# On Zero-Knowledge Proofs, MPC, and Symmetric Crytography

Christian Rechberger

Jan 9, 2026

TU Graz and TACEO

# A Zoo of FHEMPCZK-friendly concretely-efficient symmetric crypto: How many designs?

2013: -

2014: -

2015: 1

2016: 4

2017: -

2018: 3

2019: 5

2020: 5

2021: 8

2022: 10

2023: 4 until April

source: mostly IACR eprint, plus selection from IEEE Access, ToSC, arxiv

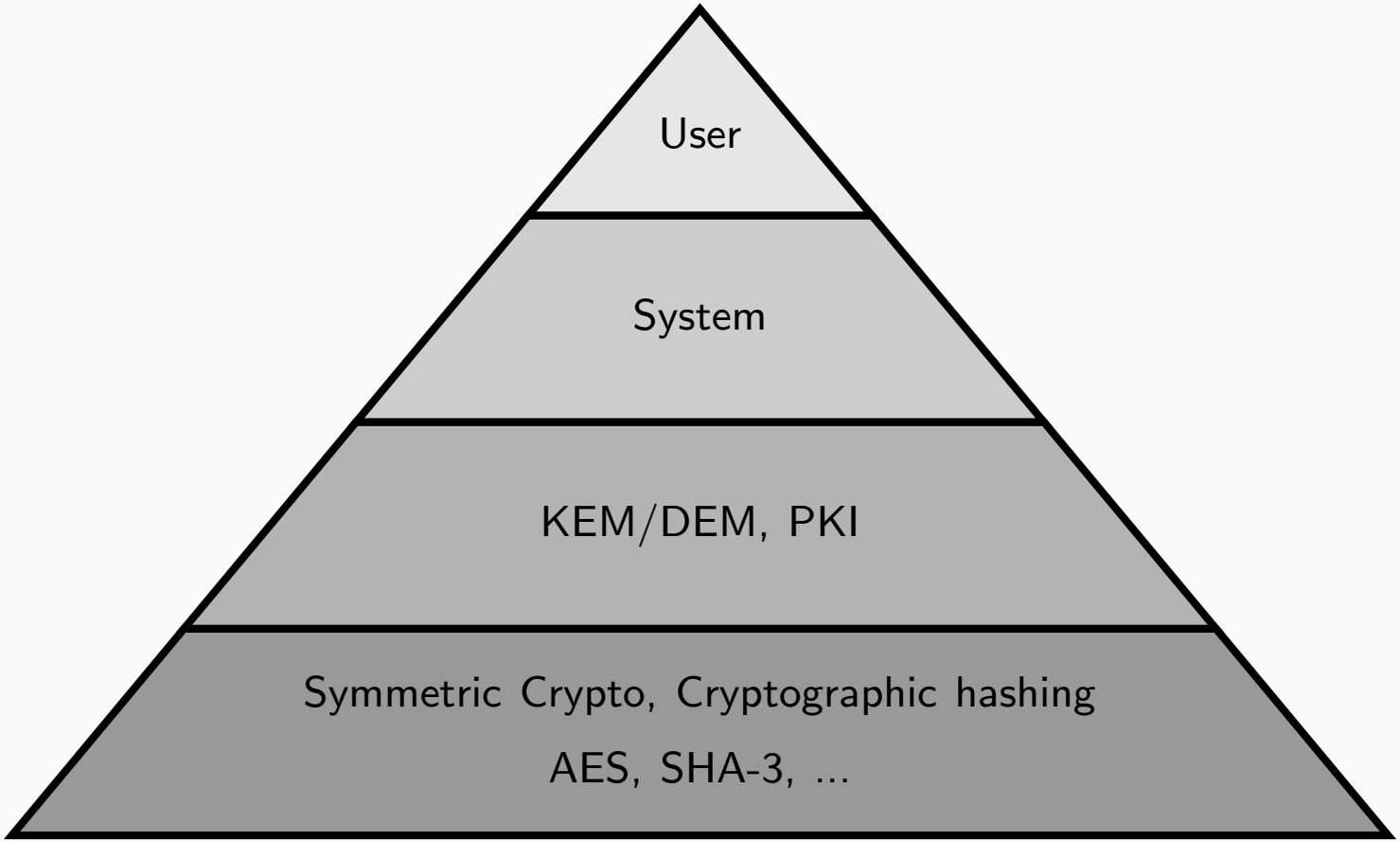
How did we get here?

# Implementation environments for symmetric cryptography

Efficiently provide confidentiality, authenticity, integrity

- until 1980s: dedicated machines, hardware implementing DES, LFSR-based approaches
- since 1990s: software implementations become more relevant in addition to hardware, see e.g. AES
- since 2010s: another boost for software-environments due to virtualization
- also since 2010s: programmable cryptography is becoming increasingly practical

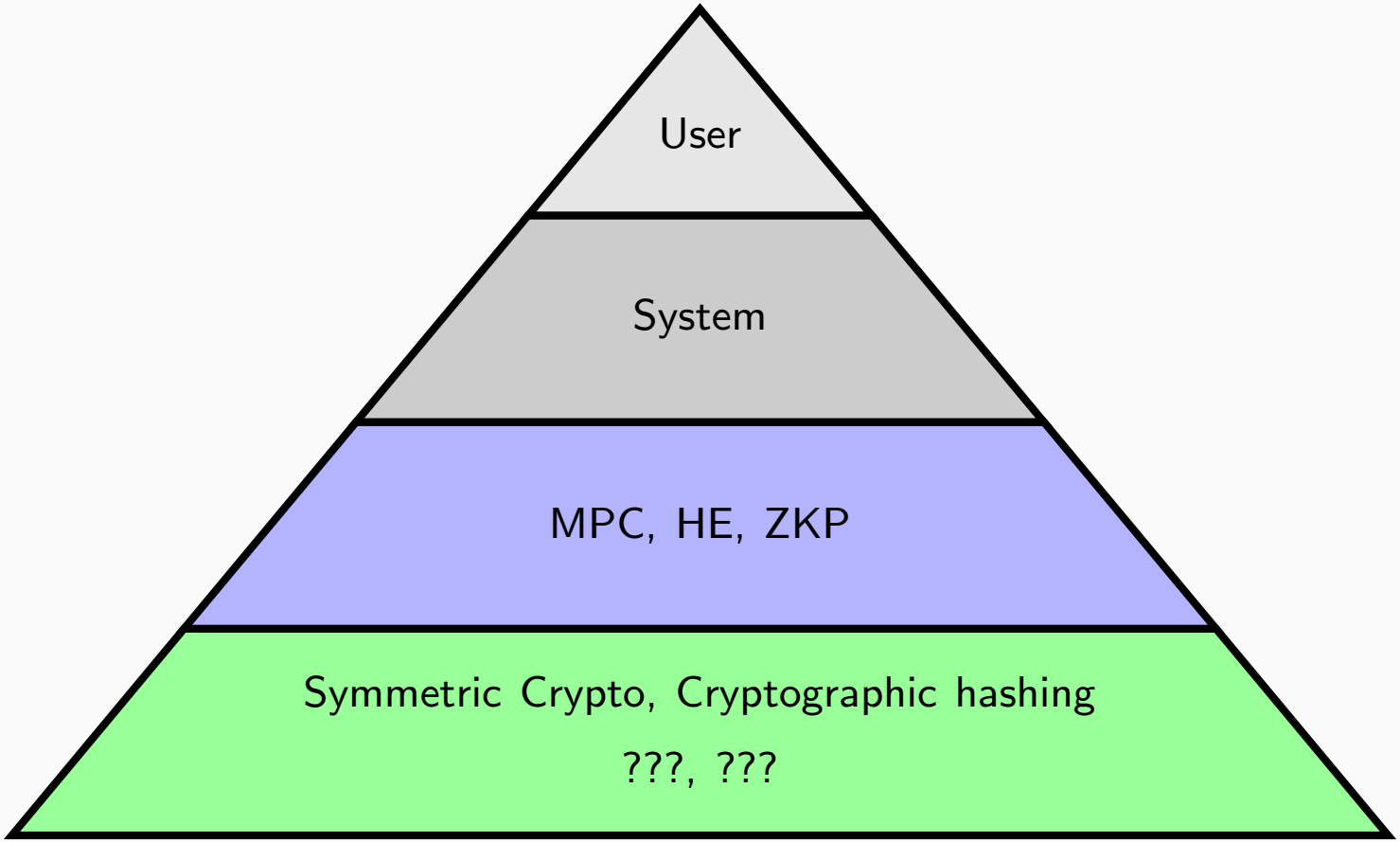
# Role of symmetric-key crypto and hashing in systems



# New cryptographic functionalities are new applications of symmetric cryptography

- FHE: Reducing ciphertext expansion, OPRFs, ...
- MPC: Distributed databases, private set intersection, data analytics, OPRFs, public-key signature schemes
- ZKP: Use-cases of zero-knowledge proofs:
  - Set Membership Proofs ("I know a private key of one of the public keys of this Merkle tree")
  - Data Commitments ("Here is the Merkle tree of the execution trace of my program, I can open it at any point").
  - "proof everything"

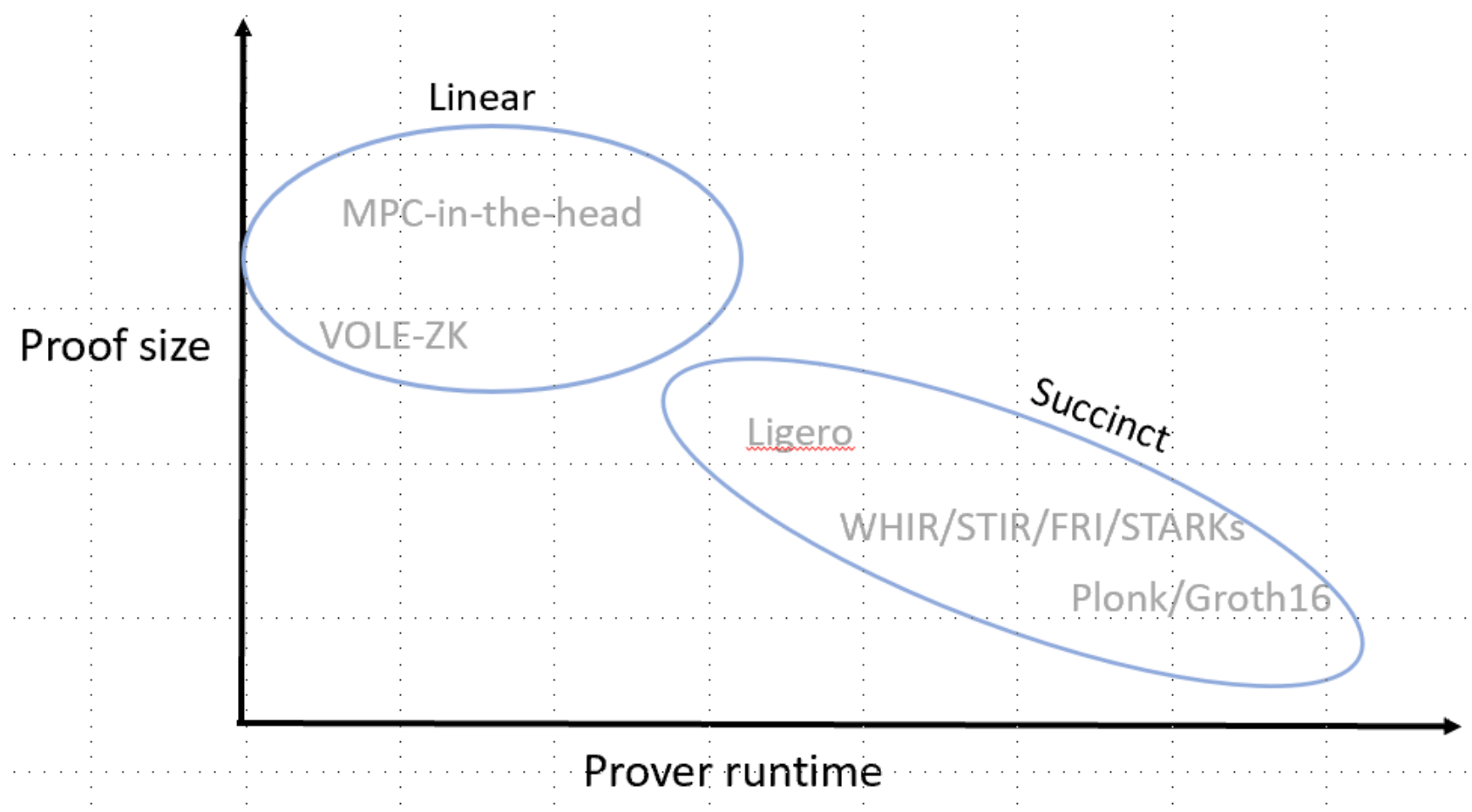
# Role of symmetric-key crypto and hashing in systems



# Transitions of use-cases in (symmetric) cryptography

- in the 1980s and 90s, there was a transition from hardware to software.
  - Hardware grew, but software grew much more.
- since the mid 2010s: we seem to be in a transition phase from direct implementations to indirect implementations within protocols aiming for "high functionality cryptography"
  - direct hardware and software implementations of course remain relevant, but the area of indirect implementations is growing fast.
  - new "virtual machines", new "metrics", co-developments of symmetric crypto with "higher/more functional" crypto layers

# Families of ZK Proofs



# The ZK(Succinct)-friendly Hash Function Zoo

## Type 1

*"low degree only"*

- Low-degree

$$y = x^d$$

- Fast in Plain
- Many rounds
- Often more constraints
- MiMC(16), GMiMC(19), Poseidon(19), Neptune (21), Poseidon2 (23), Poseidon2b(25)

# The ZK(Succinct)-friendly Hash Function Zoo

## Type 1

*"low degree only"*

- Low-degree

$$y = x^d$$

- Fast in Plain
- Many rounds
- Often more constraints
- MiMC(16), GMiMC(19), Poseidon(19), Neptune (21), Poseidon2 (23), Poseidon2b(25)

## Type 2

*"non-procedural", "fluid"*

- Low-degree equivalence

$$y = x^{1/d} \Rightarrow x = y^d$$

- Slow in Plain
- Fewer rounds
- Fewer constraints
- Friday(18), Vision (19), *Rescue*(19), Grendel(21), Griffin (22), Anemoi (22), Arion(23)

# The ZK(Succinct)-friendly Hash Function Zoo

## Type 1

*"low degree only"*

- Low-degree

$$y = x^d$$

- Fast in Plain
- Many rounds
- Often more constraints
- MiMC(16), GMiMC(19), Poseidon(19), Neptune (21), Poseidon2 (23), Poseidon2b(25)

## Type 2

*"non-procedural", "fluid"*

- Low-degree equivalence

$$y = x^{1/d} \Rightarrow x = y^d$$

- Slow in Plain
- Fewer rounds
- Fewer constraints
- Friday(18), Vision (19), *Rescue*(19), Grendel(21), Griffin (22), Anemoi (22), Arion(23)

## Type 3

*"lookups"*

- Lookup tables

$$y = T[x]$$

- Very fast in Plain
- Even fewer rounds
- Constraints depend on proof system
- `Reinforced Concrete` (21), Tip5 (23), Tip4 (23), Monolith(25), Skyscraper2(25), Polocolo(25)

# The MPC/Sharing-friendly Symmetric Crypto Zoo

2015: LowMC

2016: MiMC, LegendrePRF

2018: CryptoDarkMatter

2019: GMiMC

2020: HadesMiMC

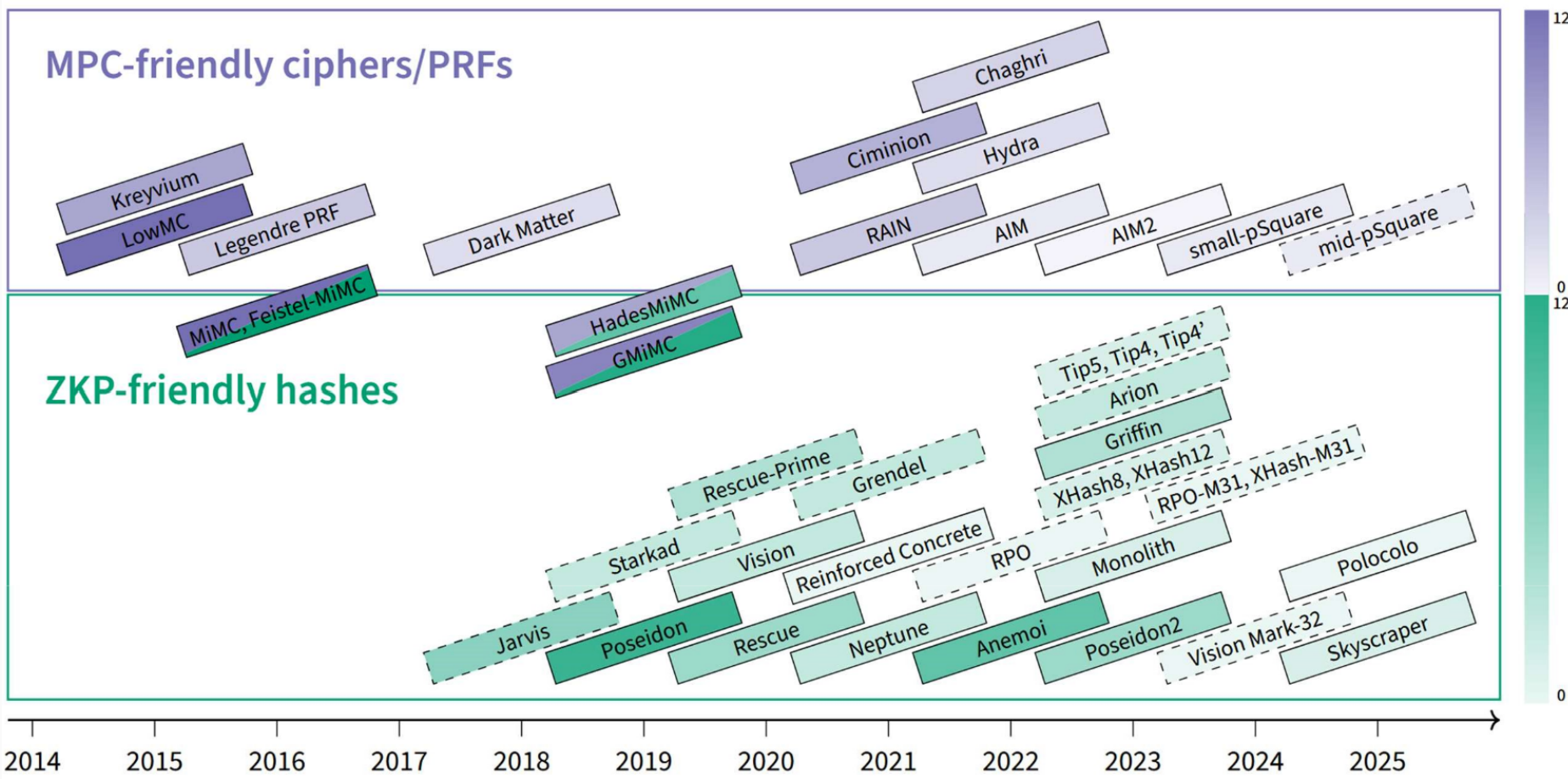
2021: Ciminion, "CryptoDarkMatter++"

2022: Rain, AIM

2023: Hydra

ongoing: GenLowMC (new: lookup aligned, dynamic generation of MPC circuit)

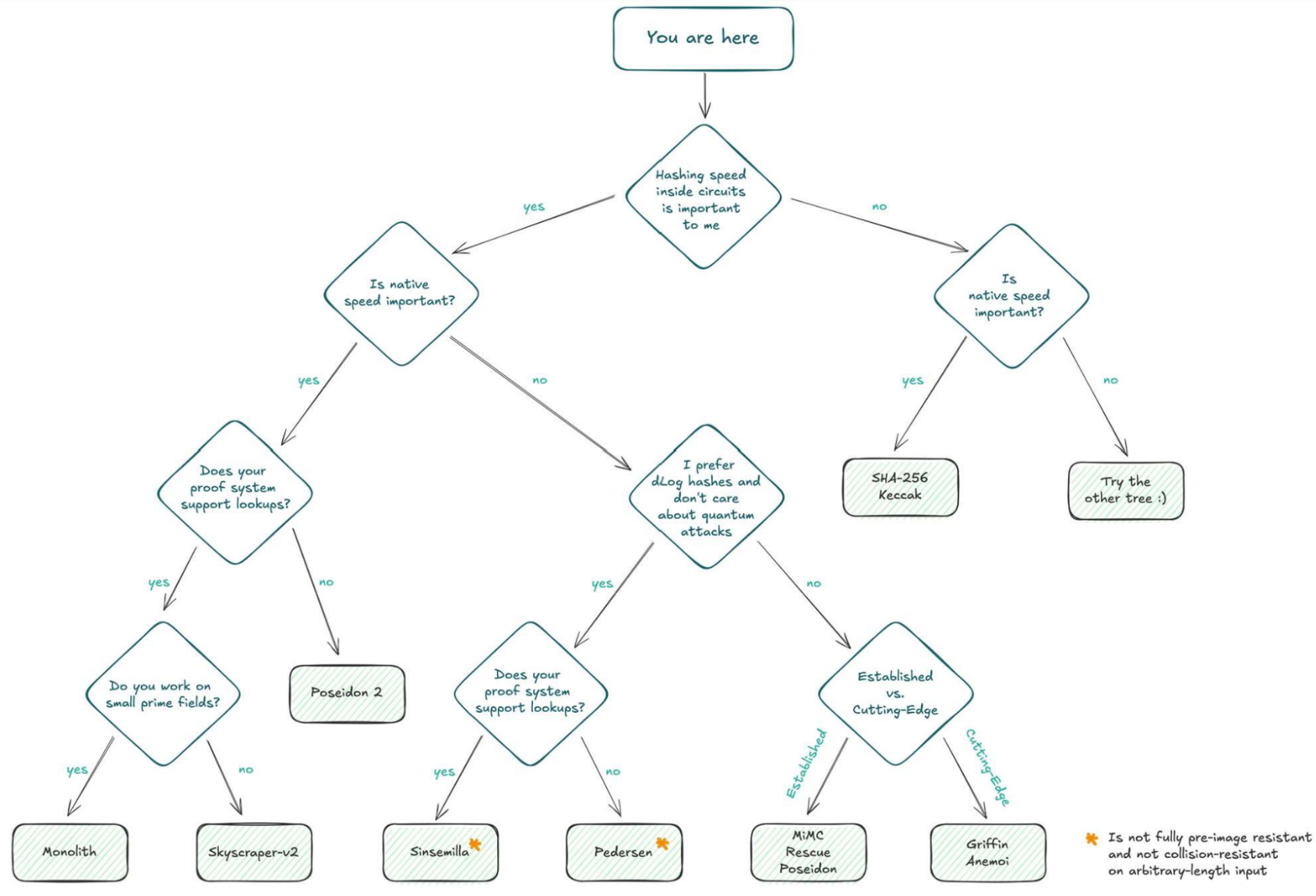
# Cryptanalysis



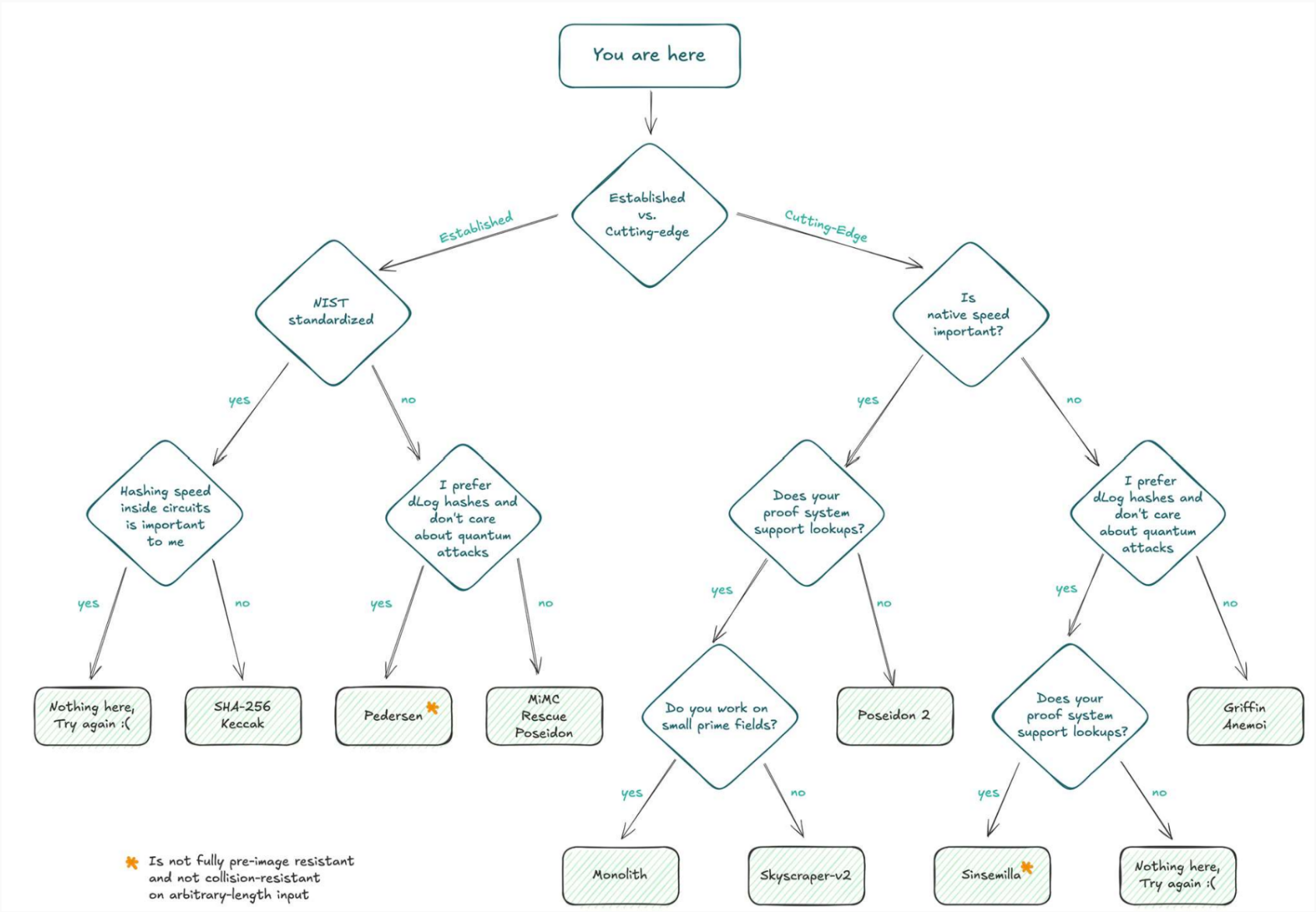
# Cryptanalysis bounties/challenges/...

- Picnic/LowMC: Three rounds of challenges since 2020-2023:
  - winners: Subhadeep Banik, Khashayar Barooti, Serge Vaudenay, Hailun Yan, F. Betül Durak, Itai Dinur
  - `https://lowmcchallenge.github.io/`
- ZKProofs-friendly hashes, 2021-2022:
  - winners: Augustin Bariant, Clémence Bouvier, Gaëtan Leurent, Léo Perrin
  - `https://www.zkhashbounties.info/`
- Ongoing: Poseidon cryptanalysis initiative (2024-2026)
  `https://www.poseidon-initiative.info/`

# How to choose? (1/2)



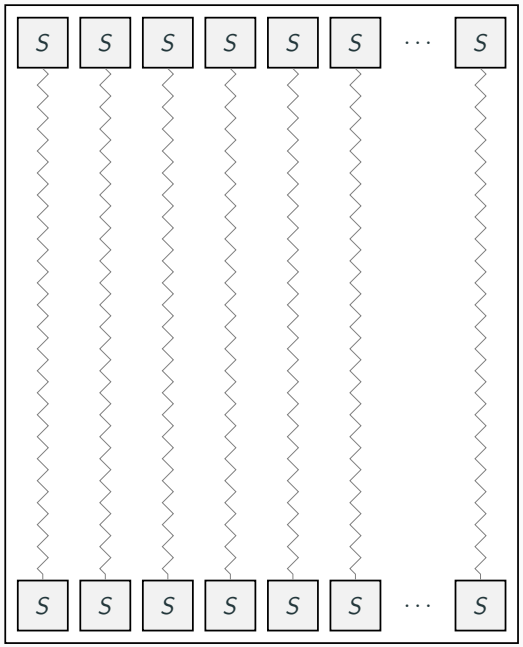
# How to choose? (2/2)



# S-Box sizes, over time. A selection

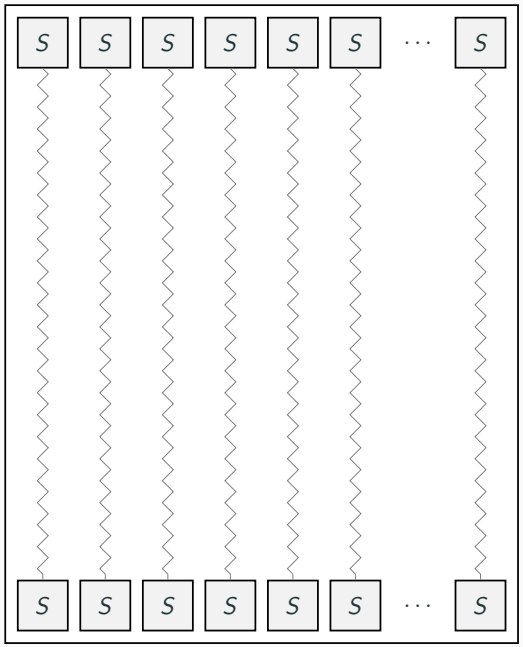
- mid 1970s, 6to4-bit: DES S-box just fits on a Chip
- mid 1990s, 8to8-bit: e.g. Rijndael/AES, attractive for good performance in both HW and SW
- since 2000, smaller, more "lightweight" S-boxes
  - 3to3-bit (e.g. Printcipher, LowMC)
  - 4to4-bit (e.g. Noekeon, Present, Klein, Prince)
  - 5to5-bit (e.g. Keccak, Ascon)
- since 2015, big and huge S-boxes
  - n to n-bit, elements in GF($2^n$)
    - for n from 100 to 1000 (e.g. MiMC, Rain)
  - n to n-bit, elements in GF(p)
    - for n from 128 to $\geq$1000 (e.g. MiMC)
    - for n from 17 to 63 (e.g. Pasta)
    - for n from 8 to 256 (most in the ZK-friendly Zoo)
  - set of size around $2^9$ to $2^{10}$ to set of same size: (elements in $\mathbb{Z}_n$) ReinforcedConcrete (RChash)

# ⩓ SPNs with Partial Nonlinear Layers

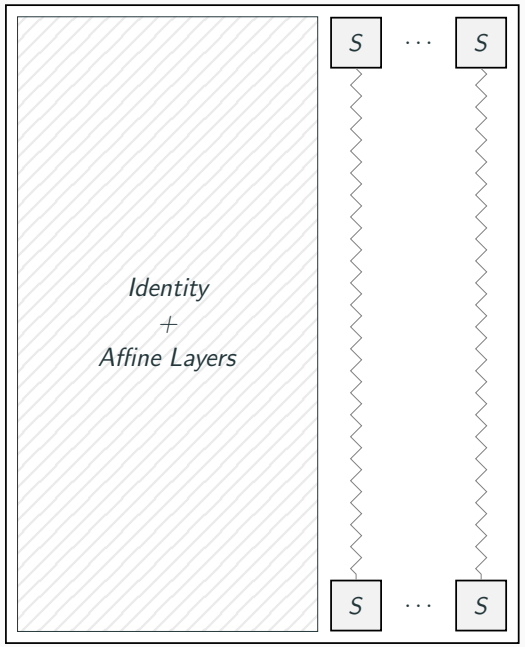



SPN
(e.g., SHARK in
1996)

# ⩓ SPNs with Partial Nonlinear Layers

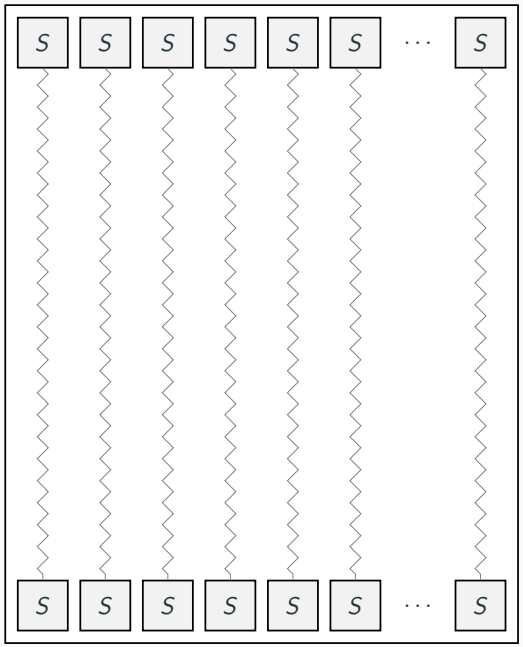



SPN
(e.g., SHARK in 1996)

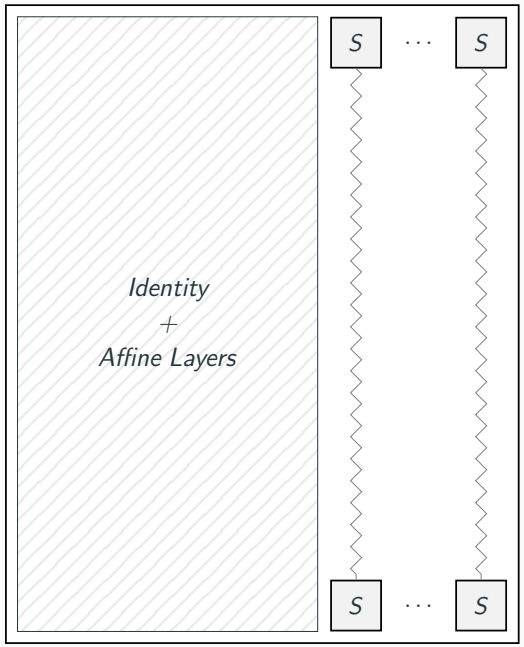



P-SPN since 2010
(e.g., ARMADILLO, Zorro, LowMC)

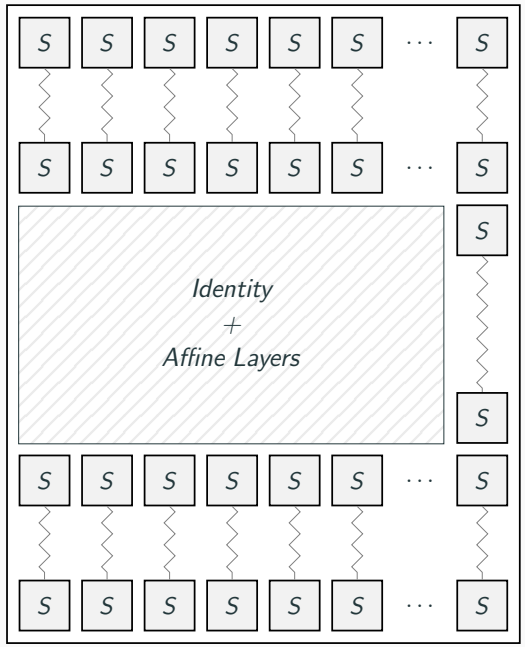
# ⩓ SPNs with Partial Nonlinear Layers




SPN
(e.g., SHARK in 1996)




P-SPN since 2010
(e.g., ARMADILLO, Zorro, LowMC)




Hades
(e.g., HadesMiMC, Poseidon)

# Thoughts on "Theory" vs. "Practice"

- Provable Security?
  - Modes of operation: do proofs carry over from $F_2$ to $F_p$?
  - SPN vs. Partial-SPN: First positive results by Guo, Standaert, Wang, Wang, Yu (FSE 22)
    - Stronger model, like indifferentiability?
  - "ZK-friendly" compression? New work by Andreeva, Bhattacharyya, Roy, Trevisani (CSF 24)
- "Asymptotic analysis" / "asymptotic designs".
  - Input: blocksize, security level
  - Output: concrete design with security claim
  - Some designs allow for it, e.g. HPC, LowMC, MiMC, Poseidon, ...
  - Pros: Flexibility
  - Cons: Less focused cryptanalysis.

# Wanted

Consolidate spezialization tree of candidate hashes

A concretely efficient primitive for *Low-depth hashing*

- ZK-friendly and simultaneously MPC-friendly

# Conclusions

- Lots of exciting new developments in "high functionality cryptography" - some are likely here to stay
- ... leading to lots of exciting research for design and analysis of symmetric crypto and hashing
- Industry interest is growing, demand for standards to support interoperability and increase trust

# On Zero-Knowledge Proofs, MPC, and Symmetric Crytography

Christian Rechberger

Jan 9, 2026

TU Graz and TACEO

# Bibliography i