Provably Secure Partially Blind Signatures

Masayuki ABE and Tatsuaki OKAMOTO

NTT Laboratories
Nippon Telegraph and Telephone Corporation
1-1 Hikari-no-oka Yokosuka-shi Kanagawa-ken, 239-0847 Japan
E-mail: {abe,okamoto}@isl.ntt.co.jp

Abstract. Partially blind signature schemes are an extension of blind signature schemes that allow a signer to explicitly include necessary information (expiration date, collateral conditions, or whatever) in the resulting signatures under some agreement with the receiver. This paper formalizes such a notion and presents secure and efficient schemes based on a widely applicable method of obtaining witness indistinguishable protocols. We then give a formal proof of security in the random oracle model. Our approach also allows one to construct secure fully blind signature schemes based on a variety of signature schemes.

Keywords: Partially Blind Signatures, Blind Signatures, Witness Indistinguishability

1 Introduction

1.1 Background

Digital signature schemes are essential for electronic commerce as they allow one to authorize digital documents that are moved across networks. Typically, a digital signature comes with not just the document body but also attributes such as "date of issue" or "valid until", which may be controlled by the signer rather than the receiver. One can find more about those attributes in PKCS #9 [23], for instance.

Blind signature schemes, first introduced by Chaum in [5], are a variant of digital signature schemes. They allow a receiver to get a signature without giving the signer any information about the actual message or the resulting signature. This blindness property plays a central role in applications such as electronic voting (e.g. [6, 12]) and electronic cash schemes (e.g. [5, 7, 4]) where anonymity is of great concern.

One particular shortcoming is that, since the singer's view is perfectly shut off from the resulting signatures, the signer has no control over the attributes except for those bound by the public key. For instance, if a signer issues blind signatures that are valid until the end of the week, the signer has to change his public key every week! This will seriously impact availability and performance. A similar

shortcoming can be seen in a simple electronic cash system where a bank issues a blind signature as an electronic coin. Since the bank cannot inscribe the value on the blindly issued coins, it has to use different public keys for different coin values. Hence the shops and customers must always carry a list of those public keys in their electronic wallet, which is typically a smart card whose memory is very limited. Some electronic voting schemes also face the same problem when an administrator issues blind signatures to authorize ballots. Since he can not include the vote ID, his signature may be used in an unintended way. This means that the public key of the administrator must be disposable. Accordingly, each voter must download a new public key for each vote.

A *partially* blind signature scheme allows the signer to explicitly include common information in the blind signature under some agreement with the receiver. For instance, the signer can attach the date of issue to his blind signatures as an attribute. If the signer issues a huge number of signatures in a day, including the date of issue will not violate anonymity. Accordingly, the attributes of the signatures can be decided independently from those of the public key.

By fixing common information to a single string, one can easily transform partially blind signature schemes into fully blind ones. However, the reverse is not that easy. One can now see that partially blind signatures are a generalized notion of blind signatures. The main subject of this paper is to consider the security of partially blind signatures and present the first secure and efficient schemes together with a formal proof of their security.

1.2 Related work

In [15], Juels, Luby and Ostrovsky gave a formal definition of blind signatures. They proved the existence of secure blind signatures assuming the one-way trapdoor permutation family. Their construction was, however, only theoretical, not practical. Before [15], Pointcheval and Stern showed the security of a certain type of efficient blind signature in the random oracle model [20]. Namely, they showed that Okamoto-Schnorr and Okamoto-Guillou-Quisquater signatures [18] are secure as long as the number of issued signatures are bounded logarithmically in the security parameter. Later, in [19], Pointcheval developed a generic approach that converts logarithmically secure schemes into polynomially secure ones at the cost of two more data transmissions between the signer and the receiver. Unfortunately, his particular construction, that based on Okamoto signatures, does not immediately lead to partially blind signature schemes.

The notion of partially blind signatures was introduced in [2]. Their construction, based on RSA, was analyzed in [1]. It also showed a construction based on Schnorr signatures that withstands a particular class of attacks. There are some other heuristic constructions in the literature. One of the authors was informed that Cramer and Pedersen independently considered the same notion and constructed a scheme, which remains unavailable in public due to an embargo [8]. All in all, no provably secure and practical partially blind signature scheme has been publicly released.

1.3 Our contribution

This paper first gives a formal definition of partially blind signature schemes. As partially blind signatures can be regarded as ones lying between ordinary non-blind digital signatures and fully blind signatures, they should satisfy the security requirements assigned to ordinary digital signatures and those of blind signatures.

We then present efficient partially blind signature schemes with a rigorous proof of security in the random oracle model [3] under the standard number theoretic intractability assumptions such as discrete-log or the RSA assumption. Since the technique developed by Pointcheval and Stern for proving the one-more-unforgeability [20] is not applicable to our scheme, we provide a new technique to prove the security of our scheme. The technique shown in this paper is more generic than that of [20] and applicable to variety of schemes based on the witness indistinguishable protocols including the ones that the technique of [20] is applicable to. As well as the result of [20, 22], our proof guarantees that the proposed scheme is secure as long as only a logarithmic number of signatures are issued. So plugging our scheme into the generic, but yet practical scheme of [19] will yield a scheme secure up to polynomial number of signatures.

For the sake of simplicity, we put off the generic description of our approach and concentrate on describing one particular scheme based on the original (i.e. not Okamoto version of) Schnorr signature scheme. One can, however, construct a scheme in a similar way based on Guillou-Quisquater signatures [14] or variants of modified ElGamal signatures [10, 21, 16] at the cost of doubling the computation and communication compared to the underlying schemes.

Although our primary goal is partially blind signatures, our approach also yields secure fully blind signatures. Thus, from a different angle, our result can be seen as a widely applicable approach that turns several secure signature schemes into secure blind signatures.

1.4 Organization

Section 2 defines the security of partially blind signatures. In Section 3 we show a partially blind signature scheme based on Schnorr signatures. Section 4 gives a proof of security.

2 Definitions

In the scenario of issuing a partially blind signature, the signer and the user are assumed to agree on a piece of common information, denoted as info. In some applications, info may be decided by the signer, while in other applications it may just be sent from the user to the signer. Anyway, this negotiation is done outside of the signature scheme, and we want the signature scheme to be secure regardless of the process of agreement. We formalize this notion by introducing function $Ag(\,)$ which is defined outside of the scheme. Function Ag is

a polynomial-time deterministic algorithm that takes two arbitrary strings info_s and info_u that belong to the signer and the user, respectively, and outputs info. To compute Ag, the signer and the user will exchange info_s and info_u with each other. However, if an application allows the signer to control info, then Ag is defined such that it depends only on info_s. In such a case, the user does not need to send info_u.

Some part of the following definitions refers to [15]. In the following, we will use the term "polynomial-time" to mean a certain period bounded by a polynomial in security parameter n.

Definition 1. *(Partially Blind Signature Scheme) A Partially blind signature scheme is a four-tuple* $(\mathcal{G}, \mathcal{S}, \mathcal{U}, \mathcal{V})$.

- $\mathcal{G}$ *is a probabilistic polynomial-time algorithm that takes security parameter* n *and outputs a public and secret key pair* (pk, sk).
- $\mathcal{S}$ *and* $\mathcal{U}$ *are a pair of probabilistic interactive Turing machines each of which has a public input tape, a private input tape, a private random tape, a private work tape, a private output tape, a public output tape, and input and output communication tapes. The random tape and the input tapes are read-only, and the output tapes are write-only. The private work tape is read-write. The public input tape of* $\mathcal{U}$ *contains* pk *generated by* $\mathcal{G}(1^n)$*, the description of* Ag*, and* info_u*. The public input tape of* $\mathcal{S}$ *contains the description of* Ag *and* info_s*. The private input tape of* $\mathcal{S}$ *contains* sk*, and that for* $\mathcal{U}$ *contains message* msg*. The lengths of* info_s*,* info_u*, and* msg *are polynomial in* n*.* $\mathcal{S}$ *and* $\mathcal{U}$ *engage in the signature issuing protocol and stop in polynomial-time. When they stop, the public output tape of* $\mathcal{S}$ *contains either completed or not-completed. If it is completed, then its private output tape contains common information* $\mathsf{info}^{(s)}$*. Similarly, the private output tape of* $\mathcal{U}$ *contains either* $\perp$ *or* $(\mathsf{info}, \mathsf{msg}, sig)$.
- $\mathcal{V}$ *is a (probabilistic) polynomial-time algorithm that takes* $(pk, \mathsf{info}, \mathsf{msg}, sig)$ *and outputs either accept or* $reject$.

Definition 2. *(Completeness) If* $\mathcal{S}$ *and* $\mathcal{U}$ *follow the signature issuing protocol, then, with probability at least* $1 - 1/n^c$ *for sufficiently large* n *and some constant* c*,* $\mathcal{S}$ *outputs completed and* $\mathsf{info} = Ag(\mathsf{info}_s, \mathsf{info}_u)$ *on its proper tapes, and* $\mathcal{U}$ *outputs* $(\mathsf{info}, \mathsf{msg}, sig)$ *that satisfies* $\mathcal{V}(pk, \mathsf{info}, \mathsf{msg}, sig) = accept$*. The probability is taken over the coin flips of* $\mathcal{G}$*,* $\mathcal{S}$ *and* $\mathcal{U}$.

We say a message-signature tuple $(\mathsf{info}, \mathsf{msg}, sig)$ is valid with regard to pk if it leads $\mathcal{V}$ to *accept*.

To define the blindness property, let us introduce the following game.

Definition 3. *(Game A) Let* $\mathcal{U}_0$ *and* $\mathcal{U}_1$ *be two honest users that follow the signature issuing protocol.*

1. $(pk, sk) \leftarrow \mathcal{G}(1^n)$.
2. $(\mathsf{msg}_0, \mathsf{msg}_1, \mathsf{info}_{u0}, \mathsf{info}_{u1}, Ag) \leftarrow \mathcal{S}^*(sk)$.
3. *Set up the input tapes of* $\mathcal{U}_0$*,* $\mathcal{U}_1$ *as follows:*

- *Select* $b \in_R \{0,1\}$ *and put* msg_b *and* $\mathsf{msg}_{\bar{b}}$ *on the private input tapes of* $\mathcal{U}_0$ *and* $\mathcal{U}_1$*, respectively (*$\bar{b}$ *denotes* $1-b$ *hereafter).*
- *Put* info_{u0} *and* info_{u1} *on the public input tapes of* $\mathcal{U}_0$ *and* $\mathcal{U}_1$*, respectively. Also put* pk *and* Ag *on their public input tapes.*
- *Randomly select the contents of the private random tapes.*

4. $\mathcal{S}^*$ *engages in the signature issuing protocol with* $\mathcal{U}_0$ *and* $\mathcal{U}_1$*.*
5. *If* $\mathcal{U}_0$ *and* $\mathcal{U}_1$ *outputs* $(\mathsf{info}_0, \mathsf{msg}_0, sig_b)$ *and* $(\mathsf{info}_1, \mathsf{msg}_1, sig_b)$*, respectively, on their private tapes, and* $\mathsf{info}_0 = \mathsf{info}_1$ *holds, then give those outputs to* $\mathcal{S}^*$*. Give* $\perp$ *to* $\mathcal{S}^*$ *otherwise.*
6. $\mathcal{S}^*$ *outputs* $b' \in \{0,1\}$*.*

We say that $\mathcal{S}^*$ *wins if* $b' = b$*.*

Definition 4. *(Partial Blindness) A signature scheme is partially blind if, for all probabilistic polynomial-time algorithm* $\mathcal{S}^*$*,* $\mathcal{S}^*$ *wins in game A with probability at most* $1/2 + 1/n^c$ *for sufficiently large* n *and some constant* c*. The probability is taken over the coin flips of* $\mathcal{G}$*,* $\mathcal{U}_0$*,* $\mathcal{U}_1$*, and* $\mathcal{S}^*$*.*

As usual, one can go for stronger notion of blindness depending on the power of the adversary and its success probability. Our scheme provides *perfect* partial blindness where any infinitely powerful adversary wins with probability exactly $1/2$.

Forgery of partially blind signatures is defined in the similar way as [15] with special care for the various pieces of common information. At first look, the forgery of a partially blind signature might be considered as forging the common information, or producing $\ell_{\mathsf{info}} + 1$ signatures with regard to info provided ℓ_{info} successful execution of the signature issuing protocol for that info. Forging the common information is actually the same as producing one-more signature with info where $\ell_{\mathsf{info}} = 0$. We define unforgeability through the following game.

Definition 5. *(Game B)*

1. $(pk, sk) \leftarrow \mathcal{G}(1^n)$*.*
2. $Ag \leftarrow \mathcal{U}^*(pk)$*.*
3. *Put* sk, Ag *and randomly taken* inf_s *on proper tapes of* $\mathcal{S}$*.*
4. $\mathcal{U}^*$ *engages in the signature issuing protocol with* $\mathcal{S}$ *in a concurrent and interleaving way. For each* info*, let* ℓ_{info} *be the number of executions of the signature issuing protocol where* $\mathcal{S}$ *outputs completed and* info *on its output tapes. (For* info *that has never appeared on the private output tape of*$\mathcal{S}$*, define* $\ell_{\mathsf{info}} = 0$*.)*
5. $\mathcal{U}^*$ *outputs a single piece of common information,* info*, and* $\ell_{\mathsf{info}}+1$ *signatures* $(\mathsf{msg}_1, sig_1), \ldots, (\mathsf{msg}_{\ell_{\mathsf{info}}+1}, sig_{\ell_{\mathsf{info}}+1})$*.*

Definition 6. *(Unforgeability) A partially blind signature scheme is unforgeable if, for any probabilistic polynomial-time algorithm* $\mathcal{U}^*$ *that plays game B, the probability that the output of* $\mathcal{U}^*$ *satisfies* $\mathcal{V}(pk, \mathsf{info}, \mathsf{msg}_j, sig_j) =$ *accept for all* $j = 1, ..., \ell_{\mathsf{info}} + 1$ *is at most* $1/n^c$ *for sufficiently large* n *and some constant* c*. The probability is taken over the coin flips of* $\mathcal{G}$*,* $\mathcal{S}$*, and* $\mathcal{U}^*$*.*

3 Construction

3.1 Key Idea

The security of signature schemes is defined so that they are secure against adaptive attacks [13]. To prove the security against such attacks, one has to simulate the signer without knowing the private signing key. Introducing a random oracle allows the simulation for ordinary signatures but does not help in the case of blind signatures. So, the simulator has to have a real signing key. Accordingly, we need to separate the signing key from the witness of the embedding intractable problem, such as the discrete logarithm problem, that we attempt to solve by using an attacker of the signature scheme. For this to be done, Pointcheval and Stern used the blind Okamoto signature scheme where the existence of a successful attacker implied extraction of the discrete logarithm of bases rather than the signing key. They also exploited the witness indistinguishable property of Okamoto signatures in a crucial way in their proof of security. Unfortunately, we do not know how to achieve partial blindness with their construction.

In [9], Cramer, Damgård and Schoenmakers presented an efficient method of constructing witness indistinguishable protocols. With their adaptation, one can turn a wide variety of signature schemes derived from public-coin honest verifier zero-knowledge into witness indistinguishable ones. Intuitively, the signer has one private key x but uses two different public keys, y and z, together to sign a message in such a way that the user can not distinguish which private key he has. By blinding the signing procedure, one can get fully blind witness indistinguishable signature schemes.

Our idea to achieve partial blindness is to put common information, say info, into one of those public keys. Suppose that $z = \mathcal{F}(\mathsf{info})$ where $\mathcal{F}$ is a sort of public hash function that transforms an arbitrary string to a random public key whose private key is not known to anybody. The signer then signs with private key x of y. Since the resulting signatures are bound to public keys y, z, the common information info is also bound to the signature. Since blinding will not cover public keys, info (i.e. z) remains unblind. This adaptation preserves witness indistinguishability which we need in our proof of security.

3.2 Preliminaries

Let $\mathcal{G}_{DL}$ be a discrete logarithm instance generator that takes security parameter n and outputs a triple (p, q, g) where p, q are large primes that satisfy $q|p-1$, and g is an element in $\mathbb{Z}_p^*$ whose order is q. Let $\langle g \rangle$ denote a subgroup in $\mathbb{Z}_p^*$ generated by g. We assume that any polynomial-time algorithm solves $\log_g h$ in $\mathbb{Z}_q$ only with negligible probability (in the size of q and coin flips of $\mathcal{G}_{DL}$ and the algorithm) when h is selected randomly from $\langle g \rangle$. All arithmetic operations are done in $\mathbb{Z}_p$ hereafter unless otherwise noted.

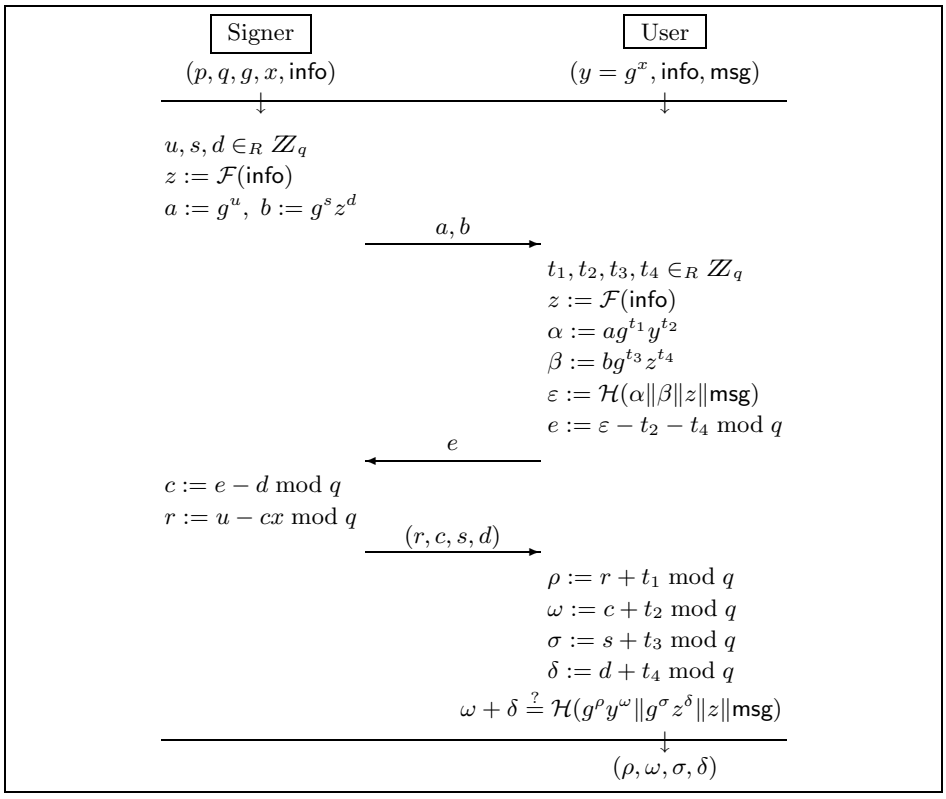

Fig. 1. Partially blind WI-Schnorr signature issuing protocol. The signer and the user are assumed to agree on info beforehand outside of the protocol. The signer can omit sending either c or d as the user can compute it himself from e.

3.3 A partially blind WI-Schnorr signature scheme

Let $\mathcal{H} : \{0,1\}^* \to \mathbb{Z}_q$ and $\mathcal{F} : \{0,1\}^* \to \langle g \rangle$ be public hash functions. Let $x \in \mathbb{Z}_q$ be a secret key and $y := g^x$ be a corresponding public key.

Signer $\mathcal{S}$ and user $\mathcal{U}$ first agree on common information info in an predetermined way. They then execute the signature issuing protocol illustrated in Figure 1. The resulting signature for message msg and common information info is a four-tuple $(\rho, \omega, \sigma, \delta)$. A signature is valid if it satisfies

$$\omega + \delta \equiv \mathcal{H}(g^\rho y^\omega \| g^\sigma \mathcal{F}(\mathsf{info})^\delta \| \mathcal{F}(\mathsf{info}) \| \mathsf{msg}) \pmod q.$$

Observe that the signature issuing protocol is witness indistinguishable. That is, the user's view has exactly the same distribution even if $\mathcal{S}$ executes the protocol with witness $w(= \log_g z)$ instead of x computing as $v, r, c \in_R \mathbb{Z}_q$, $a := g^r y^c$, $b := g^v$, $d = e - c \bmod q$, and $s := v - dw \bmod q$.

In the above description, we assumed the use of hash function $\mathcal{F}$ that maps an arbitrary string to an element of $\langle g \rangle$. This, however, would be problematic in

practice because currently available hash functions, say $\mathcal{D}$, such as SHA-1 and MD5, are of $\mathcal{D} : \{0,1\}^* \to \{0,1\}^{len}$ for some fixed len. An immediate thought would be to repeat $\mathcal{D}$ with random suffixes until the output eventually falls in $\langle g \rangle$. However, such a probabilistic strategy makes the running-time *expected* polynomial-time rather than *strict* polynomial-time. Furthermore, in practice, if q is much smaller than p as in ordinary Schnorr signatures, such a strategy is hopeless. We show two deterministic constructions of $\mathcal{F}$ assuming the use of hash function $\mathcal{D}$ with $len = |p|$.

Construction 1 Take p, q that satisfy $p = 2q + 1$. Define $\mathcal{F}$ as

$$\mathcal{F}(\mathsf{info}) \triangleq \left(\frac{\mathcal{D}(\mathsf{info})}{p}\right) \mathcal{D}(\mathsf{info}) \bmod p$$

where $\left(\frac{\mathcal{D}(\mathsf{info})}{p}\right)$ is the Jacobi symbol of $\mathcal{D}(\mathsf{info})$.

Construction 2 Take p, q that satisfy $q|p-1$ and $q^2 \nmid p-1$. Define $\mathcal{F}$ as

$$\mathcal{F}(\mathsf{info}) \triangleq \mathcal{D}(\mathsf{info})^{\frac{p-1}{q}} \bmod p.$$

The second construction is better in terms of computation as we can choose smaller q such as $|q| \approx 2^{160}$. If $\mathcal{D}$ behaves as an ideal hash function, both constructions meet our requirement for the proof of security (that is, we can assign an arbitrary element of $\langle g \rangle$ as an output of $\mathcal{F}$). For simplicity, we set aside that detail and assume $\mathcal{F}$ be an atomic function in our proof of security in section 4.

4 Security

This section proves the security of our scheme assuming the intractability of the discrete logarithm problem and ideal randomness of hash functions $\mathcal{H}$ and $\mathcal{F}$.

Lemma 1. *The proposed scheme is partially blind.*

Proof. Let $\mathcal{S}^*$ be a player of game A. For $i = 0, 1$, let a_i, b_i, e_i, r_i, c_i, s_i, d_i, info_i be data appearing in the view of $\mathcal{S}^*$ during the execution of the signature issuing protocol with $\mathcal{U}_i$ at step 4.

When $\mathcal{S}^*$ is given $\perp$ in step 6 of the game, it is not hard to see that $\mathcal{S}^*$ wins game A with probability exactly the same as random guessing of b.

Suppose that $\mathsf{info}_1 = \mathsf{info}_0$, and $\{(\rho_0, \omega_0, \sigma_0, \delta_0)\}$ and $\{(\rho_1, \omega_1, \sigma_1, \delta_1)\}$ are given to $\mathcal{S}^*$. It is sufficient to show that there exists a tuple of random factors (t_1, t_2, t_3, t_4) that maps $a_i, b_i, r_i, c_i, s_i, d_i$ to $\rho_j, \omega_j, \sigma_j, \delta_j$ for each $i, j \in \{0, 1\}$. (e_i and info_i can be omitted as c_i, d_i determines e_i, and info_i is common.) Define $t_1 := \rho_j - r_i$, $t_2 := \omega_j - c_i$, $t_3 := \sigma_j - s_i$, and $t_4 := \delta_j - d_i$. As $a_i = g^{r_i} y^{c_i}$ and $b_i = g^{s_i} z^{d_i}$ holds, we see that

$$\begin{aligned}
\omega_j + \delta_j &= \mathcal{H}(g^{\rho_j} y^{\omega_j} \| g^{\sigma_j} z^{\delta_j} \| \mathcal{F}(\mathsf{info}) \| \mathsf{msg}) \\
&= \mathcal{H}(a_i g^{-r_i} y^{-c_i} g^{\rho_j} y^{\omega_j} \| b_i g^{-s_i} z^{-d_i} g^{\sigma_j} z^{\delta_j} \| \mathcal{F}(\mathsf{info}) \| \mathsf{msg}) \\
&= \mathcal{H}(a_i g^{\rho_j - r_i} y^{\omega_j - c_i} \| b_i g^{\sigma_j - s_i} z^{\delta_j - d_i} \| \mathcal{F}(\mathsf{info}) \| \mathsf{msg}) \\
&= \mathcal{H}(a_i g^{t_1} y^{t_2} \| b_i g^{t_3} y^{t_4} \| \mathcal{F}(\mathsf{info}) \| \mathsf{msg}).
\end{aligned}$$

Thus, $a_i, b_i, r_i, c_i, s_i, d_i$ and $\rho_j, \omega_j, \sigma_j, \delta_j$ have exactly the same relation defined by the signature issuing protocol. Such t_1, t_2, t_3, t_4 always exist regardless of the values of r_i, c_i, s_i, d_i and $\rho_j, \omega_j, \sigma_j, \delta_j$. Therefore, even an infinitely powerful $\mathcal{S}^*$ wins game A of our scheme with probability exactly 1/2. □

Lemma 2. *The proposed scheme is unforgeable if* $\ell_{\mathsf{info}} < poly(\log n)$ *for all* info.

Proof. The proof is done in three steps. We first treat the common-part forgery where an attacker forges a signature with regard to common information info that has not appeared while Game B (i.e., $\ell_{\mathsf{info}} = 0$). Next we treat one-more forgery where $\ell_{\mathsf{info}} \neq 0$. For this case, we first prove the security with restricted signer $\mathcal{S}$ that issues signatures only for a fixed info. We then eliminate the restriction by showing the reduction from the unrestricted signer model to the restricted one.

We first deal with successful common-part forger $\mathcal{U}^*$ who plays game B and produces, with probability $\mu > 1/n^c$, a valid message-signature tuple $(\mathsf{info}, \mathsf{msg}, \rho, \omega, \sigma, \delta)$ such that $\ell_{\mathsf{info}} = 0$. This part of the proof follows that used for ID-reduction [17]. By using $\mathcal{U}^*$, we construct a machine $\mathcal{M}$ that forges a non-blind version of the WI-Schnorr signature in a passive environment (i.e. without talking with signer $\mathcal{S}$). We then use $\mathcal{M}$ to solve the discrete logarithm problem by exploiting the collision property.

Let q_F and q_H be the maximum number of queries asked from $\mathcal{U}^*$ to $\mathcal{F}$ and $\mathcal{H}$, respectively. Similarly, let q_S be the maximum number of invocation of signer $\mathcal{S}$ in game B. All those parameters are limited by a polynomial in n. For simplicity, we assume that all queries are different. (For all duplicated queries to $\mathcal{F}$ and $\mathcal{H}$, return formerly defined values.) Let (y, g, p, q) be the problem that we want to solve $\log_g y (= x)$ in $\mathbb{Z}_q$. Machine $\mathcal{M}$ simulates game B as follows.

1. Select $I \in_U \{1, \ldots, q_F + q_S\}$ and $J \in_U \{1, \ldots, q_H + q_S\}$.
2. Run $\mathcal{U}^*$ with $pk := (y, g, p, q)$ simulating $\mathcal{H}$, $\mathcal{F}$ and $\mathcal{S}$ as follows.
   - For i-th query to $\mathcal{F}$, return z such that
     - $z := \mathcal{F}(\mathsf{info}_I)$ (i.e. ask oracle $\mathcal{F}$) if $i = I$, or
     - $z := g^{w_i}$ where $w_i \in_U \mathbb{Z}_q$, otherwise.
   - For j-th query to $\mathcal{H}$,
     - ask $\mathcal{H}$ if $j = J$, or
     - randomly select the answer from $\mathbb{Z}_q$, otherwise.
   - For requests to $\mathcal{S}$, first negotiate the common information. Let info_k be the result of the negotiation. If $\mathcal{F}(\mathsf{info}_k)$ is not defined yet, define it as mentioned above. Then,
     - if $\mathsf{info}_k \neq \mathsf{info}_I$, simulate $\mathcal{S}$ by using witness w_k, or
     - if $\mathsf{info}_k = \mathsf{info}_I$, we expect that $\mathcal{U}^*$ aborts the session before it receives (r, c, s, d). (If $\mathcal{U}^*$ tries to complete the session, the simulation fails.) Just to simulate the state of abortion, send random (a, b) to $\mathcal{U}^*$.
3. If $\mathcal{U}^*$ eventually outputs signature $(\rho, \omega, \sigma, \delta)$ with regard to info_I and msg_J, output them.

Note that the queries to $\mathcal{F}$ and $\mathcal{H}$ may include the ones inquired during the simulation of $\mathcal{S}$. So, $\mathcal{F}$ and $\mathcal{H}$ are defined at at most $q_F + q_S$ and $q_H + q_S$ points during the simulation, respectively. The simulation of $\mathcal{S}$ for $\mathsf{info}_k \neq \mathsf{info}_I$ can be perfectly done with w_k due to witness indistinguishability. The probability that $\mathcal{U}^*$ is successful without asking $\mathcal{F}, \mathcal{H}$ in a proper way is negligible because of the unpredictability of those hash functions. Thus, the success probability of $\mathcal{M}$ is only negligibly worse than $\frac{\mu}{(q_H+q_S)(q_F+q_S)}$ which is not negligible in n. By μ', we denote the success probability of $\mathcal{M}$.

Now we use $\mathcal{M}$ to solve $\log_g y$. The trick is to simulate $\mathcal{F}$ by responding to the query from $\mathcal{M}$ with yg^γ where γ is chosen randomly from $\mathbb{Z}_q$. Note that $\mathcal{M}$ asks each of $\mathcal{F}$ and $\mathcal{H}$ only once. Furthermore, the query to $\mathcal{F}$ happens *before* the query to $\mathcal{H}$ with overwhelming probability when $\mathcal{M}$ is successful because $\mathcal{F}(\mathsf{info})$ is contained in the inputs of $\mathcal{H}$. Next, we apply the standard replay technique [11]. That is, run $\mathcal{M}$ with a random tape and a random choice of $\mathcal{H}$. $\mathcal{M}$ then outputs a valid signature, say $(\rho, \omega, \sigma, \delta)$, with probability at least $1-e^{-1}$ (here, e is base of natural logarithms) after $1/\mu'$ trials. We then rewind $\mathcal{M}$ with the same random tape and run it with a different choice of $\mathcal{H}$. By repeating this rewind-trial $2/\mu'$ times, we get another valid signature, say $(\rho', \omega', \sigma', \delta')$, with probability at least $(1 - e^{-1})/2$. After all, with constant probability and polynomial running time, we have two valid signatures whose first messages (a, b) are the same. Thus, $\rho + \omega x = \rho' + \omega' x$, $\sigma + \delta(x + \gamma) = \sigma' + \delta'(x + \gamma)$, and $\omega + \delta \neq \omega' + \delta'$ holds. Since at least $\omega \neq \omega'$ or $\delta \neq \delta'$ happens, one can get x as $x = (\rho - \rho')/(\omega' - \omega) \bmod q$ or $x = (\sigma - \sigma')/(\delta' - \delta) - \gamma \bmod q$.

Next we consider the case where the forgery is attempted against info such that $\ell_{\mathsf{info}} \neq 0$. As the first step, we consider Game B with a single info. Hence z is common for all executions of the signature issuing protocol. Accordingly, we prove the security of fully blind version of our scheme. Let $\ell = \ell_{\mathsf{info}}$.

Reduction algorithm

Assume a single-info adversary, $\mathcal{U}_F^*$, which is a probabilistic polynomial time algorithm that violates unforgeability for infinitely many sizes, n's, with the attack defined as Game B. (Let n_0 be such a size, and the success probability of $\mathcal{U}_F^*$ is at least η). Then we construct an algorithm, $\mathcal{M}$, that utilizes $\mathcal{U}_F^*$ as black-box and breaks the intractability assumption of the discrete logarithm for infinitely many n's. That is, the input to $\mathcal{M}$ is (p, q, g, z_0), and $\mathcal{M}$ tries to compute w_0 such that $z_0 = g^{w_0}$, provided $\mathcal{U}_F^*$.

First, $\mathcal{M}$ selects $b \in_U \{0, 1\}$ and assigns (y, z) as $(y, z) = (g^x, z_0 g^\gamma)$ if $b = 0$, or $(y, z) = (z_0 g^\gamma, g^w)$ if $b = 1$ by choosing γ and x (or w) randomly from $\mathbb{Z}_q$. $\mathcal{F}$ is defined so that it returns appropriate value of z according to the choice. Hereafter, without loss of generality, we assume that $b = 0$ is chosen and $(y, z) = (g^x, z_0 g^\gamma)$ is set. $\mathcal{M}$ can then simulate signer $\mathcal{S}$, since the protocol between $\mathcal{S}$ and $\mathcal{U}_F^*$ is witness indistinguishable and having $x = \log_g y$ is sufficient for $\mathcal{S}$ to complete the protocol. Let $\hat{\mathcal{S}}$ denote the signer simulated by $\mathcal{M}$.

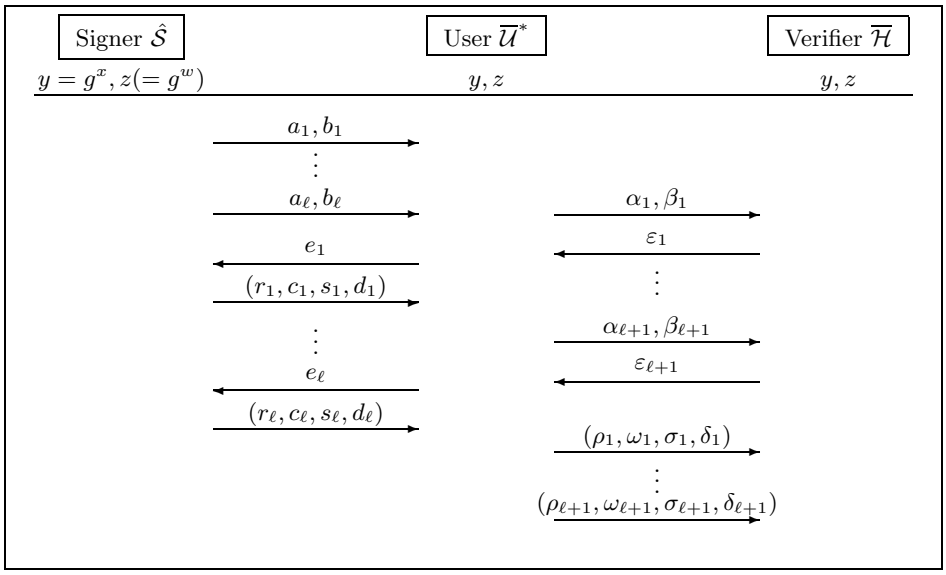

Fig. 2. Corresponding Divertible Identification Protocol.

If $\mathcal{U}_F^*$ is successful with probability at least η, we can find a random tape string for $\mathcal{U}_F^*$ and $\hat{\mathcal{S}}$ with probability at least 1/2 such that $\mathcal{U}_F^*$ with $\hat{\mathcal{S}}$ succeeds with probability at least $\eta/2$.

By employing $\mathcal{U}_F^*$ as a black-box, we can construct $\overline{\mathcal{U}}^*$ which has exactly the same interface with $\hat{\mathcal{S}}$ as $\mathcal{U}_F^*$ has, and plays the role of an impersonator in the interactive identification protocol with verifier $\overline{\mathcal{H}}$ (see Fig. 2). When $\mathcal{U}_F^*$ asks at most q_F queries to random oracle $\mathcal{H}$, $\overline{\mathcal{U}}^*$ is successful in completing the identification protocol with verifier $\overline{\mathcal{H}}$ with probability at least $\eta/2q_H^{\ell+1}$, since, with probability greater than $1/2q_H^{\ell+1}$, $\overline{\mathcal{U}}^*$ can guess a correct selection of $\ell+1$ queries that $\mathcal{U}^*$ eventually uses in the forgery.

$\mathcal{M}$ then use the standard replay technique for an interactive protocol to compute the discrete logarithm. $\mathcal{M}$ first runs $\overline{\mathcal{U}}^*$ with $\hat{\mathcal{S}}$ and $\overline{\mathcal{H}}$, and find a successful challenge tuple $(\varepsilon_1, \ldots, \varepsilon_{\ell+1})$. $\mathcal{M}$ then randomly chooses an index, $i \in \{1, \ldots, \ell+1\}$, and replay with the same environments and random tapes except different challenge tuple $(\varepsilon_1, \ldots, \varepsilon_{i-1}, \varepsilon_i', \ldots, \varepsilon_{\ell+1}')$ where the first $i-1$ challenges are unchanged. Since $\varepsilon_i \neq \varepsilon_i'$, at least either $\delta_i \neq \delta_i'$ or $\omega_i \neq \omega_i'$ happens. If $\delta_i \neq \delta_i'$, then $\mathcal{M}$ can compute $w(= \log_g z)$ as $w = (\sigma_i - \sigma_i')/(\delta_i' - \delta_i) \bmod q$. $\mathcal{M}$ then obtain $w_0 = w - \gamma \bmod q$ such that $z_0 = g^{w_0}$.

Evaluation of the success probability

Let Ω and Θ be random tape strings of $\mathcal{M}$ and $\mathcal{U}^*$, respectively. Note that Ω includes the random selection of b and random factors in the simulation of $\mathcal{S}$. Ω and Θ are assumed to be fixed throughout this evaluation. Let $\vec{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_{\ell+1})$, and $\vec{e} = (e_1, \ldots, e_\ell)$. $\mathcal{E}$ denotes the set of all $\vec{\varepsilon}$s (hence $\#\mathcal{E} = q^{\ell+1}$). The first

$i-1$ elements of $\vec{\varepsilon}$, i.e. $(\varepsilon_1, \ldots, \varepsilon_{i-1})$, is denoted by $\vec{\varepsilon}_i$, and the i-th element of $\vec{\varepsilon}$ is denoted by $\vec{\varepsilon}_{[i]}$. We define Succ a set of successful $\vec{\varepsilon}$ such that $\vec{\varepsilon} \in$ Succ iff $\vec{\varepsilon}$ is an accepted sequence of challenges between $\overline{\mathcal{U}}^*$ and $\overline{\mathcal{H}}$.

Observe that there exists different $\vec{\varepsilon}$ and $\vec{\varepsilon}'$ that yield the same transcript between $\overline{\mathcal{U}}^*$ and $\mathcal{S}$ because $\vec{e}$ is uniquely determined from $\vec{\varepsilon}$ as $\overline{\mathcal{U}}^*$ and $\mathcal{S}$ are deterministic when Ω and Θ are fixed, and $\vec{\varepsilon}$ has more variation than $\vec{e}$. We classify elements in Succ into classes so that elements in the same class yield the same transcript between $\overline{\mathcal{U}}^*$ and $\mathcal{S}$. Precisely, we introduce a mapping, $\lambda : \vec{\varepsilon} \mapsto \vec{e}$, i.e., $\lambda(\vec{\varepsilon}) = \vec{e}$, and define an equivalence relation between elements in Succ as $\vec{\varepsilon} \sim \vec{\varepsilon}'$ iff $\lambda(\vec{\varepsilon}) = \lambda(\vec{\varepsilon}')$. Let $E(\vec{\varepsilon})$ denote the equivalence class where $\vec{\varepsilon}$ belongs.

Next we classify Succ in a different way. Let $Br(\vec{\varepsilon}, \vec{\varepsilon}') = i \in \{0, \ldots, \ell+1\}$ denote the 'branching' index such that $\vec{\varepsilon}_i = \vec{\varepsilon}'_i$ and $\vec{\varepsilon}_{[i]} \neq \vec{\varepsilon}'_{[i]}$ (define $Br(\vec{\varepsilon}, \vec{\varepsilon}') = 0$ if $\vec{\varepsilon} = \vec{\varepsilon}'$). For $\vec{\varepsilon} \in$ Succ, let $Br_{\max}(\vec{\varepsilon}) = i$ denote an index where $\vec{\varepsilon}$ is most likely to branch compared with randomly taken element of $E(\vec{\varepsilon})$. Formally, for $\vec{\varepsilon} \in$ Succ, $Br_{\max}(\vec{\varepsilon}) = i$ iff

$$\#\{\vec{\varepsilon}' \in E(\vec{\varepsilon}) \mid Br(\vec{\varepsilon}, \vec{\varepsilon}') = i\} = \max_{j \in \{1, \ldots, \ell+1\}} (\#\{\vec{\varepsilon}' \in E(\vec{\varepsilon}) \mid Br(\vec{\varepsilon}, \vec{\varepsilon}') = j\})$$

(if two j's happen to give the same maximal value, define i with the larger j). Now, the elements in Succ is classified by $Br_{\max}$. Let $\mathcal{E}_{i^*}$ denotes the largest class among them. Formally, $\mathcal{E}_{i^*} = \{\vec{\varepsilon} \mid Br_{\max}(\vec{\varepsilon}) = i^*\}$ where $i^* \in \{1, \ldots, \ell+1\}$ is defined so that it satisfies $\#\{\vec{\varepsilon} \mid Br_{\max}(\vec{\varepsilon}) = i^*\} = \max_{j \in \{1, \ldots, \ell+1\}}(\#\{\vec{\varepsilon}' \mid Br_{\max}(\vec{\varepsilon}') = j\})$. Note that $i^* = 0$ does not happen since $Br_{\max}(\vec{\varepsilon}) = 0$ happens only if $\#E(\vec{\varepsilon}) = 1$ and such $\vec{\varepsilon} \in$ Succ is at most $q^\ell - 1$. From the definition, it is clear that

$$\frac{\#\mathcal{E}_{i^*}}{\#\mathcal{E}} \geq \eta_1/(\ell+2).$$

Note that $\#\mathcal{E} = q^{l+1}$.

For $\vec{\varepsilon} \in \mathcal{E}_{i^*}$, define Γ_{i^*} and $\xi_{i^*}(\vec{\varepsilon})$ as

$$\Gamma_{i^*}(\vec{\varepsilon}) = \{\varepsilon \mid {}^{\exists}\vec{\varepsilon}' \in \text{Succ} \,;\, \vec{\varepsilon}'_{i^*} = \vec{\varepsilon}_{i^*} \ \wedge \ \vec{\varepsilon}'_{[i^*]} = \varepsilon\},$$
$$\xi_{i^*}(\vec{\varepsilon}) = \frac{\#\Gamma_{i^*}(\vec{\varepsilon})}{q}.$$

Intuitively, $\Gamma_{i^*}(\vec{\varepsilon})$ is the number of good (potentially successful) choices as the i-th challenge when first i^*-1 challenges are fixed according to $\vec{\varepsilon}$. And ξ_{i^*} is its fraction. We can obtain the following claim using the standard heavy low lemma technique [11]. Note that if $\vec{\varepsilon}$ is randomly selected from $\mathcal{E}$, the probability that $\vec{\varepsilon} \in \mathcal{E}_{i^*}$ is at least $\eta_1/(\ell+2)$, where $\eta_1 = \eta/2q_H^{\ell+1}$.

Claim. $\Pr_{\vec{\varepsilon} \in \mathcal{E}_{i^*}}[\xi_{i^*}(\vec{\varepsilon}) \geq \eta_1/2(\ell+2)] > 1/2.$

Proof. Assume that there exits a fraction, F, of $\mathcal{E}_{i^*}$ such that $\#F \geq \#\mathcal{E}_{i^*}/2$ and $\forall \vec{\varepsilon} \in F$, $\xi_{i^*}(\vec{\varepsilon}) < \eta_1/2(\ell+2)$. We then obtain, for each $\vec{\varepsilon} \in F$,

$$\#\{\vec{\varepsilon}' \in \text{Succ} \mid \vec{\varepsilon}'_{i^*} = \vec{\varepsilon}_{i^*}\} < q \times (\eta_1/2(\ell+2)) \times q^{\ell - i^* + 1} = q^{\ell - i^* + 2}\eta_1/2(\ell+2).$$

Since $\sum_{\vec{\varepsilon}\in F} \#\{\vec{\varepsilon}' \in \mathrm{Succ} \mid \vec{\varepsilon}'_{i^*} = \vec{\varepsilon}_{i^*}\} \geq \#F \geq \#\mathcal{E}_{i^*}/2 = \frac{q^{\ell+1}\eta_1}{2(\ell+2)}$, the variation of the first (i^*-1) challenges of the elements in F, i.e. $\#\{\vec{\varepsilon}_{i^*} \mid \vec{\varepsilon} \in F\}$, is strictly greater than

$$\frac{q^{\ell+1}\eta_1/2(\ell+2)}{q^{\ell-i^*+2}\eta_1/2(\ell+2)} = q^{i^*-1}.$$

As i^*-1 challenges have at most q^{i^*-1} variations, this is contradiction.

For each $\vec{\varepsilon} \in \mathcal{E}_{i^*}$, we arbitrarily fix a partner of $\vec{\varepsilon}$, denoted as $\vec{\varepsilon}' = Prt(\vec{\varepsilon})$, that satisfies $\vec{\varepsilon}' \neq \vec{\varepsilon}$ and $\vec{\varepsilon}' \in E(\vec{\varepsilon})$. Let $\hat{\mathcal{E}}_{i^*}$ be a set that consists of all elements of $\mathcal{E}_{i^*}$ and their partners. That is, $\hat{\mathcal{E}}_{i^*} = \mathcal{E}_{i^*} \cup \{\vec{\varepsilon}' \mid \vec{\varepsilon}' = Prt(\vec{\varepsilon})\}$. We then call a triple, $(\vec{\varepsilon}, \vec{\varepsilon}', \vec{\varepsilon}'')$, a triangle, iff $\vec{\varepsilon} \in \mathcal{E}_{i^*}$, $\vec{\varepsilon}' = Prt(\vec{\varepsilon})$, $\vec{\varepsilon}'' \in \mathrm{Succ}$, $\vec{\varepsilon}_{i^*} = \vec{\varepsilon}''_{i^*}$, $\vec{\varepsilon}_{[i^*]} \neq \vec{\varepsilon}''_{[i^*]}$, and $\vec{\varepsilon}'_{[i^*]} \neq \vec{\varepsilon}''_{[i^*]}$. For a triangle, $(\vec{\varepsilon}, \vec{\varepsilon}', \vec{\varepsilon}'')$, we call $(\vec{\varepsilon}, \vec{\varepsilon}'')$ and $(\vec{\varepsilon}', \vec{\varepsilon}'')$ a side of the triangle, and call $(\vec{\varepsilon}, \vec{\varepsilon}')$ the base of the triangle. The number of triangles is at least

$$\#\mathcal{E}_{i^*}/3 \geq q^{\ell+1}\eta_1/(6(\ell+2)).$$

Here w.o.l.g., we assume that $y = g^x$ is chosen according to Ω. Clearly, from the definition, at least one of x and w can be calculated from $\mathcal{M}$'s view regarding a side of a triangle, $(\vec{\varepsilon}, \vec{\varepsilon}'')$ (and $(\vec{\varepsilon}', \vec{\varepsilon}'')$). We now denote $(\vec{\varepsilon}, \vec{\varepsilon}'') \rightarrow w$ iff w is extracted from $\mathcal{M}$'s view regarding $\vec{\varepsilon}$ and $\vec{\varepsilon}''$, otherwise $(\vec{\varepsilon}, \vec{\varepsilon}'') \not\rightarrow w$. It is easy to see that the following claim holds.

Claim. Let $(\vec{\varepsilon}, \vec{\varepsilon}', \vec{\varepsilon}'')$ be a triangle. Suppose that $(\vec{\varepsilon}, \vec{\varepsilon}'') \not\rightarrow w$ and $(\vec{\varepsilon}', \vec{\varepsilon}'') \not\rightarrow w$. Then $(\vec{\varepsilon}, \vec{\varepsilon}') \not\rightarrow w$.

Proof. Let δ, δ', and δ'' correspond to $\vec{\varepsilon}, \vec{\varepsilon}'$, and $\vec{\varepsilon}''$. If $(\vec{\varepsilon}, \vec{\varepsilon}'') \not\rightarrow w$, then $\delta = \delta''$. If $(\vec{\varepsilon}', \vec{\varepsilon}'') \not\rightarrow w$, then $\delta' = \delta''$. Therefore, $\delta = \delta'$. It follows that $(\vec{\varepsilon}, \vec{\varepsilon}') \not\rightarrow w$.

We then obtain the following claim:

Claim. For at least $1/5$ fraction of sides, w is extracted with probability at least $1/3$ over Ω.

Proof. If x (w resp.) is included in Ω, then w (x resp.) is called a *good* witness, which we want to extract. Suppose that a good witness is not obtained from at least $4/5$ fraction of sides with probability at least $2/3$ over Ω. It then follows from Claim 4 that a good witness is not obtained from at least $3/5$ fraction of base, $(\vec{\varepsilon}, \vec{\varepsilon}')$, with probability at least $2/3$ over Ω. When a good witness is not obtained from at least $3/5$ fraction of base, $(\vec{\varepsilon}, \vec{\varepsilon}')$, the result is (non-negligibly) biased by the witness with Ω. That is, the biased result occurs with probability at least $2/3$ over Ω. Since the information of a base, $(\vec{\varepsilon}, \vec{\varepsilon}')$, is independent of the witness the simulator already has as a part of Ω, this contradicts that a biased result should occur with probability (over Ω) less than $1/2 + 1/poly(n)$ for any polynomial $poly$.

Finally we will evaluate the total success probability of $\mathcal{M}$. The probability that i^* is correctly guessed is at least $\frac{1}{\ell+1}$. When $\vec{\varepsilon}$ is randomly selected, $\vec{\varepsilon} \in \hat{\mathcal{E}}_{i^*}$ and $\xi_{i^*}(\vec{\varepsilon}) \geq \eta_1/2(\ell+2)$ with probability at least $\frac{\eta_1}{2(\ell+2)}$. $\vec{\varepsilon}''_{[i^*]} \in \Gamma_{i^*}(\vec{\varepsilon})$ is selected with probability at least $\xi_{i^*}(\vec{\varepsilon}) \geq \eta_1/2(\ell+2)$. Then $(\vec{\varepsilon}, \vec{\varepsilon}''_{[i^*]}) \rightarrow w$ with probability greater than $1/15$ $(= (1/3) \times (1/5))$. Thus, in total, the success probability of $\mathcal{M}$ is $\frac{\eta_1^2}{60(\ell+1)(\ell+2)^2}$, where $\eta_1 = \eta/2q_H^{\ell+1}$.

Now we consider the case where the common information is not all the same. Given successful forger $\mathcal{U}_B^*$ of game B, we construct successful forger $\mathcal{U}_F^*$ of the fixed-info version of game B.

The basic strategy of constructing machine $\mathcal{U}_F^*$ is to screen the conversation between $\mathcal{U}_B^*$ and $\mathcal{S}$ except for the ones involving info that $\mathcal{U}_B^*$ will output as a result of forgery. $\mathcal{U}_F^*$ simulates $\mathcal{S}$ with regard to the blocked conversations by assigning g^w to z with randomly picked w. The simulation works perfectly thanks to the witness indistinguishability of the signature issuing protocol.

Now, we describe $\mathcal{U}_F^*$ in detail. Let q_F be the maximum number of queries for $\mathcal{F}$ from $\mathcal{U}_B^*$. Similarly, let q_S be the maximum number of queries for $\mathcal{S}$. Observe that $\mathcal{F}$ is defined at most at $q_F + q_S$ points while $\mathcal{U}_B^*$ plays game B. For simplicity, we assume that all queries to $\mathcal{F}$ are different.

1. Select J randomly from $\{1, \ldots, q_F + q_S\}$.
2. Run $\mathcal{U}_B^*$ simulating $\mathcal{F}, \mathcal{H}$ and signer $\mathcal{S}$ as follows.
   - For j-th query to $\mathcal{F}$, return z such that
     - $z := g_j^w$ where $w_j \in_R \mathbb{Z}_q$ for $j \neq J$, or
     - $z := \mathcal{F}(\mathsf{info}_J)$ (i.e. ask $\mathcal{F}$) if $j = J$.

     If z has been already defined at query point info_j, return that value.
   - For all queries to $\mathcal{H}$, ask $\mathcal{H}$.
   - If $\mathcal{U}_B^*$ initiates the signature issuing protocol with regard to info_J, $\mathcal{U}_F^*$ negotiates with $\mathcal{S}$ in such a way that they agree on info_J (this is possible because Ag is deterministic). $\mathcal{U}_F^*$ then behaves transparently so that $\mathcal{U}_B^*$ can talk with $\mathcal{S}$.
   - If $\mathcal{U}_B^*$ initiates the signature issuing protocol with regard to info_j where $j \neq J$, $\mathcal{U}_F^*$ simulates $\mathcal{S}$ by using w_j.
3. Output what $\mathcal{U}_B^*$ outputs.

Note that Ag is decided by $\mathcal{U}_B^*$ at the beginning of step 2. $\mathcal{U}_F^*$ is successful if $\mathcal{U}_B^*$ is successful and correct J is chosen so that the final output of $\mathcal{U}_B^*$ contains info_J. Therefore, the success probability of $\mathcal{U}_F^*$ is $\frac{\mu}{q_F+q_S}$ where μ is the success probability of $\mathcal{U}_B^*$. □

5 Conclusion

We have presented a formal definition of partially blind signature schemes and constructed an efficient scheme based on the Schnorr signature scheme. We then gave a proof of security in the random oracle model assuming the intractability of the discrete logarithm problem.

Although we have shown a particular construction based on Schnorr signature, the basic approach of constructing WI protocols and the proof of security do not substantially rely on the particular structure of the underlying signature scheme. Accordingly, a signature scheme derived from public-coin honest verifier zero-knowledge can be plugged into our scheme if it can be blinded. It covers, for instance, Guillou-Quisquater signature and some variants of modified ElGamal signature schemes.

As we mentioned, one can easily transform fully blind signature schemes from partially blind ones. We have shown that the reverse is possible; partially blind signature schemes can be derived from fully blind witness indistinguishable signature schemes.

References

1. M. Abe and J. Camenisch. Partially blind signatures. In the 1997 Symposium on Cryptography and Information Security, 1997.
2. M. Abe and E. Fujisaki. How to date blind signatures. In K. Kim and T. Matsumoto, editors, *Advances in Cryptology – ASIACRYPT '96*, volume 1163 of *Lecture Notes in Computer Science*, pages 244–251. Springer-Verlag, 1996.
3. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *First ACM Conference on Computer and Communication Security*, pages 62–73. Association for Computing Machinery, 1993.
4. S. Brands. Untraceable off-line cash in wallet with observers. In D. Stinson, editor, *Advances in Cryptology — CRYPTO '93*, volume 773 of *Lecture Notes in Computer Science*, pages 302–318. Springer-Verlag, 1993.
5. D. Chaum. Blind signatures for untraceable payments. In D. Chaum, R. Rivest, and A. Sherman, editors, *Advances in Cryptology — Proceedings of Crypto '82*, pages 199–204. Prenum Publishing Corporation, 1982.
6. D. Chaum. Elections with unconditionally-secret ballots and disruption equivalent to breaking RSA. In C. G. Günther, editor, *Advances in Cryptology — EUROCRYPT '88*, volume 330 of *Lecture Notes in Computer Science*, pages 177–189. Springer-Verlag, 1988.
7. D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In S. Goldwasser, editor, *Advances in Cryptology — CRYPTO '88*, volume 403 of *Lecture Notes in Computer Science*, pages 319–327. Springer-Verlag, 1990.
8. R. Cramer. personal communication, 1997.
9. R. Cramer, I. Damgård, and B. Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Y. G. Desmedt, editor, *Advances in Cryptology — CRYPTO '94*, volume 839 of *Lecture Notes in Computer Science*, pages 174–187. Springer-Verlag, 1994.
10. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology — CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer-Verlag, 1985.
11. U. Feige, A. Fiat, and A. Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1:77–94, 1988.
12. A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In J. Seberry and Y. Zheng, editors, *Advances in Cryptology —*

AUSCRYPT '92, volume 718 of *Lecture Notes in Computer Science*, pages 244–251. Springer-Verlag, 1993.

13. S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal of Computing*, 17(2):281–308, April 1988.
14. L. C. Guillou and J.-J. Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In C. G. Günther, editor, *Advances in Cryptology — EUROCRYPT '88*, volume 330 of *Lecture Notes in Computer Science*, pages 123–128. Springer-Verlag, 1988.
15. A. Juels, M. Luby, and R. Ostrovsky. Security of blind digital signatures. In B. S. Kaliski Jr., editor, *Advances in Cryptology — CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 150–164. Springer-Verlag, 1997.
16. A. Menezes, P. Oorschot, and S. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.
17. K. Ohta and T. Okamoto. On concrete security treatment of signatures derived from identification. In H. Krawczyk, editor, *Advances in Cryptology — CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 354–369. Springer-Verlag, 1998.
18. T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In E. F. Brickell, editor, *Advances in Cryptology — CRYPTO '92*, volume 740 of *Lecture Notes in Computer Science*, pages 31–53. Springer-Verlag, 1993.
19. D. Pointcheval. Strengthened security for blind signatures. In K. Nyberg, editor, *Advances in Cryptology — EUROCRYPT '98*, Lecture Notes in Computer Science, pages 391–405. Springer-Verlag, 1998.
20. D. Pointcheval and J. Stern. Provably secure blind signature schemes. In K. Kim and T. Matsumoto, editors, *Advances in Cryptology – ASIACRYPT '96*, volume 1163 of *Lecture Notes in Computer Science*, pages 252–265. Springer-Verlag, 1996.
21. D. Pointcheval and J. Stern. Security proofs for signature schemes. In U. Maurer, editor, *Advances in Cryptology — EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 387–398. Springer-Verlag, 1996.
22. D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 2000.
23. RSA Laboratories. *PKCS #9: Selected Object Classes and Attribute Types*, 2.0 edition, February 2000.