to generate the signer public/private key pair (PK_S, sk_S).
Run $\mathcal{D}$ on PK_G and (PK_S, sk_S).
Query.
Signing Queries (O_S) on $\langle m_j, C_j\rangle$ where $C_j = (c_j, W)$.
- Generate $\sigma_j = Sig(m_j, sk_S)$
- Randomly pick a_j and encrypts itself to generate the partial signature:
  $\delta_j = \langle \sigma_j \oplus h(a_j),\ Enc(PK_G, c_j, a_j),\ Com(\sigma_j, h(a_j))\rangle$
- Based on all the random coins used, run the confirmation protocol with $\mathcal{D}$.

Endorsement Queries (O_E) on (c_j, W).
- Pass all endorsement queries (c_j, W) from $\mathcal{D}$ as extraction queries on c_j to its oracle to get d_j.
- d_j is equivalent to $\sigma_W(c_j)$.

Challenge.
$\mathcal{D}$ outputs m and (c, W) to ask for a challenge.
Create a signature σ_t on a message m using Sig.
Randomly pick $\sigma_f \in \mathcal{S}_\sigma$.
Randomly pick $a_t, a_f \in \mathcal{P}_{IBE}$. Output a_t and a_f to ask for a challenge C_b where

$$C_b = \begin{cases} Enc(PK_G, c, a_t),\ b = 0 \\ Enc(PK_G, c, a_f),\ b = 1. \end{cases}$$

Flip a coin $e \in \{0,1\}$ and send the following challenge to $\mathcal{D}$:

$$\delta_e = \begin{cases} \langle \sigma_t \oplus h(a_t),\ C_b,\ Com(\sigma_t, h(a_t))\rangle, & e = 0 \\ \langle \sigma_f \oplus h(a_f),\ C_b,\ Com(\sigma_f, h(a_f))\rangle, & e = 1 \end{cases}$$

Guess. $\mathcal{D}$ outputs a guess b'. Output b' as a guess for b.

Note: $\langle \sigma_t \oplus h(a_t), Enc(PK_G, c, a_t), Com(\sigma_t, h(a_t))\rangle$ is equivalent to $\mathsf{CVSig}_S(m, C)$ and $\langle \sigma_f \oplus h(a_f), Enc(PK_G, c, a_f), Com(\sigma_f, h(a_f))\rangle$ is equivalent to $\mathsf{Fake}(C)$.

Obviously, if $\mathcal{D}$ is PPT, so is $\mathcal{D}'$ (assuming Enc, h and Com are all PPT). In the following discussion, we abuse the notation — we write $\mathcal{D}(\delta)$ instead the full notation $\mathcal{D}(\delta, m, C)$. Hence, (m, C) is always part of the input to $\mathcal{D}$ and the associated algorithms. Again, we abuse the notation by writing $Enc(PK_G, c, a)$ as $Enc(a)$.

The probability of success of $\mathcal{D}'$ is given by:

$$\begin{aligned} Pr^{IBE}_{\mathcal{D}'}[Success] &= Pr[b' = b | C_b] \\ &= \tfrac{1}{2} Pr[\mathcal{D}(\delta_e) = 0 | b = 0] + \tfrac{1}{2} Pr[\mathcal{D}(\delta_e) = 1 | b = 1] \\ &= \tfrac{1}{4} Pr[\mathcal{D}(\delta_e) = 0 | \delta_e = \langle \sigma_t \oplus h(a_t), Enc(a_t), Com(\sigma_t, h(a_t))\rangle] \\ &\quad + \tfrac{1}{4} Pr[\mathcal{D}(\delta_e) = 0 | \delta_e = \langle \sigma_f \oplus h(a_f), Enc(a_t), Com(\sigma_f, h(a_f))\rangle] \\ &\quad + \tfrac{1}{4} Pr[\mathcal{D}(\delta_e) = 1 | \delta_e = \langle \sigma_t \oplus h(a_t), Enc(a_f), Com(\sigma_t, h(a_t))\rangle] \\ &\quad + \tfrac{1}{4} Pr[\mathcal{D}(\delta_e) = 1 | \delta_e = \langle \sigma_f \oplus h(a_f), Enc(a_f), Com(\sigma_f, h(a_f))\rangle]. \end{aligned}$$

Note that

$$\begin{aligned} Pr^{Sim}_{\mathcal{D}}[Success] &= \tfrac{1}{2} Pr[\mathcal{D}(\delta_e) = 0 | \delta_e = \langle \sigma_t \oplus h(a_t), Enc(a_t), Com(\sigma_t, h(a_t))\rangle] \\ &\quad + \tfrac{1}{2} Pr[\mathcal{D}(\delta_e) = 1 | \delta_e = \langle \sigma_f \oplus h(a_f), Enc(a_f), Com(\sigma_f, h(a_f))\rangle]. \end{aligned}$$

Substituting $Pr^{IBE}_{\mathcal{D}'}[Success]$ into $Pr^{Sim}_{\mathcal{D}}[Success]$, we have

$$\begin{aligned}\tfrac{1}{2}Pr^{Sim}_{\mathcal{D}}[Success] &= Pr^{IBE}_{\mathcal{D}'}[Success] \\ &\quad -\tfrac{1}{4}Pr[\mathcal{D}(\delta_e)=0|\delta_e=\langle\sigma_f\oplus h(a_f), Enc(a_t), Com(\sigma_f, h(a_f))\rangle] \\ &\quad -\tfrac{1}{4}Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_t\oplus h(a_t), Enc(a_f), Com(\sigma_t, h(a_t))\rangle] \\ &= Pr^{IBE}_{\mathcal{D}'}[Success]-\tfrac{1}{4} \\ &\quad +\tfrac{1}{4}Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_f\oplus h(a_f), Enc(a_t), Com(\sigma_f, h(a_f))\rangle] \\ &\quad -\tfrac{1}{4}Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_t\oplus h(a_t), Enc(a_f), Com(\sigma_t, h(a_t))\rangle].\end{aligned}$$

Subtracting $\frac{1}{4}$ and then taking absolute values on both sides, we have

$$\begin{aligned}\tfrac{1}{2}Adv^{Sim}_{\mathcal{D}} \le Adv^{IBE}_{\mathcal{D}'} + \tfrac{1}{4}|&Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_f\oplus h(a_f), Enc(a_t), Com(\sigma_f, h(a_f))\rangle] \\ &-Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_t\oplus h(a_t), Enc(a_f), Com(\sigma_t, h(a_t))\rangle]|.\end{aligned}$$

Let $\varepsilon_{\mathcal{D}}$ denote $|Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_f\oplus h(a_f), Enc(a_t), Com(\sigma_f, h(a_f))\rangle]-Pr[\mathcal{D}(\delta_e)=1|\delta_e=\langle\sigma_t\oplus h(a_t), Enc(a_f), Com(\sigma_t, h(a_t))\rangle]|$. Then we could view $\varepsilon_{\mathcal{D}}$ as the advantage of $\mathcal{D}$ in distinguishing the following two distributions:

$$\begin{aligned}\Delta_f &= \{m\leftarrow\mathcal{M}; c\leftarrow\mathcal{C}; \sigma_f\leftarrow\mathcal{S}_\sigma; a, a'\leftarrow\mathcal{P}_{IBE} : (\sigma_f\oplus h(a), Enc_W(c, a'), Com(\sigma_f, h(a)))\}, \\ \Delta_t &= \{m\leftarrow\mathcal{M}; c\leftarrow\mathcal{C}; \sigma_t\leftarrow\{Sig_S(m)\}; a, a'\leftarrow\mathcal{P}_{IBE} : (\sigma_t\oplus h(a), Enc_W(c, a'), Com(\sigma_t, h(a)))\}\end{aligned}$$

We argue that $Enc_W(c, a')$ would not have useful information to help $\mathcal{D}$ in distinguishing the above two distributions as a and a' are picked independently; even if one know how to decrypt $Enc_W(c, a')$ to obtain a', a' has no useful information about a which is needed to tell whether a given δ comes from Δ_f or Δ_t. If $\varepsilon_{\mathcal{D}}$ is non-negligible, then it is straightforward to construct from $\mathcal{D}$ another algorithm D'' with an advantage $\varepsilon_{\mathcal{D}''} = \varepsilon_{\mathcal{D}}$ to distinguish the following two distributions:

$$\begin{aligned}\Pi_f &= \{m\leftarrow\mathcal{M}; \sigma_f\leftarrow\mathcal{S}_\sigma; a\leftarrow\mathcal{P}_{IBE} : (\sigma_f\oplus h(a), Com(\sigma_f, h(a)))\}, \\ \Pi_t &= \{m\leftarrow\mathcal{M}; \sigma_t\leftarrow\{Sig_S(m)\}; a\leftarrow\mathcal{P}_{IBE} : (\sigma_t\oplus h(a), Com(\sigma_t, h(a)))\}\end{aligned}$$

The idea of the construction of $\mathcal{D}''$ is when a challenge $(\sigma\oplus h(a), Com(\sigma, h(a)))$ (where σ could be equal to σ_t or σ_f) is received, $\mathcal{D}''$ randomly picks $a'\in\mathcal{P}_{IBE}$, creates $Enc_W(c, a')$, and add it to the challenge to create a new challenge $(\sigma\oplus h(a), Enc_W(c, a'), Com(\sigma, h(a)))$ for $\mathcal{D}$.

The advantage of reducing the problem of distinguishing Π_f/Π_t to that of distinguishing Δ_f/Δ_t is the adaptive queries, more specifically, the endorsement queries, in the simulatability game would not help in any way in distinguishing Π_f and Π_f. In other words, we do not need to take into account of adaptive queries while showing the indistinguishability between Π_f and Π_t. Besides, the indistinguishability between Π_f and Π_t implies that of Δ_f and Δ_t in the simulatability game.

Let ϵ_h and ϵ_{COM} be the indistinguishability coefficients of the pseudorandom generator and the commitment scheme. Recall that ϵ_h denotes the advantage of the best PPT distinguisher in distinguishing between the output distribution of a pseudorandom generator $h : \{0,1\}^{l_p}\rightarrow\{0,1\}^{l_s}$ and a uniform distribution over the output space of h, that is, between $\{x\leftarrow\{0,1\}^{l_p} : h(x)\}$ and $\{y\leftarrow\{0,1\}^{l_s} : y\}$. Whereas, ϵ_{COM} denotes the advantage of the best PPT distinguisher in distinguishing between the output distributions of the commitments of two different input values, say σ_f and σ_t, that is, between $\{r\leftarrow\{0,1\}^* : Com(\sigma_f, r)\}$ and $\{r\leftarrow\{0,1\}^* : Com(\sigma_t, r)\}$. Now, we can show the indistinguishability between Π_f and Π_t. In the following discussion, if X and Y are computationally indistinguishable, we denote $X\cong Y$. The proof below is based on the standard hybrid argument and the transitivity property

of computational indistinguishability.

$$\begin{aligned}
\Pi_t &= \{m \leftarrow \mathcal{M}; \sigma_t \leftarrow \{Sig_S(m)\}; a \leftarrow \mathcal{P}_{IBE} : (\sigma_t \oplus h(a), Com(\sigma_t, h(a)))\} \\
&\cong \{m \leftarrow \mathcal{M}; \sigma_t \leftarrow \{Sig_S(m)\}; r \leftarrow \{0,1\}^{l_s} : (\sigma_t \oplus r, Com(\sigma_t, r))\} && \text{(with } \epsilon_h\text{)} \\
&\cong \{m \leftarrow \mathcal{M}; \sigma_t \leftarrow \{Sig_S(m)\}; r, r' \leftarrow \{0,1\}^{l_s} : (r', Com(\sigma_t, r))\} \\
&\cong \{m \leftarrow \mathcal{M}; \sigma_f \leftarrow \mathcal{S}_\sigma; r, r' \leftarrow \{0,1\}^{l_s} : (r', Com(\sigma_f, r))\} && \text{(with } \epsilon_{COM}\text{)} \\
&\cong \{m \leftarrow \mathcal{M}; \sigma_f \leftarrow \mathcal{S}_\sigma; r \leftarrow \{0,1\}^{l_s} : (\sigma_f \oplus r, Com(\sigma_f, r))\} \\
&\cong \{m \leftarrow \mathcal{M}; \sigma_f \leftarrow \mathcal{S}_\sigma; a \leftarrow \mathcal{P}_{IBE} : (\sigma_f \oplus h(a), Com(\sigma_f, h(a)))\} && \text{(with } \epsilon_h\text{)} \\
&= \Pi_f
\end{aligned}$$

As a result, $\varepsilon_{\mathcal{D}} = \varepsilon_{\mathcal{D}''} < 2\epsilon_h + \epsilon_{COM}$. Substituting back, we have

$$\begin{aligned}
\tfrac{1}{2} Adv_{\mathcal{D}}^{Sim} &< Adv_{\mathcal{D}'}^{IBE} + \tfrac{1}{2}\epsilon_h + \tfrac{1}{4}\epsilon_{COM} \\
Adv_{\mathcal{D}}^{Sim} &< 2Adv_{\mathcal{D}'}^{IBE} + \epsilon_h + \tfrac{1}{2}\epsilon_{COM}.
\end{aligned}$$

If we assume COM is computationally hiding and h is a pseudorandom generator, then both ϵ_h and ϵ_{COM} should be negligible in their security parameters. Consequently, if $Adv_{\mathcal{D}}^{Sim}$ is non-negligible, the only possibility is either $Adv_{\mathcal{D}'}^{IBE}$ is non-negligible, meaning $\mathcal{D}'$ could break the semantic security of the IBE scheme (a contradiction). In other words, the semantic security of the IBE scheme implies the simulatability of the CVS construction with respect to the given construction of Fake. Since Fake is PPT, we could conclude that the given generic CVS construction is simulatable.

Single-witness Simulatability implies Multiple-witness Simulatability

We prove by contradiction. We assume the simulatability property is achieved in the single witness case. Suppose there is a PPT distinguisher D_N which can break the simulatability property for $N > 1$ where N is the number of witnesses. We show how to construct another distinguisher D_1, based on D_N, which could break the simulatability of the single-witness case.

The construction of D_1 (based on D_N) is as follows:

In the setup, D_1 asks its challenger for the signer's private and public keys, that is, sk_S and PK_S, and the witness public key $PK_1 = PK_G$. Without loss of generality, we set G as W_1 for the multiple-witness case. Then, D_1 creates the public and private keys for other witnesses $W_i, 2 \leq i \leq N$ by running CVKGS.

D_1 answer queries from D_N in the following way: When D_N makes a signing query, D_1 creates a partial signature itself as it knows the signer's private key sk_S. To answer any endorsement queries on a condition statement for witness W_1, D_1 makes an endorsement query to its challenger on the same condition statement and passes the result back to D_N. For the endorsement queries for other witnesses $W_i, 2 \leq i \leq N$, D_1 answers them itself using the private key sk_i ($2 \leq i \leq N$).

When D_N outputs a message m and a condition set $C = \{(c_i, W_i) : 1 \leq i \leq N\}$ asking for a challenge, D_1 outputs m and (c_1, W_1) as its challenge request. It is possible that (c_1, W_1) has been queried before as there is no restriction in our definition of simulatability that (c_1, W_1) has to be a new one; at least one of (c_i, W_i) not previously queried would constitute a valid challenge request. Suppose C_E denote the set of all endorsement queries made so far, the simulatability definition only requires that the current query (c_j, W_j) must satisfy that $C \backslash (C_E \cup \{(c_j, W_j)\})$ is non-empty. We will discuss later about abortion probability due to this. Let us continue assuming (c_1, W_1) is a new condition. Suppose σ_t is a valid ordinary signature on m and σ_f is just some randomly picked number in $\mathcal{S}_\sigma$. D_1 receives its challenge $\delta_b^1 = (\alpha, \beta_1, \gamma)$ where $\delta_b^1 = (\sigma_t \oplus h(a_1), Enc_{W_1}(c_1, a_1), Com(\sigma_t, h(a_1)))$ when $b = 0$ and $\delta_b^1 = (\sigma_f \oplus h(a_1), Enc_{W_1}(c_1, a_1), Com(\sigma_t, h(a_1)))$ when $b = 1$ for some unknown a_1. To create a valid challenge for D_N, D_1 randomly picks $a_2, \ldots, a_i, \ldots, a_N$ such that $a_2 \oplus a_3 \oplus \ldots \oplus a_N = 0$ and sends out the following challenge to D_N: $\delta_b^N = (\alpha, \beta_1, \beta_2, \ldots, \beta_N, \gamma)$ where $\beta_i = Enc_{W_i}(c_i, a_i), 2 \leq i \leq N$. Note that $a_1 \oplus a_2 \oplus \ldots \oplus a_N = a_1$; hence, δ_b^N is a valid challenge for D_N.

D_N could continue making signing and endorsement queries. If (c_1, W_1) is in the query, then this run fails. Otherwise, when D_N outputs its guess b' for b, D_1 outputs b' as its guess for b. Obviously, if D_N is PPT, so is D_1 and the advantage of D_1 is the same as that of D_N, that is, $Adv_{D_1}^{Sim} = Adv_{D_N}^{Sim}$, provided D_1 does not abort in the simulation. Now, it remains to find out the probability of successfully a run of D_1. Note that no matter how many queries out of the requested challenge condition set $\{(c_i, W_i) : 1 \leq i \leq N\}$ are made by D_N, D_N must answer at least one of them directly according to the definition, that is, $C \backslash C_E^{final}$ must be non-empty if C_E^{final} denotes the set of all endorsement queries made before the end of the game. In that case, if (c_1, W_1) is in the remaining subset, D_1 makes a successful run, and the probability of that is $p_{succ} = \frac{1}{N}$. Overall, the advantage of D_1 is $Adv_{D_1}^{Sim} = \frac{1}{N} Adv_{D_N}^{Sim}$. In the real cases, N would usually be a small integer, usually < 10, so the restriction would be fulfilled without mentioning. This concludes the reduction.

Note that a tighter reduction would also be possible.

Appendix D: Proof — Semantic Security of the IBE Construction from CVS

Proof of Theorem 6.

Now, we show that the above construction satisfies the conditions for IND-ID-CPA secure IBE. We assume the CVS scheme is simulatable with respect to Fake. Suppose the above constructed IBE scheme is not IND-ID-CPA secure, that is, there exists an adversary $\mathcal{D}$ which can win the IND-ID-CPA game with a non-negligible advantage $Adv_{\mathcal{D}}^{IBE}$. In other words, given a ciphertext (m, δ_b, PK_S) where δ_b is a valid/fake partial signature when $b = 0/1$, $\mathcal{D}$ could tell whether the plaintext bit $b = 0$ or $b = 1$ with a non-negligible advantage. Up to this point, it is clear that $\mathcal{D}$ could be used to break the simulatability property of the underlying CVS scheme with respect to Fake. However, for completeness, we show how to construct another adversary $\mathcal{D}'$ from $\mathcal{D}$ to tell whether a given partial signature δ_b originates from CVSig or Fake.

$\mathcal{D}'(\delta_b)$

Setup.
Get the public key PK_G of the witness from its challenger. Run $\mathcal{D}$ on PK_G.
Get the signer's public/private key pair (PK_S, sk_S). **Query.**
Extraction Query $\langle ID_j \rangle$. Pass all extraction queries from $\mathcal{D}$ to its endorsement oracle.
Challenge.
$\mathcal{D}$ outputs ID to be challenged. (Note the plaintext could only be 0 or 1.)
Randomly select a message $m \in \mathcal{M}$.
Pass m, ID to its challenger and receive the challenge δ_b.
Pass $C_b = (m, \delta_b, PK_S)$ as a challenged ciphertext to $\mathcal{D}$.
Guess.
$\mathcal{D}$ outputs a guess b'. Output b' as a guess for b.

It obvious that the advantage of $\mathcal{D}'$ with respect to CVS simulatability is the same as the advantage of $\mathcal{D}$ on breaking the semantic security of the IBE scheme. Hence, if the latter is non-negligible, so is the former, a contradiction as we assume the given CVS scheme is simulatable with respect to Fake. In conclusion, the constructed IBE scheme is semantically secure as long as the CVS scheme is simulatable.