Privacy-Enhancing Signatures

Sherman S. M. Chow

Chinese University of Hong Kong

Why this talk?

- We just heard about ring signatures "*for blockchain*".
- 1-out-of-*n* proof ('94); Ring signature ('01)
- Linkable ring sig. ('04); Traceable ring sig. ('07)
- But we have bitcoin in 2008
- Who knows what will happen next?

- Let's (re-)visit various different "flavors" of signatures!

Roadmap

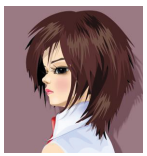
- This talk covers various privacy concerns of signatures in these 30 years!

- Verifiability Privacy
  - Undeniable Sig. ('89) → Non-Interactive Confirmer Sig. ('11)
- (Accountable) Signer Privacy
  - Group Sig. ('91) → Group Sig. w/ Event-Dependent Opening ('19)
- Message Privacy
  - Sanitizable Sig. ('05) → Unlinkable Sanitizable Sig. ('16)

- The talk will also briefly discuss 2 core pairing-based techniques, and if time permits, 2 pairing-based schemes.

I. Signatures with Verifiability Privacy

- Alice is making a signed offer to Bob
- Bob can not use Alice's offer as leverage to negotiate better terms with, say, Carol
  - We want the (verifiability of the) signature to be "private".
- Undeniable sig.: can only be verified with Alice's help
  - Cannot deny if Alice did sign (only confirm or disavow)



Confirmer Signature

- What if the signer disappear?
- Signer can appoint a *confirmer* in advance
  - Confirmer can confirm(/deny) a signature
  - Confirmer can also extract an ordinary signature out of it
- Undeniable/Confirmer signatures allow one to choose whether to engage in the confirm/disavow protocol
- Require the confirmer to be online and answer requests
- But what if an attacker sit in the middle between Alice and Bob and see everything? It will still be convinced

Designated-Verifier Proof/Signature

- "It is either Alice or Bob's signature"
  - Just like a 2-user ring signature
  - Bob knows that he didn't sign but Carol does not know about that
- But what if Alice later repudiate?
  - "It is by Bob, not by me!" i.e., no non-repudiation
- Undeniable ('89) → Confirmer ('94) → Designated-ver. ('96)
- What else have been done in these two decades?

Online-Untransferable Signature

- Bob can "transfer" the validity of the signature to Carol by interacting with Alice and Carol concurrently
- All constructions of confirmer signatures provide only *offline* untransferability [Liskov-Micali @ PKC '08]
- Their construction uses "cut-and-choose" technique
  - Prepare many "copies", reveal some of them (no privacy) and verify, hope the remaining unrevealed are well-formed.
  - But that is the source of inefficiency: For security parameter k, the signature of this scheme includes $O(k)$ ciphertext

Just don't do it online!

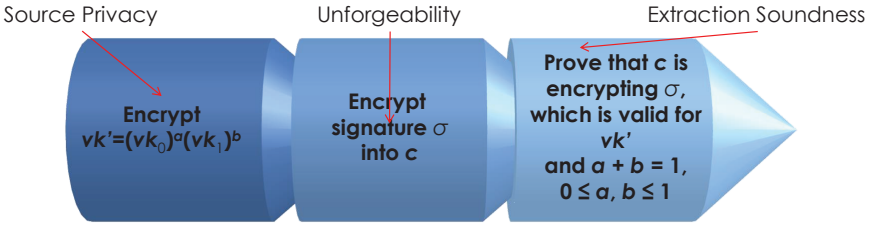
- New notion proposed by [**C**-Haralambiev @ CTRSA '11]
- *Non-interactive* confirmer signatures (NICS)
- "Confirmer" just converts an ordinary sig. to an NICS
  - Like DVS, NICS can only convince the designated verifier
  - "Confirmer" is like that in "universal" DVS, anyone can do the conversion
- No online interaction, "online-"untransferability comes naturally
- But, again, what if the true signer repudiate?
- It adds "extractability" on top of (U)DVS
  - [Steinfeld-Bull-Wang-Piperzyk @ AsiaCrypt '03]
- This proposed construction is efficient (O(1))

Traditional vs. Universal confirmer

- Traditional confirmer signature
  - Signature is ambiguous (not binding to the signer)
  - Confirmer convinces verifier about its validity
  - Confirmation requires "secret" (not universal)
    - Secret key, or randomness used in signing
- Non-interactive confirmer signature
  - Signature is an ordinary one at the first place
  - Need a step to make it ambiguous
  - Yet still convincing to the verifier
  - An ordinary one can be extracted by a (passive) "adjudicator"

Construction Idea

- "σ is a valid signature signed by either Alice or Bob"
- Confirmer does not create this by directly "signing"
- But by *converting* an ordinary signature then proving:



GS Proof [Groth-Sahai @ Eurocrypt '08]

- Fiat-Shamir heuristics for NIZK relies on the random oracle
- Can we do NIZK proof without random oracle?
- Yes in general, but inefficient
- Before proving, you need to model the computation
  - e.g., hash function mapping to a group element, as a circuit
- Groth-Sahai proof makes an NIZK proof system
  - for pairing product equations
  - e.g., $e(\underline{A}, X)e(\underline{B}, Y) = T$
  - where (A, B) is the witness to be proved about; X, Y, T are public

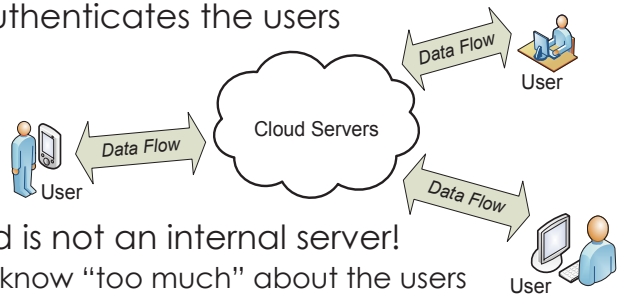
Some blockchain questions to ponder

- Everything including the signatures are put on the blockchain, perhaps "delayed" verifiability is useful?
- Integrate NICS with smart-contracts, only pay (i.e., reveal the signature) when the contract is fulfilled?

- Just some random thoughts at this stage…
- Shouldn't we build crypto but not just thinking about cryptocurrency?

II. Signer/Authenticator Privacy

- An organization employs the cloud service
- Many members belong to this organization
- The cloud authenticates the users



- But the cloud is not an internal server!
  - It shouldn't know "too much" about the users

Another application: Wikipedia

- Everyone can write on different topics.
- Writers/Reviewers may want privacy (or anonymity)
  - e.g., multiple posts are *unlinkable*
- But the wikipedia administrator needs to ban "*misbehaving*" users
  - posting advertisement, using abusive language, etc.
- In general, "Web 2.0" applications
  - relies on users participation
  - but also needs moderation

Dilemma

- Online Privacy
  - user expect actions online are unlinkable to real-world identity
  - user will not be identified (and "punished")
- Accountability
  - yet, perfect anonymity might be abused
  - what if we identify some misbehavior?
  - "someone" should be *the* judge
  - and has the power to find what other "damage" has been done
- Anonymity + *Revocability*

Balancing Privacy and Identification

- Traditional PKI-based certificate
  - No anonymity at all
  - Also, certificate may reveal privacy-related information
    - A certificate contains many different fields for other purposes
- Just gives all users the same private key
  - Unconditional anonymity may be abused
  - If any one (or at least any users) know the same private key, can it still be treated as a form of "secret"?

Signatures with Identity Privacy

- Signature is meant to be associated with a signer.
- How can one hide the identity of the signer?!
- An answer: hide it within a "group"
- The verifier only knows that 1 of the members in the group has issued the signature, but not exactly whom.
- How the group is formed? Ring sig. vs. Group sig.
- Another answer: Anonymous signature schemes
  - Without the message, one needs to try all possible public key to figure out who is the signer. [Yang-Wong-Deng-Wang '06]

Ring Signatures [Rivest-Shamir-Tauman '01]

- How to Leak a Secret, in AsiaCrypt '01
- 1-out-of-*n* Non-Interactive Zero-Knowledge (NIZK) proof
  - e.g., for DLP, I know a secret key x s.t. $y = g^x$ is in $\{y_1, y_2, \ldots y_n\}$
- Spontaneity: A signer can conscript any group of *n* users
  - This group may even not be aware that they have "joined"
- Anonymity: Verifier cannot determine who is the real signer
  - Usually unconditional! (vs. computational anonymity)

- Sometimes "linkability" can be useful
  - e.g., double-spending detection in electronic cash, or cryptocurrency (Monero) // Fujisaki's talk yesterday

Linkable Ring Sig. [Liu-Wei-Wong '04]

- Signatures from the same signer can be linkable [ACISP'04]
- Suppose there is a group element *h* denoting the "event"
  - e.g., *h* = H("event info/linkability context", "ring" of *n* public keys)
- Put "linkability tag" h^{x_i} w/ ring sig via an "AND of OR proof"
  - Anonymous under DDH assumption (becomes comp. anonymity)
  - i.e., a proof that $t = h^x$ AND $(x = x_1$ OR $x = x_2$ OR … OR $x = x_n)$

- "Escrowed" linkability [**C**-Susilo-Yuen @ VietCrypt '06]
  - Verifiable encryption of h^{x_i}
  - e.g., recipient-free e-voting [**C**-Liu-Wong @ NDSS '08]

"Verifiable" Authorship

- Verifiable Ring Signatures --- authorship can be claimed/denied.
- Any *user* can prove that s/he is the signer
  - [Lv-Wang @ DMS '03]
- Any *user* can prove that s/he *did not* sign
  - [Bultel-Lafourcade @ CANS '17]

- Accountable Ring Signatures --- signer identity can be revealed by a "trusted" *opening authority*
  - [Xu-Yung @ CARDIS '04]
  - [Bootle *et al.* @ ESORICS '15]

- You'll see them again in Group Signatures and Sanitizable Signatures.

Group Signatures

- Group-oriented signatures with anonymity
  - But with an explicit group formation (diff. from ring signature)
- A *group manager* (GM) issues credentials
- Any *member* can sign for the group
  - remain anonymous within the group
  - signatures are unlinkable
  - but, unconditional anonymity may be abused
- An *opening authority* can "open" a group signature to reveal its true signer

Applications

- Direct anonymous attestation
  - [Brickell-Camenisch-Chen @ CCS '04]
  - Authenticate an application's executable code to a server
  - Trusted Computing Group (TCG)
  - Next Generation Secure Computing Base (NGSCB a.k.a. Palladium)
- Privacy-Preserving Identity-Management
  - [**C**-He-Hui-Yiu @ ACNS '12]

Vehicular Safety Communication

Basic Algorithms of Group Signatures

- Setup
  - key pairs for the group and the opening authority
  - param = (*gpk*, *opk*), secret key = (*gsk*, *osk*)
- Join
  - interactive protocol between GM and user
  - user get the member key pair (pk_i, sk_i)
  - the GM updates the membership archive *DB* with $info_i$
- Sign(sk_i, m) → σ, Verify(σ, m) → "True"/"False"
- Open(σ, *osk*, *DB*) → ID_i // "revocable" anonymity

Design of Group Signatures

- Credential issuing
  - Using *gsk* to issue a "signature" *s* on (*ID*, pk_i)
- Proving the knowledge of credential
  - Proving about (*s*, sk_i)
  - User should have his/her own secret key for non-framability
    - a.k.a. exculpability --- not guilty of wrongdoing
- Identity is encrypted s.t. the public cannot see
  - But decryptable by the opening authority

Signature as a Credential

- GM is the Signer
- Message: Attribute of a *User*, e.g., *ID, user public-key*
- The signature certifies "Membership of a Group"

- 2-level (hierarchical) signature
  - Use the user (private) key to certify the actual message
  - Delegating the signing power

Signatures with "Efficient Protocols"

- To issue a credential, the GM signs on two things
- *Signature on a vector of messages*
  - Allow more efficient zero-knowledge proof if the components of a message vector are treated "separately"

- User secret key should be hidden in a *commitment*
- *Signature on the commitment*
  - Allow signing on the message committed in the commit
  - Allows proving the knowledge of such a message-signature pair
- Both notions can be combined:
  - i.e., signing on a vector of messages, some of them can be presented in the form of a commitment

BBS+ Signature [Au-Susilo-Mu-**C**'13]

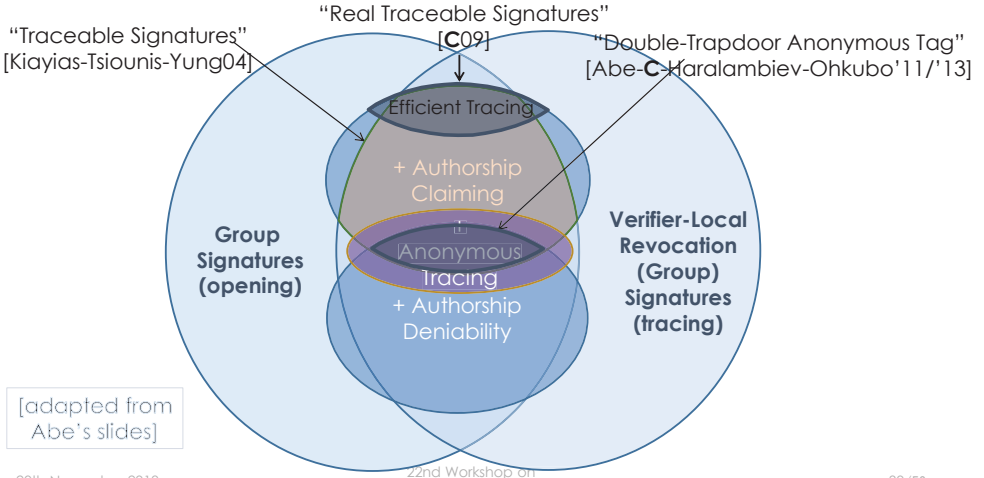
- Based on Boneh-Boyen-Shacham @ Crypto '04]
- System Parameter : (g, g_0, g_1, …, g_n) for n-block message
- Signature Requester picks a random r'
- Compute $C = g_0^{r'} \Pi g_i^{m_i}$
  - Commitment of n-block messages
- Compute PoK{(r', m_1, …$_i$, m_n): $C = g_0^{r'} \Pi g_i^{m_i}$}

- Signer picks r'' and e, define $r^* = r' + r''$
- Return $A = (g \cdot g_0^{r''} \cdot C)^{1/(\beta + e)}$
- Signature = $(A = (g \cdot g_0^{r^*} \Pi g_i^{m_i})^{1/(\beta + e)}, e, r^*)$

Identity-Escrow

- *Proof of Knowledge (PoK) of a signature*
  - without showing the GM's signature
  - the group member proves that the member private key signed by the GM is used to sign the final message
- *Proof for the correctness* of encrypting identification
  - user's public key / credential

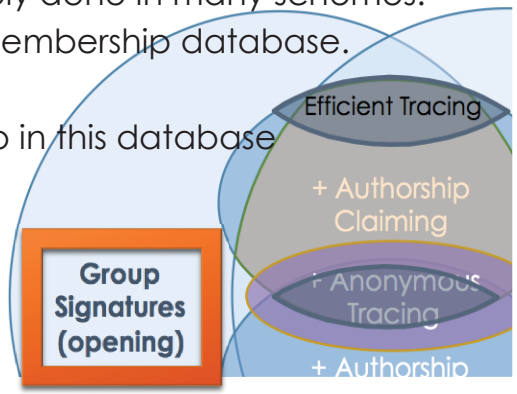
World of Group Signatures



[adapted from Abe's slides]

Subtle Issue of Opening

- Separation was not nicely done in many schemes.
- Key issuing updates a membership database.
- Opening just reveals info in this database
  - e.g., a user public key
- But not the user identity
- Need to link them back



"Catch-22" Issue of Opening

- Identified by Kiayias-Zhou at FC '07
- So, how exactly opening can be done?
- 0) The membership database is public: Not an option.
- 1) GM gives the membership database to OA.
  - OA is too powerful. Member cannot "sign in peace".
- 2) GM keeps such a membership database to itself.
  - OA talks to GM every time, GM should remains online
  - GM may even refuse to help. No separation of power.
  - This DB attracts attacker: All members are potential signers.

Hidden Identity-Based Signatures (HIBS)

- Group signature: "No-win" no matters what you do.
- The crux of the problem: member list should not exist!
- Identity-based signatures (IBS) [Shamir @ Crypto '84]
  - Private key generator (PKG) create a master key pair: (*mpk, msk*)
  - PKG generates user secret key (sk_{ID}) for an user with given its *ID*
  - Anyone can verify a signature given (*mpk*, *ID*) and the message
- Hidden identity-based signatures [Kiayias-Zhou @ FC '07]
  - Anyone can verify a signature given *mpk* and the message
  - An OA can open the signature and reveals the signer's *ID*

HIBS as a Refinement of Group Sig.

- User identity is only hidden in the signature
  - There is no membership list whatsoever
- Join
  - interactive protocol between GM and user
  - user get the member key pair (pk_i, sk_i)
  - ~~the GM updates the membership archive *DB* with *info*$_t$~~
- Opening just takes in OA's secret key and output signer ID
  - Open(*osk*, σ) → *ID*
- Supporting above features should not penalize the performance of other algorithms
  - Time and space costs for opening are independent of #members
  - "Real HIBS" based on GS-proof [**C**-Zhang-Zhang @ FC ' 17]

Two Existing HIBS Schemes

- 1. A pairing-based scheme [KZ'07] with Open() returns g^{ID}
- Requiring solving discrete logarithm (or a small ID space)
- or maintaining a mapping between *ID* and g^{ID}
- Other scheme which opens to g^{ID} also exists
  - e.g. [Boyen-Waters @ PKC '07]

- 2. A scheme based on Paillier encryption [IET-Info Sec '09]
- Rely on the Decisional Composite Residuosity assumption
- Working with an RSA modulus is not that efficient
  - Larger group elements, more involved zero-knowledge proof

Is membership list an old ('07) issue?

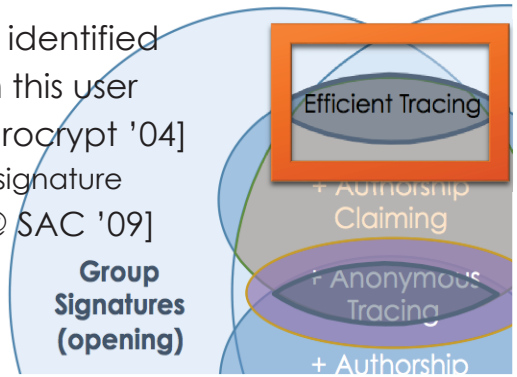
- Membership DB "affects" recent study of group signatures.

- Get Shorty via Group Signatures without Encryption [Bichsel *et al.* @ SCN '10]
- Opening/"Decryption" by referring to DB → linear in |DB|

- (Dynamic Group Signature from) Short Accountable Ring Signatures based on DDH [Bootle *et al.* @ ESORICS '15]
- The group public key is simply a list of all user public keys!

Traceable (Group) Signatures

- Opening is too powerful
- When an abusive user is identified
- Trace all signatures from this user
- Traceable Sig. [KTY @ Eurocrypt '04]
  - *Check* each candidate signature
- Real Traceable Sig. [**C** @ SAC '09]
  - "Pointing to" signatures

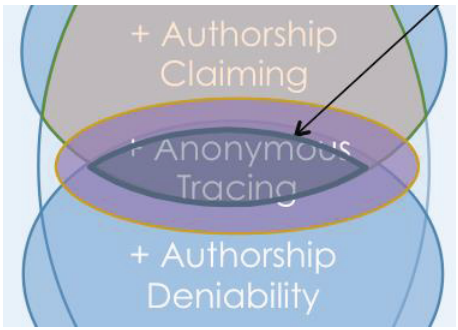


Real Traceable Signature

- Assign a seed to every member
- *Signature on a block of messages*
  - User identity, user public/private key, the seed
- Tag is $PRF_{seed}(ctr)$
  - PRF is a *pseudorandom function*
  - *ctr* is a counter maintained by the user
  - deterministic given *seed* and *ctr*
- *Range proof* ensures $ctr < N$ (a system parameter)

Modular Approach: The Missing Piece

"Traceable Signature" = "Signature" + "Revocation Mechanism"



A Modular Traceable Sig. Construction

[AFGHO10] Signatures
[BB04] One-Time Signatures

Groth-Sahai Proof System

Anonymous Tag System
[A**C**HO @ ACNS '11 or @ IJIS '13]

"Traceable Signature" = "Signature" + "Revocation Mechanism"

- Guarantees integrity of messages.
- Authenticate the signer.

- Opening, Tracing
- Claiming, Denial

Glued by NIZK that guarantees correct computation while hiding privacy related objects in each part.

[adapted from Abe's slides]

Structure-Preserving Signatures

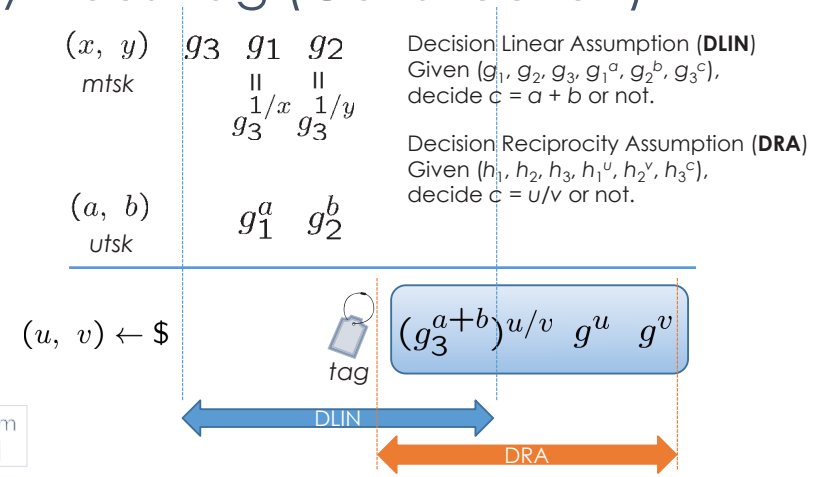
- Structure-preserving
  - [Abe-Fuchsbauer-Groth-Haralambiev-Ohkubo @ Crypto '10]:
  - Message M to be signed is a base group element
  - The signature is also formed by base group elements (not $\mathbf{G}_T$)
- GS proof cannot prove things about $\mathbf{G}_T$ elements
- Yet, signature like M^β is insecure (*cf.*, textbook RSA)
- Needs at least 2 equations to verify
  - (a proven minimum)

Double-Trapdoor Anonymous Tag

- A tag is produced by a user-secret w.r.t user public key.
- All tags of a given user are anonymous and unlinkable.
- The master-secret key can create a user-specific token.
- Token links all tags, but remain anonymous w.r.t. upk.

- With the user-secret, the user can claim the authorship.
- (And also deny the authorship of any other's tags.)
- The claim will be associated to the user public key.

Anonymous Tag (Construction)

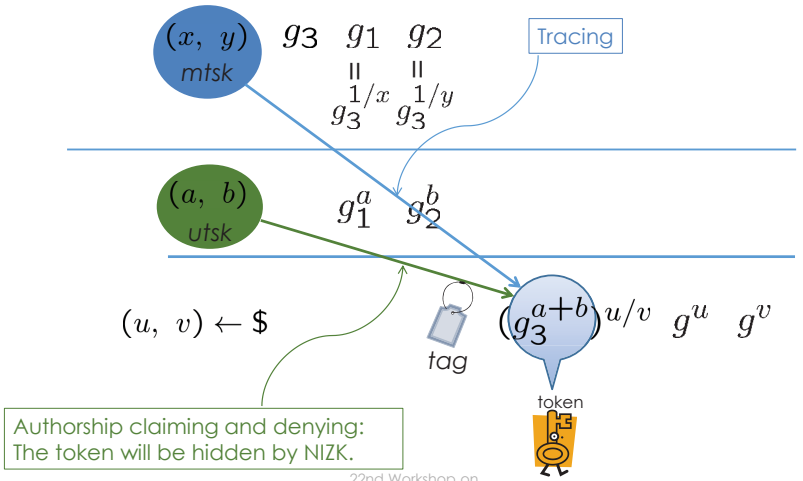


Decision Linear Assumption (**DLIN**)
Given (g_1, g_2, g_3, g_1^a, g_2^b, g_3^c), decide $c = a + b$ or not.

Decision Reciprocity Assumption (**DRA**)
Given (h_1, h_2, h_3, h_1^u, h_2^v, h_3^c), decide $c = u/v$ or not.

[adapted from Abe's slides]

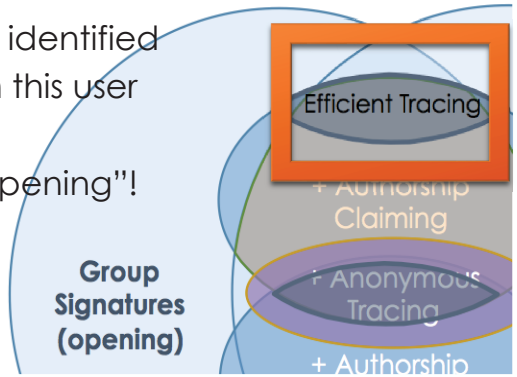
Double-Trapdoor Structure



Restrict the Power of Opening

- Opening is too powerful
- When an abusive user is identified
- Trace all signatures from this user

- This is "Tracing", NOT "Opening"!



Restrict the Power of Opening, for real

- Message-Dependent Opening (MDO) --- opening a signature on *m* needs an additional trapdoor for *m*
- The first construction is proposed by [Sakai, Emura, Hanaoka, Kawai, Matsuda, Omote @ Pairing '12]
  - Relied on *k*-resilient identity-based encryption (IBE)
- [Libert and Joye @ CTRSA '14] achieved MDO by proposing partially structure-preserving IBE.
  - The message to encrypt is in **G**, but not the identity
- The original opening key is still needed for opening.

Group Sig. w/ Event-Dependent Opening

- Event-dependent opening (EDO) decouples the opening criteria from the signed message.
  - e.g., the event is for e-voting
  - cf. "linkability context" in linkable ring signature
- Event-dependent trapdoor is derived by opening key.
  - Opening key is no longer an input for the opening algorithm.
- [Zhang-Wu-**C**@CTRSA'19] proposed structure-preserving certificateless encryption and group signature w/ EDO.

III. Message Privacy

- For outsourcing database, say, for further processing, not all data should be revealed.
  - E.g. 1: Personal identification information for a medical record should be sanitized.
  - E.g. 2: Secure routing [Ateniese *et al.* @ ESORICS '15]
- It is desirable to sanitize sensitive *signed* information without asking the original signer to sign again, before releasing the information to public.
- *Computation over data signed by multiple signers, see [Lai-Tai-Wong-**C** @ AsiaCrypt '18]

Sanitizable Signatures

- Proposed by [Ateniese *et al.* @ ESORICS '15]
- Signer signs the fixed part and modification allowed
- Support "controlled malleability"
- A designated sanitizer can sanitize a signature
  - without the help of the original owner
- Signer Accountability: Signer cannot accuse sanitizer.
- Sanitizer Accountability: Sanitizer can't accuse signer.

Accountability vs. Privacy

- Accountability is easy to achieve without privacy.
- Transparency: Sanitized and fresh signatures should be indistinguishable.

- Unlinkability: Sanitized signature from different sources should be indistinguishable[Brzuska *et al.* @ PKC '10]
- $\mathrm{Sig}(m') \rightarrow \sigma' \approx \sigma \leftarrow \mathrm{San}(m, \sigma, \mathrm{Mod})$
  - where $m' = \mathrm{Mod}(m)$
- 2 modular approaches (with new building blocks) are proposed by [Lai-Zhang-**C**-Schröder @ ESORICS '16]

Sanitizable Sig. from Rerandomizable Tag

- A new building block proposed by [LZ**C**S@ESORICS '16]
- Issuer generates a tag using its secret key
  - w.r.t. a user public key.
- Issuer can claim the authorship of the tag.
- User can use its own secret key to rerandomize the tag.
- Randomized tags are indistinguishable from issuer's one.
- Issuer can then deny the authorship of the tag.
  - The original embedded randomness is "spoiled".
- It's a dual notion of double-trapdoor anonymous tag.
- We only know how to construct it with lattice.

Sanitizable Sig. from Accountable Ring Sig.

- The first construction is transparent but not unlinkable.
- The signer ring-signs with the ring = {signer, sanitizer}.
- The signer signs the fixed part with a regular signature.
- To sanitize, ring-signs with a new message.

- Accountability features reveals the true signer.
- In [LZ**C**S @ ESORICS '16], signer is opening authority.
- In [Bultel-Lafourcade @ CANS '17], the sanitizer can prove that s/he *didn't* sanitize.

References (1/2)

- Sherman S. M. Chow, Willy Susilo, Tsz Hon Yuen: Escrowed Linkability of Ring Signatures and Its Applications. VIETCRYPT 2006: 175-192
- Sherman S. M. Chow, Joseph K. Liu, Duncan S. Wong: Robust Receipt-Free Election System with Ballot Secrecy and Verifiability. NDSS 2008
- Sherman S. M. Chow: Real Traceable Signatures. Selected Areas in Cryptography 2009: 92-107
- Sherman S. M. Chow, Kristiyan Haralambiev: Non-interactive Confirmer Signatures. CT-RSA 2011: 49-64
- Sherman S. M. Chow, Yi Jun He, Lucas Chi Kwong Hui, Siu-Ming Yiu: SPICE - Simple Privacy-Preserving Identity-Management for Cloud Environment. ACNS 2012: 526-543

References (2/2)

- Masayuki Abe, Sherman S. M. Chow, Kristiyan Haralambiev, Miyako Ohkubo: Double-trapdoor anonymous tags for traceable signatures. Int. J. Inf. Sec. 12(1): 19-31 (2013)
- Man Ho Au, Willy Susilo, Yi Mu, Sherman S. M. Chow: Constant-Size Dynamic k-Times Anonymous Authentication. IEEE Systems Journal 7(2): 249-261 (2013)
- Russell W. F. Lai, Tao Zhang, Sherman S. M. Chow, Dominique Schröder: Efficient Sanitizable Signatures Without Random Oracles. ESORICS (1) 2016: 363-380
- Sherman S. M. Chow, Haibin Zhang, Tao Zhang: Real Hidden Identity-Based Signatures. Financial Cryptography 2017: 21-38
- Russell W. F. Lai, Raymond K. H. Tai, Harry W. H. Wong, Sherman S. M. Chow: Multi-Key Homomorphic Signatures Unforgeable under Insider Corruption. ASIACRYPT 2018. To appear.
- Tao Zhang, Huangting Wu, Sherman S. M. Chow: Structure-Preserving Certificateless Encryption. CT-RSA 2019. To appear.

Summary

- We quickly went through part of the 30-years history of signatures with privacy concerns.
- We briefly discussed 2 core pairing-based techniques: Groth-Sahai proof and Structure-Preserving Signature.
- We briefly discussed 2 pairing-based constructions: BBS+ signatures and double-trapdoor anonymous tag.

Q&A: sherman@ie.cuhk.edu.hk

- Verifiability Privacy
  - Undeniable Signatures
  - (Universal) Designated Verifier Signatures
  - (Non-Interactive) Confirmer Signatures
- (Accountable) Signer Privacy
  - Ring Signatures (with (Escrowed) Linkability)
  - Verifiable Ring Signatures
  - Accountable Ring Signatures
  - Hidden ID-Based Signatures (or No-Member-Lis Group S atures)
  - (Real) Traceable (Group) Signatures (with De l Proof)
- Message Privacy
  - Unlinkable and (Strongly) Accountable Saniti ble Signatures