TalonG and P-DualTS

Two-Round Lattice-Based Threshold Signatures

NIST Threshold Call Preview Talks #3

Speaker: Liming Gao — Institute of Information Engineering, CAS, China

Team: Octopus



Team members:
Man Ho Au, Robert H. Deng, **Liming Gao**, Bowen Jiang, Xianhui Lu, Tian Qiu, Guofeng Tang, Ruida Wang, Haiyang Xue, Guomin Yang, Chengru Zhang

Presented on 2026-10-06

Lattice-Based Two-Round Threshold Signatures

1 Why Threshold?

Distributed Trust



t parties → signature

No single party holds the signing key

2 Why Lattices?

Post-Quantum Security

RSA / ECC

↓ quantum threat

MLWE · MSIS · NTRU

post-quantum foundation

Candidate foundation for post-quantum cryptography

3 Why Two Rounds?

Low Interaction



Lower latency and coordination overhead

Our goal

Make two-round lattice-based threshold signatures practical across different deployment scales.

Two Schemes, Two Target Settings

The common goal is two-round post-quantum threshold signing.

TalonG — large committees

- up to 1,024 parties, general $2 \leq t \leq n$
- Focus: Low per-party communication
- Key idea: Semi-commitment + NTRU-trapdoor
- Trade-off: optimize communication at large n

P-DualTS — small committees

- e.g., $n \leq 16$, general $2 \leq t \leq n$
- Focus: Compact PK, signature & communication
- Key ideas: Parallel-dual simulation + replicated sharing
- Trade-off: optimize overall efficiency at small n

Lattice-Based Signature — Fiat–Shamir

A Fiat–Shamir Framework for Lattice-Based Signatures.

Key Generation

$$\mathbf{A} \leftarrow \boldsymbol{U}(\mathcal{R}_q^{m*n})$$
$$\boldsymbol{s} \leftarrow \boldsymbol{\chi_s}$$
$$\boldsymbol{b} = [\mathbf{I}\ \ \mathbf{A}]\boldsymbol{s}$$
$$\text{output } (\text{pk}, \text{sk}) = (\boldsymbol{b}, \boldsymbol{s})$$

Verification

$$\text{for } (\text{msg}, \text{sig} = (\boldsymbol{z}, c))$$
check
1. $\|\boldsymbol{z}\|$ short
2. $c = H([\mathbf{I}\ \ \mathbf{A}]\boldsymbol{z} - \boldsymbol{b}c||\text{msg})$

Signing

$$\boldsymbol{y} \leftarrow \boldsymbol{\chi_y}$$
$$\boldsymbol{w} = [\mathbf{I}\ \ \mathbf{A}]\boldsymbol{y}$$
$$c = H(\boldsymbol{w}||\text{msg})$$
$$\boldsymbol{z} = \boldsymbol{y} + \boldsymbol{s}c$$
$$\text{output sig} = (\boldsymbol{z}, c)$$

- $\boldsymbol{s}$: **secret key**
- $\mathbf{A}$: **public matrix**
- $\boldsymbol{y}$: **ephemeral randomness**
- c: **Fiat–Shamir challenge**
- $\boldsymbol{z}$: **response**

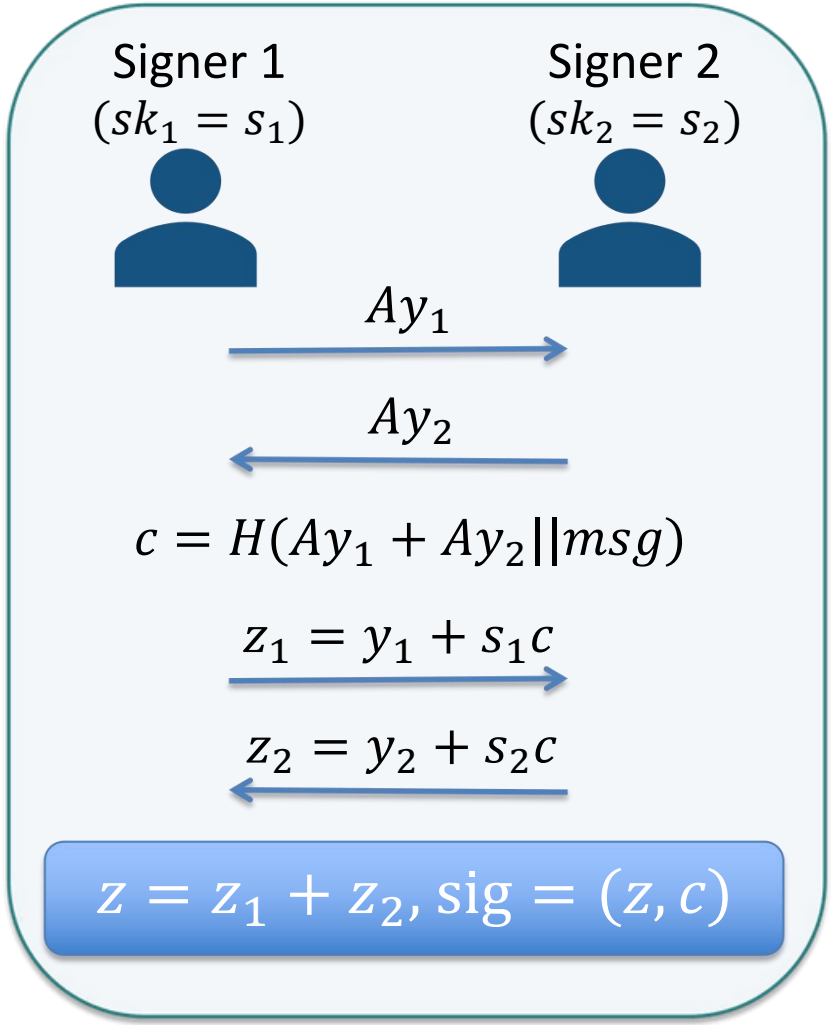
Security

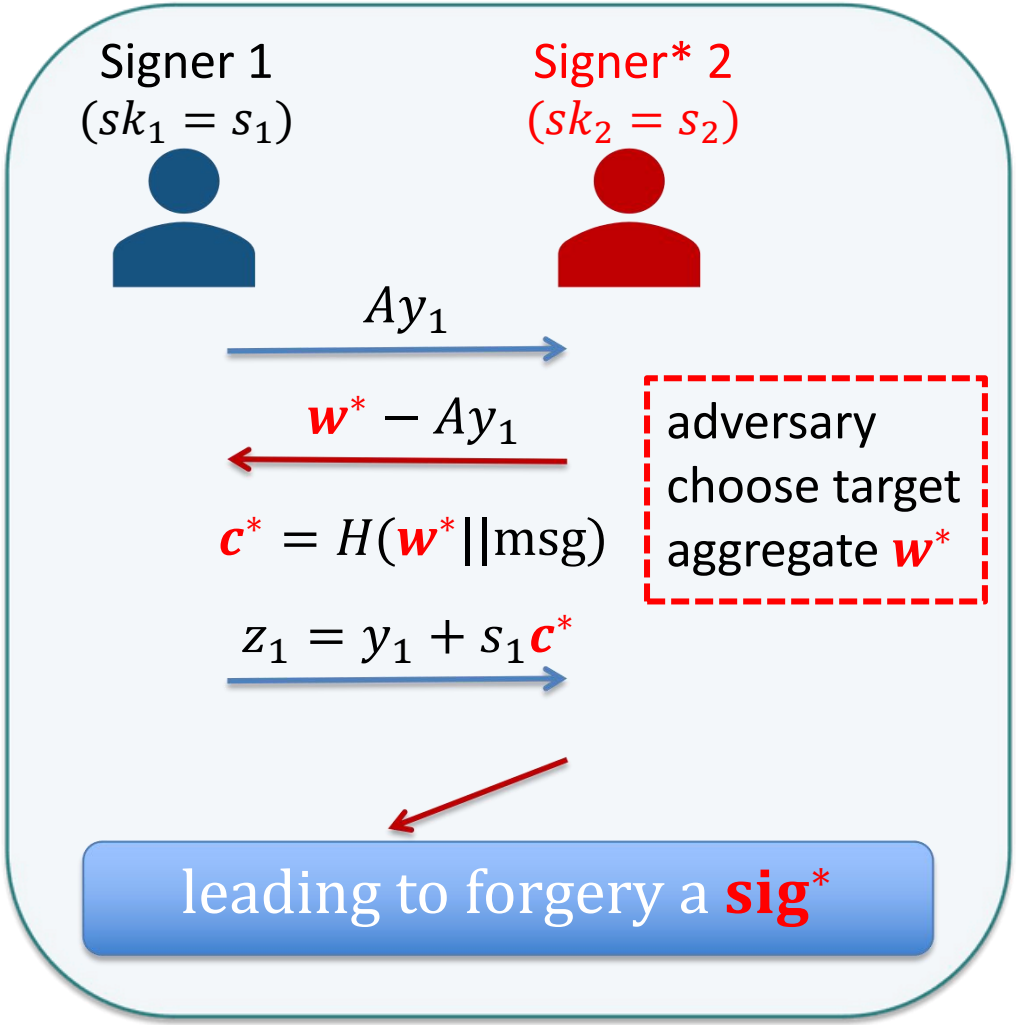
Security reduces to standard lattice problems(e.g., MLWE / MSIS)

Security typically proved in the Random Oracle Model

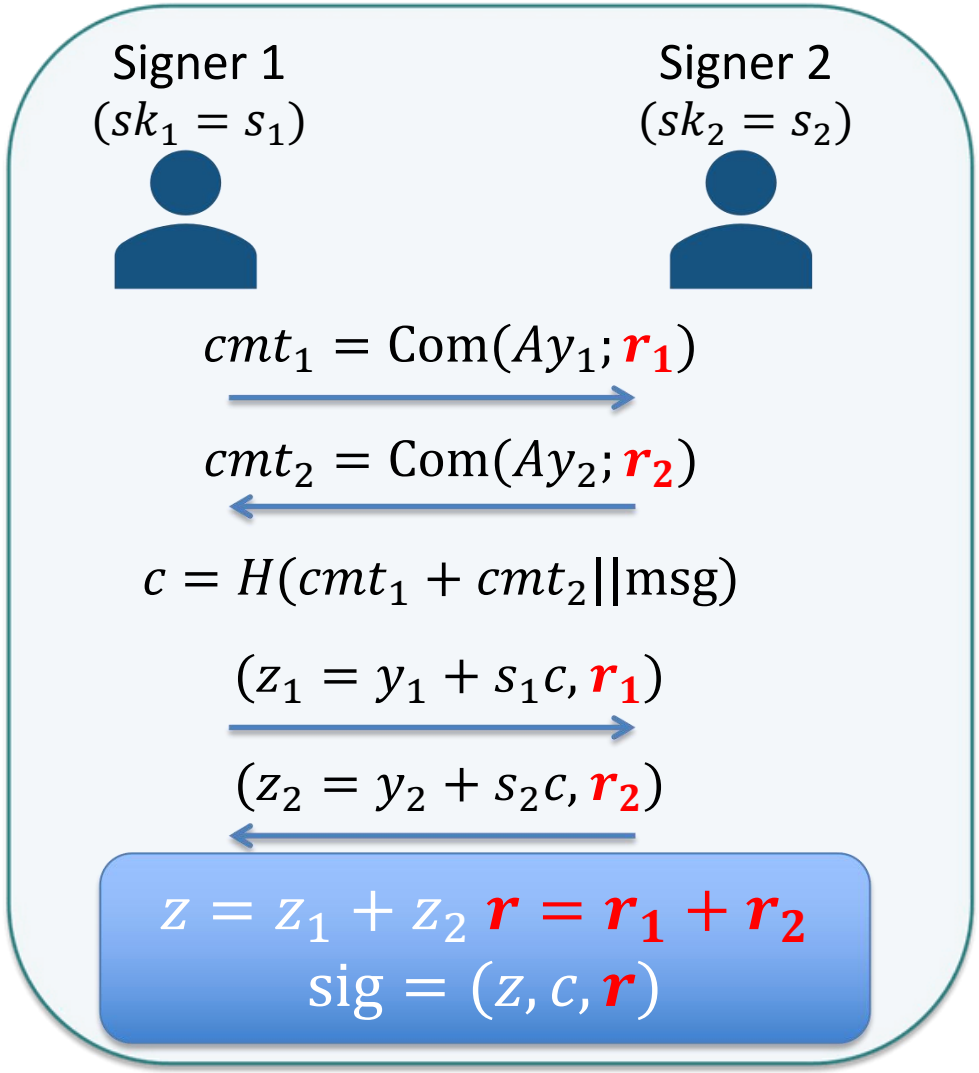
A Naive Two-Round Construction Is Not Secure

Two rounds are the known minimum for Fiat–Shamir-based lattice threshold signing.





Commit Before the Challenge



Naive

see w_1

↓

choose w_2

↓

control $w_1 + w_2$

Commit

see $\mathrm{Com}(w_1)$

↓

w_1 still hidden

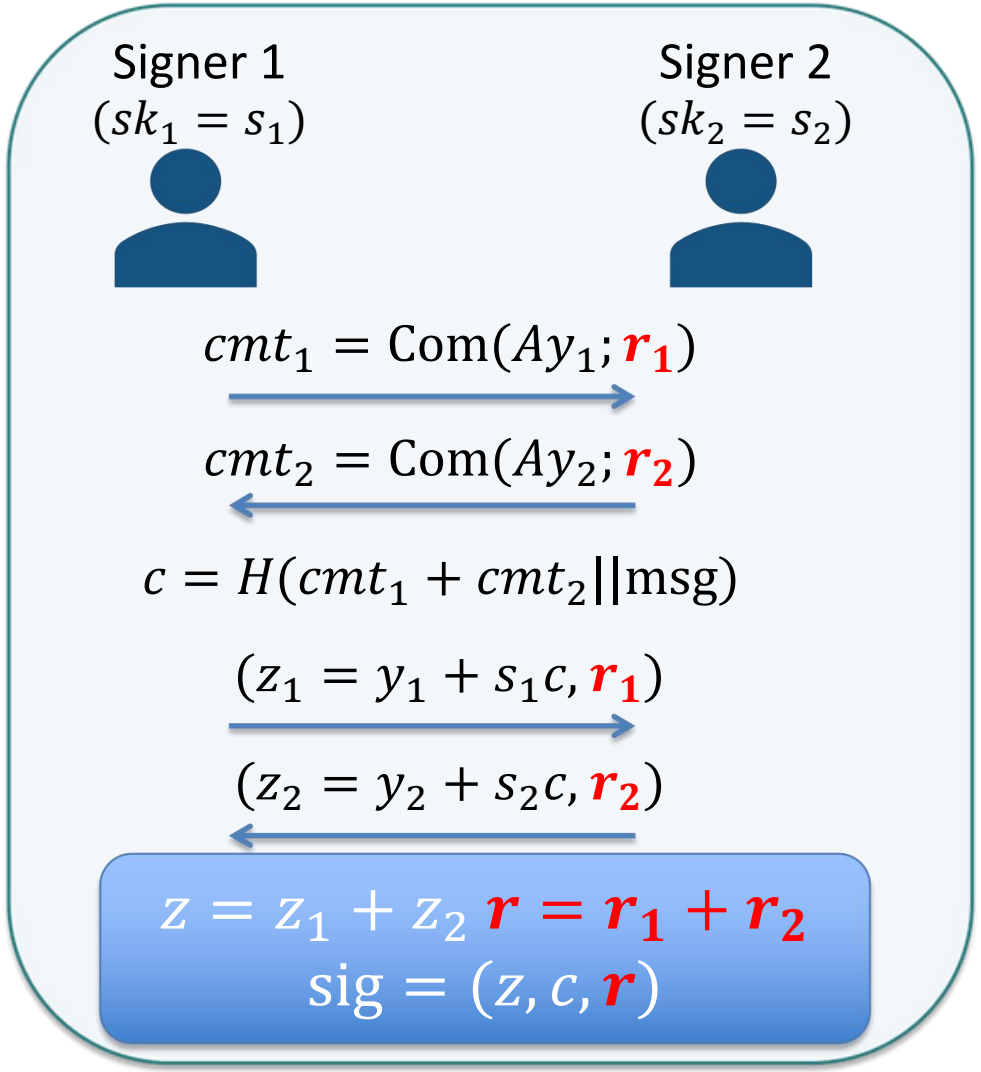
↓

choose $\mathrm{Com}(w_2)$

↓

CANNOT control $w_1 + w_2$

Two Rounds Come with a New Cost



Advantage

√ two rounds

√ less communication

(One commitment per signer in first round)

Limitation

☐ signature must carry the opening randomness

(Random tape becomes long due to **Trapdoor-setting**)

Our First Scheme: TalonG

eprint 2026/303

TalonG: We Do Not Need Full Binding

Key Observation:
Commitments DO NOT need binding

[DOTT21]

$$\mathrm{Com}(w;\boldsymbol{\rho}) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,l} \\ 0 & 1 & \cdots & a_{2,l} \end{bmatrix} \cdot \boldsymbol{\rho} + \begin{bmatrix} 0 \\ w \end{bmatrix}$$

Full Binding + MP Trapdoor

$$\boldsymbol{l = O(\log q)}$$

TalonG

$$\mathrm{Com}(w;\boldsymbol{\rho}) = [1 \quad h] \cdot \boldsymbol{\rho} + w$$

$$\boldsymbol{l = 2}$$

Weak-Binding + NTRU Trapdoor

TalonG: Two-Round Signing and Compression

The protocol combines semi-commitment, zero-sum masking, and modulus switching.

Round 1

Each signer broadcasts

$$\widetilde{w_i} = ay_i + e_i^{'} + r_{i,1} + hr_{i,2}$$

One polynomial commitment per signer.

Round 2

Signers compute challenge from aggregate commitment and message.

$$c = H(\lfloor \sum \widetilde{w_i} \rceil_p || \text{msg})$$

Response shares are protected by zero-sum masks.

$$z_i = y_i + \lambda_i s_i c + ZeroShare_i$$

Broadcasts $(z_i, r_{i,2})$

Compression

$$\sum \widetilde{w_i} = ay + hr + \underbrace{e' + r_1}_{\text{small-norm}}$$

Use modulus switching to drop the low-order information

Combine & Verify

$z = \sum z_i, r = \sum r_{i,2}$

$\Delta = \lfloor \sum \widetilde{w_i} \rceil_p - \lfloor az - bc + hr \rceil_p$

sig $= (z, r, c, \Delta)$ satisfying

$$c = H(\lfloor az - bc + hr \rceil_p + \Delta || \text{msg})$$

TalonG: performance at t = 1024

128-bit parameter set.

Communication

26.9 KB / signer

Public key

2.0 KB

Signature

17.7 KB

Signing time

140.7 ms
(Intel Core i7-13700 processor 2.10 GHz)

Security

Selective, TS-UF-0
128-bit classical security,
Hint-RLWE, RSIS, NTRU

Trade-offs

Large signatures; online rounds; not adaptive

Our Second Scheme: P-DualTS

P-DualTS: A Different Target

Small committees, standard assumptions, and compact signatures.

Deployment

Small-scale deployments

Practical range: $n \leq 16$

NIST profiles: Two, Three, Small, and Medium

Security

Fiat–Shamir with Aborts paradigm

Standard MLWE + MSIS

Selective/static corruption up to $t-1$

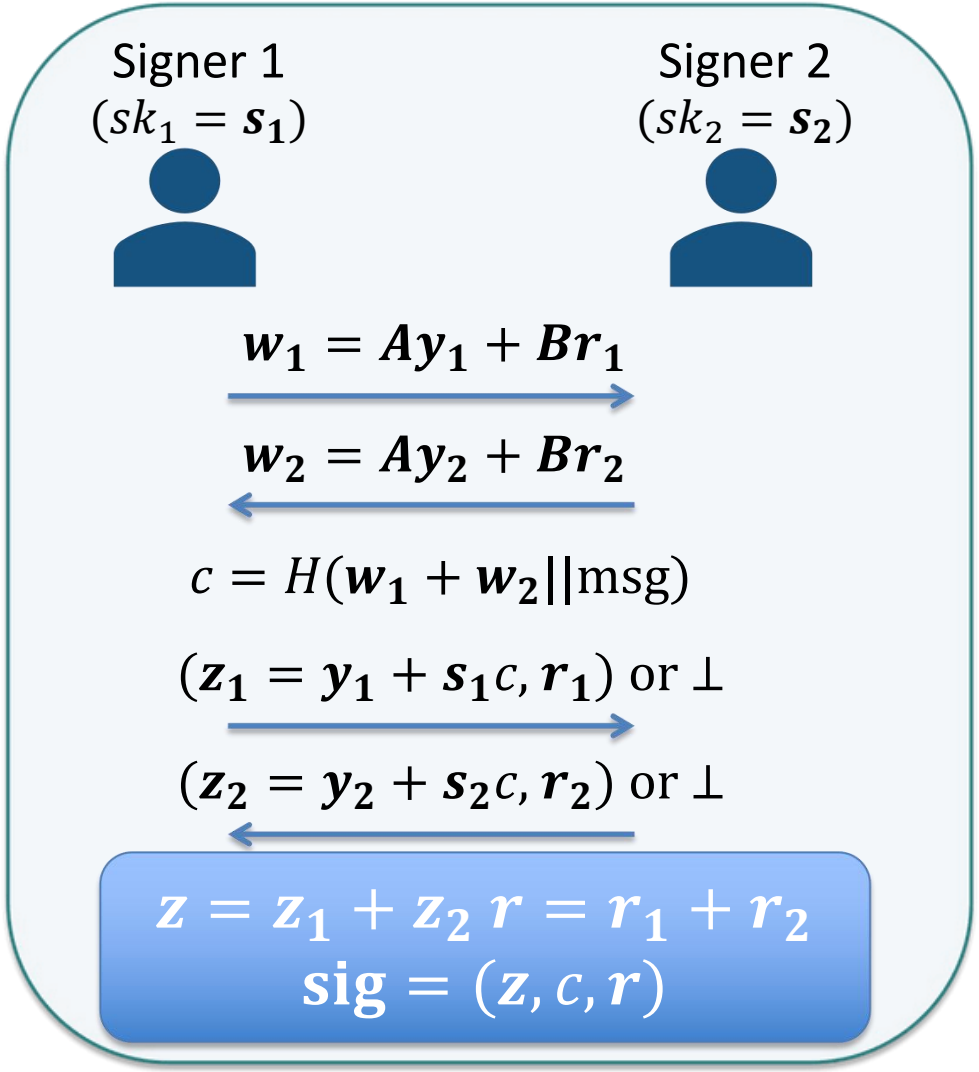
Goal

Two rounds

Compact public key and signature

Low communication

Our Base work: DualMS [Chen23]



Key Technique

Rejection Sampling + **Dual Simulation**

Dual Simulation

require $\mathbf{A}s_i = \mathbf{B}u_i$

Real transcript

$w_i = \mathbf{A}y_i + \mathbf{B}p_i$

$z_i = y_i + s_i c$

$r_i = p_i$

$\approx$

Simulated transcript

$w_i = \mathbf{A}y_i + \mathbf{B}p_i$

$z_i = y_i$

$r_i = p_i + u_i c$

Thresholdizing DualMS: Two Key Techniques

1 Parallel-Dual Simulation

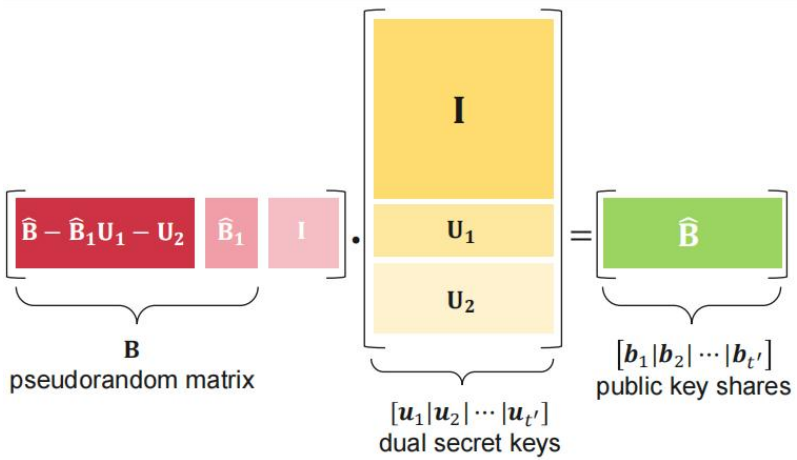
DualMS: one simulated party

for one i, simulated
$(\mathbf{B}, \boldsymbol{u_i})$ s.t. $\mathbf{A}\boldsymbol{s_i} = \mathbf{B}\boldsymbol{u_i}$

↓ Threshold setting

DualTS: multiple simulated parties

for multiple i, simulated
$(\mathbf{B}, \{\boldsymbol{u_i}\}_i)$ s.t. $\mathbf{A}\boldsymbol{s_i} = \mathbf{B}\boldsymbol{u_i}$ for any i



2 Replicated Secret Sharing

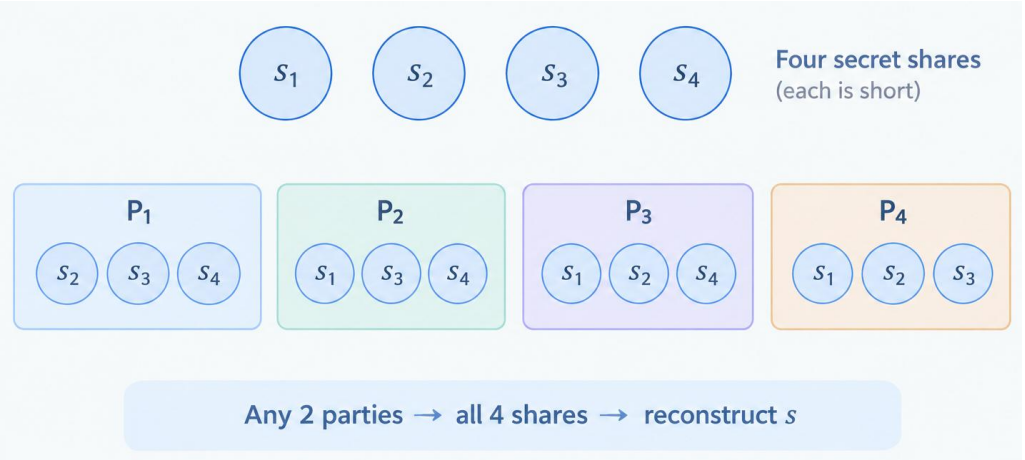
DualMS: n-out-of-n secret share

$\boldsymbol{s} = \boldsymbol{\Sigma_i s_i}$ and any $\boldsymbol{s_i}$ short

↓ Threshold setting

DualTS: t-out-of-n secret share

$\boldsymbol{s_i} = \boldsymbol{\Sigma_{J \in set_i} s_J}$ and any $\boldsymbol{s_J}$ short



P-DualTS: Two-Round Signing

The protocol combines replicated secret share, rejection sampling, and modulus switching.

Round 1

Each signer broadcasts

$$\boldsymbol{w_i} = \mathbf{A}\boldsymbol{y_{i,1}} + \boldsymbol{y_{i,2}} + \mathbf{B}\boldsymbol{r_{i,1}} + \boldsymbol{r_{i,2}}$$

One polynomial vector commitment per signer.

Round 2

Signers compute challenge from aggregate commitment and message.

$$c = H(\lfloor \sum \boldsymbol{w_i} \rceil_{\boldsymbol{p}} || \text{msg})$$

Output response shares

$$(\boldsymbol{z_{i,1}} = \boldsymbol{y_{i,1}} + \boldsymbol{s_i} c, \boldsymbol{r_{i,1}})$$

with probablity $\frac{D_\sigma(\boldsymbol{z_i})}{M * D_\sigma(\boldsymbol{y_i})}$

Otherwise, output $\perp$

Compression

$$\sum \boldsymbol{w_i} = \mathbf{A}\boldsymbol{y_1} + \mathbf{B}\boldsymbol{r_1} + \underbrace{\boldsymbol{y_2} + \boldsymbol{r_2}}_{\text{small-norm}}$$

Use modulus switching to drop the low-order information

Combine & Verify

$$\boldsymbol{z} = \sum \boldsymbol{z_{i,1}}, \boldsymbol{r} = \sum \boldsymbol{r_{i,1}}$$
$$\Delta = \lfloor \sum \boldsymbol{w_i} \rceil_{\boldsymbol{p}} - \lfloor \mathbf{A}\boldsymbol{z} - \boldsymbol{b}c + \mathbf{B}\boldsymbol{r} \rceil_{\boldsymbol{p}}$$

sig $= (\boldsymbol{z}, \boldsymbol{r}, c, \Delta)$ satisfying

$$c = H(\lfloor \mathbf{A}\boldsymbol{z} - \boldsymbol{b}c + \mathbf{B}\boldsymbol{r} \rceil_p + \Delta || \text{msg})$$

P-DualTS: performance at (t,n)=(5,8)

128-bit parameter set.

Communication

10.72 KB / signer

Public key

1.72 KB

Signature

7.67 KB

Signing time

6.960 ms
(Including rejection restarts, accept rate≈ $1/4$)

Security

Selective, TS-UF-0
128-bit classical security, MLWE, MSIS

Trade-offs

Only small committees; online rounds; not adaptive

P-DualTS vs TalonG

TalonG **Fiat–Shamir-without-Abort**

Raccoon-style approach

- No rejection sampling
- NTRU-type trapdoor semi-commitment
- Designed for large-scale threshold settings
- Focus: low per-party communication

P-DualTS **Fiat–Shamir-with-Abort**

- **ML-DSA-style** approach
- Rejection sampling
- Replicated secret sharing+
  Parallel-dual simulation
- Designed for small-party threshold settings
- Focus: compact overall size

Common Limitation: No Functional Interchangeability

Neither is Functionally Interchangeable with the underlying signature scheme.

Security, implementation, and submission status

What will be delivered in the NIST submission.

Security

Both:

- selective EUF-CMA

  Static ≤ t−1 corruptions
- ROM
- 128-bit classical security

TalonG: RLWE/RSIS/NTRU

P-DualTS: MLWE/MSIS

Implementation

- Go prototype available
- Modular framework
- Correctness +

  benchmarks
- Local execution

Final submission

Update code to NIST requirements

Release optimized implementations

Provide detailed benchmarks and comparisons

Apache License 2.0

Thank you!

Questions?

Team: Octopus

