Efficient MtA from Joye-Libert and Its Application to Threshold ECDSA

Haiyang Xue, Man Ho Au, Mengling Liu, Kwan Yin Chan,
Handong Cui, Xiang Xie, Tsz Hon Yuen, Chengru Zhang

THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

香 港 大 學
THE UNIVERSITY OF HONG KONG

PADO

ECDSA

- Elliptic Curve Digital Signature Algorithm issued by NIST

- Widely deployed, especially in cryptocurrency, Bitcoin etc.

ECDSA

- Elliptic Curve Digital Signature Algorithm issued by NIST

- Widely deployed, especially in cryptocurrency, Bitcoin etc.

Loss of ECDSA secret key SK = loss of money

ECDSA

- Elliptic Curve Digital Signature Algorithm issued by NIST

- Widely deployed, especially in cryptocurrency, Bitcoin etc.

Loss of ECDSA secret key SK = loss of money

This is known as the problem of single-point-of-failure

ECDSA

- Elliptic Curve Digital Signature Algorithm issued by NIST

- Widely deployed, especially in cryptocurrency, Bitcoin etc.

Loss of ECDSA secret key SK = loss of money

This is known as the problem of single-point-of-failure

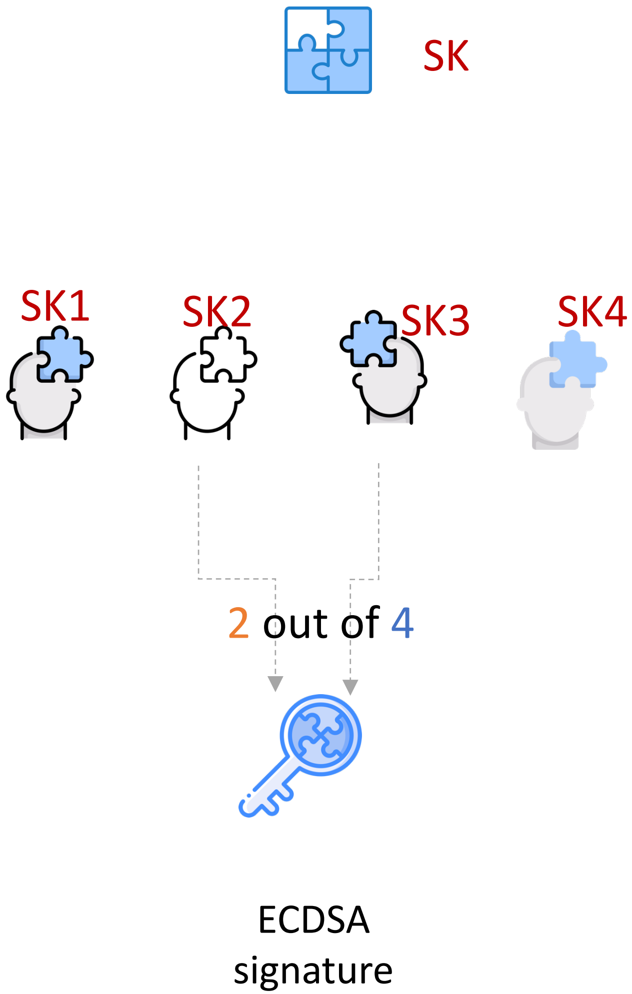
"Threshold digital signature"[DF89] aims to address this problem

[DF89] Yvo Desmedt and Yair Frankel. Threshold cryptosystems. In CRYPTO 1989

Threshold ECDSA signature

Goal: An ECDSA signature that looks like it was produced
by a single party, yet the key is stored in shared form

Naïve idea: divide key into shares



An t out of n threshold ECDSA requires
at least t parties of n signers (holding shares)
to sign an ECDSA signature

Threshold ECDSA signature

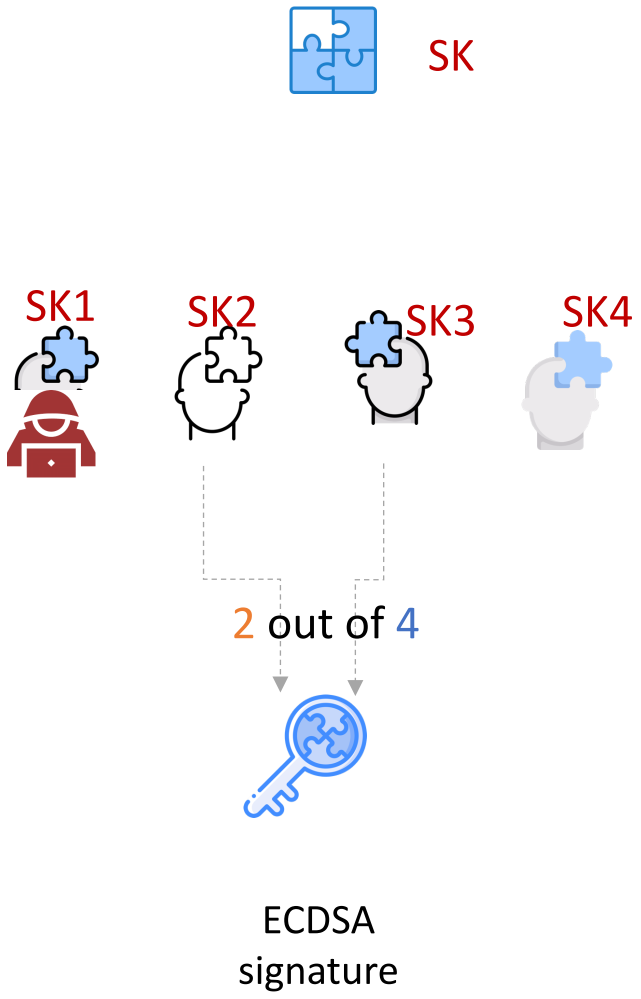
Goal: An ECDSA signature that looks like it was produced

by a single party, yet the key is stored in shared form

Naïve idea: divide key into shares

An t out of n threshold ECDSA requires
at least t parties of n signers (holding shares)
to sign an ECDSA signature

Even t-1 parties are compromised, Still Secure!



More about ECDSA

Input: secret key: $\mathrm{SK} = x$, message m

Output: Output (r, s) where

$$s = k^{-1}(H(m) + x \cdot r),$$

k is the randomness,
$H(m)$ is the hash of message m,
r can derived from $k \cdot P$

More about ECDSA

Input: secret key: $\mathrm{SK} = x$, message m

Output: Output (r, s) where

$$s = k^{-1}(H(m) + x \cdot r),$$

k is the randomness,
$H(m)$ is the hash of message m,
r can derived from $k \cdot P$

- $H(m)$ and r can be public
- k and x should be kept secret

Challenge of Threshold ECDSA

Input: secret key: $\mathrm{SK} = x$, message m

Output: Output (r, s) where

$k_1, k_2, \cdots, k_t$ $\quad$ $x_1, x_2, \cdots, x_t$

$$s = k^{-1}(H(m) + x \cdot r),$$

r is derived from $k \cdot P$

Challenge of Threshold ECDSA

Input: secret key: $\mathrm{SK} = x$, message m

Output: Output (r, s) where

$k_1, k_2, \cdots, k_t$ $\qquad$ $x_1, x_2, \cdots, x_t$

$$s = k^{-1}(H(m) + x \cdot r),$$

r is derived from $k \cdot P$

Challenge When Thresholding

How to securely compute (**non-linear**) k^{-1} and $k^{-1}x$
from shares of x and k

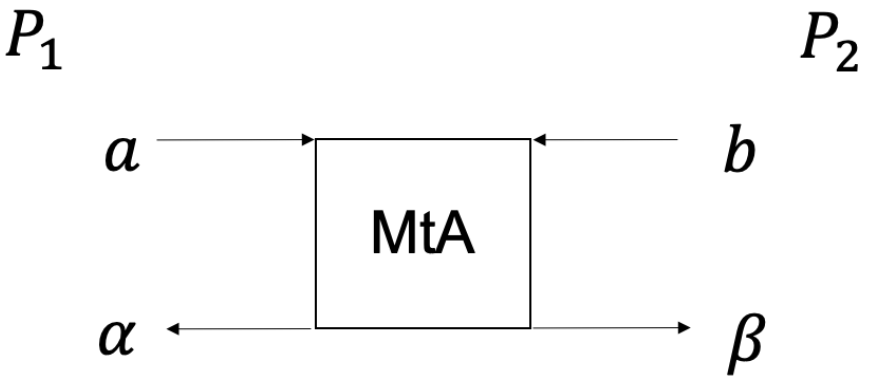
Even worse, some of the parties are controlled by the adversary

Main Building Block: MtA

Several works have been done to address the challenge

[GG18], [LN18], [DKLs18], [DKLs19], [CCL+19], [CCL+20], [CGG+21], [XAX+21], …

Multi-to-Add (MtA):



such that $\alpha + \beta = a \cdot b \bmod q$

State-of-the-art: MtA

MtA is costly

Tools ($\lambda = 128$)	Computation (ms)	Communication (KB)
Oblivious Transfer (OT) [DKLs18-19]	~10	90
Castagnos-Laguillaumie Enc [CCL+19,20]	~1600	~2
Paillier Enc [LN18,GG18]	~250	7.5

State-of-the-art: MtA

MtA is costly

Tools ($\lambda = 128$)	Computation (ms)	Communication (KB)
Oblivious Transfer (OT) [DKLs18-19]	~10	90
Castagnos-Laguillaumie Enc [CCL+19,20]	~1600	~2
Paillier Enc [LN18,GG18]	~250	7.5

MtA dominates the cost of threshold ECDSA

92%~98% cost of computation and/or communication

State-of-the-art: MtA

MtA is costly

Tools ($\lambda = 128$)	Computation (ms)	Communication (KB)
Oblivious Transfer (OT) [DKLs18-19]	~10	90
Castagnos-Laguillaumie Enc [CCL+19,20]	~1600	~2
Paillier Enc [LN18,GG18]	~250	7.5

MtA dominates the cost of threshold ECDSA

92%~98% cost of computation and/or communication

Our goal is to find better candidates of MtA (than Paillier in both comp. and comm.)?

Our work: Better MtA from Joye-Libert

MtA is costly

Tools ($\lambda = 128$)	Computation (ms)	Communication (KB)
Oblivious Transfer (OT) [DKLs18-19]	~10	90
Castagnos-Laguillaumie Enc [CCL+19,20]	~1600	~2
Paillier Enc [LN18,GG18]	~250	7.5
Our work Joye-Libert	~200	4.1

When $\lambda = 192, 256$, the improvement over Paillier-base MtA achieves 48% (resp. 44%) in communication (resp. computation)

MtA from additive HE (semi-honest)

P_1 $\qquad\qquad\qquad\qquad$ P_2

a $\qquad\qquad\qquad\qquad$ b

$$\xleftarrow{\mathrm{C}_2 = \mathrm{Enc}\,(b)}$$

$$\alpha \quad \xrightarrow{\mathrm{C}_1 = \mathrm{Enc}\,(a \cdot b - \alpha)} \quad \beta = \mathrm{Dec}\,(C_1)\; mod\; q$$

Such that $\alpha + \beta = a \cdot b \bmod q$

MtA from additive HE (semi-honest)

P_1 P_2

a b

$C_2 = \text{Enc}\,(b)$ ←

$C_1 = \text{Enc}\,(a \cdot b - \alpha)$ →

α $\beta = \text{Dec}\,(C_1)\ mod\ q$

Such that $\alpha + \beta = a \cdot b \bmod q$

Simple but insecure against adaptive adv

MtA from additive HE (semi-honest)

P_1		P_2
a		b
	$\xleftarrow{C_2 = \mathrm{Enc}\,(b)}$	
α	$\xrightarrow{C_1 = \mathrm{Enc}\,(a \cdot b - \alpha)}$	$\beta = \mathrm{Dec}\,(C_1)\; mod\; q$
		Such that $\alpha + \beta = a \cdot b \bmod q$

Simple but insecure against adaptive adv

Alpha-rays attack [TS21] due to the mismatch of message space (Paillier ~3096 bits) and q (ECDSA~256 bits)

[TS21] Dmytro Tymokhanov and Omer Shlomovits. Alpha-rays: Key extraction attacks on threshold ecdsa implementations, eprint 2021.

MtA from Paillier

Schemes	over	(Message) Space
ECDSA	$mod\ q$ (~256 bits)	$mod\ q$ (~256 bits)
Paillier	$mod\ N^2$	$mod\ N$

Mismatch

To guarantee security (e.g. against Alpha-rays Attack [TS21])

- P_2 needs to prove: $R_2 = \{C_2; b \mid C_2 = Enc(b), b \in [0, q]\}$
- P_1 needs to prove: $R_1 = \{C_1; a, \alpha \mid C_1 = Enc(ab - \alpha), a, \alpha \in [0, q]\}$

MtA from Paillier

Schemes	over	(Message) Space
ECDSA	$mod\ q$ (~256 bits)	$mod\ q$ (~256 bits)
Paillier	$mod\ N^2$	$mod\ N$

Mismatch

To guarantee security (e.g. against Alpha-rays Attack [TS21])

- P_2 needs to prove: $R_2 = \{C_2; b \mid C_2 = Enc(b), b \in [0, q]\}$
- P_1 needs to prove: $R_1 = \{C_1; a, \alpha \mid C_1 = Enc(ab - \alpha), a, \alpha \in [0, q]\}$

zero-knowledge range proof

Simple Observation:

There is waste message space in Paillier

Expensive zero-knowledge range proof is required

MtA from Joye-Libert Enc

Schemes	over	(Message) Space
ECDSA	$mod\ q$ (~256 bits)	$mod\ q$ (~256 bits)
Paillier	$mod\ N^2$	$mod\ N$
Joye-Libert Enc	$mod\ N$	$mod\ 2^k$

Mismatch

- P_2 needs to prove: $R_2 = \{C_2; b \mid C_2 = Enc(b), b \in [0, q]\}$
- P_1 needs to prove: $R_1 = \{C_1; a, \alpha \mid C_1 = Enc(ab - \alpha), a, \alpha \in [0, q]\}$

[BSJL17] Fabrice Benhamouda, Javier Herranz Sotoca, Marc Joye, and Benoit Libert. Efficient cryptosystems from $2k$-th power residue symbols. Journal of cryptology 2017

MtA from Joye-Libert Enc

Schemes	over	(Message) Space
ECDSA	$mod\ q$ (~256 bits)	$mod\ q$ (~256 bits)
Paillier	$mod\ N^2$	$mod\ N$
Joye-Libert Enc	$mod\ N$	$mod\ 2^k$

Mismatch

- P_2 needs to prove: $R_2 = \{C_2; b \mid C_2 = Enc(b), b \in [0, q]\}$
- P_1 needs to prove: $R_1 = \{C_1; a, \alpha \mid C_1 = Enc(ab - \alpha), a, \alpha \in [0, q]\}$

The challenge is

no standard zero-knowledge (range) proof for Joye-Libert with large challenge space

[BSJL17] Fabrice Benhamouda, Javier Herranz Sotoca, Marc Joye, and Benoit Libert. Efficient cryptosystems from $2k$-th power residue symbols. Journal of cryptology 2017

MtA from Joye-Libert Enc

Standard zero-knowledge (range) proof for Joye-Libert with large challenge space

$$(e - e')m = z_m - z'_m \; mod \; 2^k$$

MtA from Joye-Libert Enc

Standard zero-knowledge (range) proof for Joye-Libert with large challenge space

$$(e - e')m = z_m - z'_m \ mod \ 2^k$$

Current solutions:

- Small challenge space, $e, e' \in \{0, 1\}$; inefficient
- Non-standard soundness [CRFG20]: do not extract all the bits of m

MtA from Joye-Libert Enc

Standard zero-knowledge (range) proof for Joye-Libert with large challenge space

$$(e - e')m = z_m - z'_m \; mod \; 2^k$$

Current solutions:

- Small challenge space, $e, e' \in \{0, 1\}$; inefficient
- Non-standard soundness [CRFG20]: do not extract all the bits of m

Our solution:

- Modified Joye-Libert: $\mathrm{Enc}(m) = \mathrm{C} = y^m \, h^r \; mod \; N$

MtA from Joye-Libert Enc

Standard zero-knowledge (range) proof for Joye-Libert with large challenge space

$$(e - e')m = z_m - z'_m \ mod \ 2^k$$

Current solutions:

- Small challenge space, $e, e' \in \{0, 1\}$; inefficient
- Non-standard soundness [CRFG20]: do not extract all the bits of m

Our solution:

- Modified Joye-Libert: $\mathrm{Enc}(m) = \mathrm{C} = y^m \, h^r \ mod \ N$
- It is an encryption, and at the same time, an integer commitment

MtA from Joye-Libert Enc

Standard zero-knowledge (range) proof for Joye-Libert with large challenge space

$$(e - e')m = z_m - z'_m \; mod \; 2^k$$

Current solutions:

- Small challenge space, $e, e' \in \{0, 1\}$; inefficient
- Non-standard soundness [CRFG20]: do not extract all the bits of m

Our solution:

- Modified Joye-Libert: $\mathrm{Enc}(m) = \mathrm{C} = y^m \, h^r \; mod \; N$
- It is an encryption, and at the same time, an integer commitment
- Standard soundness is based on the strong-RSA assumption

More Applications

- Three-party TLS handshake
  - for the Decentralized Oracle authenticating TLS data
- Naor-Yung CCA secure encryption
  - Two Joye-Libert Encs with zero-knowledge Range Proof
- Multiparty Computation (MPC)
  - SPDZ_2^k

Open Problems

- Efficient proof on the correctness of Joye-Libert modulus
  - $N = (2^k p' + 1)(2^k q' + 1)$
- More applications of our MtA
- Other candidate of MtA?

Summary

- Another choice of MtA from Joye-Libert
  - besides those based on Paillier, CL, and Oblivious Transfer
- Applications
  - Threshold ECDSA, Naor-Yung CCA, Three-party TLS, etc.
- Zero-knowledge range proof for Joye-Libert cryptosystem

Eprint: 2023/1312

Thanks

Q & A