Private Mutual Authentication and Conditional Oblivious Transfer

Stanisław Jarecki and Xiaomin Liu

University of California, Irvine

Abstract. A bi-directional Private Authentication, or *Unlinkable Secret Handshake*, allows two parties to authenticate each other as certified by given certification authorities (i.e. affiliated with given groups), in a *mutually private way*, in the sense that the protocol leaks no information about either participant to a party which does not satisfy that participant's authentication policy. In particular, the protocol hides what group this participant belongs to, and protocol instances involving the same participant are unlinkable. We construct the first realization of such private authentication using $O(1)$ exponentiations and bilinear maps, secure under Strong Diffie-Hellman and Decisional Linear assumptions.

Our protocols rely on a novel technical tool, a family of efficient Private Conditional Oblivious Transfer (COT) protocols, secure under DDH, for languages defined by modular arithmetic constraints (e.g. equality, inequality, sums, products) on discrete-log representations of some group elements. (Recall that $(w_1, ..., w_n)$ is a representation of C in bases $(g_1, ..., g_n)$ if $C = g_1^{w_1} ... g_n^{w_n}$.) A COT protocol for language L allows sender S to encrypt message m "under" statement x so that receiver R gets m only if R holds a witness for membership of x in L, while S learns nothing. A *private* COT for L hides not only message m but also statement x from any R that does not know a witness for x in L.

1 Introduction

Authentication Privacy and Mutual Authentication. It seems evident that if party A authenticates itself to some verifier then A must necessarily reveal some information about itself in the process. At the minimum, an authentication protocol seemingly needs to reveal that A is credentialed by a given Certification Authority (CA), because the goal of (policy-based) uni-directional authentication is to let any verifier learn whether A holds valid credentials from a given CA. However, in the case of a *mutual* authentication, where A authenticates itself to B as certified by a CA of B's choice but it cares to do so only if B is itself appropriately certified by the CA of A's choice, we can ask whether we can protect each party's privacy fully, including its affiliation with particular CA, against any entity which does not satisfy this party's authentication policy. In other words, we ask for a protocol which mimics the following ideal private mutual authentication functionality, or an *(unlinkable) Secret Handshake* (SH):

A and B input their certificates and authentication policies, and the functionality returns 1 if A's certificate matches B's policy and B's certificate matches A's policy, and 0 otherwise. To be practical, SH scheme should match the performance of a standard non-private PKI system (or a Group Signature Scheme): The certificate of each user should be short and re-usable, the CA's public key should be short, authentication protocol should take $O(1)$ rounds and public-key operations, revocation information should be at most linear in the number of revoked players, and the scheme should support escrow, i.e. protocol participants should be efficiently traceable by a trusted party from protocol transcripts.

Applications of private mutual authentication range from peer-to-peer groups to law-enforcement agencies who might be concerned with their privacy in the sense of not wanting to publicly advertise the fact of their membership in a given group. They might want to do this due to privacy concerns, e.g. in the case of parties or clubs, to business concerns, e.g. a company who wants its employees or trading partners to be unrecognizable by competition, or due to security concerns, e.g. in the case of members of some law-enforcement agency whose safety is enhanced if their membership in the agency is not advertised. Secret Handshakes allow members in any such group to constrain the dissemination of the fact of their group membership only to other group members. Any group member can still identify other members by engaging them in an authentication protocol, but using privacy escrow and revocation mechanisms a group can revoke any member who poses a privacy risk to others.

Related Work. There exist efficient *linkable* Secret Handshakes which hide the participants' policy and source of certificates [BDS+03,CJT04,JKT07,JL08], but they publicly reveal unique tokens assigned to each certificate, thus making protocol instances executed by the same party linkable. SH's can also be thought of as a bidirectional counterpart to private *uni-directional* authentication, i.e. identity-escrow [KP97], group signatures [CvH91], or unlinkable credentials [CL01], but uni-directional authentication unconditionally reveals prover's affiliation to any verifier. An SH scheme without key escrow would be implied by *key-private* broadcast encryption, whose ciphertext cannot be linked to the broadcast encryption key. The two parties could then privately establish an authenticated key by encrypting nonces under broadcast encryption keys associated with their CA's. However, the ciphertexts of existing broadcast encryption schemes, e.g. [NNL02,BGW05], can be easily linked to the revocation lists corresponding to their encryption keys. The key-private broadcast encryption of [BBW06] has limited applicability because its ciphertext size is linear in group size, while the key-private broadcast encryption of [JL07] is stateful, and the corresponding SH scheme works only if group members have roughly synchronized certificate revocation lists. (Private) attribute-based encryption [GPSW06,BSW07] allows the sender to encrypt a nonce so that it can be decrypted only by a holder of a certificate for specified attributes issued by some public key, and the ciphertext can hide this attribute from anyone who does not have the corresponding certificate. However, the ciphertext in these schemes hides only the attributes and not the public key that issues the certificates.

Our Contributions. Our first contribution is the first practical private bi-directional authentication scheme a.k.a. an *(unlinkable) Secret Handshake* (SH), and the first practical private envelope scheme, a.k.a. an *Anonymous Credential* scheme (AC), i.e. an envelope scheme with the privacy properties corresponding to SH's. Namely, the receiver can recover an encrypted message if and only if its certificate matches sender's authorization policy, but the protocol hides the sender's policy from any un-authorized receiver and it hides all information about the receiver from the sender. All our schemes support certificate revocation and privacy escrow, i.e. the group manager can recover otherwise hidden identity of protocol participants from protocol transcripts. Our SH protocol has $O(1)$ communication rounds (3 in ROM) and requires about 40 exponentiations and 6 bilinear maps per player, with additional $2r$ bilinear maps if r is the size of the revocation list, and our AC scheme has twice smaller costs because it is actually just a one-sided version of our SH scheme.

Our technical contribution is an enabling tool of our SH and AC schemes, a family of efficient *Conditional Oblivious Transfer* (COT) protocols for certain cryptographically useful class of relations. A COT protocol *for relation* $\mathcal{R}$ is a protocol between a sender S and a receiver R, which allows S, running on input a statement x and a message m, to disclose m to R if and only if R holds a *witness* for x in the language associated with relation $\mathcal{R}$, i.e. a string w s.t. $(x, w) \in \mathcal{R}$. The protocol is oblivious in the sense that S does not learn anything, not even whether there *exists* w which is a valid witness for sender's statement x. We call such COT protocol *private* if it also hides S's statement x from any receiver who does not hold a valid witness for x. COT is implied by secure two-party computation [Yao86], but it was introduced as a primitive in [COR99], extended to private COT in [Cre00], and later considered e.g. in [AIR01,BK04,LL07]. All these works used slightly different terminology than ours, calling inputs x and w just bitstrings and not *statement* and *witness* as we do, and they constructed COT protocols at the cost of $O(1)$ modular exponentiations for an equality relation on bitstrings [AIR01], i.e. $x = w$, and at the $O(k)$-exponentiations cost for monotonic Boolean formulas of size k [COR99,Cre00,AIR01,LL07,BK04].

We show practical COT protocols for relations that commonly appear in various cryptographic protocols e.g. group signatures, e-cash schemes, or threshold schemes. Recall that a *representation* of a group element C in bases $\mathbf{G} = (g_1, ..., g_m)$ is a vector $\mathbf{w} = (w_1, ..., w_m)$ s.t. $C = \prod_{i=1}^{m} (g_i)^{w_i}$. We exhibit two private COT protocols, one perfectly secure for the receiver and the other perfectly secure for the sender, with the computationally protected side in both protocols secure under the DDH assumption, for any relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$ of the following form: The relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$, for a predicate Φ on an $n \times m$ matrix $\mathbf{w}$, consists of pairs $((\mathbf{C}, \mathbf{G}), \mathbf{w})$ s.t. $\Phi(\mathbf{w}) = 1$ and the i-th row of $\mathbf{w}$ is a representation of the i-th element in $\mathbf{C}$ in bases formed by the i-th row of $\mathbf{G}$. Both protocols we propose use a single execution of a ZKPK of values in the same matrix $\mathbf{w}$ committed using the Pedersen commitment scheme [Ped91] (or its computationally-private but perfectly-binding modification) s.t. $\Phi(\mathbf{w}) = 1$. The cost of our COT protocols is the cost of the ZKPK plus about $(4 \cdot |\mathbf{w}|)$ expo-

nentiations for either party. Note that there exist ZKPK's for various conditions on values committed in Pedersen commitments, e.g. equality or inequality of modular sums, products, or inverses, all using $O(1)$ exponentiations. Our results transform any such ZKPK proof system into a private COT protocol for the same relation, at the cost comparable to the cost of the ZKPK. Previous COT constructions do not enable efficient COT protocols for such relations, and since many cryptographic applications rely on efficiently provable relations on committed values, practical COT protocols for such relations might enable new privacy-protecting mechanisms beyond our SH and AC schemes.

Note that private COT forms an encryption counterpart to a zero-knowledge proof of knowledge: The verifier can use a COT protocol to encrypt some message m "under" a statement x, and the COT protocol ensures that the prover can decrypt m only if she holds a valid witness w for x. However, private COT's can enable higher level that what is zero-knowledge proofs achieve: Consider a server who wants to grant access to some resource m to a client if and only if the client's credential cert satisfies the sender's authorization policy Pol, i.e. $\mathsf{Ver}(\mathsf{Pol}, \mathsf{cert}) = 1$. If a client proves in zero knowledge that $\mathsf{Ver}(\mathsf{Pol}, \mathsf{cert}) = 1$, this reveals the fact that the client holds a certificate which satisfies policy Pol to *any* party who engages the client in this zero-knowledge proof as a verifier. Moreover, the server who engages in a proof system as a verifier on its statement Pol, even if this proof system is zero-knowledge, might also end up revealing this statement to any party, whether or not this party holds a valid witness for this statement. In contrast, the privacy of both parties is protected if the server sends m to the client in a private COT protocol for relation Ver. Thus private COT protocols for relation Ver would enable (fully) private envelopes, and a bi-directional version of this envelope would make a private authentication scheme, and in particular this is how our AC and SH schemes are constructed.

Organization. We start with a technical roadmap in Section 2. We set notation in Section 3. We define private COT in Section 4. In Section 5 we construct a private COT for relations on discrete logarithm representations with perfect security for the receiver. (For lack of space we have to omit from these proceedings our alternative protocol with perfect security for the sender.) In Section 6 we define SH schemes and construct such scheme on the basis of a group signature by Boneh and Shacham [BS04] and a COT protocol like that of Section 5.

2 Technical Roadmap

We construct an unlinkable secret handshake using a group signature scheme and a COT protocol on an appropriate relation. One possible way of doing this could be as follows. Party A issues a group signature on a challenge message, and B sends a nonce to A via a private COT protocol, s.t. A receives it if and only if A's commitment opens to a group signature which verifies under the key specified in B's authentication policy. Then the roles of the two parties are reversed: B creates and commits to its signature, and A sends its nonce to B via the COT protocol on the same condition applied to B's commitment and A's authentication policy. The first technical challenge lies in handling revocations:

It's not clear how to check whether a signature is issued by some revoked member when the signature is hidden behind the commitment, unless the signer (receiver in the COT protocol) also attaches a proof that the committed signature is *not* issued by anyone in the *receiver's* revocation list. This seems hard because the revocation list assumed by each party should also be hidden, or otherwise the affiliation of that party is immediately revealed.

To avoid the problem caused with revocation we turn to the group signature (GS) scheme with verifier-local revocation (VLR) introduced by Boneh and Shacham [BS04]. In a VLR-GS scheme the signer's certificate consists of two parts: The first is a random revocation token, unrelated to the group's public key, and the second is essentially a group manager's signature on this revocation token. In the VLR-GS scheme the signer first commits to its token using a commitment scheme which is *unlinkable* without the knowledge of the committed token, but is *traceable* given the token. (This latter property enables efficient revocation.) The group signature then consists of this committed token and a Zero-Knowledge Proof of Knowledge (ZKPK) of group manager's signature on this committed token, made non-interactive using the Fiat-Shamir heuristic.

We construct an unlinkable SH scheme using the same components of the VLR-GS scheme, but replacing the above NI-ZKPK proof with a private COT scheme for the same relation. Namely party A commits to its token, B uses the traceability procedure to check if the committed token has not been revoked, and if the check passes then B sends a nonce to A via a COT protocol s.t. A receives the nonce if and only if A has a group manager's signature on the committed token, and then the roles are reversed. The reason this yields an efficient SH construction when the VLR-GS scheme is instantiated with the scheme of [BS04] is that the relation involved in the above COT protocol belongs to the class $\mathcal{R}_{\mathsf{REP}(\Phi)}$ of relations on discrete-log representations satisfying some arithmetic constraints. In other words, the commitment to a token and the group public key can be represented as a vector $\mathbf{C}$ and a matrix $\mathbf{G}$ of group elements, while the decommitment and the group manager's signature on the committed token form a matrix of exponents $\mathbf{w}$ s.t. $\mathbf{w}$ satisfies certain set of arithmetic constraints Φ *and* each row of $\mathbf{w}$ is a discrete-log representation of a corresponding element in $\mathbf{C}$ in a vector of bases form by the same row of $\mathbf{G}$.

Technically, the security argument for the above SH scheme follows easily from the unforgeability of the group manager's signatures *if* the COT protocol guarantees extractability of a witness for the receiver's statement from a receiver which tells some information about the transferred message: In such case an adversary which breaks the security of the authentication scheme immediately implies efficient computation of a forgery of the group manager's signature, since the witness to the sender's statement must be a valid signature on an unrevoked token, and an unrevoked token is an unsigned message from adversary's point of view. A privacy argument for this SH scheme will be similarly aided if the COT protocol also guarantees extractability of a witness from a receiver which tells any information about the sender's statement. This is why the security and

privacy notion we give for a COT protocol in Section 4 requires extraction of inputs from a "successful" receiver.

Finally, we sketch our COT construction for relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$. The receiver cannot just run a ZKPK of $\mathbf{w}$ s.t. the arithmetic constraint Φ is satisfied and $\mathbf{w}$ is the representation of $\mathbf{C}$ in bases $\mathbf{G}$, where $(\mathbf{C}, \mathbf{G})$ is the statement assumed by the receiver, because this would reveal this part of the receiver's inputs to any sender, and in particular it could reveal the CA who issued the receiver's certificate. Instead, the receiver can independently commit to vector $\mathbf{w}$ and perform a zero-knowledge proof of knowledge of a committed $\mathbf{w}$ which satisfies constraint Φ. This protects all information about $\mathbf{w}$ (except that $\Phi(\mathbf{w}) = 1$, but this is presumably true of any party engaging in this protocol) and it also ensures efficient extraction of some $\mathbf{w}$ s.t. $\Phi(\mathbf{w}) = 1$ from any malicious receiver. If the receiver's proof verifies then the sender follows an encryption-like procedure - somewhat reminiscent of Cramer-Shoup's projective hash [CS01] – which transfers sender's message M to the receiver but ensures that the receiver gets no information about either the sender's message M or its statement $(\mathbf{C}, \mathbf{G})$ unless the committed matrix $\mathbf{w}$ is a representation of $\mathbf{C}$ in bases $\mathbf{G}$. Looking ahead, in the COT protocol of Figure 2 this additional commitment to $\mathbf{w}$ is denoted $\mathbf{D}$, and the encryption-like procedure outputs $\mathbf{E}$, $\mathbf{F}$, $\hat{K}$ on inputs $\mathbf{C}, \mathbf{G}, \mathbf{D}$ and M.

3 Cryptographic Setting and Notation

Throughout the paper we assume that $\mathbb{G}$ is a multiplicative group of prime order q, and that g is its generator. Our security statements in section 5 assume an exact security version of the DDH assumption, i.e. we say that DDH is (t, ϵ)-hard in group $\mathbb{G}$ if any t-time algorithm $\mathcal{A}$ has at most ϵ advantage in distinguishing distributions $\{(g, g^a, g^b, g^{ab})\}_{a,b \leftarrow_R \mathbb{Z}_q}$ and $\{(g, g^a, g^b, g^c)\}_{a,b,c \leftarrow_R \mathbb{Z}_q}$. If $\mathcal{A}$ is a probabilistic algorithm then $A(x; r)$ denotes an output of A on x and random tape r. We use bold letters to denote vectors or matrices. We write $\mathbf{w} \in S^{n \times m}$ to denote a matrix $\mathbf{w}$ with dimensions $n \times m$ and elements in set S. We use $\mathbf{w}[i, j]$ to designate an element in the i-th row and j-th column of $\mathbf{w}$.

4 Definition of Private Conditional Oblivious Transfer

A COT protocol for message space $\mathcal{M}$ and relation $\mathcal{R}$ (and the language $L_{\mathcal{R}}$ implied by $\mathcal{R}$ as well as an implicit universe of "statement-looking" strings $\mathcal{U}_{\mathcal{R}} \supseteq L_{\mathcal{R}}$) consists of two probabilistic interactive algorithms S and R, which execute on S's private inputs a message M in $\mathcal{M}$ and a bitstring x, and on R's private input a bitstring w. At the end of the interaction, R outputs message M if and only if $(x, w) \in \mathcal{R}$, and S has no output. (See Figure 1.) A COT protocol must meet the following basic properties:

Definition 1 (Completeness). *A COT protocol for relation $\mathcal{R}$ and message space $\mathcal{M}$ is* complete *if for any $(x, w) \in \mathcal{R}$ and any $M \in \mathcal{M}$, at the end of the interaction between $\mathsf{S}(x, M)$ and $\mathsf{R}(w)$, R outputs M.*

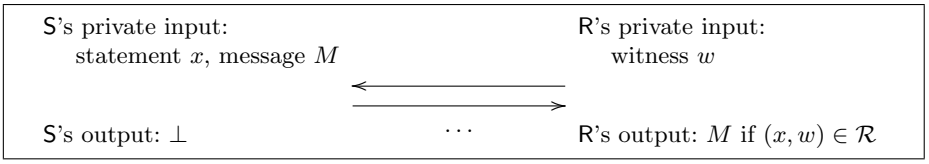

Fig. 1. Functionality of a COT scheme for relation $\mathcal{R}$ between sender S and receiver R

Definition 2 (Security). *A COT protocol for relation $\mathcal{R}$ and message space $\mathcal{M}$ is* (t, ϵ)-secure *if for any $x \notin L_{\mathcal{R}}$, any $M_0, M_1 \in \mathcal{M}$, any t-time algorithm $\mathcal{A}$, and any auxiliary information z,*

$$\left| Pr[\mathcal{A}^{\mathsf{S}(x,M_0)}(x, M_0, M_1, z) = 1] - Pr[\mathcal{A}^{\mathsf{S}(x,M_1)}(x, M_0, M_1, z) = 1] \right| \leq \epsilon$$

where the probabilities are taken over the randomness of $\mathcal{A}$ and S.

Definition 3 (Receiver Privacy). *A COT protocol for relation $\mathcal{R}$ is* (t, ϵ)-receiver private *if for any t-time algorithm $\mathcal{A}$, any w_0, w_1, and any auxiliary information z,*

$$\left| Pr[\mathcal{A}^{\mathsf{R}(w_0)}(w_0, w_1, z) = 1] - Pr[\mathcal{A}^{\mathsf{R}(w_1)}(w_0, w_1, z) = 1] \right| \leq \epsilon$$

where the probabilities are taken over the randomness of $\mathcal{A}$ and R.

However, the above security property has several limitations. First, it allows the protocol to reveal sender's message to any receiver if the sender's statement x *is* in the language. A more useful notion would require that the message is revealed only to the receiver who holds a valid witness for x. This requirement can be captured via extractability, i.e. if the receiver distinguishes the execution of $S(x, M_0)$ and $S(x, M_1)$ then a witness w for x can be efficiently extracted from this receiver. Moreover, a *private* COT protocol should protect sender's statement in a similar way, i.e. if the receiver distinguishes the execution of $S(x_0, M)$ and $S(x_1, M)$ then a witness for *either* x_0 *or* x_1 can be extracted from this receiver. We capture both of these properties in a notion of *strong security and sender privacy* defined below. Technically, we define this notion in terms of distinguishing between a "real" sender $S(x, M)$ and a "simulated" sender $S(x', M')$ which runs on any statement x' and a *random* message M': An adversary can distinguish between these two only if a witness for x in $L_{\mathcal{R}}$ can be extracted from this adversary. Note that this notion implies the intuitive security and sender privacy properties discussed above. Moreover, this notion is convenient for arguing security of applications of a private COT because it implies that if an adversary distinguishes real and simulated protocols then the reduction can extract a witness for the real sender's statement.

Definition 4 (Strong Security and Sender Privacy). *A COT protocol for relation $\mathcal{R}$, statement universe $\mathcal{U}$, and message space $\mathcal{M}$ is* strongly secure and sender private with soundness error δ *if there exists an efficient extractor algorithm* Ext *and a polynomial $p(\cdot)$ s.t. for any $x, x' \in \mathcal{U}$, any $M \in \mathcal{M}$, any*

efficient probabilistic algorithm $\mathcal{A}$, any auxiliary information z (w.l.o.g. z contains x, x', M), and any randomness vector r, if

$$\epsilon_{\mathcal{A},z,r} \stackrel{\triangle}{=} \left| Pr_{\{\$_{\mathsf{S}}\}}[\mathcal{A}^{\mathsf{S}(x,M)}(z;r)=1] - Pr_{\{\$_{\mathsf{S}}, M' \leftarrow_R \mathcal{M}\}}[\mathcal{A}^{\mathsf{S}(x',M')}(z;r)=1] \right| > \delta$$

then $$Pr_{\{\$_{\mathsf{Ext}}\}}\left[(x,w)\in\mathcal{R} |\ w \leftarrow \mathsf{Ext}^{\mathcal{A}(z;r)}\right] \geq p(\epsilon_{\mathcal{A},z,r}-\delta)$$

where $\$_{\mathsf{S}}$ and $\$_{\mathsf{Ext}}$ are the randomness of S and Ext respectively.
In concrete security terms, we call a COT protocol $(t, t_{ext}, q_{ext}, d, e)$-strongly secure and sender private with soundness error δ *if the above requirement is satisfied for any t-time adversary $\mathcal{A}$, for polynomial $p(\epsilon) = d\epsilon^e$, and for algorithm Ext running in time t_{ext} and making at most q_{ext} calls to $\mathcal{A}$.*

5 Private COT Protocol for Relations on Representations

We give two constructions of a private COT protocol for any relation on so-called representations of group elements. Our first construction relies on a witness-indistinguishable proof of knowledge (WIPoK) for the same relation on values committed using Pedersen commitment scheme [Ped91]. The second construction needs a Strong WIPoK for the same relation on values committed in the following simple perfectly binding but computationally hiding commitment scheme: $Com_{g,h,y}(m) = (g^r, y^r h^m)$. Security and sender privacy of the first COT protocol construction relies on the DDH assumption and the strong soundness of the WIPoK proof system, while receiver privacy relies on witness-indistinguishability of the WIPoK. For the second construction, security and sender privacy relies on strong soundness of the SWIPoK, while receiver privacy relies on DDH assumption and strong witness-indistinguishability of the SWIPoK. We show the first construction below, while for lack of space we relegate our second construction to the full version of this paper.

Let $\mathbb{G}$ be a multiplicative group of prime order q. A *representation* of group element $C \in \mathbb{G}$ in bases $(g_1, ..., g_n) \in \mathbb{G}^n$ is any vector $(w_1, ..., w_n) \in (\mathbb{Z}_q)^n$ s.t. $C = \prod_{i=1}^{n} (g_i)^{w_i}$. Let Φ be a relation on sets of $n \times m$ elements in $\mathbb{Z}_q$, i.e. $\Phi : (\mathbb{Z}_q)^{n\times m} \rightarrow \{0,1\}$. Assume that Φ is satisfiable, i.e. there exists $\mathbf{w}$ s.t. $\Phi(\mathbf{w}) = 1$. Moreover, assume that there exists an efficient procedure to find (any) $\mathbf{w}$, s.t. $\Phi(\mathbf{w}) = 1$.

We define a language $\mathsf{REP}(\Phi)$ as a set of pairs $(\mathbf{G}, \mathbf{C})$,

$$\mathbf{G} = \begin{pmatrix} \mathbf{G}[1,1],\ \mathbf{G}[1,2],\ \ldots,\ \mathbf{G}[1,m] \\ \mathbf{G}[2,1],\ \mathbf{G}[2,2],\ \ldots,\ \mathbf{G}[2,m] \\ \cdots \\ \mathbf{G}[n,1],\ \mathbf{G}[n,2],\ \ldots,\ \mathbf{G}[n,m] \end{pmatrix} \in \mathbb{G}^{n\times m} \quad , \quad \mathbf{C} = \begin{pmatrix} \mathbf{C}[1] \\ \mathbf{C}[2] \\ \cdots \\ \mathbf{C}[n] \end{pmatrix} \in \mathbb{G}^n$$

s.t. $\exists\ \mathbf{w} \in (\mathbb{Z}_q)^{n\times m}$ s.t. $\Phi(\mathbf{w}) = 1$ and $\mathbf{C}[i] = \prod_{j=1}^{m} (\mathbf{G}[i,j])^{\mathbf{w}[i,j]}$ for all $i \in [1..n]$.

$\mathcal{R}_{\mathsf{REP}(\Phi)}$ is a relation corresponding to this language, i.e. set of pairs $((\mathbf{G}, \mathbf{C}), \mathbf{w})$ which satisfy the above conditions, and $\mathcal{U}_{\mathsf{REP}(\Phi)}$ includes all $(\mathbf{G}, \mathbf{C}) \in \mathbb{G}^{n\times m} \times \mathbb{G}^n$.

Construction of Private COT for Relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$. We construct a private COT protocol for $\mathcal{R}_{\mathsf{REP}(\Phi)}$ given a witness-indistinguish- able proof of knowledge (WIPoK) for the following language:

$$\mathsf{PedREP}_{g,h}(\Phi) \triangleq \left\{ \begin{array}{l} \mathbf{D} \in \mathbb{G}^{n\times m} \text{ s.t. } \exists \mathbf{w}, \mathbf{r} \in (\mathbb{Z}_q)^{n\times m} \text{ s.t. } \Phi(\mathbf{w}) = 1 \\ \qquad\qquad \text{and } \forall_{(i,j)\in[1..n]\times[1..m]}\ \mathbf{D}[i,j] = g^{\mathbf{w}[i,j]} \cdot h^{\mathbf{r}[i,j]} \end{array} \right\}$$

Note that PedREP is a trivial language, i.e. every $\mathbf{D} \in \mathbb{G}^{n\times m}$ is in PedREP. However, we require a (non-trivial) proof of knowledge of $(\mathbf{w}, \mathbf{r})$, given $\mathbf{D}$, s.t. $(\mathbf{D}, (\mathbf{w}, \mathbf{r})) \in \mathcal{R}_{\mathsf{PedREP}}$ where $\mathcal{R}_{\mathsf{PedREP}}$ is a relation corresponding to this language, i.e. set of pairs $(\mathbf{D}, (\mathbf{w}, \mathbf{r}))$ which satisfy above conditions.

Practical ZKPK (and WIPoK) proofs exist for languages $\mathsf{PedREP}_{g,h}(\Phi)$ for many useful constraints Φ. First, note that there exist efficient HVZKPK proof systems with special HVZK and properties for many constraints involving values committed in Pedersen commitment, e.g. linear equations, i.e. $\phi(\mathbf{w}) = 1$ if $a_1\mathbf{w}[i_1, j_1] + a_2\mathbf{w}[i_2, j_2] = a_3\mathbf{w}[i_3, j_3]$ for some $i_1, i_2, i_3 \in [1..n]$ and $j_1, j_2, j_3 \in [1..m]$ and constants a_1, a_2, a_3, or quadratic equations, i.e. $\mathbf{w}[i_1, j_1] = \mathbf{w}[i_2, j_2] \cdot \mathbf{w}[i_3, j_3]$ (see e.g. [CM99]), a "less than" relation [Bou00], i.e. $\phi(\mathbf{w}) = 1$ iff $\mathbf{w}[i_1, j_1] \leq \mathbf{w}[i_2, j_2]$, or an inequality relation, i.e. $\phi(\mathbf{w}) = 1$ iff $\mathbf{w}[i_1, j_1] \neq \mathbf{w}[i_2, j_2]$. Secondly, by results of[CDS94], such HVZKPK's can be "compiled" into an efficient HVZKPK's for any constraint Φ formed by conjunctions and disjunctions of such constraints. Finally, all such HVZKPK proof systems can be compiled, with negligible overhead, into ZKPK proof systems, non-interactive in ROM model (using Fiat-Shamir heuristic), 3-round in CRS model [Dam00], or 5-round in the standard model [MP03].

The protocol proceeds given group $\mathbb{G}$ with generator g, on sender's private inputs an instance $(\mathbf{G}, \mathbf{C}) \in \mathbb{G}^{n\times m} \times \mathbb{G}^n$ and a message $M \in \mathbb{G}$, and on receiver's private input $\mathbf{w}$. First the sender S sends to the receiver R a random h in $\mathbb{G} \setminus \{1\}$. R aborts if $h = 1$. If $\Phi(\mathbf{w}) \neq 1$, then R picks $\mathbf{w}'$, s.t. $\Phi(\mathbf{w}') = 1$, and sets $\mathbf{w} \leftarrow \mathbf{w}'$. Then R sends to S Pedersen commitments to all $\mathbf{w}[i, j]$'s in $\mathbf{w}$: picks $\mathbf{r} \leftarrow_R \mathbb{G}^{n\times m}$, creates $\mathbf{D}$, s.t. $\mathbf{D}[i, j] = g^{\mathbf{w}[i,j]} h^{\mathbf{r}[i,j]}$, and proves using the WIPoK proof system for $\mathsf{PedREP}_{g,h}(\Phi)$ that the committed values $\mathbf{w}$ satisfy the Φ relation, i.e. that $(\mathbf{D}, (\mathbf{w}, \mathbf{r})) \in \mathcal{R}_{\mathsf{PedREP}}$. If R passes the proof, S uses the instance $(\mathbf{G}, \mathbf{C})$ and commitments $\mathbf{D}$ to encrypt M as follows: S picks random s_i's in $\mathbb{Z}_q$ for every $i \in [1..n]$ and random $t_{i,j}$'s in $\mathbb{Z}_q$ for every $(i, j) \in [1..n] \times [1..m]$, and sends to R the sets $\mathbf{E}, \mathbf{F}$ of ciphertexts $\mathbf{E}[i, j]$ and $\mathbf{F}[i, j]$,

$$\forall_{(i,j)\in[1..n]\times[1..m]} \quad \mathbf{E}[i, j] = (\mathbf{G}[i, j])^{s_i} g^{t_{i,j}} \quad \text{and} \quad \mathbf{F}[i, j] = h^{t_{i,j}}$$

together with value $\hat{K} = \prod_{i=1}^{n} K_i \cdot M$, where

$$\forall_{i\in[1..n]} \quad K_i = (\mathbf{C}[i])^{s_i} \cdot \prod_{j=1}^{m} (\mathbf{D}[i, j])^{t_{i,j}}$$

Finally R decrypts $(\mathbf{E}, \mathbf{F}, \hat{K})$ as $M' \leftarrow \hat{K} \cdot K'_1 \cdot ... \cdot K'_n$, where

$$\forall_{i\in[1..n]} \quad K'_i = \prod_{j=1}^{m} (\mathbf{E}[i, j])^{\mathbf{w}[i,j]} \cdot (\mathbf{F}[i, j])^{\mathbf{r}[i,j]}$$

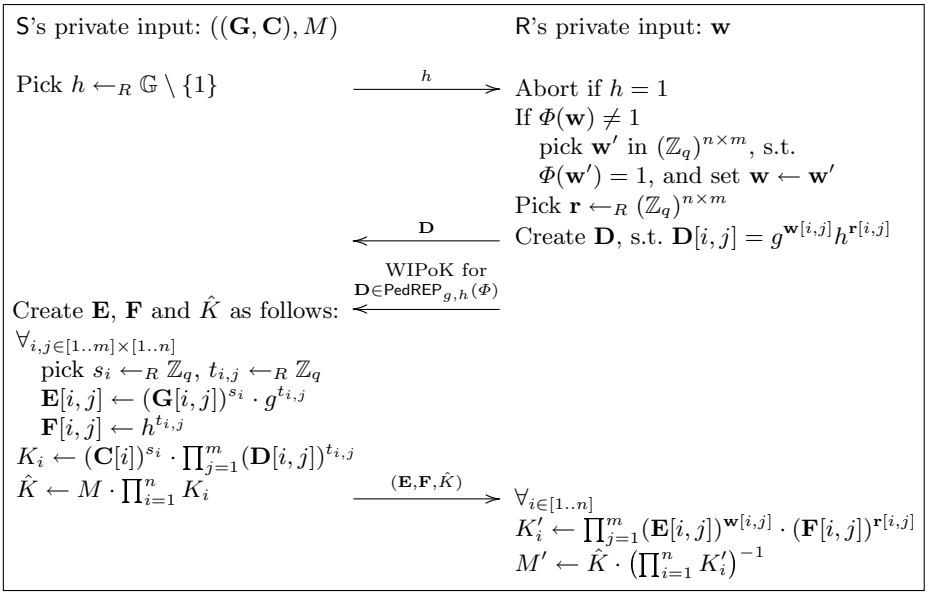

Fig. 2. COT Protocol for Relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$ (and Message Space $\mathbb{G}$)

Theorem 1. *If DDH problem is $(t_{ddh}, \epsilon_{ddh})$-hard, and the proof system in the construction is (t_{ext}, q_{ext}, d, e)-strongly-sound with soundness error δ, then the above construction of COT protocol for $\mathcal{R}_{\mathsf{REP}(\Phi)}$ is $(t_{adv}, t_{ext}, q_{ext}, d', e',)$-strongly-secure-and-sender-private, with soundness error δ', where $t_{adv} = t_{ddh} - O(nm)t_{exp}$, $d' = \frac{1}{2^{e+1}}d$, $e' = e + 1$, $\delta' = 2\delta + 4nm\epsilon$, and t_{exp} is the time for one exponentiation operation.*

Proof sketch: First, by splitting lemma, if adversary $\mathcal{A}$ has $\epsilon_{\mathcal{A}}$ advantage in distinguishing the real game from the random game, i.e., between $\mathsf{S}(x, M)$ and $\mathsf{S}(x', M')$ for random M' in $\mathbb{G}$, then for $\epsilon_{\mathcal{A}}/2$ portion of the h values sent by S in the first round, $\mathcal{A}$ has $\epsilon_{\mathcal{A}}/2$ advantage in distinguishing the two games, where in both games h is fixed to this chosen value. Suppose h sent by S is from this portion. By strong soundness of the proof system, a pair $(\mathbf{w}, \mathbf{r})$ can be extracted from $\mathcal{A}$, s.t. $\mathbf{D}[i,j] = g^{\mathbf{w}[i,j]} \cdot h^{\mathbf{r}[i,j]}$. So what remains to argue is that the extracted $\mathbf{w}$ is the witness for the real sender $\mathsf{S}(x, M)$'s statement $x = (\mathbf{G}, \mathbf{C})$. Note that (1) for each (i, j) pair, $(\mathbf{F}[i,j], \mathbf{E}[i,j])$ is an ElGamal encryption of element $(\mathbf{G}[i,j])^{s_i}$ under "key" h. Therefore by DDH assumption $(\mathbf{E}, \mathbf{F})$ is indistinguishable from $U_{\mathbb{G}^{n\times m}\times\mathbb{G}^{n\times m}}$, where U_G denotes uniform distribution over group G. Hence $(\mathbf{E}', \mathbf{F}', \hat{K}')$ sent by $\mathsf{S}(x', M')$ in the random game is indistinguishable from $U_{\mathbb{G}^{n\times m}\times\mathbb{G}^{n\times m}\times\mathbb{G}}$ because M' is random in $\mathbb{G}$; and (2) $K_i = \mathbf{C}[i]^{s_i} \cdot \prod_{j=1}^{m}(\mathbf{D}[i,j])^{t_{i,j}}$ computed by $\mathsf{S}((\mathbf{G}, \mathbf{C}), M)$ is indeed $\prod_{j=1}^{m}((\mathbf{E}[i,j])^{\mathbf{w}[i,j]} \cdot (\mathbf{F}[i,j])^{\mathbf{r}[i,j]}) \cdot (\mathbf{C}[i] \cdot (\prod_{j=1}^{n}(\mathbf{G}[i,j])^{-\mathbf{w}[i,j]}))^{s_i}$. If $\mathbf{w}$ is *not* the witness for $(\mathbf{G}, \mathbf{C})$, then there exists at least one i, s.t. $\mathbf{C}[i] \neq \prod_{j=1}^{n}(\mathbf{G}[i,j])^{\mathbf{w}[i,j]}$.

Then because s_i is random in $\mathbb{Z}_q$, K_i is random in $\mathbb{G}$ and so is $\hat{K}$. Hence $(\mathbf{E}, \mathbf{F}, \hat{K})$ sent by the real sender $\mathsf{S}((\mathbf{G}, \mathbf{C}), M)$ is also indistinguishable from $U_{\mathbb{G}^{n\times m}\times\mathbb{G}^{n\times m}\times\mathbb{G}}$. Therefore, the only way $\mathcal{A}$ can tell a difference between the real and random games is either by breaking the DDH assumption or by feeding the "correct" witness for the real sender $\mathsf{S}(x, M)$'s statement and then the extractor can extract it with large enough probability. □

The proof of the following theorem is simple, so we omit it for lack of space:

Theorem 2. *If the proof system for* PedREP *is* (t, ϵ)*-witness-indistinguishable, then the constructed COT protocol is* (t, ϵ)*-receiver-private.*

6 Construction of Unlinkable Secret Handshake Scheme

We construct an unlinkable SH scheme from so-called "Verifier-Local Revocable" Group Signatures (VLR GS), introduced and realized under Strong Diffie-Hellman and Decisional Linear assumptions by Boneh and Shacham [BS04]. Below we define unlinkable secret handshakes, specify the properties of a VLR-GS scheme that are useful to us, and show a construction of an SH scheme using such VLR-GS scheme and private COT protocol.

6.1 (Unlinkable) Secret Handshakes: Definition

An (Unlinkable) Secret Handshake Scheme (SH) is an authenticated key exchange protocol which operates in an environment with many groups, each managed by some group manager GM, and N users $P_1, ..., P_N$, each of which can be a member of several groups. Each GM plays a role of the Certificate Authority for its group, issuing certificates to any user it wants to admit to its group, and publishing revocation tokens for any user it wants to revoke from it. An SH scheme consists of three algorithms Setup, KGen, and Trace, and an interactive procedure Handshake, s.t.

- Setup on security parameter κ outputs parameters par (and key space $\mathcal{K}$).
- KGen, executed by a group manager GM on input par, outputs a group public key gpk and a vector of user keys $\mathbf{usk} = (\mathbf{usk}[1], \mathbf{usk}[2], \ldots, \mathbf{usk}[N])$ and revocation tokens $\mathbf{urt} = (\mathbf{urt}[1], \mathbf{urt}[2], \ldots, \mathbf{urt}[N])$. For notational simplicity we assume that user P_i is given key $\mathbf{usk}[i]$ for every group it belongs to.
- Handshake is an interactive protocol between two users, where each P_i runs on its private inputs $(\mathbf{usk}[i], \mathsf{gpk})$, and outputs a pair (k, tr) where $k \in \mathcal{K}$ is a key material to be used for subsequent secure communication with the protocol counterparty and tr is an escrow of that counterparty's identity.
- Trace, on inputs $(\mathsf{tr}, \mathbf{urt}[i])$ outputs 1 if tr is linked to $\mathbf{usk}[i]$, 0 otherwise.

Remark on Trace usage: Algorithm Trace has two uses: First, it can be used by the group manager to de-escrow the identity of a player involved in any protocol instance. Second, the intended usage of the above Handshake protocol, in which player P_i always outputs some (k, tr) pair, is to be followed by a verification that tr does not open to any revocation tokens included in the revocation list. If it does, P_i throws away the created key k.

Remark on privacy of revoked users: The revocation tokens are kept secret by the group manager, and published only to revoke a given player from a group. Therefore all past transcripts of a given user can be linked to this user, via the Trace algorithm, once the user is revoked. Such privacy limitation is a feature the verifier-local revocation group signatures [BS04]. A stronger privacy model, where past transcripts of revoked users remain private, can be supported by group signature schemes using accumulators, e.g. [CL01,BBS04]. It is an open question whether similar privacy can be efficiently achieved by an unlinkable SH scheme. (The major difficulty stems from the fact that two communicating players might assume different revocation epochs, and hence run the SH protocol on incompatible accumulators.)

Properties of SH Scheme. An SH scheme must meet the following properties:

Completeness: For every par output by Setup and every $(\mathsf{gpk}, \mathbf{usk}, \mathbf{urt})$ output by KGen on par, if any two players P_i and P_j honestly execute the $\mathsf{Handshake}$ protocol with inputs $(\mathbf{usk}[i], \mathsf{gpk})$ and $(\mathbf{usk}[j], \mathsf{gpk})$ respectively, then their respective outputs (k_i, tr_i) and (k_j, tr_j) satisfy $k_i = k_j$.[1] (It will follow from the security definition below that also $\mathsf{Trace}(\mathsf{tr}_j, \mathbf{urt}[i]) = 1$ and $\mathsf{Trace}(\mathsf{tr}_i, \mathbf{urt}[j]) = 1$.)

Security (Traceability): Security of an SH scheme is similar to traceability in a group signature scheme. Namely, it requires that if some player successfully authenticates itself to some player P_i then P_i's transcript of this protocol can be linked to that player's identity. Formally, security of an SH scheme is defined via the following game between an adversary $\mathcal{A}$ and a challenger $\mathcal{CH}^{\mathsf{sec}}$, on input any par output by Setup:

- **Init**. The challenger $\mathcal{CH}^{\mathsf{sec}}$, on input par and a bit b, runs $\mathsf{KGen}(\mathsf{par})$, which defines $(\mathsf{gpk}, \mathbf{usk}, \mathbf{urt})$, sends $(\mathsf{gpk}, \mathbf{urt})$ to $\mathcal{A}$, and sets $\mathsf{Cor} \leftarrow \emptyset$.
- **Queries**. $\mathcal{A}$ can make the following queries, where each query is serviced by the challenger *sequentially*, which disallows man in the middle attacks:
  - **Handshake**(i). $\mathcal{CH}^{\mathsf{sec}}$ on this query performs the $\mathsf{Handshake}$ protocol on inputs $(\mathbf{usk}[i], \mathsf{gpk})$, interacting with $\mathcal{A}$.
  - **Corruption**(i). $\mathcal{CH}^{\mathsf{sec}}$ sends $\mathbf{usk}[i]$ to $\mathcal{A}$ and adds i to Cor.
  - **Challenge**(i). (Allowed only once.) $\mathcal{CH}^{\mathsf{sec}}$ acts as in the **Handshake**(i) query, but denotes its outputs as (k, tr). If $\mathsf{Trace}(\mathsf{tr}, \mathbf{urt}[j]) = 1$ for any $j \in \mathsf{Cor}$, the game stops. Else, $\mathcal{CH}^{\mathsf{sec}}$ assigns $k_0 \leftarrow k$, picks $k_1 \leftarrow_R \mathcal{K}$, and gives k_b to $\mathcal{A}$.
- **Guess**. $\mathcal{A}$ outputs b' as its guess of b.

Let $p_b = \Pr[\mathcal{A}^{\mathcal{CH}^{\mathsf{sec}}(b,\mathsf{par})}(\mathsf{par}) = 1]$, where the probability goes over the randomness of both $\mathcal{A}$ and $\mathcal{CH}^{\mathsf{sec}}$, and let $\mathsf{Adv\text{-}Sec}(\mathcal{A}, \mathsf{par}) = |p_1 - p_0|$. We say that an SH scheme is (t, q_{sh}, N, ϵ)-*secure* on parameters par if in a universe with N users, for any t-time adversary $\mathcal{A}$ making at most q_{sh} **Handshake** queries, $\mathsf{Adv\text{-}Sec}(\mathcal{A}, \mathsf{par}) \leq \epsilon$.

[1] For notational simplicity, we present the completeness definition in the "symmetric" setting where two players authenticate each other if they are in the same group. However, our constructions generalize to the "asymmetric" setting, i.e. if $\mathbf{usk}[i]$ is issued under gpk_j and $\mathbf{usk}[j]$ is issued under gpk_i, then $k_i = k_j$.

Upgrade to Authenticated Key Agreement : If the Handshake protocol meets the above notion then it should be straightforward to convert it to an Authenticated Key Exchange (AKE) protocol secure against man-in-the-middle attacks. However, since modeling of AKE protocols requires introduction of an extended formalism, e.g. [BCK01], such compilation is out of scope of this paper.

Privacy: The privacy property covers both the anonymity property of group signatures, i.e. that no one except the group manager can detect if two instances of the SH protocol are executed by the same user, together with the affiliation-hiding property of secret handshake protocols (e.g. [BDS+03,CJT04]), i.e. that no one can detect which group a given player belongs to except of non-revoked members of the same group. Formally, we define privacy via the following game between an adversary $\mathcal{A}$ and the challenger $\mathcal{CH}^{\mathsf{pri}}$, on input any parameters par output by Setup:

- **Init**. The challenger $\mathcal{CH}^{\mathsf{pri}}$, on input par and a bit b, runs $\mathsf{KGen}(\mathsf{par})$ for every group G, with outputs denoted $(\mathsf{gpk}_G, \mathbf{usk}_G, \mathbf{urt}_G)$, gives gpk_G for all groups G to $\mathcal{A}$, and sets $\mathsf{Cor} \leftarrow \emptyset$ and $\mathsf{Chosen} \leftarrow \emptyset$.
- **Queries**. $\mathcal{A}$ can make the following types of queries. As in the security game, the challenger services each query sequentially:
  - **Handshake**(i, G). $\mathcal{CH}^{\mathsf{pri}}$ runs $\mathsf{Handshake}$ on $(\mathbf{usk}_G[i], \mathsf{gpk}_G)$, interacting with $\mathcal{A}$.
  - **Corruption**(i). If $i \notin \mathsf{Chosen}$ then $\mathcal{A}$ gets $\mathbf{usk}_G[i], \mathbf{urt}_G[i]$ for every G. Let $\mathsf{Cor} \leftarrow \mathsf{Cor} \cup \{i\}$.
  - **Challenge**(i_0, G_0, i_1, G_1). (Allowed only once.) Set $\mathsf{Chosen} \leftarrow \{i_0, i_1\}$. If $\mathsf{Chosen} \cap \mathsf{Cor} \neq \emptyset$ then the game stops. Otherwise $\mathcal{CH}^{\mathsf{pri}}$ runs $\mathsf{Handshake}$ on input $(\mathbf{usk}_{G_b}[i_b], \mathsf{gpk}_{G_b})$, interacting with $\mathcal{A}$.
- **Guess**. $\mathcal{A}$ outputs b' as its guess of b.

Let $p_b = \Pr[\mathcal{A}^{\mathcal{CH}^{\mathsf{pri}}(b,\mathsf{par})}(\mathsf{par}) = 1]$, where the probability goes over the randomness of both $\mathcal{A}$ and $\mathcal{CH}^{\mathsf{pri}}$, and let $\mathsf{Adv\text{-}Pri}(\mathcal{A}, \mathsf{par}) = |p_1 - p_0|$. We say that a SH scheme is (t, q_{sh}, N, ϵ)-*private* on parameters par if in a universe with N users, for any t-time adversary $\mathcal{A}$ making at most q_{sh} **Handshake** queries, $\mathsf{Adv\text{-}Sec}(\mathcal{A}, \mathsf{par}) \leq \epsilon$.

6.2 Verifier-Local Revocable Group Signature (VLR-GS)

A VLR-GS scheme consists of the following algorithms: A setup procedure $\mathsf{Setup}_{\mathsf{GS}}$ which creates public parameters par, an unforgeable *certificate scheme* $\Pi_{cert} = (\mathsf{KeyGen}, \mathsf{Cert}_{\mathsf{par}}, \mathsf{Ver}_{\mathsf{par}})$, a non-interactive zero-knowledge proof for relation $\mathcal{R}_{\mathsf{AUTH}}$ which we define below, and two additional procedures $\mathsf{Com}_{\mathsf{par}}$ and $\mathsf{TraceCom}_{\mathsf{par}}$. The functionality of the certificate scheme Π_{cert} is that if $(\mathsf{sk}, \mathsf{pk})$ is an output by $\mathsf{KeyGen}(\mathsf{par})$ then each run of $\mathsf{Cert}_{\mathsf{par}}(\mathsf{sk})$ generates a new token/secret pair $(\mathsf{tk}, \mathsf{scr})$ s.t. $\mathsf{Ver}_{\mathsf{par}}(\mathsf{pk}, \mathsf{tk}, \mathsf{scr}) = 1$. To enable an efficient VLR-GS scheme, three conditions must be met:

I. The outputs of Com must be *traceable* in the sense that (1) $\mathsf{TraceCom}_{\mathsf{par}}(C, \mathsf{tk})$ $= \mathsf{TraceCom}_{\mathsf{par}}(C, \mathsf{tk}') = 1$ implies $\mathsf{tk} = \mathsf{tk}'$, and (2) $\mathsf{TraceCom}_{\mathsf{par}}$ $(C, \mathsf{tk}) = 1$ if and only if $\exists\, r$ s.t. $C = \mathsf{Com}_{\mathsf{par}}(\mathsf{tk}; r)$, and that .

II. There must exist an efficient non-interactive ZKPK proof system for language

$$\mathsf{AUTH}(\mathsf{par}) = \left\{ \begin{array}{c} (C, \mathsf{pk}) \;\; \text{s.t. } \exists\, (\mathsf{tk}, \mathsf{scr}, r) \text{ s.t. } C = \mathsf{Com}_{\mathsf{par}}(\mathsf{tk}; r) \\ \text{and } \mathsf{Ver}_{\mathsf{par}}(\mathsf{pk}, \mathsf{tk}, \mathsf{scr}) = \mathsf{accept} \end{array} \right\}$$

III. The certificate scheme Π_{cert} must be existentially unforgeable:

Definition 5. *We say that the certificate scheme Π_{cert} is $(t, \hat{q}, \epsilon)$-*unforgeable *on parameters* par *if for any t-time adversary $\mathcal{A}$, the probability of the following event is at most ϵ: First* $(\mathsf{sk}, \mathsf{pk})$ *is generated by* $\mathsf{KeyGen}(\mathsf{par})$*, then* $\mathsf{Cert}_{\mathsf{par}}(\mathsf{sk})$ *is executed $\hat{q}$ times to generate $\hat{q}$ token/secret pairs* $(\mathsf{tk}_i, \mathsf{scr}_i)$*, and then $\mathcal{A}$ on input* par*,* pk*, and* $\{\mathsf{tk}_i, \mathsf{scr}_i\}_{i=1,..,\hat{q}}$*, outputs* $(\mathsf{tk}^*, \mathsf{scr}^*)$ *s.t.* $\mathsf{Ver}_{\mathsf{par}}(\mathsf{pk}, \mathsf{tk}^*, \mathsf{scr}^*) = 1$ *and* $\mathsf{tk}^* \neq \mathsf{tk}_i$ *for all i. The probability in this experiment runs over the randomness of $\mathcal{A}$ and procedures* KeyGen *and* Cert*.*

Under the above conditions a VLR-GS scheme works as follows. The group public key is pk output by KeyGen, and each group member's signature key is $(\mathsf{tk}, \mathsf{scr})$ output by $\mathsf{Cert}_{\mathsf{par}}$ on the corresponding sk. A signature under group key pk consists of $C = \mathsf{Com}_{\mathsf{par}}(\mathsf{tk})$ and a non-interactive ZKPK for $(C, \mathsf{pk}) \in \mathsf{AUTH}(\mathsf{par})$. Any user can be revoked by the group manager adding the token part tk in his/her key to the CRL. A verifier then checks if $\mathsf{TraceCom}_{\mathsf{par}}(C, \mathsf{tk}) = 1$ for each tk in the CRL. However, to enable our SH construction a VLR-GS scheme must meet two more properties:

IV. Token tk in pair $(\mathsf{tk}, \mathsf{scr})$ output by $\mathsf{Cert}_{\mathsf{par}}(\mathsf{sk})$ must be uniformly distributed in some set $\mathbb{U}_t$ defined by par and independent of key sk.

V. Values C output by $\mathsf{Com}_{\mathsf{par}}(\mathsf{tk})$ hide the tk value, not in the sense of semantic security, because knowledge of tk enables linking C to tk via $\mathsf{TraceCom}$, but in the following sense:

Definition 6. *We say that the algorithm* Com *is (t, q_{com}, ϵ)-*private *on parameters* par*, if for any t-time adversary $\mathcal{A}$ with at most q_{com} oracle accesses to procedures* $\mathsf{Com}(\mathsf{tk}_0)$ *and* $\mathsf{Com}(\mathsf{tk}_1)$*, we have $|p_0 - p_1| \leq \epsilon$ where*

$$p_b \stackrel{\triangle}{=} \Pr[\mathcal{A}^{\mathsf{Com}_{\mathsf{par}}(\mathsf{tk}_0), \mathsf{Com}_{\mathsf{par}}(\mathsf{tk}_1)}(\mathsf{par}, C_b) = 1 \mid \mathsf{tk}_0, \mathsf{tk}_1 \leftarrow_R \mathbb{U},\; C_b \leftarrow \mathsf{Com}_{\mathsf{par}}(\mathsf{tk}_b)]$$

where the probability additionally goes over the randomness of $\mathcal{A}$ and Com*.*

6.3 Construction of SH's from VLR-GS and Private COT

Assume we have a VLR-GS scheme consisting of procedures $\mathsf{Setup}_{\mathsf{GS}}$, KeyGen, Cert, Ver, Com, and $\mathsf{TraceCom}$ which satisfy all the above requirements. Assume also a private COT protocol for relation $\mathcal{R}_{\mathsf{AUTH}(\mathsf{par})}$ and message space $\mathcal{M}$ corresponding to this VLR-GS scheme. An SH scheme (Setup, KGen, $\mathsf{Handshake}$, Trace) is constructed as follows:

– Setup is the same as $\mathsf{Setup}_{\mathsf{GS}}$, and keyspace $\mathcal{K}$ is the message space $\mathcal{M}$.
– KGen, on input par, first computes $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par})$, then computes $(\mathsf{tk}_i, \mathsf{scr}_i) \leftarrow \mathsf{Cert}_{\mathsf{par}}(\mathsf{sk})$ for $i = 1, .., N$, and outputs $(\mathsf{gpk}, \mathbf{usk}, \mathbf{urt})$ where $\mathsf{gpk} = \mathsf{pk}$, and for all i we assign $\mathbf{usk}[i] \leftarrow (\mathsf{tk}_i, \mathsf{scr}_i)$ and $\mathbf{urt}[i] \leftarrow \mathsf{tk}_i$.
– Protocol $\mathsf{Handshake}$, executed between players P_i and P_j: Player P_i runs the protocol on inputs $(\mathsf{usk}_i, \mathsf{gpk}_i)$ for some group in which P_i is a member. Similarly P_j runs it on $(\mathsf{usk}_j, \mathsf{gpk}_j)$ for some group in which P_j is a member. The protocol proceeds as in Figure 3.

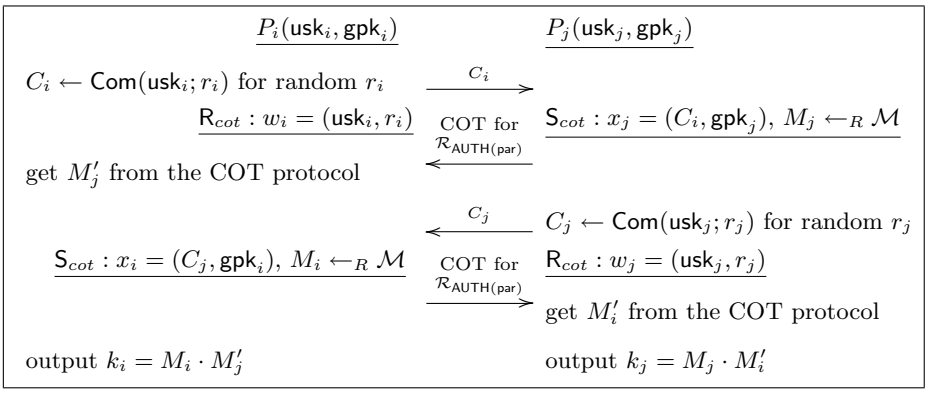

Fig. 3. $\mathsf{Handshake}$ between P_i and P_j with inputs $(\mathsf{usk}_i, \mathsf{gpk}_i)$ and $(\mathsf{usk}_j, \mathsf{gpk}_j)$

– Trace, on inputs tr and tk, outputs $\mathsf{TraceCom}_{\mathsf{par}}(\mathsf{tr}, \mathsf{tk})$.

For lack of space we omit the proofs of the following theorems:

Theorem 3 (SH Security). *For any* par *outputted by* $\mathsf{Setup}_{\mathbf{GS}}$*, if* $\mathsf{Com}_{\mathsf{par}}$ *is traceable,* Π_{cert} *is* (t_1, q_1, ϵ_1)*-unforgeable on* par*, and if the COT protocol for* $\mathcal{R}_{\mathsf{auth}(\mathsf{par})}$ *is* $(t_2, t_{ext}, q_{ext}, d, e)$*-strongly-secure and sender-private with soundness error* δ*, and* (t_3, ϵ_3)*-receiver-private, then the above SH scheme is* $(t', q_{sh}, q_1, \epsilon')$*-secure, where* $t' = \min\{t_2, (t_1 - t_{ext})/(q_{ext} + 1), t_3\}$*,* $\epsilon' = (q_1 + 1)(\epsilon_1/d)^{1/e} + \delta) + q_1 \cdot q_{sh} \cdot \epsilon_3$*.*

Theorem 4 (SH Privacy). *For any* par *outputted by* $\mathsf{Setup}(\kappa)$*, if the* $\mathsf{Com}_{\mathsf{par}}$ *algorithm is* (t_1, q_1, ϵ_1)*-private and is uniquely traceable, if* P_{cert} *is* (t_2, q_2, ϵ_2)*-unforgeable and if the COT protocol for* $\mathcal{R}_{\mathsf{auth}(\mathsf{par})}$ *is* $(t_3, t_{ext}, q_{ext}, d, e)$*-strongly-secure-and-sender-private with soundness error* δ*,and* (t_4, ϵ_4)*-receiver-private, then the above SH scheme is* $(t', q_{sh}, q_2, \epsilon')$*-private, where* $t' = \min\{t_1, (t_2 - t_{ext})/(q_{ext} + 1), t_3, t_4\}$*,* $\epsilon' = \epsilon_1 + ((q_2 + 1)(\epsilon_2/d)^{1/e} + \delta) + (q_2 \cdot q_{sh} \cdot \epsilon_4) + \epsilon_4$*, and* $q_{sh} \leq q_1$*.*

Note that the SH scheme presented above is a generic construction from appropriate VLR-GS components and an associated private COT protocol. For lack

of space we omit from these proceedings a description of a concrete implementation where all components are instantiated with those used in the VLR-GS scheme of [BS04]. However, it is easy to see that relation $\mathcal{R}_{\mathsf{AUTH}}$ defined by these components can be transformed to a special case of relation $\mathcal{R}_{\mathsf{REP}(\Phi)}$ of Section 5, and therefore an efficient private COT protocol for this relation is implied by the private COT for $\mathcal{R}_{\mathsf{REP}(\Phi)}$ given in Figure 2.

Acknowledgments. The authors want to thank Anna Lysyanskaya and Yuval Ishai for stimulating discussions, and Gene Tsudik for frequent productive discussions and for a crucial suggestion which triggered this work, namely to look into converting some group signature into a "group-signature-based" envelope.

References

[AIR01] William Aiello, Yuval Ishai, and Omer Reingold. Priced oblivious transfer: How to sell digital goods. In *EUROCRYPT*, pages 119–135, 2001.

[BBS04] D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *Proceedings of Crypto '04*, 2004.

[BBW06] Adam Barth, Dan Boneh, and Brent Waters. Privacy in encrypted content distribution using private broadcast encryption. In Giovanni Di Crescenzo and Aviel D. Rubin, editors, *Financial Cryptography*, volume 4107 of *Lecture Notes in Computer Science*, pages 52–64. Springer, 2006.

[BCK01] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. A modular approach to the design and analysis of authentication and key-exchange protocols. In *Symposium on Theory of Computing*, 2001.

[BDS+03] Dirk Balfanz, Glenn Durfee, Narendar Shankar, Diana K. Smetters, Jessica Staddon, and Hao-Chi Wong. Secret handshakes from pairing-based key agreements. In *IEEE Symposium on Security and Privacy*, pages 180–196, 2003.

[BGW05] Dan Boneh, Craig Gentry, and Brent Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *CRYPTO*, pages 258–275, 2005.

[BK04] Ian F. Blake and Vladimir Kolesnikov. Strong conditional oblivious transfer and computing on intervals. In *ASIACRYPT*, pages 515–529, 2004.

[Bou00] Fabrice Boudot. Efficient proofs that a committed number lies in an interval. In *EUROCRYPT*, pages 431–444, 2000.

[BS04] Dan Boneh and Hovav Shacham. Group signatures with verifier-local revocation. In *ACM Conference on Computer and Communications Security*, pages 168–177, 2004.

[BSW07] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *IEEE Symposium on Security and Privacy*, pages 321–334, 2007.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *CRYPTO*, pages 174–187, 1994.

[CJT04] Claude Castelluccia, Stanislaw Jarecki, and Gene Tsudik. Secret handshakes from ca-oblivious encryption. In *ASIACRYPT*, pages 293–307, 2004.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *EUROCRYPT*, pages 93–118, 2001.

[CM99] Jan Camenisch and Markus Michels. Proving in zero-knowledge that a number is the product of two safe primes. In *EUROCRYPT*, pages 107–122, 1999.

[COR99] Giovanni Di Crescenzo, Rafail Ostrovsky, and Sivaramakrishnan Rajagopalan. Conditional oblivious transfer and timed-release encryption. In *EUROCRYPT*, pages 74–89, 1999.

[Cre00] Giovanni Di Crescenzo. Private selective payment protocols. In *Financial Cryptography*, pages 72–89, 2000.

[CS01] Ronald Cramer and Victor Shoup. Universal hash proofs and and a paradigm for adaptive chosen ciphertext secure public-key encryption. *Electronic Colloquium on Computational Complexity (ECCC)*, 8(072), 2001.

[CvH91] David Chaum and Eugène van Heyst. Group signatures. In *EUROCRYPT*, pages 257–265, 1991.

[Dam00] Ivan Damgard. Efficient concurrent zero-knowledge in the auxiliary string model. In *EuroCrypt*, 2000.

[GPSW06] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM Conference on Computer and Communications Security*, pages 89–98, 2006.

[JKT07] Stanislaw Jarecki, Jihye Kim, and Gene Tsudik. Group secret handshakes or affiliation-hiding authenticated group key agreement. In *CT-RSA*, pages 287–308, 2007.

[JL07] Stanislaw Jarecki and Xiaomin Liu. Unlinkable secret handshakes and key-private group key management schemes. In *proceedings of ACNS*, 2007.

[JL08] Stanislaw Jarecki and Xiaomin Liu. Affiliation-hiding envelope and authentication schemes with efficient support for multiple credentials. In *ICALP (2)*, pages 715–726, 2008.

[KP97] Joe Kilian and Erez Petrank. Identity escrow. In *Proceedings of Crypto 97*, pages 169–185. Springer-Verlag, 1997.

[LL07] Sven Laur and Helger Lipmaa. A new protocol for conditional disclosure of secrets and its applications. In *ACNS*, pages 207–225, 2007.

[MP03] Daniele Micciancio and Erez Petrank. Simulatable commitments and efficient concurrent zero-knowledge. In *EUROCRYPT*, pages 140–159, 2003.

[NNL02] Dalit Naor, Moni Naor, and Jeffery Lotspiech. Revocation and tracing schemes for stateless receivers. *Electronic Colloquium on Computational Complexity (ECCC)*, 043, 2002.

[Ped91] Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *CRYPTO*, pages 129–140, 1991.

[Yao86] Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167, 1986.