Standards for Zero-Knowledge Proofs and their Relevance to the NIST Threshold Call

Mary Maller, Ethereum Foundation and PQShield

MPTS 2023: NIST Workshop on Multi-party Threshold Schemes 2023 - 27 Sept 2023

ZKPROOF

What is a Zero-Knowledge Proof

The digital language of truth

- Everything I say in zero-knowledge is true.
- I can choose to say nothing at all.
- Everything I do not say is perfectly hidden.

What is a Zero-Knowledge Proof

The digital language of truth

- Everything I say in zero-knowledge is true.
- I can choose to say nothing at all.
- Everything I do not say is perfectly hidden.

Privacy

Building Blocks

Scalability

What is a Zero-Knowledge Proof

The digital language of truth

- Everything I say in zero-knowledge is true.
- I can choose to say nothing at all.
- Everything I do not say is perfectly hidden.

Privacy

Building Blocks

Scalability

Companies usually resort to trusted hardware.
Zero-knowledge would be better solution but is currently viewed as experimental technology.

ZKPs or Trusted Hardware?

	ZKPs	Trusted Hardware
Attacks Detectable	Nope	Nope
No Trusted Third Party	Yes	Nope
Easy Bug Fixes	Sometimes	Nope
Safe from Side Channels	Sometimes	Nope
Open Source	Yes	Nope
Widely Used	Nope	Yes

Disclaimer: I may be a little biased here.

Applications

Scalability

- Verifiable FHE
- Verifiable outsourced computation
- Verifiable mixnets
- Attested sensors
- Verifiable formal verification
- Scalable blockchains

Building Blocks

- Actively secure MPC
- Code based digital signatures
- Random beacons
- Range proofs
- Membership proofs
- Blind signatures

Privacy

- Online games
- Anonymous cryptocurrency
- Whistleblowers
- Solvency proofs
- Compliant closed source algorithms
- Anonymous credentials

Scalability + Privacy

- Blocklists
- Machine learning checks and balances
- Storage proofs
- Captcha
- Persistent pseudonyms
- Proof of exploits

Applications

Scalability	Building Blocks	Privacy	Scalability + Privacy
Verifiable FHE	Actively secure MPC	Online games	Blocklists
Verifiable outsourced computation	Code based digital signatures	Anonymous cryptocurrency	Machine learning checks and balances
Verifiable mixnets	Random beacons	Whistleblowers	Storage proofs
Attested sensors	Range proofs	Solvency proofs	Captcha
Verifiable formal verification	Membership proofs	Compliant closed source algorithms	Persistent pseudonyms
Scalable blockchains	Blind signatures	Anonymous credentials	Proof of exploits

Many of todays engineering efforts are targeting Scalable Blockchains.

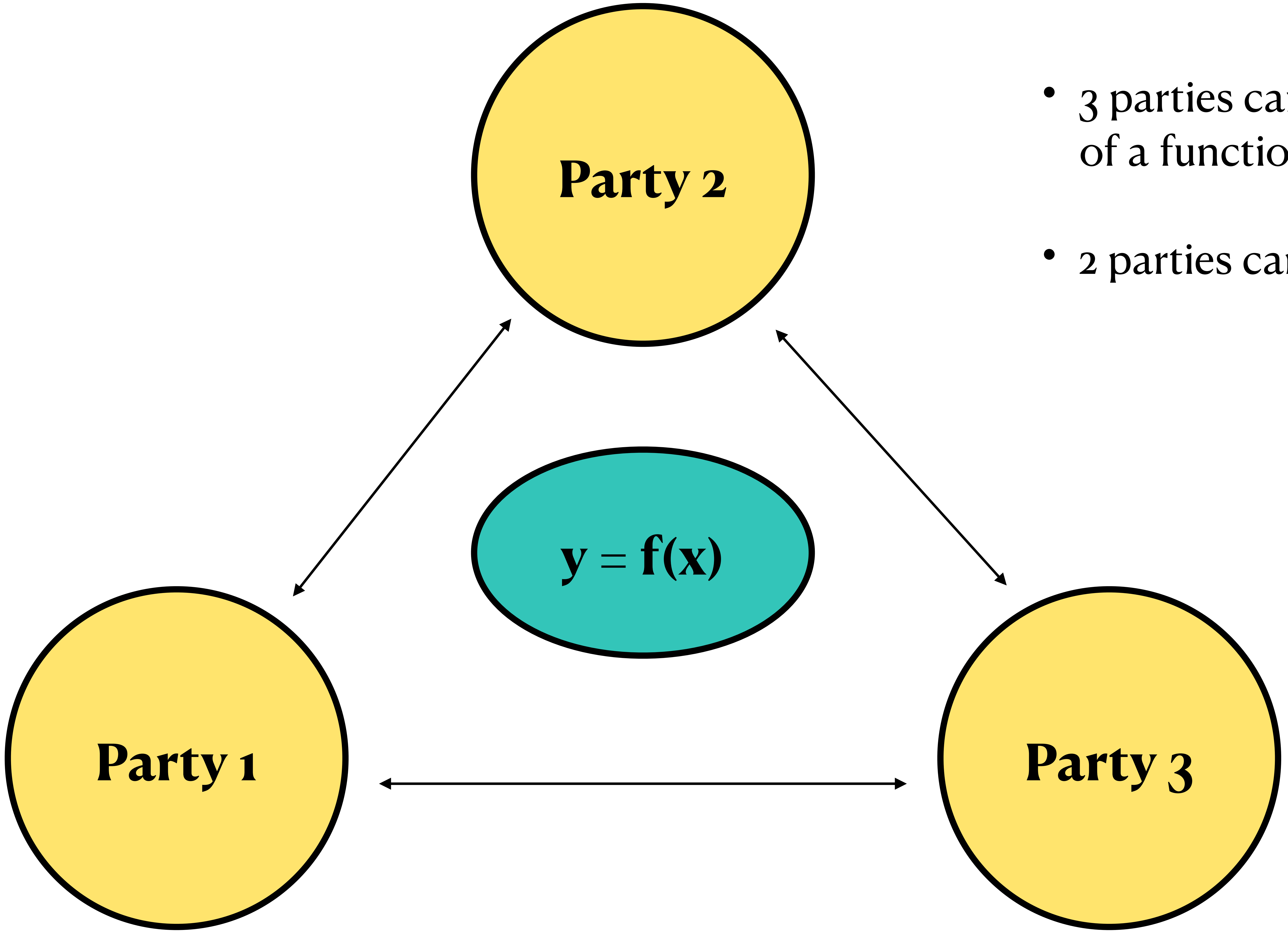
Applications

Building Blocks

Verifiable FHE	Actively secure MPC
Anonymous credentials	Code based digital signatures
Blind signatures	Random beacons
Group signatures	Range proofs
Aggregate signatures	Membership proofs
Broadcast channels	Delay encryption
CCA Encryption	E-Voting

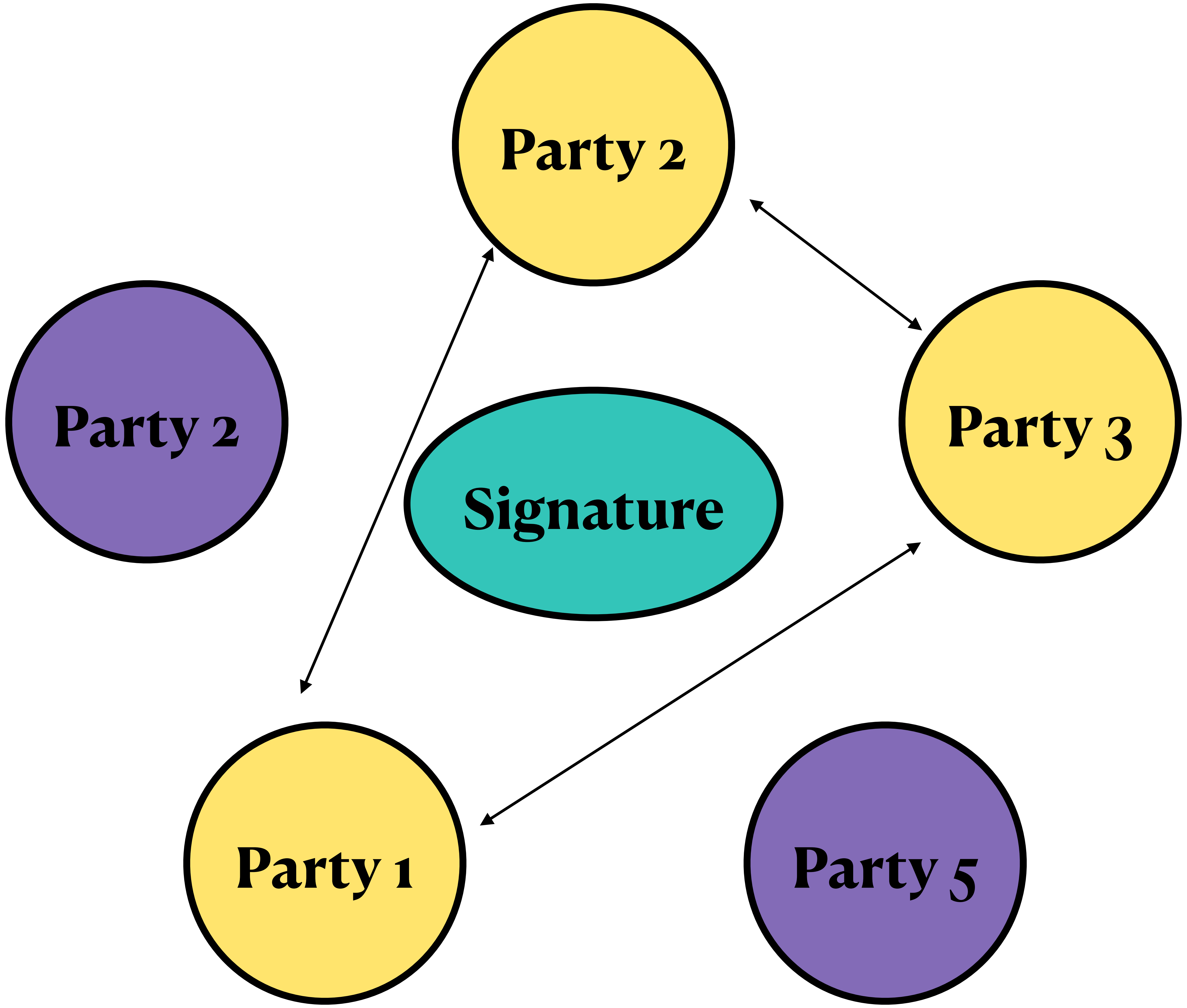
- Today I am focusing on ZKPs in the context of MPC;
- Outside academia, industries, governments and NIST are thinking about advanced cryptographic primitives;
- Many cryptographic primitives rely on zero-knowledge.

Multiparty Computation (MPC)



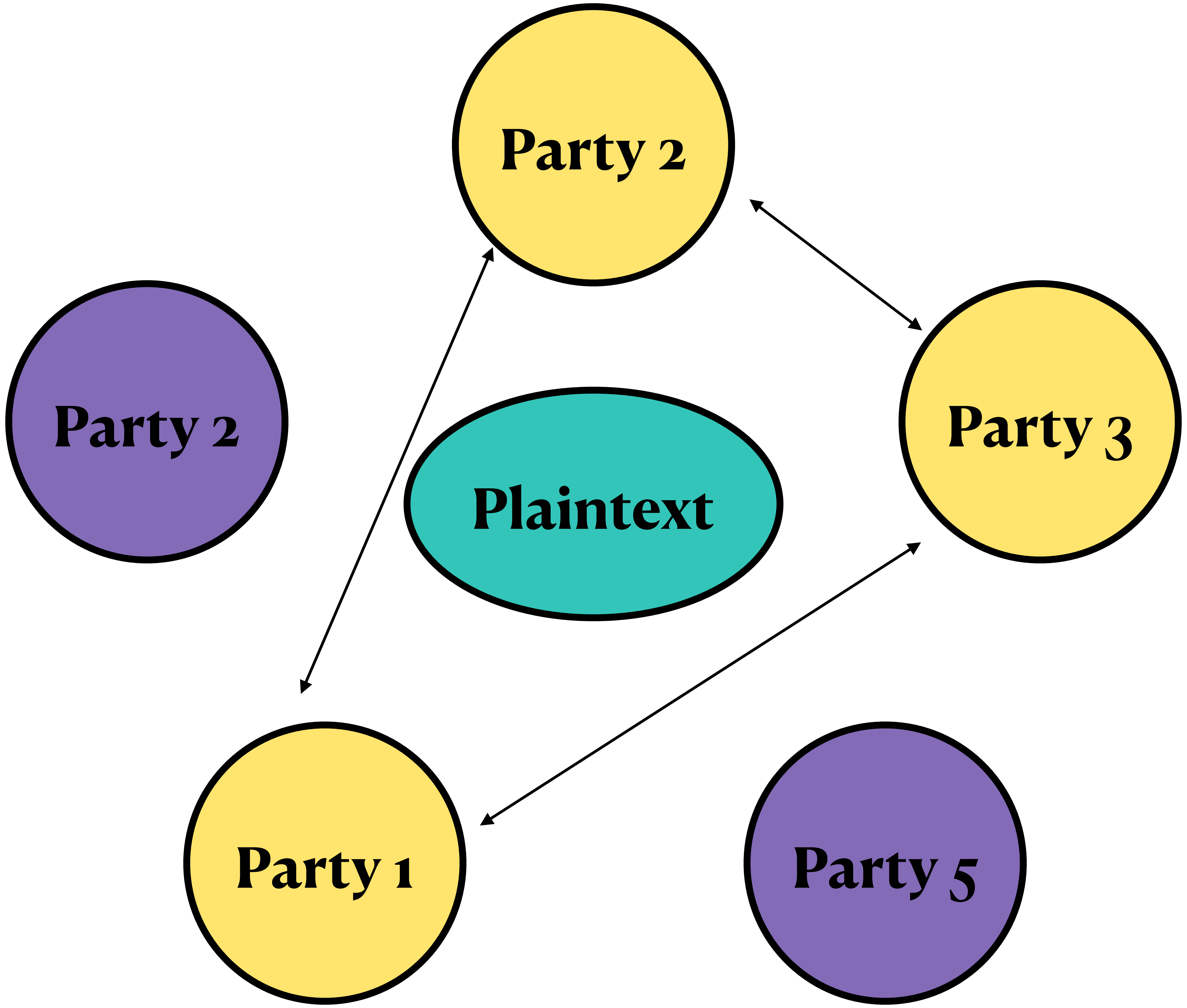
- 3 parties can compute the output of a function.
- 2 parties cannot

Threshold Signatures = MPC Special Case



- 3 parties can compute the output of a function.
- 2 parties cannot

Threshold Decryption = MPC Special Case



- 3 parties can compute the output of a function.
- 2 parties cannot

Desirable Properties for Threshold Schemes

Easy Key Generation

Active Security

Identifiable abort

Desirable Properties for Threshold Schemes

Easy Key Generation | Active Security | Identifiable abort

- Trusted key generation: Sometimes acceptable and sometimes not acceptable.
- Distributed key generation: Usually tricky.
- ZKPs: Multisignatures with Proof of Possession of secret key are usually easy.

Desirable Properties for Threshold Schemes

Easy Key Generation

Active Security

Identifiable abort

- Passive security: Output is correct and private when all parties follow the protocol.
- Active security: Output is correct and private even if parties behave badly.
- ZKPs: If all parties prove honest behaviour in zk in a passively secure scheme, then output is actively secure.

Desirable Properties for Threshold Schemes

Easy Key Generation

Active Security

Identifiable abort

- Liveness: Want protocol to terminate. If doesn't terminate want to know why.
- ZKPs: If all parties prove honest behaviour in zk in a passively secure scheme, then aborts can only be caused by not saying anything at all.

NIST Draft Threshold Call

- Table 12 (Page 53): Explicitly expressed interest in zero-knowledge proofs of knowledge of secret key for a selection of schemes.
- Option 1: Give special purpose proving scheme for each of the relations.
- Option 2: Give general purpose proving scheme and just specialise the constraints.

Table 12. Example ZKPoKs of interest related to Cat1 primitives

Related type	**Related (sub)sub-category: Primitive**	**Example ZKPoK (including consistency with public commitments of secret-shares, when applicable)**
Keygen	C1.5.1: ECC keygen	of discrete-log (s or d) of pub key Q
|	C1.5.2: RSA keygen	of factors (p, q), or group order ϕ, or decryption key d
|	C1.5.3: AES keygen	of secret key k (with regard to secret-sharing commitments)
PKE	C1.2.1: RSA encryption	of secret plaintext m (encrypted)
|	C1.2.2: RSA decryption	of secret-shared plaintext m (after SSO-threshold decryption)
Symmetric	C1.4.1: AES enciphering	of secret key k (with regard to plaintext/ciphertext pair)
|	C1.4.2: Hashing in KDM	of secret pre-image Z

Special Purpose

- Fast: No need to do arithmetisation.
- More work: Standardisation process will only be useful for one primitive.

General Purpose

- Slow: Pay full cost of arithmetisation.
- Less work: Standardisation process is useful for all ZKP applications.

But general purpose ZKPs are now fast enough that we can afford it.

ZK Implementations are Becoming Fast

RISC Zero Datasheet

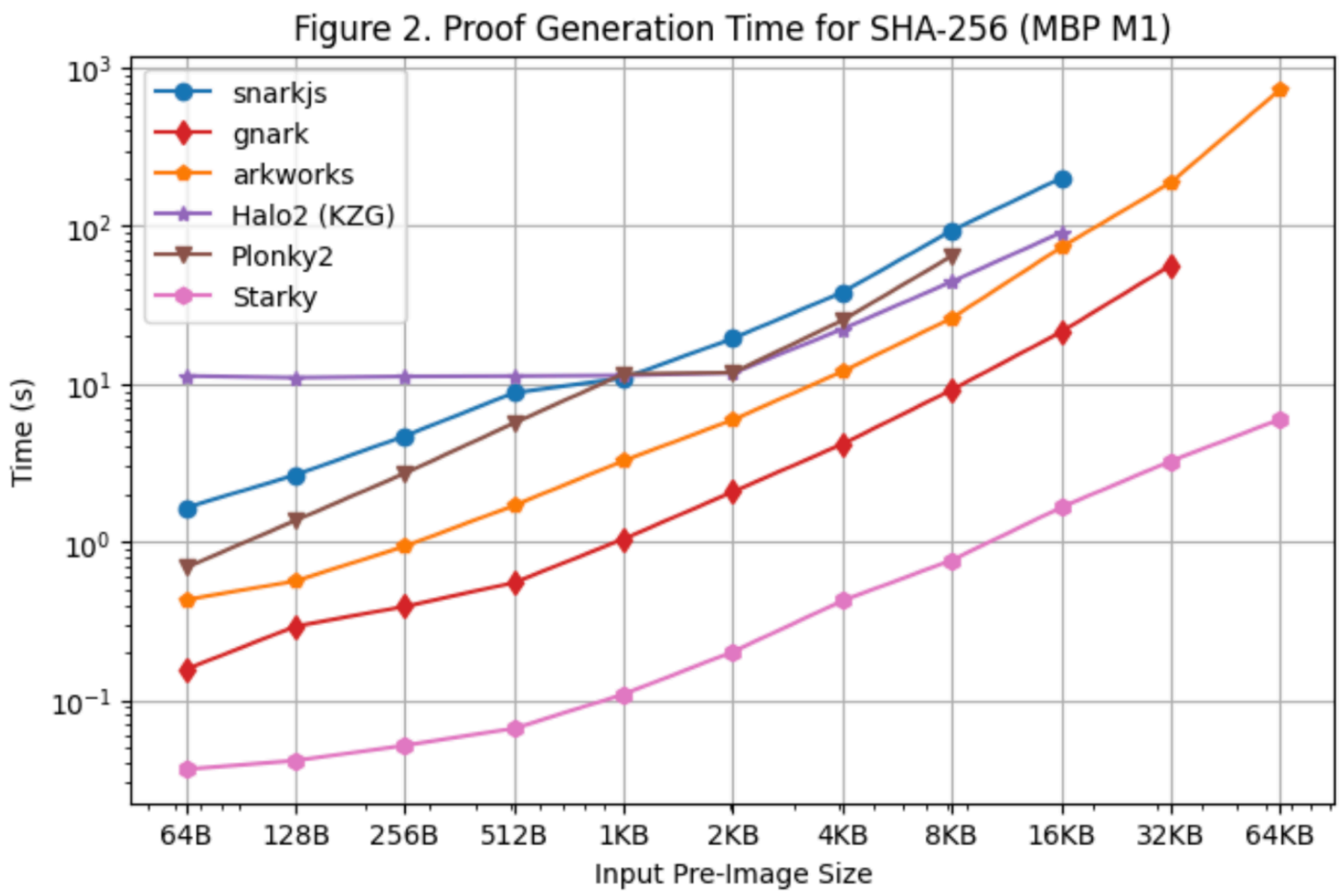
(April 2023 -- as of commit cd1a37e)

Example	Cycles
Factors	32 k
Chess	256 k
Digital Signature	64 k
EVM	2048 k
JSON	64 k
Password Checker	64 k
SHA	64 k
Waldo	8192 k
Wordle	64 k

Metal on M1 MacBook

Cycles	Prover Time	RAM	Proof Size	Speed
32 k	1.38 s	234.4 MB	201.3 kB	23.7 kHz
64 k	1.87 s	468.7 MB	213 kB	35 kHz
128 k	2.80 s	937.4 MB	236 kB	46.9 kHz
256 k	4.97 s	1.87 GB	247.7 kB	52.7 kHz
512 k	9.49 s	3.75 GB	259.9 kB	55.3 kHz
1024 k	17.96 s	7.5 GB	273.2 kB	58.4 kHz
2048 k	50.13 s	15 GB	297.8 kB	41.8 kHz
4096 k	1:51.2	30 GB	311.1 kB	37.7 kHz

Benchmarking ZKP Development Frameworks: the Pantheon of ZKP



Mo Dong
2nd March 2023

ZKProof Standardisation Effort

- Global movement to standardise and mainstream advanced cryptography by building a community-driven trust ecosystem.
- Formed in 2018 after top researchers and developers saw technology becoming advanced enough for standards.
- I joined the editorial team in 2021.
- We expect this to be a long process as the community jointly learn best practices.

Education

Standards

Community

ZKProof Standardisation Effort

- Most ZKPs are formalised only in research papers.
- Research paper != formal specification suitable for deployment.
- Collaboration of industry developers and academics are in the process of writing specifications for a full general purpose proving system.
- This is a lot of work.
- Hopeful that if we can pull it off, then it should be directly applicable to proofs of possession and other threshold related applications.

Final Remarks

- Easier to get support for specification drafts with formal industry support for the application.
- If you are seriously considering using ZKPs in your threshold application then we would love to hear from you.
- Contact us at contact@zkproof.org