Threshcon

Multi-Party Threshold Ascon

NIST Threshold Call - Preview Talk

Categories N3.2, N3.3

Peter Schwarz, Aysajan Abidin - COSIC, KU Leuven

TCPT2 - July 8, 2026

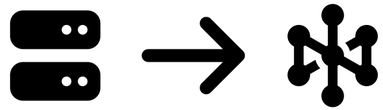
A little dry? I hope not

Contents

1. Intro & Motivation
2. Ascon (in MPC)
3. Package submission
4. Performance & Conclusion

Why Care about Multi-Party Threshold Cryptography?

Generic to Concrete

what do we want to achieve? 

Why Care about Multi-Party Threshold Cryptography?

Generic to Concrete

what do we want to achieve?

- **Distribution** of **trust** in **cryptographic primitive** execution

Why Care about Multi-Party Threshold Cryptography?

Generic to Concrete

what do we want to achieve?

- **Distribution** of **trust** in **cryptographic primitive** execution
- Singular **entity splits** compromise/availability **risk** to sub-entities

Why Care about Multi-Party Threshold Cryptography?

Generic to Concrete

what do we want to achieve?

- **Distribution** of **trust** in **cryptographic primitive** execution
- Singular **entity splits** compromise/availability **risk** to sub-entities
- **Oblivious decryption**: yield plaintext w/o full key

Why Care about (MPT) Ascon?

Why Care about (MPT) Ascon?

- Ascon: NIST **lightweight standard** (incl. AEAD, Hash, [C]XOF)
  - Major use-cases: Lightweight device → MPC committee

Why Care about (MPT) Ascon?

- Ascon: NIST **lightweight standard** (incl. AEAD, Hash, [C]XOF)
  - Major use-cases: Lightweight device → MPC committee
- **AEAD**: Sensor/Smartcard encrypts data via Ascon, requiring obl. decr.
  -
  - Logs, Medical, Data for analysis/re-mixing

Why Care about (MPT) Ascon?

- Ascon: NIST **lightweight standard** (incl. AEAD, Hash, [C]XOF)
  - Major use-cases: Lightweight device → MPC committee
- **AEAD**: Sensor/Smartcard encrypts data via Ascon, requiring obl. decr.
  -
  - Logs, Medical, Data for analysis/re-mixing
- **Hash**: **secret/split** message and **verify** w/ **lightweight** device aswell

Why Care about (MPT) Ascon?

- Ascon: NIST **lightweight standard** (incl. AEAD, Hash, [C]XOF)
  - Major use-cases: Lightweight device → MPC committee
- **AEAD**: Sensor/Smartcard encrypts data via Ascon, requiring obl. decr.
  -
  - Logs, Medical, Data for analysis/re-mixing
- **Hash**: **secret/split** message and **verify** w/ **lightweight** device aswell
- **[C]XOF**: **K**ey **D**erivation **F**unction, **P**seudo-**R**andom **N**umber **G**enerator

What Happened, What's New?

What Happened, What's New?

- **WPES'25 / MPTS'26 :** 1st Ascon in MPC optimization study
  - Theoretical circuit benchmarks and MP-SPDZ baselines

What Happened, What's New?

- **WPES'25 / MPTS'26 :** 1st Ascon in MPC optimization study
  - Theoretical circuit benchmarks and MP-SPDZ baselines
- **Since then → Threshcon package plan:**
  - Formalized as **named crypto-systems** (Threshcon-AEAD128, Hash256, [C]XOF128)
  - Concrete **I/O secret-sharing interfaces**
  - **Rust reference implementation** in active development
  - Full **security characterization**

Scope

Element	**Threshcon Plan**
Class / Categories	**N3.2** (Ascon-AEAD Enc/Dec), **N3.3** (Hash, XOF, CXOF)
Reference primitive	**Ascon** (NIST SP 800-232)
Interoperability	**Strictly Interchangable**

Ascon Family Spec’d:

Ascon Family Spec’d:

- Authenticated Encryption with Associated Data (**AEAD**) - N3.2:
  - Ascon-AEAD128

Ascon Family Spec'd:

- Authenticated Encryption with Associated Data (**AEAD**) - N3.2:
  - Ascon-AEAD128
- **Hash**/eXtendable-Output-Functions (**XOF**) - N3.3:
  - Ascon-Hash256
  - Ascon-XOF128
  - Ascon-CXOF128

~~Ascon~~ Threshcon Family Spec'd:

- Authenticated Encryption with Associated Data (**AEAD**) - N3.2:
  - Ascon-AEAD128 $\rightarrow$ **`Threshcon-AEAD128`**
- **Hash**/eXtendable-Output-Functions (**XOF**) - N3.3:
  - Ascon-Hash256 $\rightarrow$ **`Threshcon-Hash256`**
  - Ascon-XOF128 $\rightarrow$ **`Threshcon-XOF128`**
  - Ascon-CXOF128 $\rightarrow$ **`Threshcon-CXOF128`**
- **Unifying technique:** Ascon permutation eval via RMFE-packing over optimized Bool. circuit

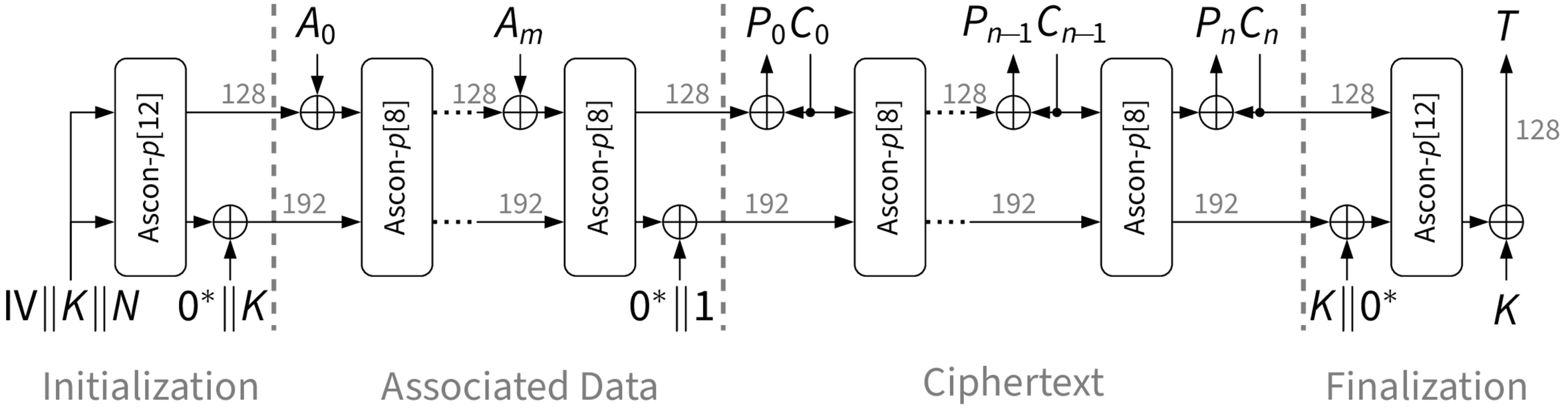
System & Threat Model

Aspect	**Threshcon Model**
Adversary behavior	**Active-malicious**
Corruption threshold	$t < n/3$ (honest supermajority)
Security type	**Information-theoretic**
Setup assumption	**Preprocessing model**
Network model	Synchronous; complete network of secure, authenticated channels

Background - Ascon Structure

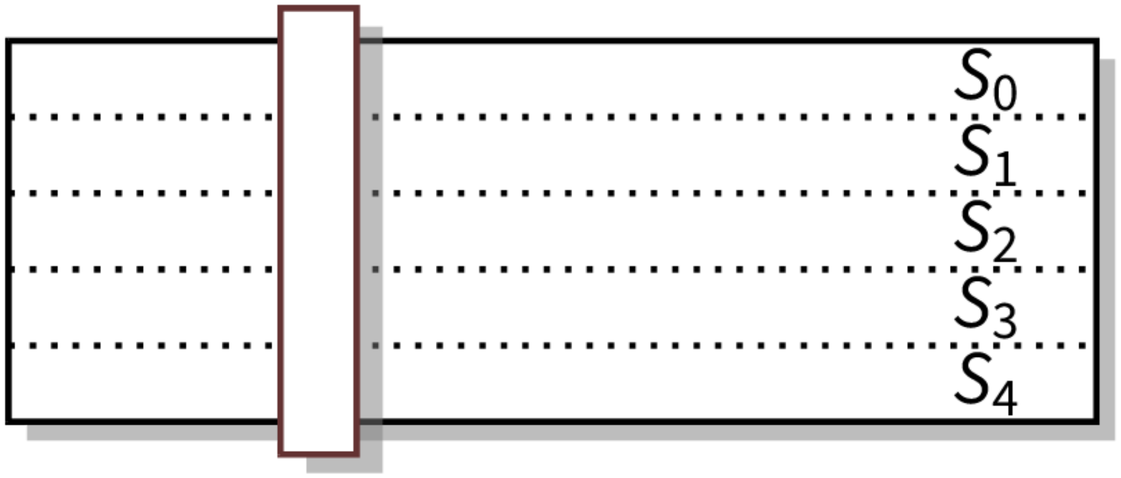
Built around Permutation: $p = p_L \circ p_S \circ p_C$

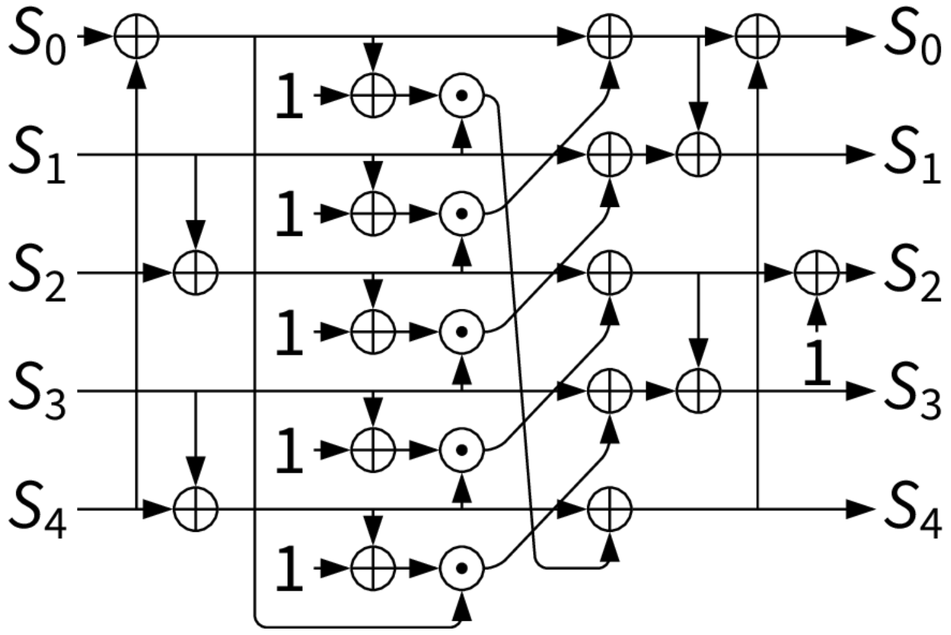
Ascon-AEAD128 authenticated decryption mode:



Background - Ascon Structure

Substitution layer, state structure and function:





Linear layer:

$$\Sigma_0(S_0) = S_0 \oplus (S_0 \ggg 19) \oplus (S_0 \ggg 28)$$

....

Background - Reverse-Multiplication Friendly Embeddings (RMFEs)

A Set of ϕ, ψ mappings called a $(k, m)_q$ RMFE "imitates" a homomporhism between vectors and field elements.

Core Approach: RMFE-Packed Ascon (Recap)

Core Approach: RMFE-Packed Ascon (Recap)

- **Bottleneck:** Ascon's 5-bit S-box requires 320 AND gates per permutation round
  - Each AND gate = communication round in Boolean MPC → bandwidth wall

Core Approach: RMFE-Packed Ascon (Recap)

- **Bottleneck:** Ascon's 5-bit S-box requires 320 AND gates per permutation round
  - Each AND gate = communication round in Boolean MPC → bandwidth wall
- **RMFE packing:** Embed k Boolean bits into $\mathbb{F}_{2^m}$ field elements
  - With $(3, 5)_2$ RMFE: **3 packed S-boxes per field multiplication**
  - Significantly reduces non-linear layer operations

Core Approach: RMFE-Packed Ascon (Recap)

- **Bottleneck:** Ascon's 5-bit S-box requires 320 AND gates per permutation round
  - Each AND gate = communication round in Boolean MPC → bandwidth wall
- **RMFE packing:** Embed k Boolean bits into $\mathbb{F}_{2^m}$ field elements
  - With $(3, 5)_2$ RMFE: **3 packed S-boxes per field multiplication**
  - Significantly reduces non-linear layer operations
- **Linear layer:** Blinded unpack → rotate → repack

Core Approach: RMFE-Packed Ascon (Recap)

- **Bottleneck:** Ascon's 5-bit S-box requires 320 AND gates per permutation round
  - Each AND gate = communication round in Boolean MPC → bandwidth wall
- **RMFE packing:** Embed k Boolean bits into $\mathbb{F}_{2^m}$ field elements
  - With $(3, 5)_2$ RMFE: **3 packed S-boxes per field multiplication**
  - Significantly reduces non-linear layer operations
- **Linear layer:** Blinded unpack → rotate → repack
- **Result:** $\sim$38% communication reduction for $n{=}13$; up to 50% for larger n

Core Approach: RMFE-Packed Ascon (Recap)

- **Bottleneck:** Ascon's 5-bit S-box requires 320 AND gates per permutation round
  - Each AND gate = communication round in Boolean MPC → bandwidth wall
- **RMFE packing:** Embed k Boolean bits into $\mathbb{F}_{2^m}$ field elements
  - With $(3, 5)_2$ RMFE: **3 packed S-boxes per field multiplication**
  - Significantly reduces non-linear layer operations
- **Linear layer:** Blinded unpack → rotate → repack
- **Result:** $\sim$38% communication reduction for $n{=}13$; up to 50% for larger n
- *Full details: WPES'25 Paper (**ia.cr/2025/1538**)*

Specification Strategy

Component	**Approach**
Security definition style	Ideal functionality for Ascon permutation; game-based for composite modes
Modularity	Layered architecture (RMFE → MPC → Ascon → Modes)
Parameterization	Configurable $(k, m)_2$ RMFE parameters; threshold profile (t, n)

I/O Secret-Sharing Modes

Per NIST Call Requirements (§10.3, Req. 10.9)

AEAD Mode	**Key K**	**Nonce N**	**Assoc. Data A**	**Plaintext PT**	**Ciphertext CT**	**Tag T**
Encryption	→ ■	→ □	→ □	→ □	← □	← □
Decryption	→ ■	→ □	→ □	← □	→ □	→ □

Hash Mode	**Message M**	**Length L**	**Customization Z**	**Output H**
Hash	→ ■	-	-	← □
XOF	→ ■	→ □	-	← □
CXOF	→ ■	→ □	→ □	← □

Legend: ■ = shall be secret-shared - □ = may be secret-shared - → input - ← output

Threshcon-AEAD128

Function Interface (Category N3.2)

Core operations:

- $\texttt{Threshcon-AEAD128.enc}([K], N, A, PT) \rightarrow (CT, T)$ or $([CT], [T])$
- $\texttt{Threshcon-AEAD128.dec}([K], N, A, CT, T) \rightarrow PT$ or $[PT]$

Sharing functions:

- $\texttt{ShareKey}(K, P_i) \rightarrow [K]$; $\texttt{ShareNonce}(N, P_i) \rightarrow [N]$; $\texttt{ShareAssoc}(A, P_i) \rightarrow [A]$; $\texttt{SharePT}(PT, P_i) \rightarrow [PT]$; $\texttt{ShareCT}(CT, P_i) \rightarrow [C]$; $\texttt{ShareTag}(T, P_i) \rightarrow [T]$
- **Design principle:** Minimal required signature + flexibility

Threshcon-Hash256 & [C]XOF128

Function Interface (Category N3.3)

Core operations:

- $\texttt{Threshcon-Hash256}([M]) \to H$ or $[H]$
- $\texttt{Threshcon-XOF128}([M], L) \to H$ or $[H]$
- $\texttt{Threshcon-CXOF128}([M], L, Z) \to H$ or $[H]$

Sharing functions:

- $\texttt{ShareMsg}(M, P_i) \to [M]$; $\texttt{ShareLen}(L, P_i) \to [L]$; $\texttt{ShareCust}(Z, P_i) \to [Z]$
- Hash/XOF modes keyless, but **message is SSI** per Req. 10.9
- CXOF's customization parameter Z, length may be secret-shared (domain separation secrecy)

Protocol Modules

Layered Architecture (Spec Organization)

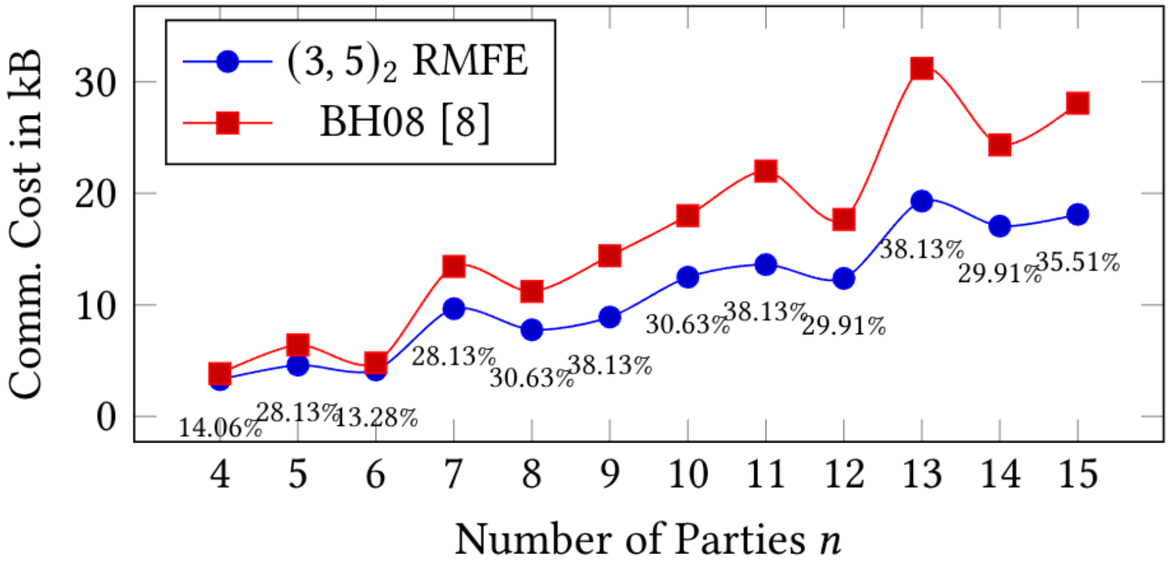
Layer	Component	Basis
Foundation	RMFE encoding/decoding	CCXY18 (Cascudo et al., CRYPTO'18)
MPC Framework	Arithmetic MPC with player elimination	BH08 (Beerliová-Trubíniová & Hirt, TCC'08)
Ascon-specific	Permutation, Packed linear layer (bit rotations)	SPAP25 (Schwarz et al., WPES'25)
Mode adaptation	AEAD / Hash / XOF / CXOF circuits	New (this package)

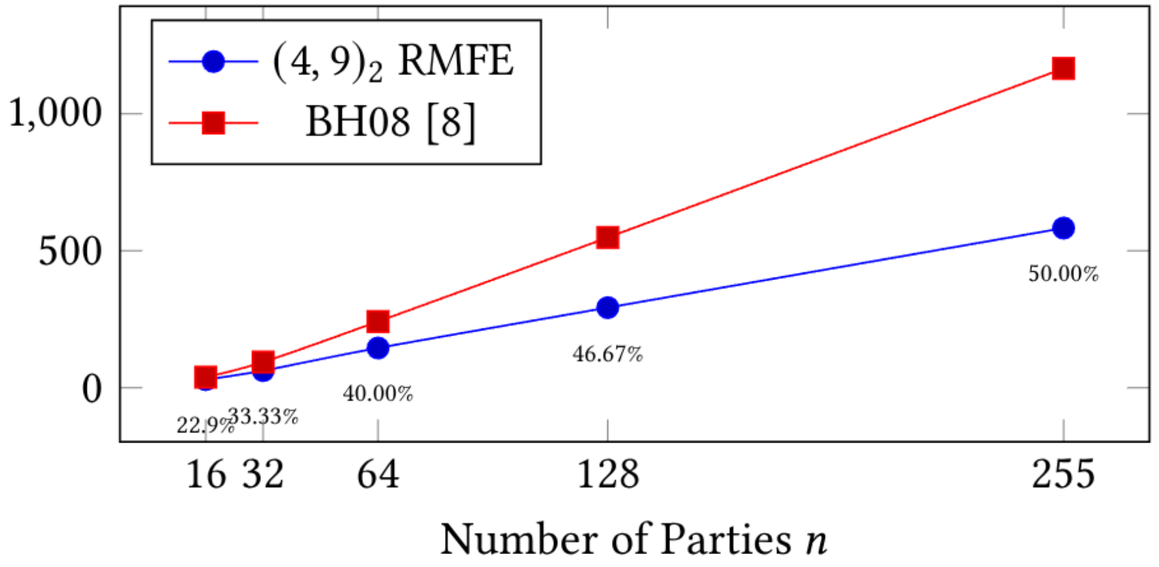
Sub-protocols: `ReconsPriv`, `ReconsPubl`, `Input`, `XOR`, `AND`, `MultiRightRotate`

Reference Implementation Status

Aspect	Status
Language	**Rust** (cargo build system)
Repository	**Active development**; public release planned with package submission
License	EUPL v1.2
Architecture	**Crypto** engine (Ascon + CCXY18 + BH08) **separated** from networking (QUIC/TLS)
Key dependencies	`tokio` (async runtime), `quinn` (QUIC transport), `rand`, `criterion`
Testing	KATs against Ascon test vectors; malformed-share injection; network fault simulation

Performance: Theoretical Results





- **Baseline:** Boolean BH08 (per-bit Shamir sharing)
- **Gain source:** SIMD amortization - ~~individual AND gates~~ field multiplications
- **Full empirical benchmarks:** Work in progress (see Evaluation Plan)

Preprocessing Communication Costs

Comparing RMFE packing vs. naive Boolean evaluation (BH08) Parameters: $n = 13, t = 4, (3, 5)_2$ *RMFE*

Concrete Communication Sent (Bits):

Workload	Baseline (BH08)	Threshcon (RMFE)	Reduction
1 Permutation (p)	583,680 bits	358,140 bits	**~38%**
1 Block (p^8)	~4.6 MB	~2.7 MB	**~41%**
1 KiB Data ($p^{8 \cdot 64}$)	~298.8 MB	~173.6 MB	**~41%**

- **Why it matters:** RMFE packing amortizes randomness generation for p_S, p_L layers.

Experimental Evaluation Plan

Metric	Target Platform
Online communication (bits/round)	16-core / 64 GB (NIST baseline)
Latency (ms/permutation)	Local loopback, LAN/WAN sim.
Throughput (bits/sec)	Local loopback, LAN/WAN sim.
Scalability (n)	$n \in \{4, 7, 13, 31, 63, 127, 255\}$

- Comparison Baseline: BH08-MP-SPDZ baseline, theoretical eval, pot. N3 submissions
- **Includes:** All-honest configuration + corrupted-party configurations per Call Req.
- **Amortization study:** Multi-block throughput for AEAD (128-bit rate) vs. Hash (64-bit rate)

IP / Funding

IP / Funding

- **Known patents (team members):** None ✅

IP / Funding

- **Known patents (team members):** None ✅
- **Known patents (third parties):** None known to the team ✅

IP / Funding

- **Known patents (team members):** None ✅
- **Known patents (third parties):** None known to the team ✅
- **Funding:** CyberSecurity Research Flanders, FWO SBO MOZAIK, EU dAIEDGE

Roadmap

Roadmap

- **Done ✅:** Theoretical analysis, MP-SPDZ baselines; WPES'25 publication; core engine

Roadmap

- **Done ✅:** Theoretical analysis, MP-SPDZ baselines; WPES'25 publication; core engine
- **In progress ⏳:** Full AEAD/Hash/XOF mode circuits; QUIC networking layer; KAT suite; protocol optimizations

Roadmap

- **Done ✅:** Theoretical analysis, MP-SPDZ baselines; WPES'25 publication; core engine
- **In progress ⏳:** Full AEAD/Hash/XOF mode circuits; QUIC networking layer; KAT suite; protocol optimizations
- **Outstanding 🚧:** Empirical WAN benchmarks; security proof writeup; spec finalization

Takeaways

1. **Threshcon:** bandwidth-optimized thresholdization of Ascon family (AEAD, Hash, [C]XOF), strictly interchangeable
2. **Info-theoretic active-adaptive security** at $t < n/3$, no computational assumptions, achieved via RMFE-packed Boolean evaluation
3. **Rust-based reference implementation** until October '26

Writeup: Available via NIST-MPTC

Contact: threshcon@esat.kuleuven.be

Paper: ia.cr/2025/1538

Questions?