MPTS 2023: NIST Workshop on Multi-Party Threshold Schemes 2023

26 September 2023

coinbase

Diversity and Tradeoffs in MPC Standardization

Yehuda Lindell

Head of Cryptography at Coinbase

MPC Settings and Parameters

- **Adversary:** semi-honest, covert, malicious, rational
- **Threshold:** dishonest majority, honest majority
  - For honest majority: guaranteed output delivery, security with abort
- **Corruption:** static, adaptive (with/out erasures), proactive-static, proactive-adaptive
- **Security model:** game-based, simulation
  - Game based: which definition?
  - Simulation: empty signing functionality, signature-generation functionality, other functionalities (e.g., DKLs18)
  - Does it matter? Related keys, general composition,…

MPC Settings and Parameters

- **Composition:** UC (no rewinding), stand-alone
- **Proof model:** plain, CRS, ROM, GGM-Shoup, GGM-Maurer, AGM
- **Assumptions:** minimal (signing scheme itself), almost minimal (DLOG/DDH), standard (Paillier, RSA, lattice), non-standard (who determines – DDH?), interactive/non-falsifiable
- **Post quantum security or not**
- **Efficiency optimization:** low rounds (e.g., 2), low bandwidth (for mobile upload), higher bandwidth/fast computation
- **Other consideration:** protocol simplicity (ease of implementation), proof simplicity (ease of verifying security), protocol legacy (new/old, reviewed/not reviewed)

MPC Settings and Parameters

#choices

- 3 **Adversary:** semi-honest, covert, malicious, rational
- 3 **Threshold:** dishonest majority, honest majority
    - For honest majority: guaranteed output delivery, security with abort
- 4 **Corruption:** static, adaptive (with/out erasures), proactive-static, proactive-adaptive
- 4 **Security model:** game-based, simulation
    - Game based: which definition?
    - Simulation: empty signing functionality, signature-generation functionality, other functionalities
- 2 **Composition:** UC (no rewinding), stand-alone
- 6 **Proof model:** plain, CRS, ROM, GGM-Shoup, GGM-Maurer, AGM
- 4 **Assumptions:** minimal (signing scheme itself), almost minimal (DLOG/DDH), standard (Paillier, RSA, lattice), non-standard, interactive/non-falsifiable
- 2 **Post quantum security or not**
- 3 **Efficiency optimization:** low rounds (e.g., 2), low bandwidth (for mobile upload), higher bandwidth/fast computation
- 8 **Other consideration:** protocol simplicity, proof simplicity, protocol legacy

Total = 3 x 3 x 4 x 4 x 2 x 6 x 4 x 2 x 3 x 8 = 331,776

MPC Settings and Parameters – “Likely”

Proposal: limit possible choices to the likely ones – but the choice below will force UC (no plain Fiat-Shamir), and will not allow GGM (and will also not consider a plain or CRS model protocol an advantage) ☹

- 2 **Adversary:** semi-honest, covert, malicious, rational
- 2 **Threshold:** dishonest majority, honest majority
  - For honest majority: guaranteed output delivery, security with abort
- 2 **Corruption:** static, adaptive (with/out erasures), proactive-static, proactive-a
- 4 **Security model:** game-based, simulation
  - Game based: which definition?
  - Simulation: empty signing functionality, signature-generation functionality, other functiona
- 1 **Composition:** UC/concurrent composition (no rewinding), stand-alone
- 1 **Proof model:** plain, CRS, ROM, GGM-Shoup, GGM-Maurer, AGM
- 4 **Assumptions:** minimal (signing scheme itself), almost minimal (DLOG/DDH), standard (Paillier, RSA, lattice), non-standard, interactive/non-falsifiable
- 2 **Post quantum security or not**
- 3 **Efficiency optimization:** low rounds, low bandwidth, higher bandwidth/fast computation
- 4 **Other consideration:** protocol simplicity, proof simplicity, protocol legacy (new/old)

Yes, I know I’m over-counting (there is no sense for a complex protocol and complex proof combination)

Total = 2 x 2 x 2 x 4 x 1 x 1 x 4 x 2 x 3 x 4 = 3,072

The Concern

- **We standardize protocols and…**
  - An organization with a lower risk appetite (e.g., protecting billions with each key) and without a need for optimal performance (e.g., custody use case versus wallet use case) will be questioned by customers about why they aren't using the standard?
- **We standardize protocols and…**
  - My business use case needs to optimize something else (I'm serving weak mobiles in developing countries) but there's no standard protocol for that
- **We standardize protocols and…**
  - A year later, we have a much better protocol that isn't standard
- **We standardize protocols and…**
  - A year later, we find a gap in the proof of a standardized protocol which can be fixed in the GGM or under much stronger assumptions (or with a minor change to the protocol that increases the cost)

Mitigations

- **Standardize tools and not (just) protocols**
  - VSS, Sigma protocols and NIZK transformation, garbled circuits, oblivious transfer and extension, basic primitives (commitments, coin tossing, Paillier, etc.), common zero-knowledge proofs (e.g., range proofs)
- **Standardize methodology and not (just) protocols**
  - If you do X,Y,Z in your development, then also OK
  - That’s not practical, but NIST can have a committee to approve it
- **Provide a relaxed interpretation of the standard**
  - If I add **standardized** ZK proofs to a protocol, it’s still considered standard
  - If key generation is done differently (?) then it’s still considered standard

Mitigations

- **Encourage modular submissions with flexibility**
    - Protocol with two-round and three-round versions and with and without ZK versions
        - If you run two-round then OMDL
        - If you run three-round then empty signature functionality
        - If you run three-round with ZK then signature-generation functionality
    - Protocol with OT-based and Paillier-based multiplication subprotocol
        - One achieves lower computation / higher bandwidth, and the other the reverse
    - Garbled circuits that can be instantiated with *any encryption scheme* meeting conditions A,B,C

Mitigations

- **Look far back at well-established and simple constructions (don’t just ask for new submissions)**
  - Threshold encryption for RSA and for EC (TDH2) by Shoup
  - Oblivious transfer of PVW
  - Feldman VSS
- **To the extent possible, do not run a one-shot process but have an ongoing standardization initiative**
  - Each year, choose something to standardize
    - Don’t try to do everything at once
    - Start with basic primitives and simple protocols, and build up
  - Allow adding new standards to prior year’s standards, if the gain is considerable

Thank You