MQ on my Mind (MQOM)

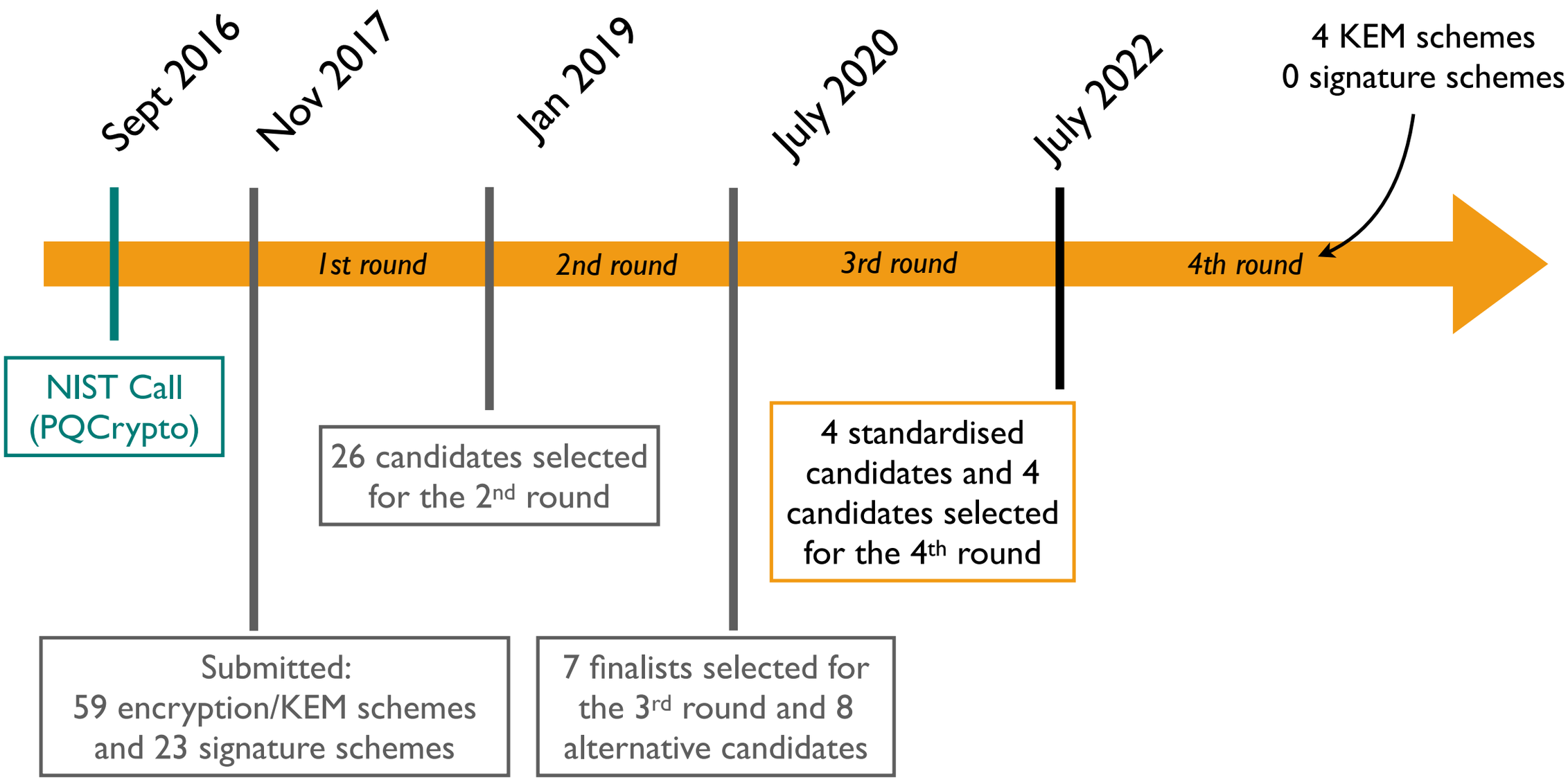
Post-Quantum Signatures from the Non-Structured Multivariate Quadratic Problem

Ryad Benadjila, Thibauld Feneuil, Matthieu Rivain

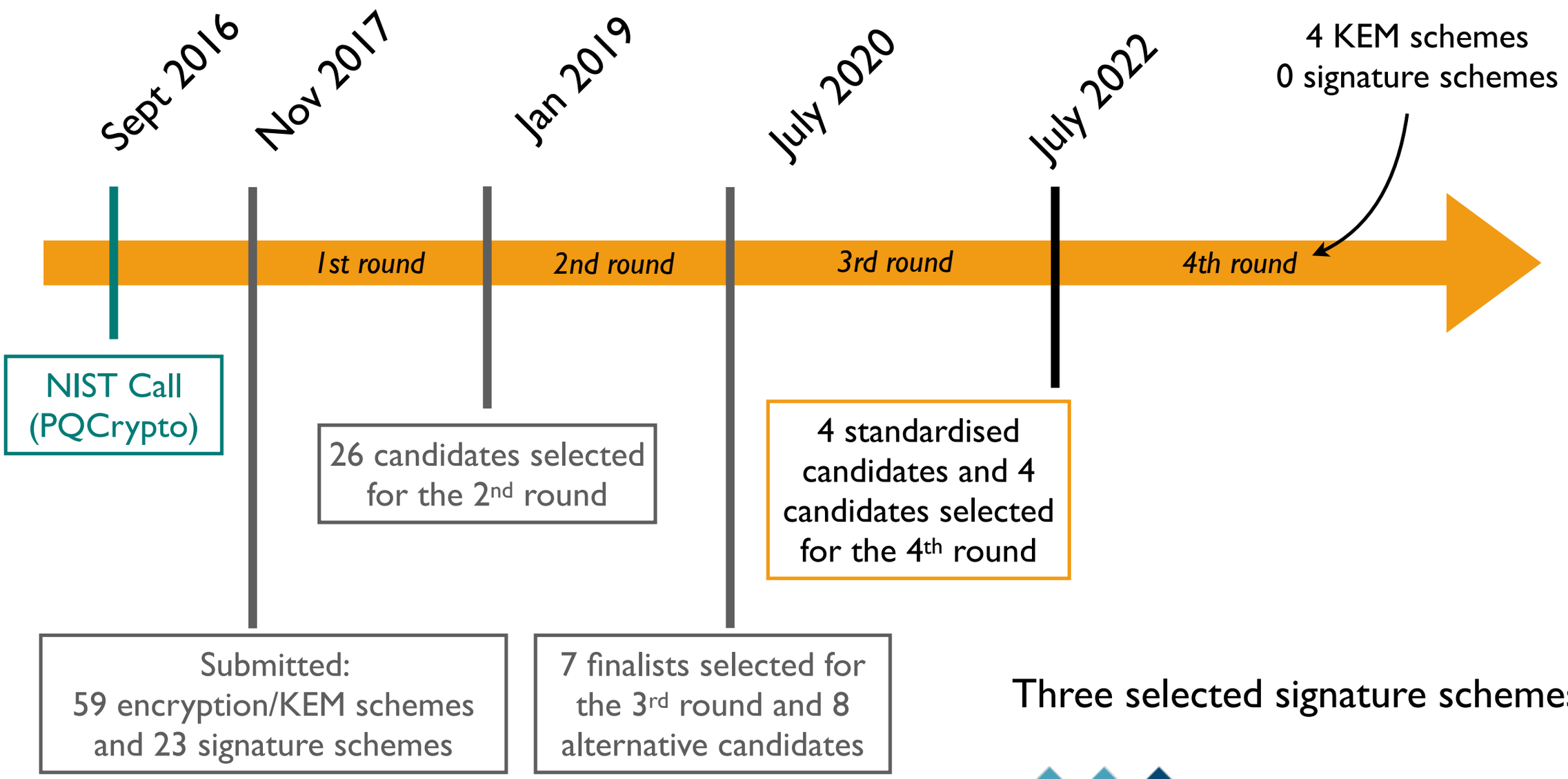
EuroS&P 2024

July 10, 2024, Vienna

Context


Sept 2016
Nov 2017
Jan 2019
July 2020
July 2022
4 KEM schemes
0 signature schemes
1st round
2nd round
3rd round
4th round
NIST Call (PQCrypto)
26 candidates selected for the 2nd round
4 standardised candidates and 4 candidates selected for the 4th round
Submitted: 59 encryption/KEM schemes and 23 signature schemes
7 finalists selected for the 3rd round and 8 alternative candidates

Context



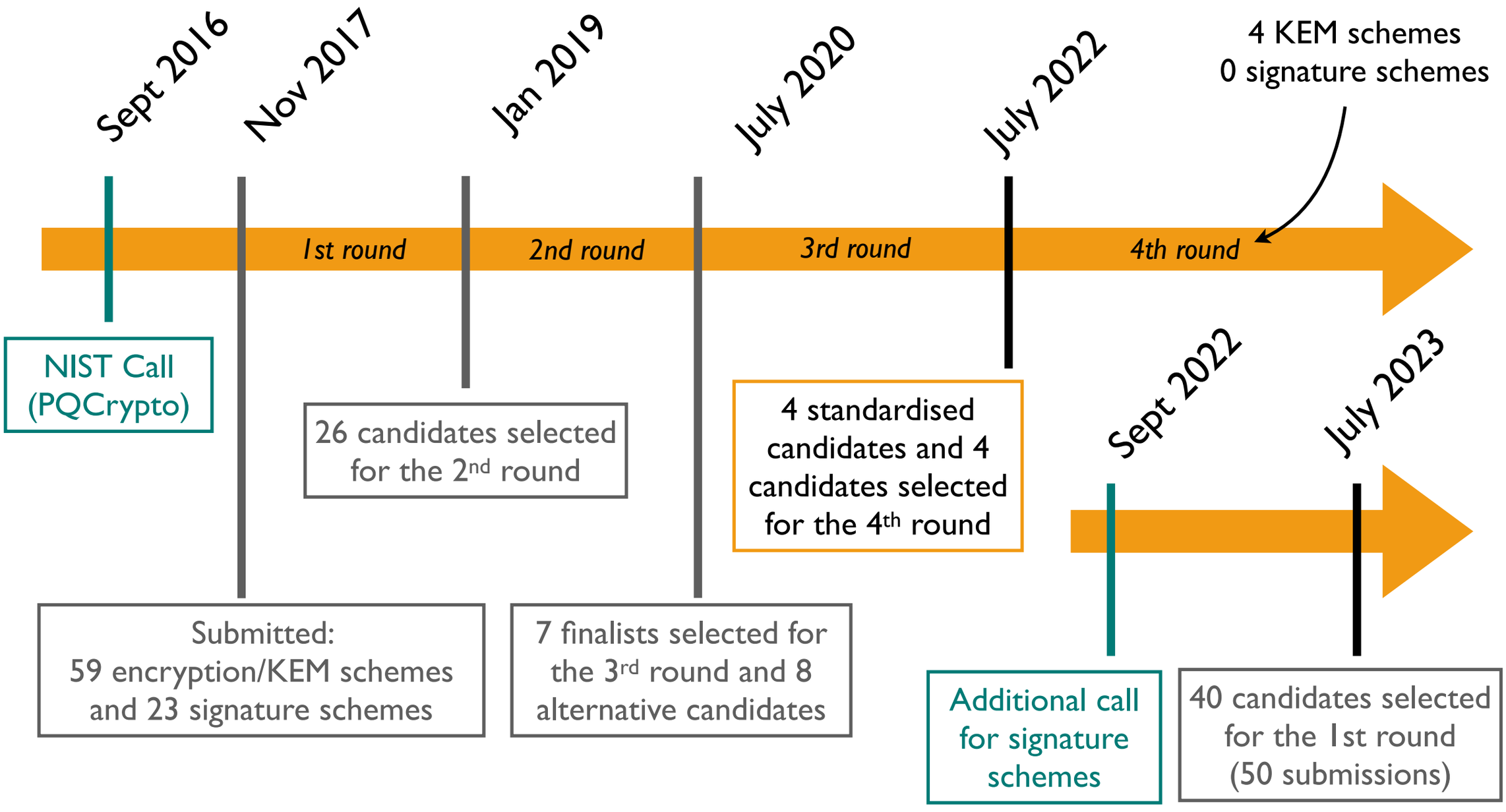
Three selected signature schemes:

DILITHIUM

FALCON

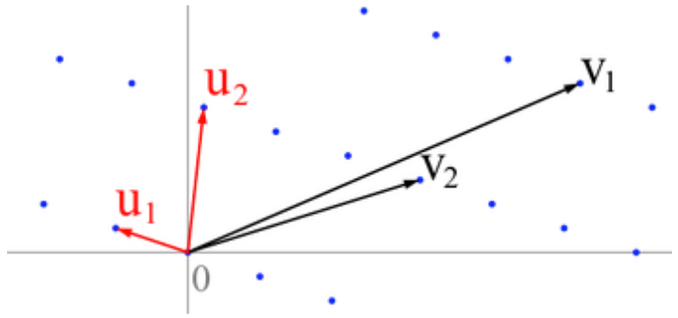
SPHINCS+

Context

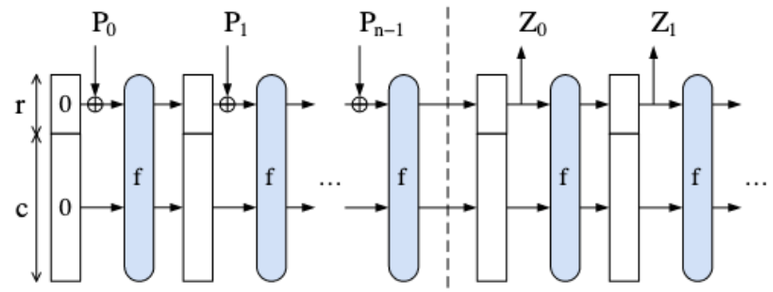

Sept 2016
Nov 2017
Jan 2019
July 2020
July 2022
4 KEM schemes
0 signature schemes
1st round
2nd round
3rd round
4th round
NIST Call (PQCrypto)
26 candidates selected for the 2nd round
4 standardised candidates and 4 candidates selected for the 4th round
Sept 2022
July 2023
Submitted: 59 encryption/KEM schemes and 23 signature schemes
7 finalists selected for the 3rd round and 8 alternative candidates
Additional call for signature schemes
40 candidates selected for the 1st round (50 submissions)

Which Security Assumptions

Lattice-based Cryptography



Symmetric Cryptography



Multivariate Cryptography

$$3x^2 - 43y = 0$$
$$xy - 10 = 0$$

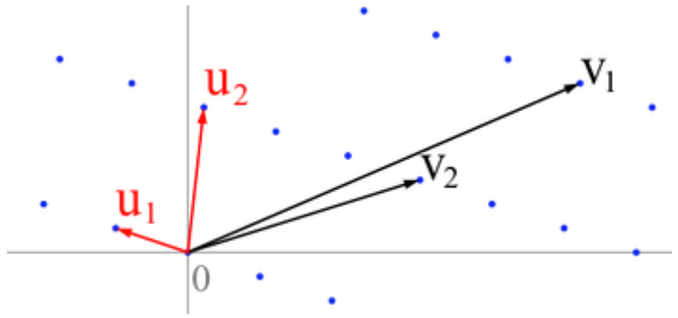
Code-based Cryptography

Isogeny-based Cryptography

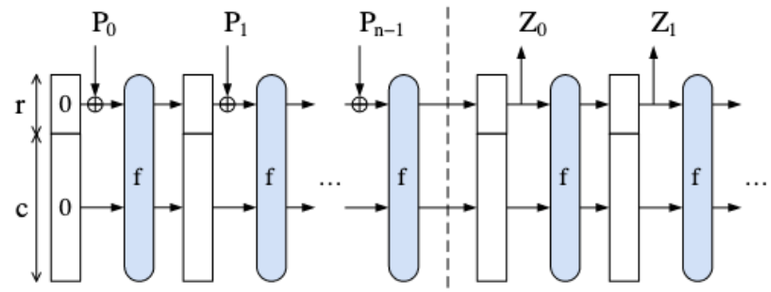
. . .

Which Security Assumptions

Lattice-based Cryptography



Symmetric Cryptography



Multivariate Cryptography

$$3x^2 - 43y = 0$$
$$xy - 10 = 0$$

Code-based Cryptography

Isogeny-based Cryptography

. . .

Design Strategy

- Rely on the MQ problem

Multivariate Quadratic Problem

From $\{a_{i,j,k}\}_{i,j,k}, \{b_{i,j}\}_{i,j}$ and $\{y_i\}_i$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ such that

$$\begin{cases} y_1 &= \sum_{j,k} a_{1,j,k} x_j x_k + \sum_j b_{1,j} x_j, \\ &\vdots \\ y_m &= \sum_{j,k} a_{m,j,k} x_j x_k + \sum_j b_{m,j} x_j \,. \end{cases}$$

Design Strategy

- Rely on the MQ problem

Multivariate Quadratic Problem

From $\{a_{i,j,k}\}_{i,j,k}, \{b_{i,j}\}_{i,j}$ and $\{y_i\}_i$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ such that

$$\begin{cases} y_1 &= \sum_{j,k} a_{1,j,k} x_j x_k + \sum_j b_{1,j} x_j, \\ &\vdots \\ y_m &= \sum_{j,k} a_{m,j,k} x_j x_k + \sum_j b_{m,j} x_j \,. \end{cases}$$

Multivariate Quadratic Problem (matrix form)

From $(A_1, \ldots, A_m, b_1, \ldots, b_m, y_1, \ldots, y_m)$, find $x \in \mathbb{F}_q^n$ such that

$$\forall i \leq m, \quad y_i = x^T A_i x + b_i^T x.$$

Design Strategy

- Rely on the MQ problem

Multivariate Quadratic Problem

From $\{a_{i,j,k}\}_{i,j,k}, \{b_{i,j}\}_{i,j}$ and $\{y_i\}_i$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ such that

$$\begin{cases} y_1 &= \sum_{j,k} a_{1,j,k} x_j x_k + \sum_j b_{1,j} x_j, \\ &\vdots \\ y_m &= \sum_{j,k} a_{m,j,k} x_j x_k + \sum_j b_{m,j} x_j \,. \end{cases}$$

Multivariate Quadratic Problem (matrix form)

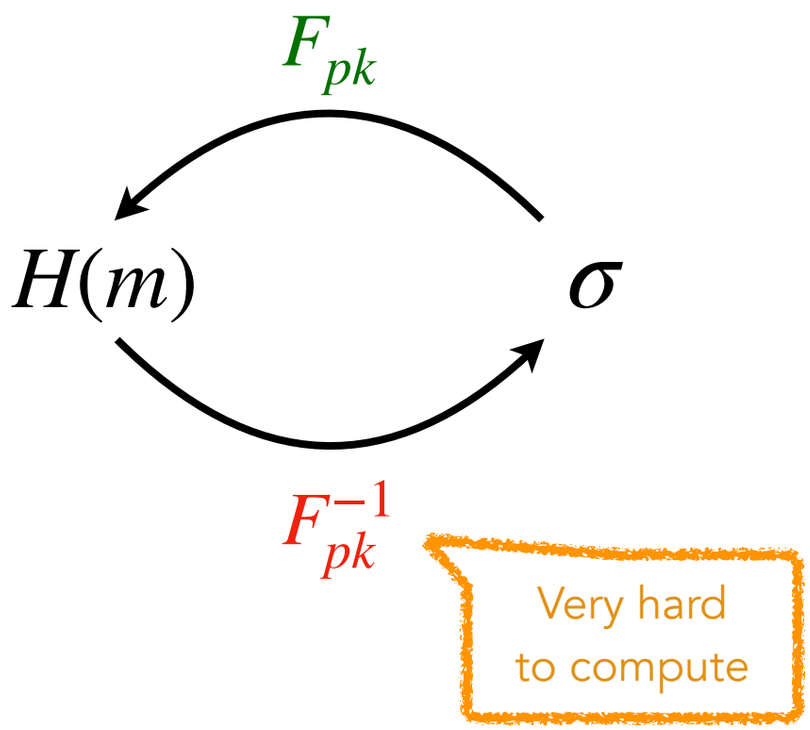
From $(A_1, \ldots, A_m, b_1, \ldots, b_m, y_1, \ldots, y_m)$, find $x \in \mathbb{F}_q^n$ such that

$$\forall i \leq m, \quad y_i = x^T A_i x + b_i^T x.$$

- Build the **more conservative** multivariate signature scheme

How to build signature schemes?

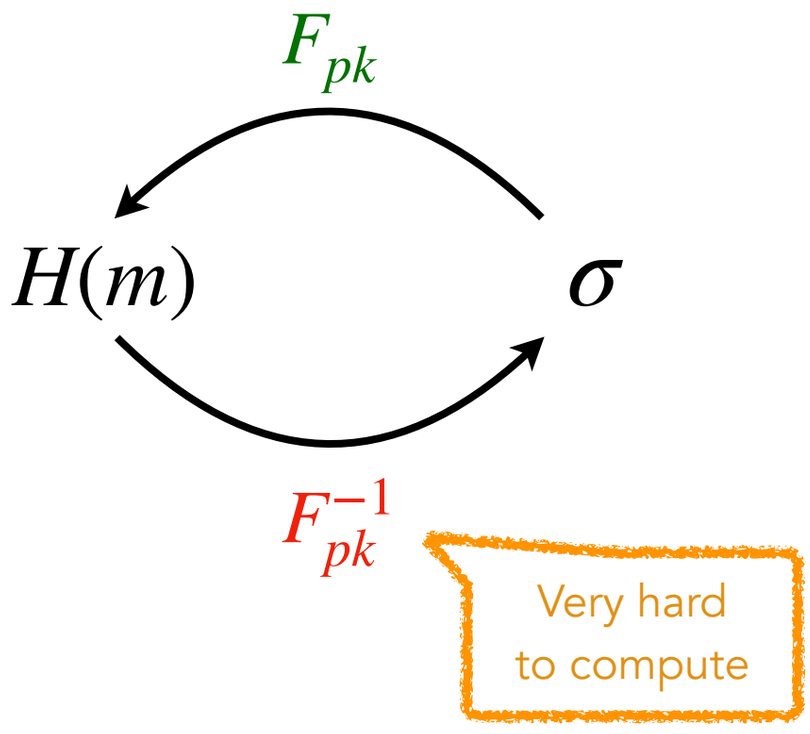
Hash & Sign



- Short signatures
- “Trapdoor” in the public key

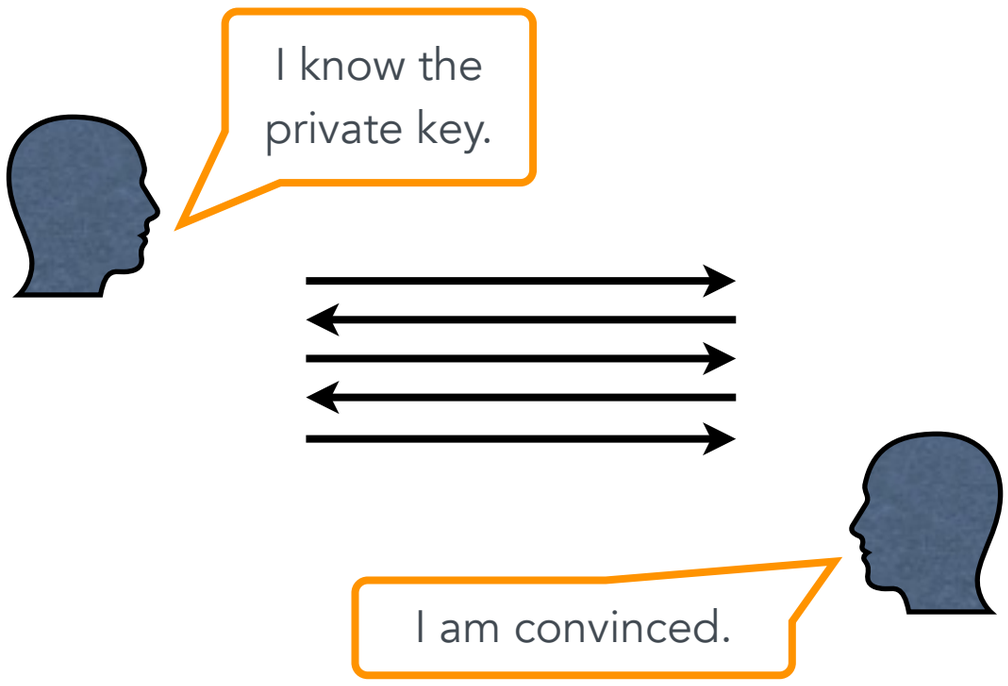
How to build signature schemes?

Hash & Sign



- Short signatures
- "Trapdoor" in the public key

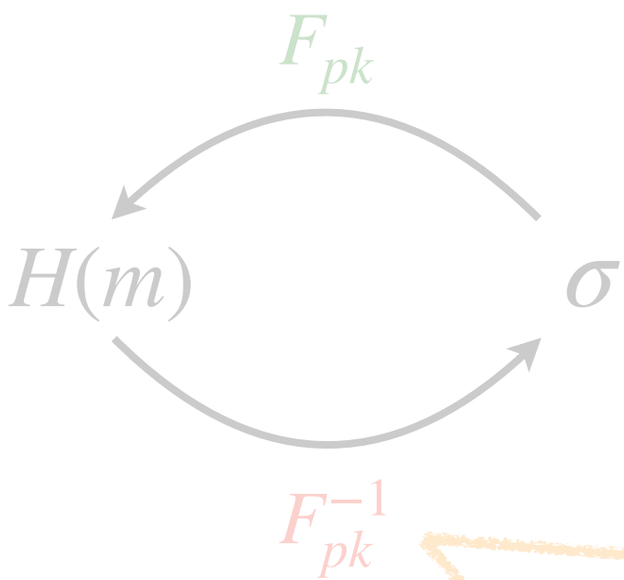
From an identification scheme



- Large(r) signatures
- Short public key

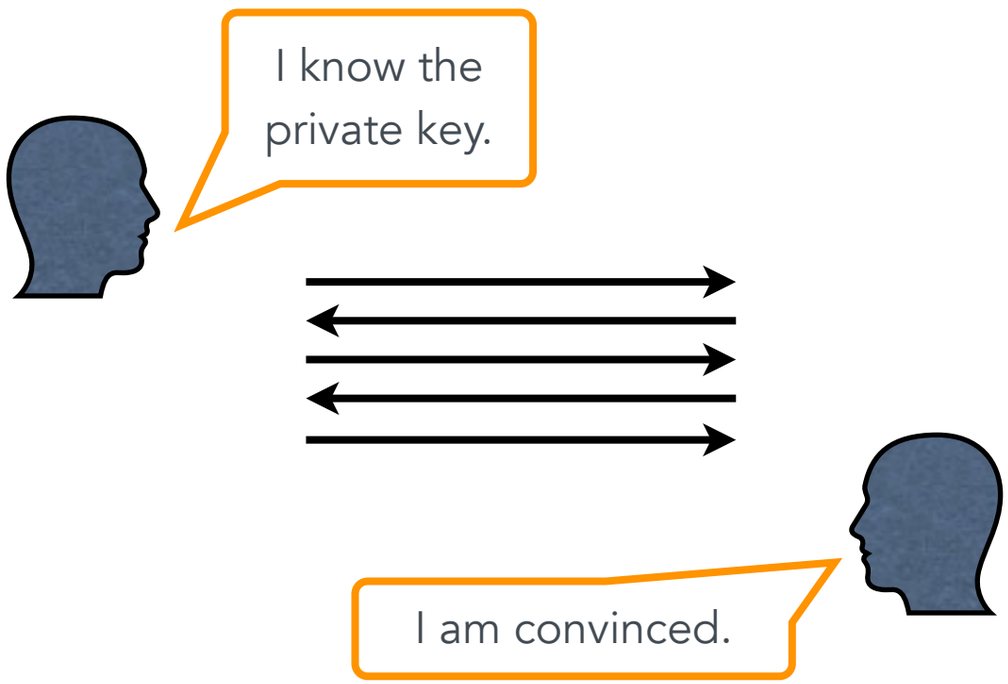
How to build signature schemes?

Hash & Sign



- Short signatures
- "Trapdoor" in the public key

From an identification scheme

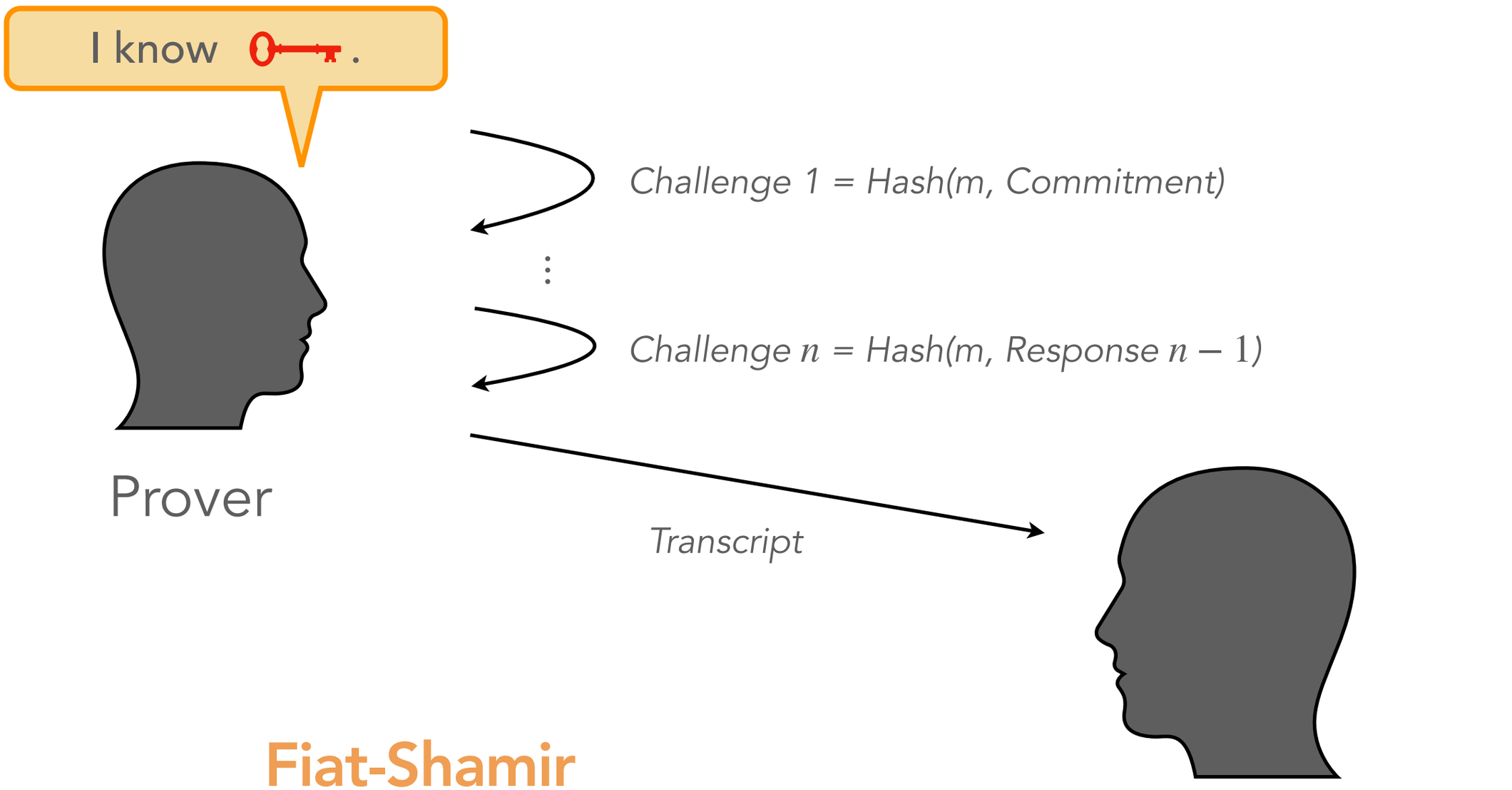


- Large(r) signatures
- Short public key

Identification Scheme

- **Completeness:** Pr[verif ✓ | honest prover] = 1
- **Soundness:** Pr[verif ✓ | malicious prover] $\leq \varepsilon$ (e.g. 2^{-128})
- **Zero-knowledge:** verifier learns nothing on .

Identification Scheme



Fiat-Shamir Transformation

m: message to sign

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

[IKOS07] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, Amit Sahai. *Zero-knowledge from secure multiparty computation.* STOC 2007

[KKW18] Katz, Kolesnikov, Wang. *Improved non-interactive zero knowledge with applications to post-quantum signatures.* ACM CCS 2018.

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

Example: we want to prove that $x \cdot y = 55 \pmod{p}$

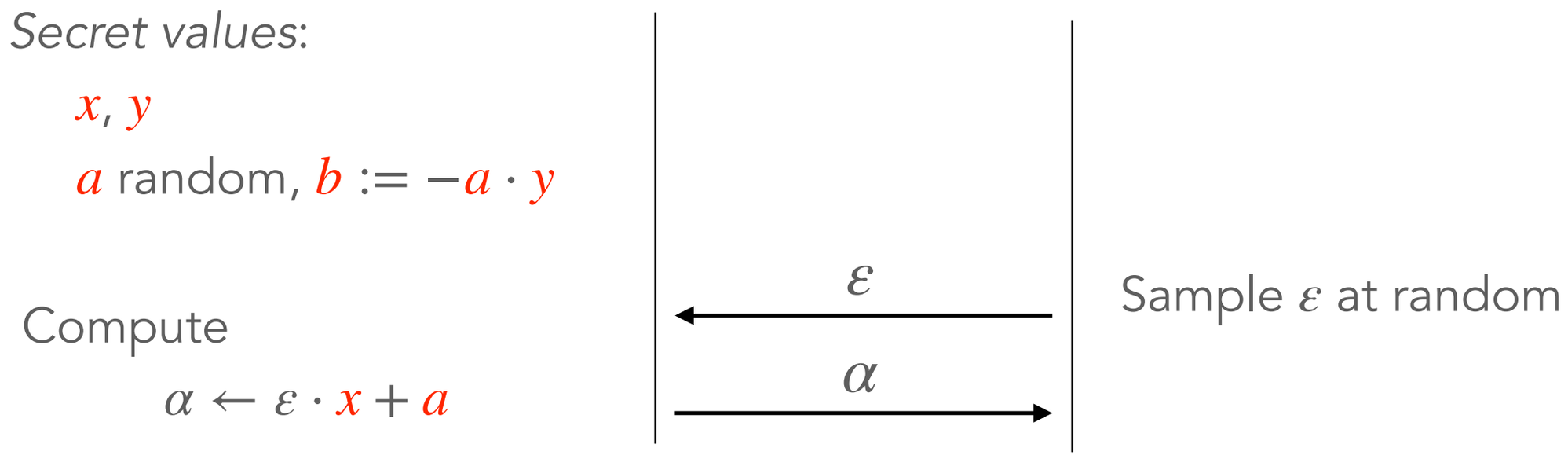
[BN20] Baum, Nof. *Concretely-efficient zero-knowledge arguments for arithmetic circuits and their application to lattice-based cryptography. PKC 2020.*

[KZ22] Kales, Zaverucha. *Efficient Lifting for Shorter Zero-Knowledge Proofs and Post-Quantum Signatures. ePrint 2022.*

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

Example: we want to prove that $x \cdot y = 55 \pmod{p}$



Prove that $\begin{cases} 0 = \varepsilon \cdot x + a - \alpha \pmod{p} \\ 0 = \alpha \cdot y + b - 55 \cdot \varepsilon \pmod{p} \end{cases}$

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

Example: we want to prove that $x \cdot y = 55 \pmod{p}$

Prover | ***Verifier***

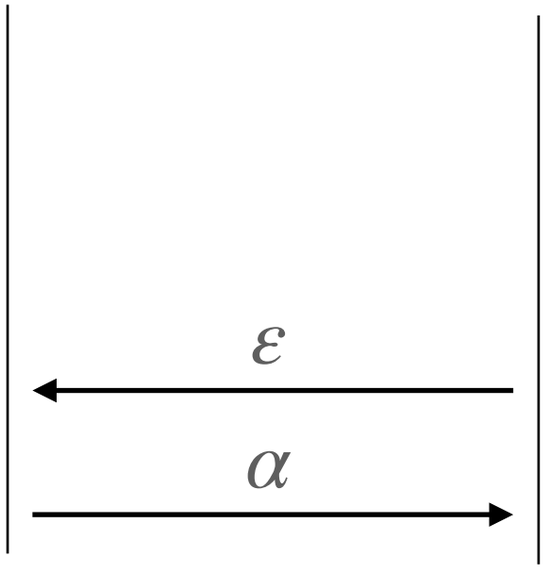
Secret values:

x, y

a random, $b := -a \cdot y$

Compute

$\alpha \leftarrow \varepsilon \cdot x + a$



Sample ε at random

Prove that $\begin{cases} 0 = \varepsilon \cdot x + a - \alpha \pmod{p} \\ 0 = \alpha \cdot y + b - 55 \cdot \varepsilon \pmod{p} \end{cases}$

$$\Rightarrow \varepsilon \cdot (x \cdot y - 55) + (ay + b) = 0$$

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

Example: we want to prove that $x \cdot y = 55 \pmod{p}$

Prover | ***Verifier***

Secret values:

x y

If $x \cdot y \neq 55$, the relation is satisfied only with probability $1/p$.

Sample ε at random

Prove that $\begin{cases} 0 = \varepsilon \cdot x + a - \alpha \pmod{p} \\ 0 = \alpha \cdot y + b - 55 \cdot \varepsilon \pmod{p} \end{cases}$

$\Rightarrow \varepsilon \cdot (x \cdot y - 55) + (ay + b) = 0$

Additive-based MPC-in-the-Head

- Framework to prove **linear relations** over secret values

Example: we want to

Pr

Secret values:

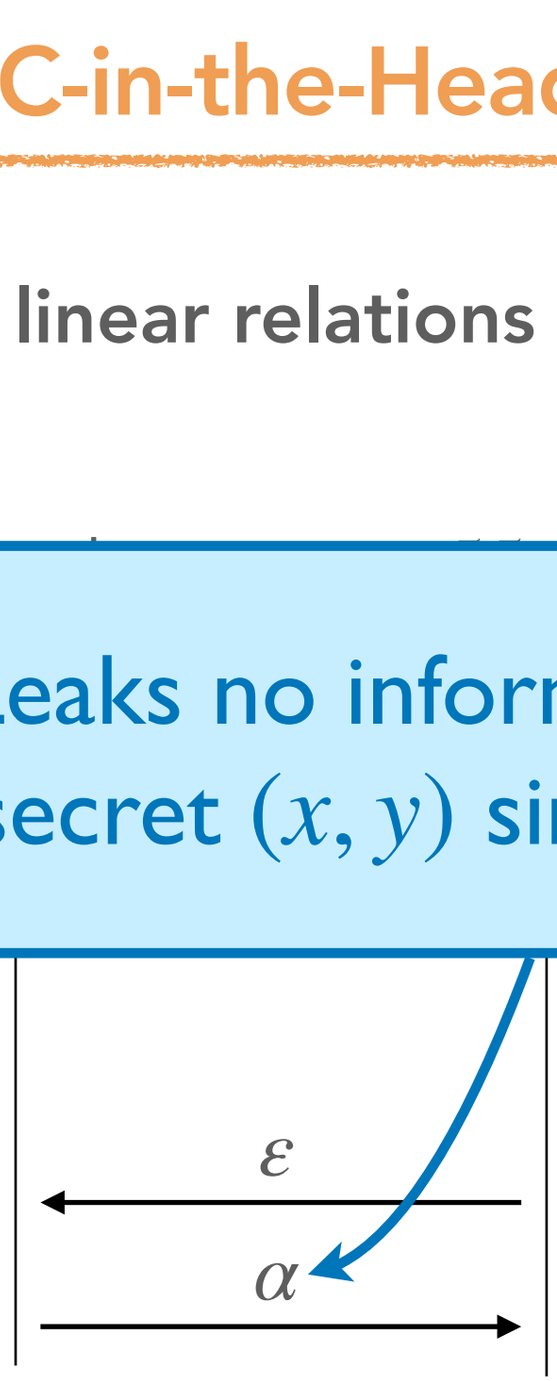
x, y

a random, $b := -a \cdot y$

Compute

$$\alpha \leftarrow \varepsilon \cdot x + a$$

Leaks no information about the secret (x, y) since a is random.



Sample ε at random

Prove that $\begin{cases} 0 = \varepsilon \cdot x + a - \alpha \pmod p \\ 0 = \alpha \cdot y + b - 55 \cdot \varepsilon \pmod p \end{cases}$

$$\Rightarrow \varepsilon \cdot (x \cdot y - 55) + (ay + b) = 0$$

Step 1: batching MQ equations

- *Goal*: prove that $x = (x_1, \ldots, x_n)$ satisfies

$$\forall i \in [1 : m], \quad y_i - x^T A_i \, x - b_i^T = 0$$

Step 1: batching MQ equations

- *Goal*: prove that $x = (x_1, \ldots, x_n)$ satisfies

$$\forall i \in [1 : m], \quad y_i - x^T A_i \, x - b_i^T = 0$$

- *Batched check*: prove that $x = (x_1, \ldots, x_n)$ satisfies

$$\sum_{i=1}^{m} \gamma_i \left(y_i - x^T A_i \, x - b_i^T \right) = 0$$

where $\gamma_1, \ldots, \gamma_m$ are uniformly at random in $\mathbb{F}_{q^\eta}$

Extension of degree η

$\Rightarrow$ *False positive probability:* $p_1 = \dfrac{1}{q^\eta}$

$\dfrac{n(n+1)}{2}$ multiplications $\longrightarrow$ $n\eta$ multiplications

(e.g. 946) (e.g. 172)

Step 1: batching MQ equations

- *Goal*: prove that $x = (x_1, \ldots, x_n)$ satisfies

$$\forall i \in [1 : m], \quad y_i - x^T A_i \, x - b_i^T = 0$$

- *Batched check*: prove that $x = (x_1, \ldots, x_n)$ satisfies

$$\sum_{i=1}^{m} \gamma_i \left(y_i - x^T A_i \, x - b_i^T \right) = 0$$

where $\gamma_1, \ldots, \gamma_m$ are uniformly at random in $\mathbb{F}_{q^\eta}$

Extension of degree η

$\Rightarrow$ *False positive probability:* $p_1 = \dfrac{1}{q^\eta}$

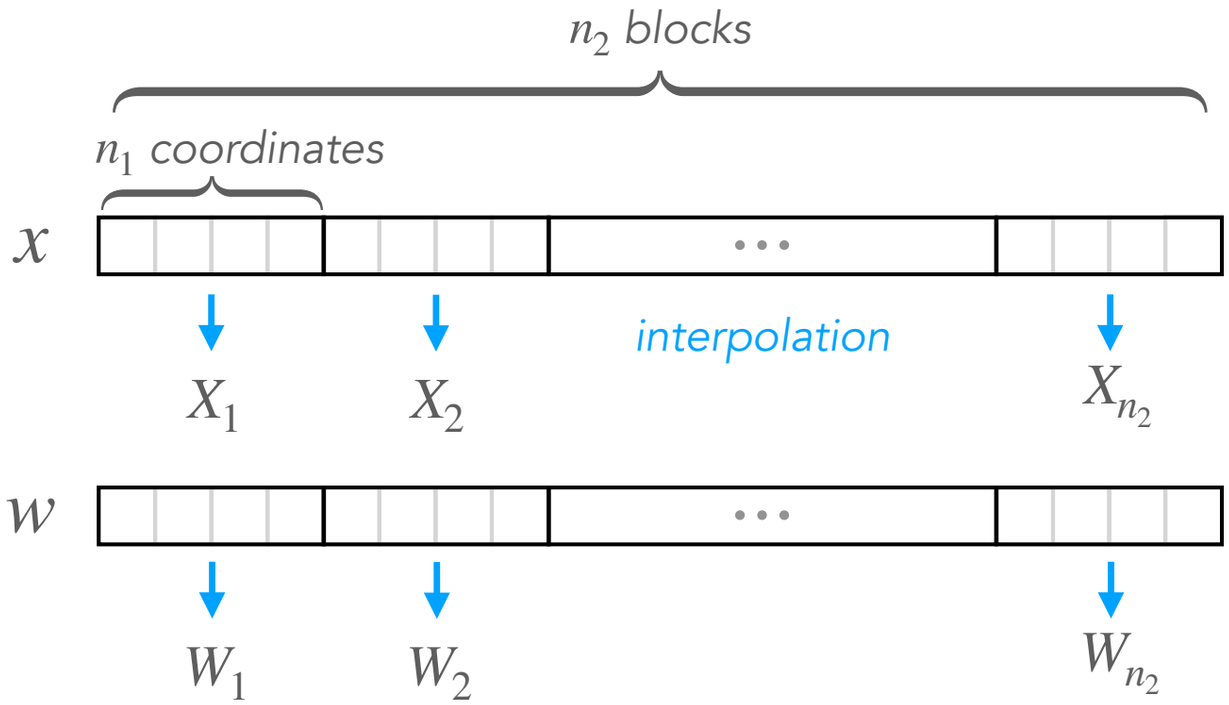
- Rewrite as $\langle x, w \rangle = z$

$$\left.\begin{aligned} z &:= \sum_{i=1}^{m} \gamma_i (y_i - b_i^T x) \\ w &:= \left(\sum_{i=1}^{m} \gamma_i A_i \right) x \end{aligned}\right\}$$

Linear (affine) functions of x

Step 2: inner product check

- *Goal*: prove that (x, w, z) satisfies $\langle x, w \rangle = z$



$$\begin{cases} X_1(f_1) & = x_1 \\ & \vdots \\ X_1(f_{n_1}) & = x_{n_1} \end{cases}$$

$$\begin{cases} X_2(f_1) & = x_{n_1+1} \\ & \vdots \\ X_2(f_{n_1}) & = x_{2n_1} \end{cases}$$

$$\vdots$$

Same for $W_1, W_2, \ldots$

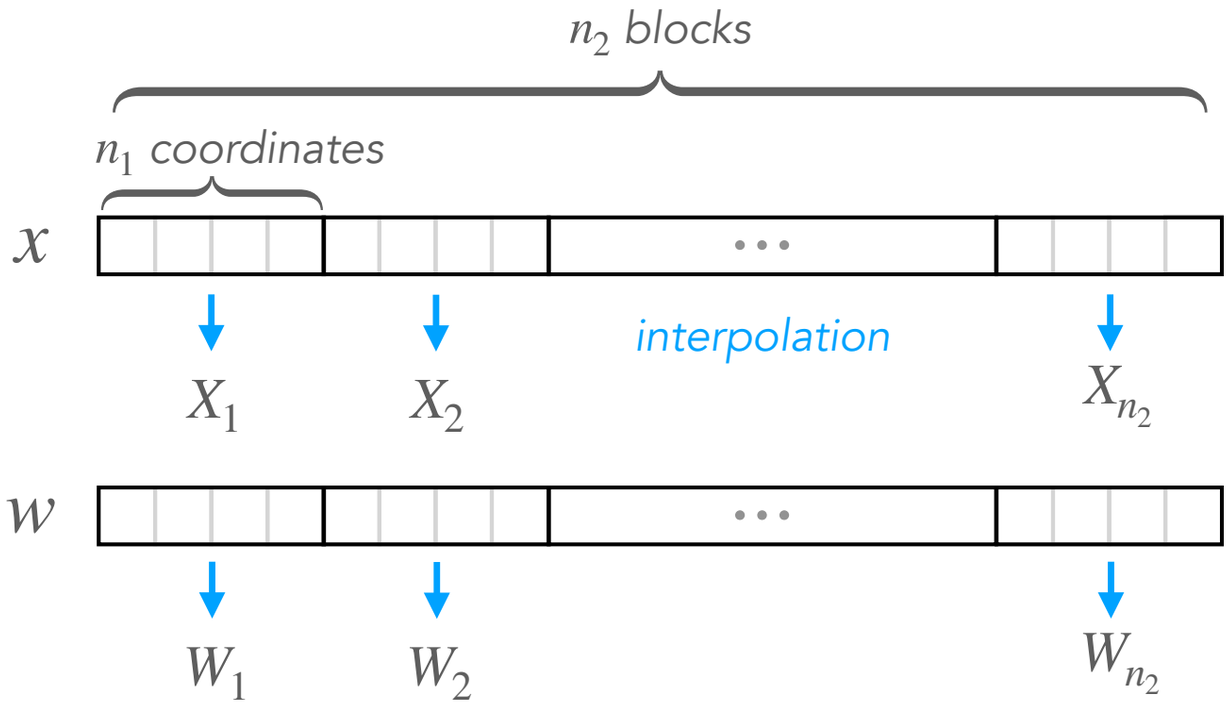
We have

$$z = \langle x, w \rangle = \sum_{i=0}^{n_1-1} \sum_{j=0}^{n_2-1} x_{i \cdot n_2 + j + 1} \cdot w_{i \cdot n_2 + j + 1}$$

$$= \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} X_j(f_i) W_j(f_i)$$

Step 2: inner product check

- *Goal*: prove that (x, w, z) satisfies $\langle x, w \rangle = z$



$$\begin{cases} X_1(f_1) & = x_1 \\ & \vdots \\ X_1(f_{n_1}) & = x_{n_1} \end{cases}$$

$$\begin{cases} X_2(f_1) & = x_{n_1+1} \\ & \vdots \\ X_2(f_{n_1}) & = x_{2n_1} \end{cases}$$

$\vdots$

Same for $W_1, W_2, \ldots$

We have $z = \langle x, w \rangle = \sum_{i=1}^{n_1} Q_0(f_i)$ with $Q_0 = \sum_{j=1}^{n_2} X_j W_j$

Step 2: inner product check

- ~~Goal: prove that (x, w, z) satisfies $\langle x, w \rangle = z$~~

  Goal: prove that (x, w, z) and Q_0 satisfy

$$z = \sum_{i=1}^{n_1} Q_0(f_i)$$

$$Q_0 = \sum_{j=1}^{n_2} X_j W_j$$

Step 2: inner product check

- ~~Goal: prove that (x, w, z) satisfies $\langle x, w \rangle = z$~~

  Goal: prove that (x, w, z) and Q_0 satisfy

$$z = \sum_{i=1}^{n_1} Q_0(f_i)$$

Linear relation

$$Q_0 = \sum_{j=1}^{n_2} X_j W_j$$

Check the relation into a random point r

$$Q_0(r) = \sum_{j=1}^{n_2} X_j(r) \cdot W_j(r)$$

⇒ *False positive probability:* $p_2 = \dfrac{2n_1 - 1}{q^\eta - n_1}$

Step 2: inner product check

- ~~Goal: prove that (x, w, z) satisfies $\langle x, w \rangle = z$~~

  Goal: prove that (x, w, z) and Q_0 satisfy

$$z = \sum_{i=1}^{n_1} Q_0(f_i)$$

Linear relation

$$Q_0 = \sum_{j=1}^{n_2} X_j W_j$$

Check the relation into a random point r

$n\eta$ multiplications (e.g. 172)

$\longrightarrow n_2\eta$ multiplications (e.g. 44)

$$Q_0(r) = \sum_{j=1}^{n_2} X_j(r) \cdot W_j(r)$$

$\Rightarrow$ *False positive probability:* $p_2 = \dfrac{2n_1 - 1}{q^\eta - n_1}$

MQOM: Summary

- Checking a MQ solution can be efficiently expressed as linear equations.

- Using MPCitH paradigm, we obtain an efficient identification scheme.

- Using the Fiat-Shamir transformation, we obtain a signature scheme.

Implementation

- *MQOM*: A candidate to the NIST call for additional post-quantum signatures.

  https://csrc.nist.gov/projects/pqc-dig-sig/round-1-additional-signatures

- Website MQOM, with specification:

  https://mqom.org/

- Two open-source source codes
  - **Reference code**: generic C implementation
  - **Optimized for Intel processor**: implementation which has been optimized using AVX2 instruction set.

Performances

MQOM Variants	NIST Security		MQ Parameters		MPC Parameters					Sig. size (Bytes)		Sig. perf.		Verif. perf.	
	Category	Bits	q	$m = n$	$N = 2^D$	n_1	n_2	η	τ	Avg.	Max.	Time (ms)	Cycles (Mc)	Time (ms)	Cycles (Mc)
MQOM-L1-gf31-short	I	143	31	49	256	5	10	10	20	6348	6352	11.7	44.3	11.0	41.7
MQOM-L1-gf31-fast	I	143	31	49	32	5	10	6	35	7621	7657	4.6	17.6	4.1	15.5
MQOM-L1-gf251-short	I	143	251	43	256	4	11	5	22	6575	6578	7.5	28.5	7.2	27.3
MQOM-L1-gf251-fast	I	143	251	43	32	4	11	4	34	7809	7850	3.0	11.5	2.7	10.2
MQOM-L3-gf31-short	III	207	31	77	256	6	13	11	30	13837	13846	28.5	108.1	27	102.2
MQOM-L3-gf31-fast	III	207	31	77	32	6	13	7	51	16590	16669	14.8	56.3	13.5	51.2
MQOM-L3-gf251-short	III	207	251	68	256	5	14	7	30	14257	14266	18.3	69.5	17.3	65.5
MQOM-L3-gf251-fast	III	207	251	68	32	5	14	4	52	17161	17252	8.6	32.8	7.8	29.6
MQOM-L5-gf31-short	V	272	31	106	256	6	18	10	42	24147	24158	59.2	224.4	56.3	213.6
MQOM-L5-gf31-fast	V	272	31	106	32	6	18	8	66	28917	29036	41.2	156.2	38.5	146.2
MQOM-L5-gf251-short	V	272	251	93	256	6	16	7	41	24926	24942	39.0	148.0	37.5	142.2
MQOM-L5-gf251-fast	V	272	251	93	32	6	16	5	66	29919	30092	21.5	81.5	19.9	75.6

- Sig sizes:
  - Cat I (128-bit): 6.3 - 7.8 KB
  - Cat III (192-bit): 14 - 17 KB
  - Cat V (256-bit): 24 - 30 KB
- Key sizes:
  - Cat I (128-bit): $|pk|, |sk| \leq 100$ B
  - Cat III (192-bit): $|pk|, |sk| \leq 160$ B
  - Cat V (256-bit): $|pk|, |sk| \leq 220$ B
- Timings: one to few dozen Mc (megacycles)

Comparison (multivariate crypto)

Multivariate signatures that are NIST candidates to the second call

	Signature Size	***Public Key Size***	***Security Assumption***
3WISE	32 B	187 000 B	MQ with **hidden** structure
MAYO	321 B	1 168 B	MQ with **hidden** structure
PROV	160 B	68 326 B	MQ with **hidden** structure
QR-UOV	157 B	23 657 B	MQ with **hidden** structure
SNOVA	248 B	1 016 B	MQ with **hidden** structure
TUOV	112 B	42 608 B	MQ with **hidden** structure
UOV	128 B	43 576 B	MQ with **hidden** structure
VOX	102 B	9 104 B	MQ with **hidden** structure
HPPC	21 B	129 000 B	MQ with **hidden** structure
Biscuit	4 758 B	50 B	MQ with **public** structure
MQOM	6 352 B	47 B	MQ with **no structure**

What Next?

- Recent works in MPC-in-the-Head (MPCitH):
  - Threshold-based MPCitH [BBDK+23,FR23]:

    Framework to prove **polynomial relations** over secret values
  - Recent works to optimize the efficiency of the MPCitH paradigm [BCD23,BBMO+24]

[BBDK+23] Baum, Braun, Delpech, Klooß, Orsini, Roy, Scholl. *Publicly Verifiable Zero-Knowledge and Post-Quantum Signatures From VOLE-in-the-Head*. Crypto 2023

[FR23] Feneuil, Rivain. *Threshold Computation in the Head: Improved Framework for Post-Quantum Signatures and Zero-Knowledge Arguments. ePrint 2023/1573*

[BCD23] Bui, Cong, Delpech. *Improved All-but-One Vector Commitment with Applications to Post-Quantum Signature. ePrint 2024/097*

[BBMO+24] Baum, Beullens, Mukherjee, Orsini, Ramacher, Rechberger, Roy, Scholl. *One Tree to Rule Them All: Optimizing GGM Trees and OWFs for Post-Quantum Signatures. ePrint 2024/490*

What Next?

- Recent works in MPC-in-the-Head (MPCitH):
  - Threshold-based MPCitH [BBDK+23,FR23]:

    Framework to prove **polynomial relations** over secret values
  - Recent works to optimize the efficiency of the MPCitH paradigm [BCD23,BBMO+24]

Simpler, shorter, faster! 🥳

What Next?

- Recent works in MPC-in-the-Head (MPCitH):
  - Threshold-based MPCitH [BBDK+23,FR23]:

    Framework to prove **polynomial relations** over secret values
  - Recent works to optimize the efficiency of the MPCitH paradigm [BCD23,BBMO+24]

Simpler, shorter, faster! 🥳

Towards MQOMv2:

Signature size: around 2.6-3.6 KB, instead of 6.3-7.8 KB

Timings: around 0.5-6.0 ms, instead of 3.0-11.0 ms

What Next?

- Recent works in MPC-in-the-Head (MPCitH):
  - Threshold-based MPCitH [BBDK+23,FR23]:

    Framework to prove **polynomial relations** over secret values
  - Recent works to optimize the efficiency of the MPCitH paradigm [BCD23,BBMO+24]

Simpler, shorter, faster! 🥳

Towards MQOMv2:

Signature size: around 2.6-3.6 KB, instead of 6.3-7.8 KB

Timings: around 0.5-6.0 ms, instead of 3.0-11.0 ms

Thank you for your attention.

MQOM: Parameter Selection

$\lambda \in \{128,192,256\}$: security level (in bits)

MQ parameters:

q: field size

n: numbers of variables

m: number of equations

MPCitH parameters:

N: number of parties

τ: numbers of repetitions

η, n_1, n_2: proof parameters

MQOM: Parameter Selection

$\lambda \in \{128,192,256\}$: security level (in bits)

MQ parameters:

q: field size

n: numbers of variables

m: number of equations

MPCitH parameters:

N: number of parties

τ: numbers of repetitions

η, n_1, n_2: proof parameters

We take $n = m$, since it corresponds to the harder MQ instances.
We choose n and m such that solving the MQ problem takes respectively 2^{143}, 2^{207} and 2^{272} bit operations.

MQOM: Parameter Selection

$\lambda \in \{128,192,256\}$: security level (in bits)

MQ parameters:

q: field size

n: numbers of variables

m: number of equations

MPCitH parameters:

N: number of parties

τ: numbers of repetitions

η, n_1, n_2: proof parameters

We take N, τ and η such that forging a signature without the secret key takes respectively 2^{128}, 2^{192} and 2^{256} hash operations, while minimizing the signature size.

Comparison (unstructured MQ)

	Security	*Signature Size*	*Public Key Size*	*Running Times*
MQ-DSS	141	28 400 B	46 B	≈ 3-5 Mc
MudFish	149	14 400 B	38 B	≈ 15 Mc
Mesquite - Fast	149	9 492 B	38 B	≈ 12-15 Mc
Mesquite - Compact	149	8 844 B	38 B	≈ 24-31 Mc
[Fen22] - gf251 - Fast	135	8 488 B	56 B	≈ 8 Mc
[Fen22] - gf251 - Short	135	7 114 B	56 B	≈ 23 Mc
MQOM - gf251 - Fast	144	7 809 B	59 B	≈ 11 Mc
MQOM - gf251 - Short	144	6 575 B	59 B	≈ 28 Mc
MQOM - gf31 - Fast	143	7 621 B	47 B	≈ 17 Mc
MQOM - gf31 - Short	143	6 348 B	47 B	≈ 44 Mc

Comparison (MPCitH)

	Signature	*Public Key*	*Running Time*	*Security Assumption*
Picnic3	13 802 B	46 B	≈ 3-5 Mc	LowMC cipher
AIMer	4 176 B	32 B	≈ 15 Mc	AIM one-way function
Biscuit	4 758 B	50 B	≈ 12-15 Mc	Structured MQ
FAEST	5 006 B	32 B	≈ 12-15 Mc	AES cipher
MIRA	5 640 B	84 B	≈ 24-31 Mc	MinRank
MiRitH	5 673 B	38 B	≈ 8 Mc	MinRank
PERK	6 060 B	240 B	≈ 23 Mc	Permuted Kernel
RYDE	5 956 B	86 B	≈ 11 Mc	Rank Syndrome Decoding
SDitH	8 260 B	120 B	≈ 28 Mc	Syndrome Decoding
MQOM - gf251 - Short	6 575 B	59 B	≈ 28 Mc	Non-structured MQ
MQOM - gf31 - Short	6 348 B	47 B	≈ 44 Mc	Non-structured MQ