UC Non-Interactive, Proactive, Threshold ECDSA w/ Identifiable Aborts

Ran Canetti (Boston University), Rosario Gennaro (City College, CUNY),
Steven Goldfeder (Cornell Tech), **Nikolaos Makriyannis** (Fireblocks),
Udi Peled (Fireblocks)

To appear in
CCS'20

Fireblocks

Background (MPC)

Secure Multiparty Computation

Distrustful parties compute correlated outputs on their (secret) inputs and **only** reveal what the outputs suggest.

☺ Powerful Feasibility Results

Yao’82, Goldreich-Micali-Widgerson’86,

Chaum-Crepeau-Damgard’88, Ben Or-Goldwasser-Wigderson’88

☺ Any traditional signature scheme can be “thresholdized”, *in principle*

☹ MPC theory is not a panacea

Desiderata

➢ Non-Interactive Signing

Signature generation boils down to a single message (w/ preprocess).

Especially relevant for
“cold wallets”.

Desiderata

- **Non-Interactive Signing**

  Signature generation boils down to a single message (w/ preprocess).

- **Accountability**

  Faulty/malicious signatories are identified in case of failure.

Known as security
w/ identifiable abort
in MPC literature.

Desiderata

- Non-Interactive Signing

  Signature generation boils down to a single message (w/ preprocess).

- Accountability

  Faulty/malicious signatories are identified in case of failure.

- Proactive Security

  Long-haul security against adaptive adversaries.

Adaptive vs Static Adversaries

Desiderata

- **Non-Interactive Signing**
  Signature generation boils down to a single message (w/ preprocess).
- **Accountability**
  Faulty/malicious signatories are identified in case of failure.
- **Proactive Security**
  Long-haul security against adaptive adversaries.
- **UC Security**
  Security preserved under composition.

Even when multiple different sessions are occurring simultaneously.

Desiderata

- ➢ Non-Interactive Signing
  Signature generation boils down to a single message (w/ preprocess).
- ➢ Accountability
  Faulty/malicious signatories are identified in case of failure.
- ➢ Proactive Security
  Long-haul security against adaptive adversaries.
- ➢ UC Security
  Security preserved under composition.

We show how to achieve all of these properties in one protocol!

Previous/Concurrent Work on t-ECDSA

Honest Majority:

Gennaro-Jarecki-Krawcyk-Rabin’96

Two-Party Dishonest Majority:

Mackenzie-Reiter’01

Lindell’17, Doerner-Shelat’18, Castagnos-Catalano-Laguillaumie-Savasta-Tucker’19

Multiparty Dishonest Majority:

Gennaro-Goldfeder-Narayanan’16, Boneh-Gennaro-Goldfeder’17

Lindell-Nof’19, Gennaro-Goldfeder’19, Doerner-Kondi-Lee-Shelat’20

Castagnos-Catalano-Laguillaumie-Savasta-Tucker’20

Previous/Concurrent Work on t-ECDSA

Honest Majority:

Gennaro-Jarecki-Krawcyk-Rabin'96

Damgard-Jakobsen-Nielsen-Pagter-Ostergaard'20

Two-Party Dishonest Majority:

Mackenzie-Reiter'01

Lindell'17, Doerner-Shelat'18, Castagnos-Catalano-Laguillaumie-Savasta-Tucker'19

Multiparty Dishonest Majority:

Gennaro-Goldfeder-Narayanan'16, Boneh-Gennaro-Goldfeder'17

Lindell-Nof'19, Gennaro-Goldfeder'19, Doerner-Kondi-Lee-Shelat'20

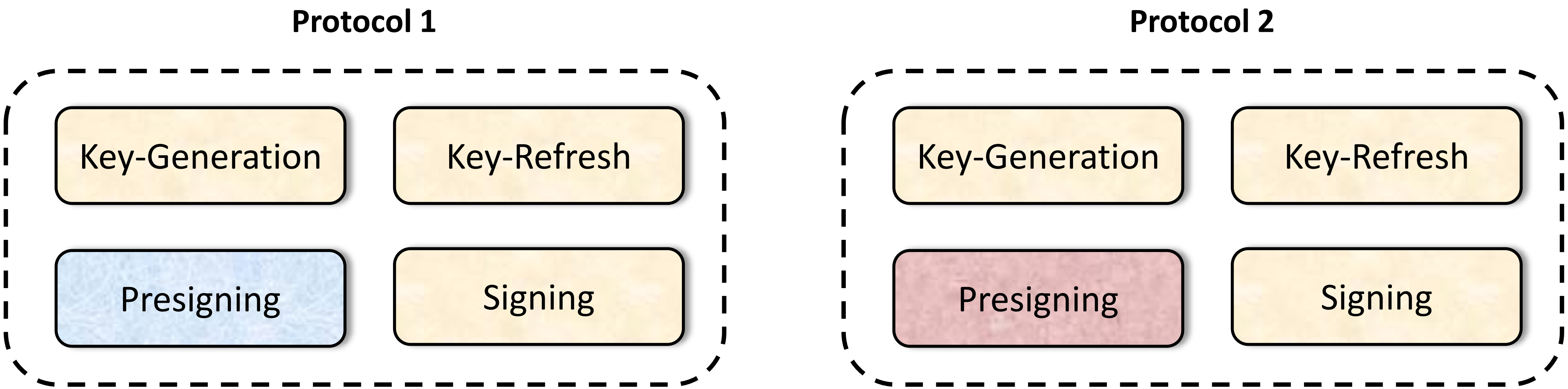
Castagnos-Catalano-Laguillaumie-Savasta-Tucker'20

Dalskov-Keller-Orlandi-Shrishak-Shulman'20

Gagol-Kula-Straszak-Swietek'20

Our Results

We present **two** related protocols for threshold ECDSA.



Communication Model:

We rely on synchronous broadcast channel

Our Results (cont’d)

We present **two** related protocols for threshold ECDSA.

	PROTOCOL 1	PROTOCOL 2
Non-Interactive Signing	✓	✓
Full Proactive Security	✓	✓
Accountability	✓	✓
UC - Security	✓	✓

Our Results (cont'd)

We present **two** related protocols for threshold ECDSA.

	PROTOCOL 1	PROTOCOL 2
Non-Interactive Signing	✓	✓
Full Proactive Security	✓	✓
Accountability	✓	✓
UC - Security	✓	✓
Round-Complexity (Signing)	4 i.e. $3+1$	7 i.e. $6+1$
Accountability Overhead	$O(n^2)$	$O(n)$

Overhead kicks in only when a fault is detected

Comparison

Most
Round-Efficient

Signing Protocol	*Rounds*	*Group Ops*	*Ring Ops*	*Communication*	*Proactive*	*ID Abort*	*UC*
Gennaro and Goldfeder [30]	9	$10n$	$50n$	$10\kappa + 20N$ (7 KiB)	✗	✗	✗
Lindell et al. [45] (Paillier)†‡	8	$80n$	$50n$	$50\kappa + 20N$ (7.5 KiB)	✗	✗	✓
Lindell et al. [45] (OT)†	8	$80n$	0	50κ (190 KiB)	✗	✗	✓
Doerner et al. [27]	$\log(n) + 6$	5	0	$10 \cdot \kappa^2$ (90 KiB)	✗	✗	✓
Castagnos et al. [20]*	8	$15n$	0	$100 \cdot \kappa$ (4.5 KiB)	✗	✗	✗
This Work: *Interactive*§	4 *or* 7	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Presign*§	3 *or* 6	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Sign*	1	0	0	κ (256 bits)	✓	✓	✓

Comparison

Most
Round-Efficient

Signing Protocol	*Rounds*	*Group Ops*	*Ring Ops*	*Communication*	*Proactive*	*ID Abort*	*UC*
Gennaro and Goldfeder [30]	9	$10n$	$50n$	$10\kappa + 20N$ (7 KiB)	✗	✗	✗
Lindell et al. [45] (Paillier)†‡	8	$80n$	$50n$	$50\kappa + 20N$ (7.5 KiB)	✗	✗	✓
Lindell et al. [45] (OT)†	8	$80n$	0	50κ (190 KiB)	✗	✗	✓
Doerner et al. [27]	$\log(n) + 6$	5	0	$10 \cdot \kappa^2$ (90 KiB)	✗	✗	✓
Castagnos et al. [20]*	8	$15n$	0	$100 \cdot \kappa$ (4.5 KiB)	✗	✗	✗
This Work: *Interactive*§	4 *or* 7	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Presign*§	3 *or* 6	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Sign*	1	0	0	κ (256 bits)	✓	✓	✓

Comparison

Most Round-Efficient

~2 as expensive in comp & com compared to the most com-efficient protocols

Signing Protocol	*Rounds*	*Group Ops*	*Ring Ops*	*Communication*	*Proactive*	*ID Abort*	*UC*
Gennaro and Goldfeder [30]	9	$10n$	$50n$	$10\kappa + 20N$ (7 KiB)	✗	✗	✗
Lindell et al. [45] (Paillier)†‡	8	$80n$	$50n$	$50\kappa + 20N$ (7.5 KiB)	✗	✗	✓
Lindell et al. [45] (OT)†	8	$80n$	0	50κ (190 KiB)	✗	✗	✓
Doerner et al. [27]	$\log(n) + 6$	5	0	$10 \cdot \kappa^2$ (90 KiB)	✗	✗	✓
Castagnos et al. [20]*	8	$15n$	0	$100 \cdot \kappa$ (4.5 KiB)	✗	✗	✗
This Work: *Interactive*§	4 *or* 7	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Presign*§	3 *or* 6	$10n$	$90n$	$10\kappa + 50N$ (15 KiB)	✓	✓	✓
This Work: *Non-Int. Sign*	1	0	0	κ (256 bits)	✓	✓	✓

TECHNICAL OVERVIEW

Background

Preliminaries (Notation)

For $T \in \mathbb{N}$, let $\pm T$ denote $\{-T, \ldots, 0, \ldots, T\}$.

Non Standard Notation!!

Index disappearance denotes summation

e.g. if x_i, k_j, δ_ℓ ... becomes x, k, δ ... it means $\sum_i x_i, \sum_j k_j, \sum_\ell \delta_\ell$ ...

Also for double indices!

Preliminaries (ECDSA)

- Parameters:
  - $(\mathbb{G}, g, q)$ group-generator-order and hash $\mathcal{H}: \{0,1\}^* \rightarrow \mathbb{F}_q$.
- Algorithms:
  - $\mathrm{keygen}() = \left(x \leftarrow \mathbb{F}_q,\ X = g^x \in \mathbb{G}\right)$
  - $\mathrm{sign}_x(\mathrm{msg}) = (r, \sigma)$ s.t.

$$r = g^{k^{-1}}|_{\mathrm{x-axis}}$$ and $\sigma = k(m + rx)$.

where $k \leftarrow \mathbb{F}_q$ and $m = \mathcal{H}(\mathrm{msg})$.

Preliminaries (ECDSA)

- Parameters:
  - $(\mathbb{G}, g, q)$ group-generator-order and hash $\mathcal{H}: \{0,1\}^* \rightarrow \mathbb{F}_q$.
- Algorithms:
  - $\text{keygen}() = \left(x \leftarrow \mathbb{F}_q,\ X = g^x \in \mathbb{G}\right)$
  - $\text{sign}_x(\text{msg}) = (r, \sigma)$ s.t.

$$r = g^{k^{-1}}|_{\text{x-axis}} \text{ and } \boxed{\sigma = k \cdot m + r(k \cdot x).}$$

where $k \leftarrow \mathbb{F}_q$ and $m = \mathcal{H}(\text{msg})$.

Preliminaries (ECDSA)

- Parameters:
  - $(\mathbb{G}, g, q)$ group-generator-order and hash $\mathcal{H}: \{0,1\}^* \to \mathbb{F}_q$.
- Algorithms:
  - $\text{keygen}() = \left(x \leftarrow \mathbb{F}_q,\ X = g^x \in \mathbb{G}\right)$
  - $\text{sign}_x(\text{msg}) = (r, \sigma)$ s.t.

$$r = g^{k^{-1}}|_{\text{x-axis}} \text{ and } \sigma = k \cdot m + r(k \cdot x).$$

where $k \leftarrow \mathbb{F}_q$ and $m = \mathcal{H}(\text{msg})$.

(Gist of) MPC sign:
Sample shares $k_1 \ldots k_n$ of k and compute **shares** of $k \cdot x$ **via pairwise multiplication** with $x_1 \ldots x_n$.

Preliminaries (ECDSA)

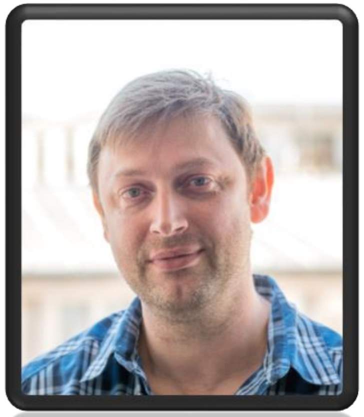
- Parameters:
  - $(\mathbb{G}, g, q)$ group-generator-order and hash $\mathcal{H}: \{0,1\}^* \to \mathbb{F}_q$.
- Algorithms:
  - $\text{keygen}() = \left(x \leftarrow \mathbb{F}_q,\ X = g^x \in \mathbb{G}\right)$
  - $\text{sign}_x(\text{msg}) = (r, \sigma)$ s.t.

$$r = g^{k^{-1}}|_{\text{x-axis}} \text{ and } \sigma = k \cdot m + r(k \cdot x).$$

  - $\text{vrfy}_X(\text{msg};\ r, \sigma) = 1$ if and only if $(\text{g}^m \cdot X^r)^{\sigma^{-1}}|_{\text{x-axis}} = r$.

where $k \leftarrow \mathbb{F}_q$ and $m = \mathcal{H}(\text{msg})$.

Preliminaries (Paillier Encryption)



- Algorithms:
  - ➢ $\mathrm{keygen}()$ = RSA Modulus & Factors $(N; p_1, p_2)$
  - ➢ $\mathrm{enc}_N(m \in \mathbb{Z}_N) = (1+N)^m \cdot \rho^N \bmod N^2$
  - ➢ $\mathrm{dec}_{\varphi(N)}\left(C \in \mathbb{Z}_{N^2}^*\right) = \frac{\mathrm{C}^{\varphi(N)-1 \bmod N^2}}{N} \cdot \phi(N)^{-1} \bmod N$

Where $\rho \leftarrow \mathbb{Z}_N^*$

Easy to deduce m knowing $\varphi(N)$

Preliminaries (Paillier Encryption)

- Algorithms:
  - $\text{keygen}()$ = RSA Modulus & Factors $(N; p_1, p_2)$
  - $\text{enc}_N(m \in \mathbb{Z}_N) = (1+N)^m \cdot \rho^N \bmod N^2$

Where $\rho \leftarrow \mathbb{Z}_N^*$

  - $\text{dec}_{\varphi(N)}\left(C \in \mathbb{Z}_{N^2}^*\right) = \frac{C^{\varphi(N)-1 \bmod N^2}}{N} \cdot \phi(N)^{-1} \bmod N$

Easy to deduce m knowing $\varphi(N)$

- Paillier is additive homomorphic:

$$\text{enc}_N(m_1 + m_2) = \text{enc}_N(m_1) \cdot \text{enc}_N\,(m_2)$$

$$\text{enc}_N(\alpha \cdot m) = \text{enc}_N(m)^\alpha$$

Preliminaries (Multiplication via Paillier)

$\mathcal{A}$ and $\mathcal{B}$ wish to compute $(a, b) \mapsto (s_1, s_2)$ such that

$$s_1 + s_2 = a \cdot b$$

$\mathcal{A}$ is associated with Paillier public key N

1. $\mathcal{A}$ sends $C = \mathrm{enc}(a)$

2. $\mathcal{B}$ samples s_2 and replies with $D = C^b \cdot \mathrm{enc}(-s_2)$

$\mathrm{dec}(D) = ab - s_2$

Output: $\mathcal{A}$ outputs $s_1 = \mathrm{dec}(D)$ and $\mathcal{B}$ outputs s_2.

Protocol (Honest-But-Curious)

From $\mathcal{P}_i$ perspective - Each $\mathcal{P}_i$ holds secret key-share x_i

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.
2. For each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ for $\beta_{i,j} \leftarrow \pm 2^{\ell} \cdot q$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ for $\beta'_{i,j} \leftarrow \pm 2^{\ell} \cdot q$

   Send $(D_{j,i}, D'_{j,i})$ to $\mathcal{P}_j$.
3. Set $\Gamma_i = g^{\gamma_i}$ and send (Γ_i, δ_i) to all
4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all.

Write $\chi_{i,j}$ and $\delta_{i,j}$
for $\mathcal{P}_i$'s output in each mult.
NB $\rightarrow$ $\delta = k \cdot \gamma$ and $\chi = k \cdot x$

$\gamma \cdot \delta^{-1} = k^{-1}$

Output (r, σ).

Malicious Security Challenges

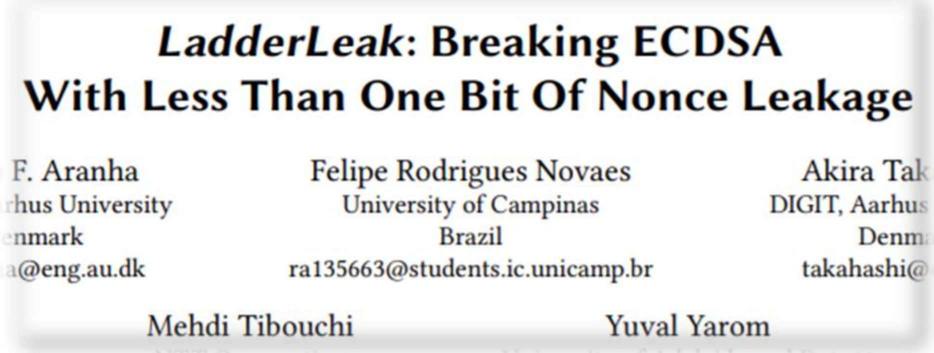
We are embedding values of $\mathbb{F}_q$ into $\mathbb{Z}_N$ (q & N are coprime)

$$\mathrm{enc}(\gamma \cdot k\ + \beta \mod q) \overset{?}{=} \mathrm{enc}(\gamma \cdot k\ + \beta) \bmod q \qquad (\dagger)$$

In case of equality → signature verifies

Otherwise → signature **does not** verify

Carefull choice of γ & β reveals a bit of information per protocol execution.



***LadderLeak*: Breaking ECDSA With Less Than One Bit Of Nonce Leakage**

F. Aranha
hus University
nmark
@eng.au.dk

Felipe Rodrigues Novaes
University of Campinas
Brazil
ra135663@students.ic.unicamp.br

Akira Tak
DIGIT, Aarhus
Denm
takahashi@

Mehdi Tibouchi

Yuval Yarom

Malicious Security Challenges

We are embedding values of $\mathbb{F}_q$ into $\mathbb{Z}_N$ (q & N are coprime)

$$\mathrm{enc}(\gamma \cdot k + \beta \mod q) \stackrel{?}{=} \mathrm{enc}(\gamma \cdot k + \beta) \mod q \qquad (\dagger)$$

In case of equality → signature verifies

Otherwise → signature **does not** verify

Carefull choice of γ & β reveals a bit of information per protocol execution.

☺ **Solution:** Enforce a "range policy" on all secret data

i.e. values can only be chosen from some range $\pm 2^{\ell} \ll N$

Also in Lindell-Nof'18 and Gennaro-Goldfeder'18

ZK-Proofs for $\mathcal{R} = \{(N, C;\ x) | C = \mathrm{enc}_N(x) \wedge x \in \pm 2^{\ell}\}$

Our Protocol(s)

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.
2. For each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j\left(\beta_{i,j}\right)$ for $\beta_{i,j} \leftarrow \pm 2^{\ell} \cdot q$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j\left(\beta'_{i,j}\right)$ for $\beta'_{i,j} \leftarrow \pm 2^{\ell} \cdot q$

   Send $\left(D_{j,i}, D'_{j,i}\right)$ to $\mathcal{P}_j$.
3. Set $\Gamma_i = g^{k_i}$ and send (Γ_i, δ_i) to all
4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

Prove that k_i is small.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using **small** values

Verify that R is well-formed

Our Protocol

Prove that k_i is small.

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.

2. For each $j \neq i$ do

   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ for $\beta_{i,j} \leftarrow \pm 2^{\ell} \cdot q$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ for $\beta'_{i,j} \leftarrow \pm 2^{\ell} \cdot q$

   Send $(D_{j,i}, D'_{j,i})$ to $\mathcal{P}_j$.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using **small** values

3. Set $\Gamma_i = g^{k_i}$ and send (Γ_i, δ_i) to all

4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

NEW!
Special algebraic check for R.

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.

2. Set $\Gamma_i = g^{\gamma_i}$ and for each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ for $\beta_{i,j} \leftarrow \pm 2^{\ell} \cdot q$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ for $\beta'_{i,j} \leftarrow \pm 2^{\ell} \cdot q$

   Send $(\Gamma_i, D_{j,i}, D'_{j,i})$ to $\mathcal{P}_j$.

3. Set $\Delta_i = \left(\prod_j \Gamma_j\right)^{k_i}$ and send (Δ_i, δ_i) to all

4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

Prove that k_i is small.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using **small** values

NEW!
Special algebraic check for R.

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \text{enc}_i(k_i)$ to all.

Prove that k_i is small.

2. Set $\Gamma_i = g^{\gamma_i}$ and for each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \text{enc}_j(\beta_{i,j})$ for $\beta_{i,j} \leftarrow \pm 2^{\ell} \cdot q$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \text{enc}_j(\beta'_{i,j})$ for $\beta'_{i,j} \leftarrow \pm 2^{\ell} \cdot q$

   Send $(\Gamma_i, D_{j,i}, D'_{j,i})$ to $\mathcal{P}_j$.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using **small** values

Prove that you use the right k_i.

3. Set $\Delta_i = \left(\prod_j \Gamma_j\right)^{k_i}$ and send (Δ_i, δ_i) to all

Check that $g^{\delta} = \prod_j \Delta_j$

4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.

Prove that k_i is small.

2. Set $\Gamma_i = g^{\gamma_i}$ and for each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ and $F_{j,i} = \mathrm{enc}_i(\beta_{i,j})$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ and $F'_{j,i} = \mathrm{enc}_i(\beta'_{i,j})$

   Send $(\Gamma_i, D_{j,i}, D'_{j,i}, F_{j,i}, F'_{j,i})$ to $\mathcal{P}_j$.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using **small** values

Prove that you use the right k_i.

3. Set $\Delta_i = \left(\prod_j \Gamma_j\right)^{k_i}$ and send (Δ_i, δ_i) to all

Check that $g^{\delta} = \prod_j \Delta_j$

4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

Output (r, σ) if it's a valid sig

Accountability

Accountability

Fault Attribution Process(es)

- ☺ If zk-proof fails, attribute fault to relevant party. 🔫
- ☹ Parties verify only parts of the transcript.
- ☹ Offline GMW-Style accountability is wasteful.

$O(n^2)$ comp/comm overhead for "GMW-style accountability"

Accountability

Fault Attribution Process(es)

If nonce R is malformed:

a) Open* all the ciphertexts $\{D'_{i,j}\}_{j \neq i}$.

b) Verify which party sent the wrong δ_j.

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.

   Prove that k_i is small.

2. Set $\Gamma_i = g^{\gamma_i}$ and for each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ and $F_{j,i} = \mathrm{enc}_i(\beta_{i,j})$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ and $F'_{j,i} = \mathrm{enc}_i(\beta'_{i,j})$

   Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using small values

   Send $(\Gamma_i, D_{j,i}, D'_{j,i}, F_{j,i}, F'_{j,i})$ to $\mathcal{P}_j$.

   Prove that you use the right k_i.

3. Set $\Delta_i = \left(\prod_j \Gamma_j\right)^{k_i}$ and send (Δ_i, δ_i) to all

   Check that $g^\delta = \prod_j \Delta_j$

4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

   Output (r, σ) if it's a valid sig

Accountability

Fault Attribution Process(es)

If signature-string does not verify

☹ Not possible to reveal the underlying plaintexts.

Includes long-term secrets $x_1 \dots x_n$

☺ Our Solution for Protocol 2

a) Reveal $S_j = R^{k_j}$ and $Y_j = R^{\chi_j}$ during presigning.

Check that they are well-formed**.

b) Once m is known check $R^{\sigma_i} = S_i^m \cdot Y_i^r$.

Incurs a round-complexity penalty.

$O(n)$ comp/comm overhead!

Our Protocol

1. Sample $k_i, \gamma_i \leftarrow \mathbb{F}_q$ and send $K_i = \mathrm{enc}_i(k_i)$ to all.

Prove that k_i is small.

2. Set $\Gamma_i = g^{\gamma_i}$ and for each $j \neq i$ do
   - Set $D_{j,i} = K_j^{x_i} \cdot \mathrm{enc}_j(\beta_{i,j})$ and $F_{j,i} = \mathrm{enc}_i(\beta_{i,j})$
   - Set $D'_{j,i} = K_j^{\gamma_i} \cdot \mathrm{enc}_j(\beta'_{i,j})$ and $F'_{j,i} = \mathrm{enc}_i(\beta'_{i,j})$

   Send $(\Gamma_i, D_{j,i}, D'_{j,i}, F_{j,i}, F'_{j,i})$ to $\mathcal{P}_j$.

Prove that $D_{j,i}$ and $D'_{j,i}$ were computed as prescribed using small values

Prove that you use the right k_i.

3. Set $\Delta_i = \left(\prod_j \Gamma_j\right)^{k_i}$ and send (Δ_i, δ_i) to all
4. Set $R = \left(\prod_j \Gamma_j\right)^{\delta^{-1}}$ and send $\sigma_i = k_i\, m + r\chi_i$ to all

Check that $g^{\delta} = \prod_j \Delta_j$

Output (r, σ) if it's a valid sig

Security Analysis

Security Analysis

Analysis in ROM

Previous works show security either via

1. Secure FE of ECDSA (in standalone or UC-framework)
2. Standalone reduction to unforgeability of ECDSA

THIS WORK (New)

Our protocol(s) UC-realize an ideal threshold signature functionality.

1. *Authorized sets can generate valid signatures.*
2. *Unauthorized sets cannot generate valid signatures.*

Crux of the proof:
UC simulation is indistinguishable unless non-threshold ECDSA is forgeable.

Scheme is provably secure against **adaptive** adversary

Conclusion

- We leverage Paillier Encryption as a commitment scheme

    Reduces round-complexity and enables concurrent signings.

- We devise a special-purpose technique for fault attribution.

    Reduces complexity penalty for accountability.

- Completely new approach for obtaining UC-security.

    Security against adaptive adv. to gain full proactive security.