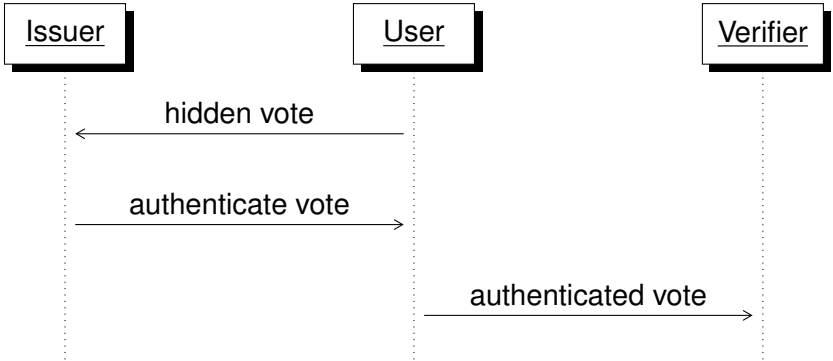

DTU

Carsten Baum, **Marvin Beckmann**, Ward Beullens, Shibam Mukherjee, Christian Rechberger

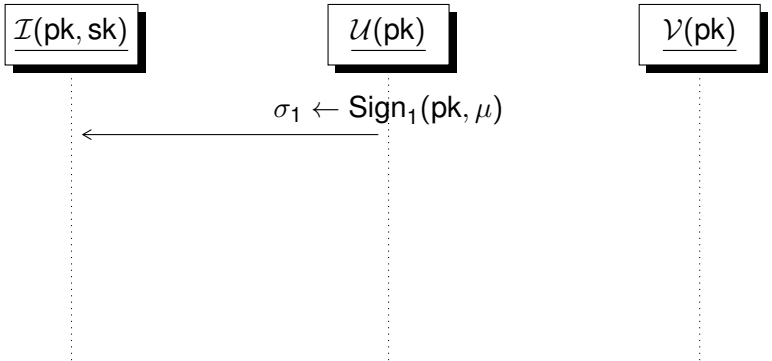
Blind Signatures Based on VOLEitH and MAYO

Example: E-Voting with Blind Signatures

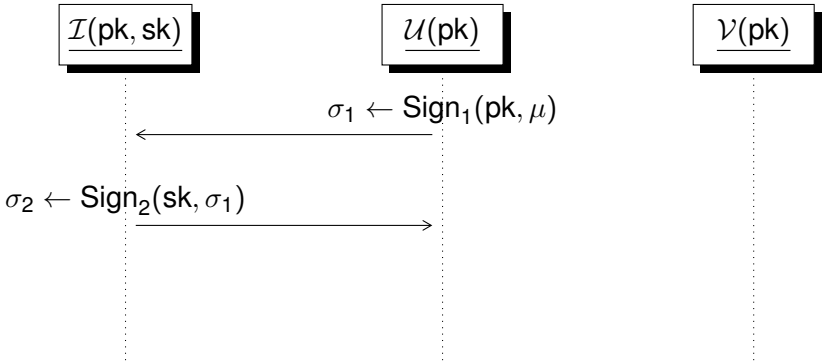


DTU

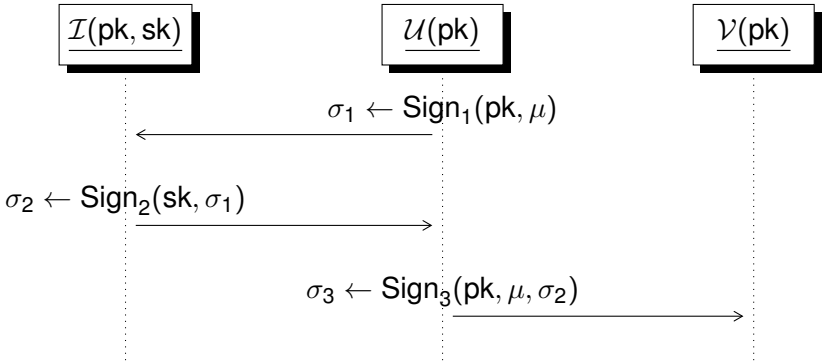
Blind Signatures



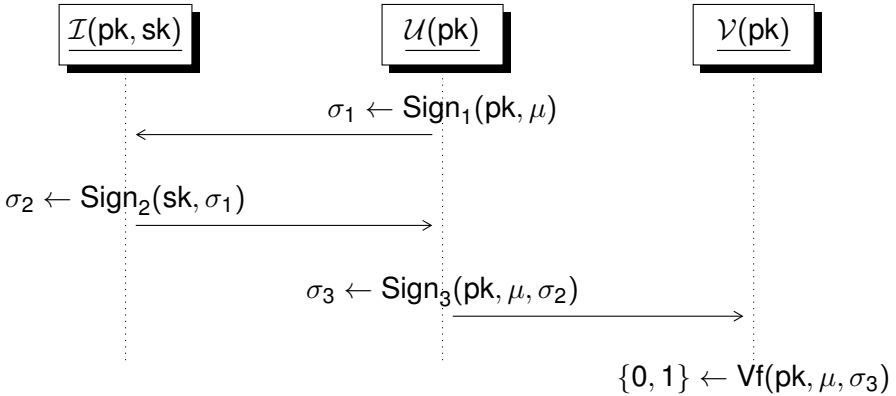
Blind Signatures



Blind Signatures

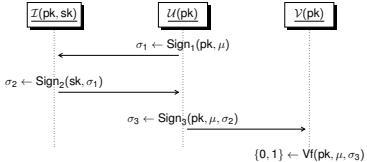


Blind Signatures



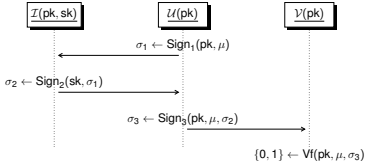
Blindness Property of Blind Signatures

- A cooperating issuer and verifier can associate the first message with the final signature



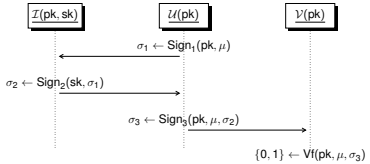
Blindness Property of Blind Signatures

- A cooperating issuer and verifier can associate the first message with the final signature
- $\mathcal{A}$ takes the role of $\mathcal{I}$ and $\mathcal{V}$ and for m_0, m_1, and $\mathcal{A}$ can not associate $\sigma_{1,m_0/m_1}$ with respective final signatures $\sigma_{3,m_0/m_1}$



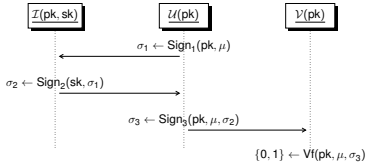
Fischlin's Blueprint for Blind Signatures

- $(\mathsf{pk}, \mathsf{sk})$ are from a normal signature scheme



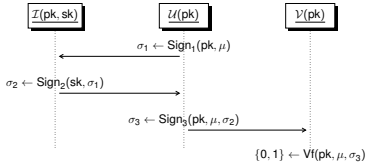
Fischlin's Blueprint for Blind Signatures

- $(\mathsf{pk}, \mathsf{sk})$ are from a normal signature scheme
- $\sigma_1 \leftarrow \mathit{Com}(\mu, r)$ where $r \leftarrow \{0,1\}^{1^\lambda}$



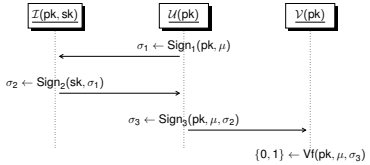
Fischlin's Blueprint for Blind Signatures

- (pk, sk) are from a normal signature scheme
- $\sigma_1 \leftarrow \mathit{Com}(\mu, r)$ where $r \leftarrow \{0,1\}^{1^\lambda}$
- $\sigma_2 \leftarrow \mathsf{Sign}(\mathsf{sk}, \sigma_1)$



Fischlin's Blueprint for Blind Signatures

- (pk, sk) are from a normal signature scheme
- $\sigma_1 \leftarrow \mathit{Com}(\mu, r)$ where $r \leftarrow \{0,1\}^{1^\lambda}$
- $\sigma_2 \leftarrow \mathsf{Sign}(\mathsf{sk}, \sigma_1)$
- $\sigma_3 \leftarrow (\mu, z)$ where $z \leftarrow \mathit{ZKP}(\mathsf{pk}, \sigma_2, \mu, r)$



High-Level Description of our Blind Signature

For a fixed μ, and fixed MAYO keys $(\mathsf{pk}, \mathsf{sk})$

1 : $r \leftarrow\!\!\$\ \{0,1\}^{\lambda}$

2 : $c \leftarrow \text{SHAKE256}(r \| \mu)$

3 : $salt \leftarrow\!\!\$\ \{0,1\}^{\lambda}$

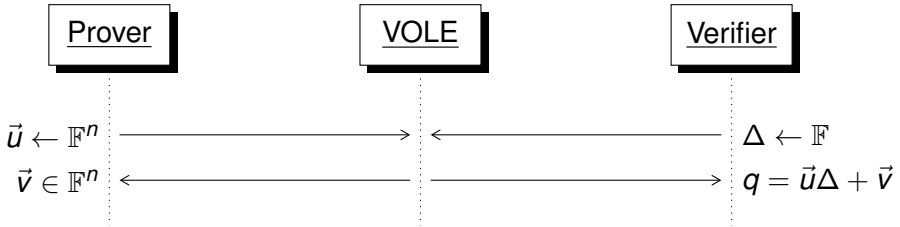
4 : $t \leftarrow \text{SHAKE256}(salt \| c)$

5 : $\sigma \leftarrow \text{MAYO.SamplePre}(\mathsf{sk}, \mathsf{pk}, t)$

6 : $\text{VOLEitH-ZKP}(\mu, r, salt, \sigma)$

Vector Oblivious Linear Evaluation (VOLE)

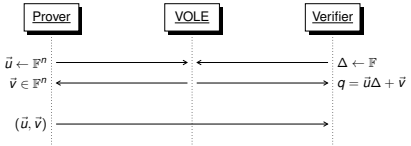
Vector Oblivious Linear Evaluation (VOLE)



▷ The verifier evaluates the polynomial $p(x) = \vec{u}x + \vec{v}$ at position Δ.

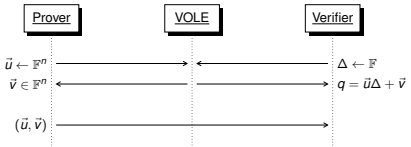
View VOLE as a Commitment Scheme to $\vec{u}$

- Prover can send $(\vec{u}, \vec{v})$ to open



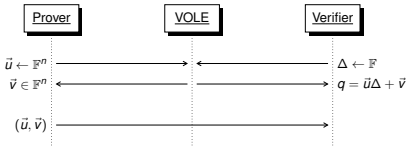
View VOLE as a Commitment Scheme to $\vec{u}$

- Prover can send $(\vec{u}, \vec{v})$ to open
- Verifier checks $q = \vec{u}\Delta + \vec{v}$



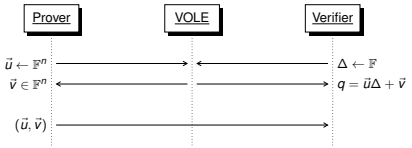
View VOLE as a Commitment Scheme to $\vec{u}$

- Prover can send $(\vec{u}, \vec{v})$ to open
- Verifier checks $q = \vec{u}\Delta + \vec{v}$
- *Hiding*: $\vec{v}$ is random



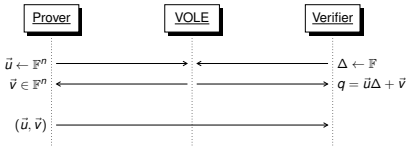
View VOLE as a Commitment Scheme to $\vec{u}$

- Prover can send $(\vec{u}, \vec{v})$ to open
- Verifier checks $q = \vec{u}\Delta + \vec{v}$
- *Hiding*: $\vec{v}$ is random
- *Binding*: opening to $u' \neq u$ requires guessing Δ $(u'\Delta + \vec{v}' = u\Delta + \vec{v} = q \Rightarrow \Delta = (\vec{v} - \vec{v}')/(\vec{u}' - \vec{u}))$

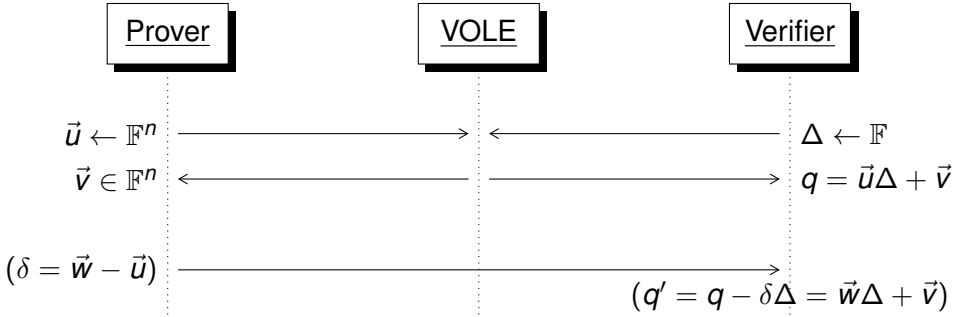


View VOLE as a Commitment Scheme to $\vec{u}$

- Prover can send $(\vec{u}, \vec{v})$ to open
- Verifier checks $q = \vec{u}\Delta + \vec{v}$
- *Hiding*: $\vec{v}$ is random
- *Binding*: opening to $u' \neq u$ requires guessing Δ $(u'\Delta + \vec{v}' = u\Delta + \vec{v} = q \Rightarrow \Delta = (\vec{v} - \vec{v}')/(\vec{u}' - \vec{u}))$
- Evaluable as a circuit with n multiplications and n additions

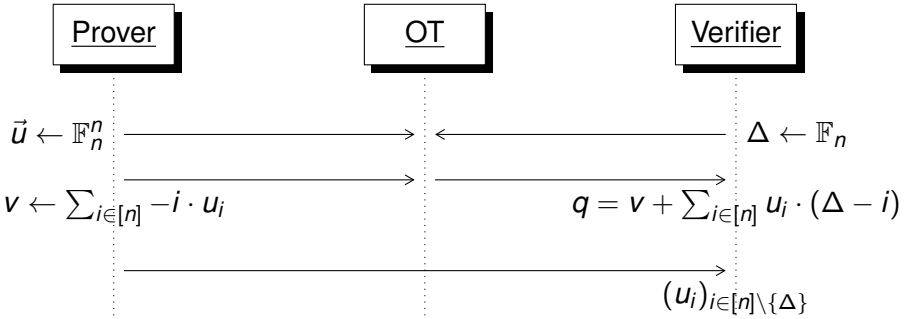


Cast Commitment Scheme to a Witness $\vec{w}$



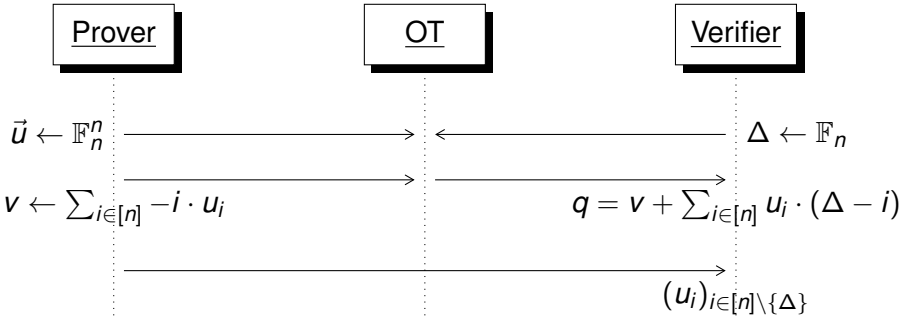
Casting VOLE to get Public Verifiability

- Generate VOLE from a n-out-of-n secret sharing scheme and OT
- Δ now only determines which $n-1$ challenges need to be opened



Casting VOLE to get Public Verifiability

- Generate VOLE from a n-out-of-n secret sharing scheme and OT
- Δ now only determines which $n-1$ challenges need to be opened
- Derive Δ using hash of commitments and cast in Fiat-Shamir style



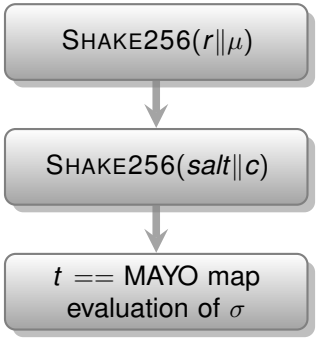
Properties of VOLEitH as a Commit&Proof Framework

- linear: adding two commitments gives commitment to sum of witnesses
- multiplication: multiplying gives you a degree increase
- degree decrease: at the cost of n additional commitments
- zero-check

Our ZKP using VOLEitH

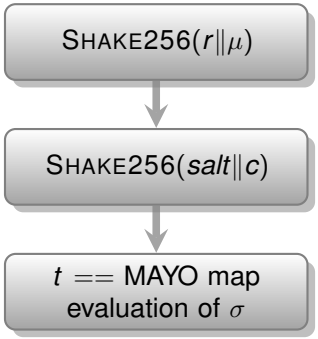
ZKP Strategy

1. bit-commit to $r \| \mu$
2. SHAKE256 with bit-commit and VOLEitH properties
3. SHAKE256 with previous result and bit-commitment to salt
4. $\mathbb{F}_{16}$ commitments of σ
5. MAYO map evaluation with VOLEitH properties
6. Comparison using 0-check

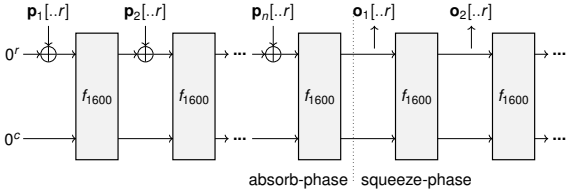


ZKP Strategy

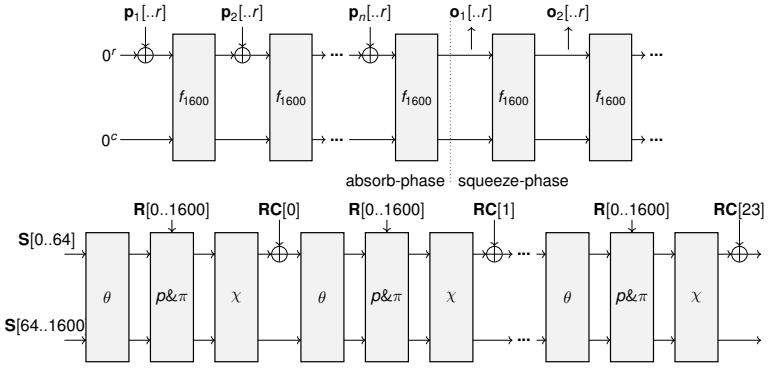
1. bit-commit to $r \| \mu$
2. SHAKE256 with bit-commit and VOLEitH properties
3. SHAKE256 with previous result and bit-commitment to salt
4. $\mathbb{F}_{16}$ commitments of σ
5. MAYO map evaluation with VOLEitH properties
6. Comparison using 0-check



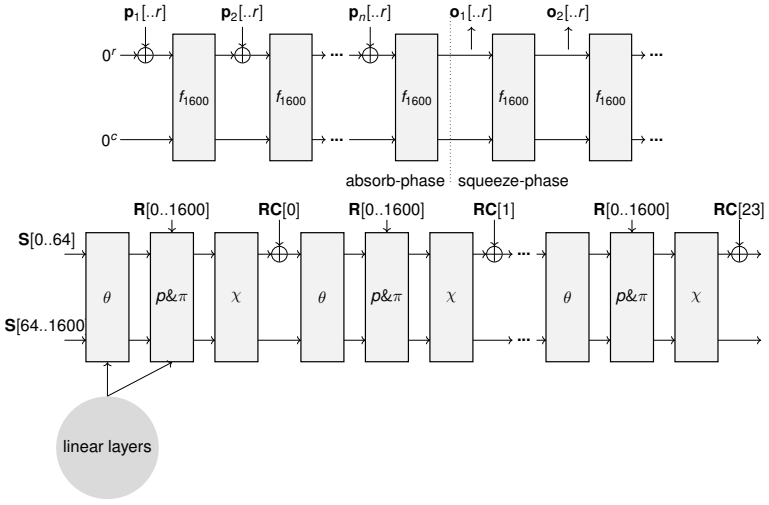
SHAKE256


$\mathbf{p}_1[..r]$
$\mathbf{p}_2[..r]$
$\mathbf{p}_n[..r]$
$\mathbf{o}_1[..r]$
$\mathbf{o}_2[..r]$
0^r
0^c
f_{1600}
absorb-phase
squeeze-phase

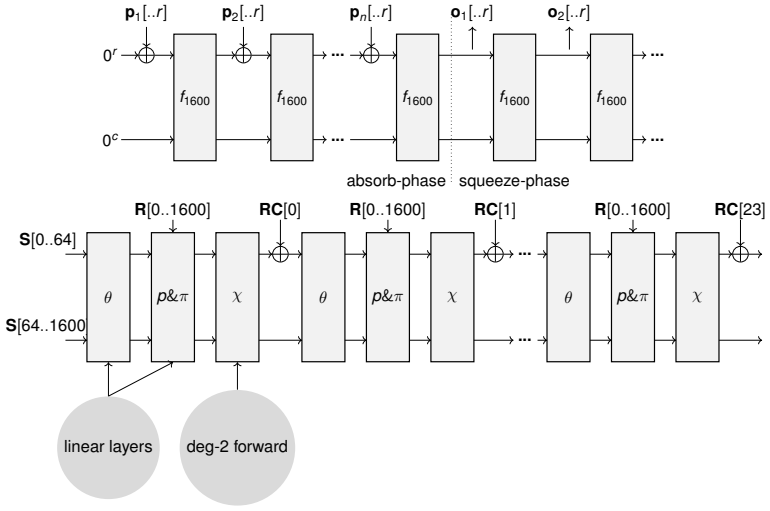
SHAKE256


$\mathbf{p}_1[..r]$
$\mathbf{p}_2[..r]$
$\mathbf{p}_n[..r]$
$\mathbf{o}_1[..r]$
$\mathbf{o}_2[..r]$
0^r
0^c
f_{1600}
absorb-phase
squeeze-phase
$\mathbf{R}[0..1600]$
$\mathbf{RC}[0]$
$\mathbf{RC}[1]$
$\mathbf{RC}[23]$
$\mathbf{S}[0..64]$
$\mathbf{S}[64..1600]$
θ
$p\&\pi$
χ

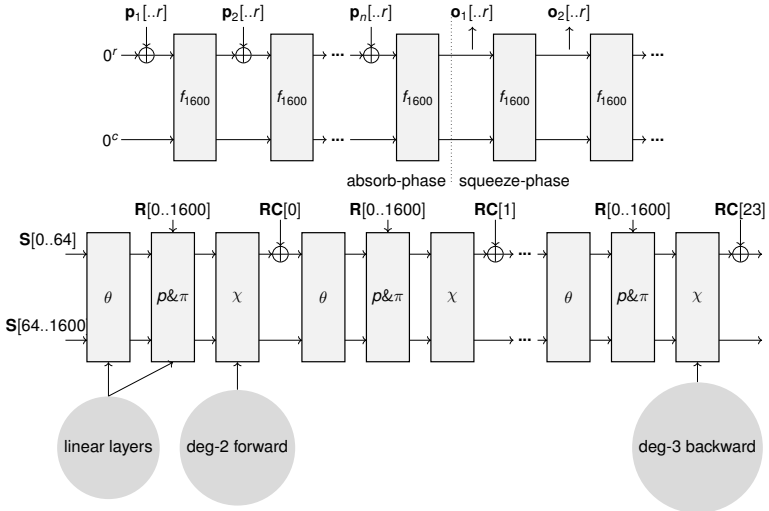
SHAKE256


p1[..r]
p2[..r]
pn[..r]
o1[..r]
o2[..r]
0^r
0^c
f1600
absorb-phase
squeeze-phase
R[0..1600]
RC[0]
RC[1]
RC[23]
S[0..64]
S[64..1600]
θ
p&π
χ
linear layers

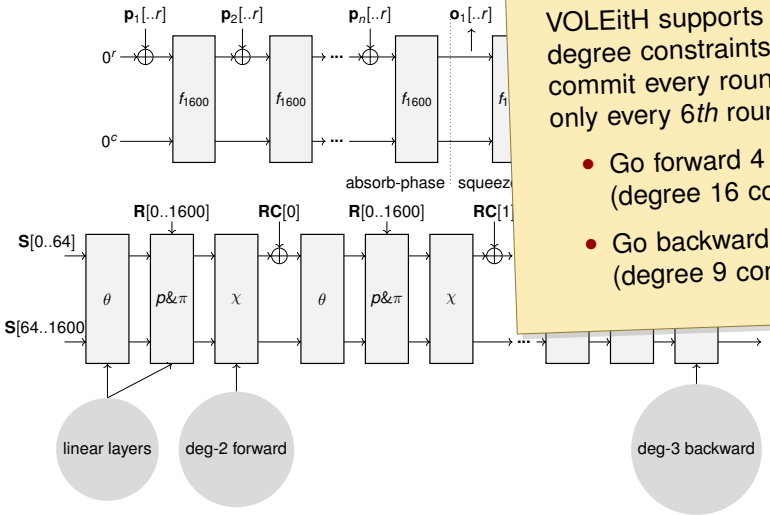
SHAKE256


$\mathbf{p}_1[..r]$
$\mathbf{p}_2[..r]$
$\mathbf{p}_n[..r]$
$\mathbf{o}_1[..r]$
$\mathbf{o}_2[..r]$
0^r
0^c
f_{1600}
absorb-phase
squeeze-phase
$\mathbf{R}[0..1600]$
$\mathbf{RC}[0]$
$\mathbf{RC}[1]$
$\mathbf{RC}[23]$
$\mathbf{S}[0..64]$
$\mathbf{S}[64..1600]$
θ
$p\&\pi$
χ
linear layers
deg-2 forward

SHAKE256


$\mathbf{p}_1[..r]$
$\mathbf{p}_2[..r]$
$\mathbf{p}_n[..r]$
$\mathbf{o}_1[..r]$
$\mathbf{o}_2[..r]$
0^r
0^c
f_{1600}
absorb-phase
squeeze-phase
$\mathbf{R}[0..1600]$
$\mathbf{RC}[0]$
$\mathbf{RC}[1]$
$\mathbf{RC}[23]$
$\mathbf{S}[0..64]$
$\mathbf{S}[64..1600]$
θ
$p\&\pi$
χ
linear layers
deg-2 forward
deg-3 backward

SHAKE256



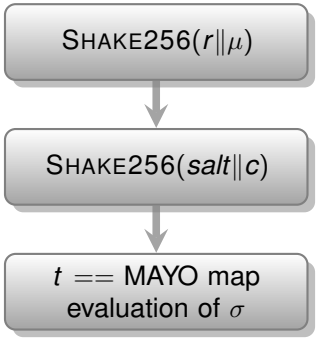
Optimization Strategy:

VOLEitH supports higher-degree constraints: Do not commit every round, but only every 6*th* round.

- Go forward 4 steps (degree 16 constraints)
- Go backward 2 steps (degree 9 constraints)

ZKP Strategy

1. bit-commit to $r \| \mu$
2. SHAKE256 with bit-commit and VOLEitH properties
3. SHAKE256 with previous result and bit-commitment to salt
4. $\mathbb{F}_{16}$ commitments of σ
5. MAYO map evaluation with VOLEitH properties
6. Comparison using 0-check



MAYO Map Evaluation

- There are n public matrices $\boldsymbol{M}_1, ..., \boldsymbol{M}_n \in \mathbb{F}_q^{m \times m}$
- The signature $\vec{s}$ is generated such that

$$\forall i \in [n] : v_i = \vec{s}^\top \boldsymbol{M}_i \vec{s}$$

where $H(\mu) = \vec{v} \in \mathbb{F}_q^n$

▷ Verifying the signature is only a degree-2 operation on the signature $\vec{s}$

Competitiveness

Scheme	Assumptions	Signing (in KB)	Showing (in KB)	Runtime Showing (in ms)	Security (in bits)
Agrawal et al. [AKSY22]	One-More-ISIS	1.4	45	–	109
del Pino[a] & Katsumata [dPK22]	MSIS/MLWE/NTRU	851	102.6	–	128
Lyubashevsky[b] et al. [LNP22]	MSIS/MLWE	$\gg 1000$	150	–	128
Bootle et al. [BLNS23b]	ISIS-f/NTRU	> 100	> 100	–	128
Policharla et al. [PWFW23]	MSIS/MLWE+Poseidon	2.4	$85 - 173$	$304 - 4822$	115
Lyubashevsky[c] et al. [LSS24]	ISIS-f/NTRU	1.3	29	< 200	128
Argo et al. [AGJ+24] (implementing [JRS23])	MSIS/MLWE	45	79.6	< 500	128
Beullens[d] et al. [BLNS23a]	MSIS/MLWE/NTRU	60	22	–	128
Jeudy and Sanders [JS24]	MSIS/MLWE	60	41	–	128
Baldimtsi[e] et al. [BCGY24]	One-More-ISIS	1	68	–	128
SHAKE256-deg16+MAYO-$128_{s/f}$	UOV/WMQ	0.486	24.3/41.6	178.1/75.6	128
RainHash+MAYO-$128_{s/f}$	UOV/WMQ	0.486	16.5/27.8	114.4/49.8	128
One-More-MAYO-$128_{s/f}$	UOV/One-More-WMQ	0.469	6.7/10.4	42/40	128

[a] May be improved upon using more recent lattice NIZKs. [b] Bounded number of signatures, 1.3MB public key. [c] Implements a version of [BLNS23b].
[d] Uses a generic NIZK scheme. [e] Non-interactive construction with pseudorandom messages.

Competitiveness

2 additional constructions:

RainHash+MAYO-128$_{s/f}$ Uses RAINHASH instead of SHAKE256, which is friendlier for ZKPs.

One-More-MAYO-128$_{s/f}$ Uses stronger assumption on underlying signature, but does not have any hash evaluation in the blind signature

Scheme					
Agrawal et al. [AKSY22] del Pino[a] & Katsumata [dPK22] Lyubashevsky[b] et al. [LNP22] Bootle et al. [BLNS23b] Policharla et al. [PWFW23] Lyubashevsky[c] et al. [LSS24] Argo et al. [AGJ+24] (implementing [JRS23]) Beullens[d] et al. [BLNS23a] Jeudy and Sanders [JS24] Baldimtsi[e] et al. [BCGY24]					
SHAKE256-deg16+MAYO-128$_{s/f}$	UOV/WMQ	0.486	24.3/41.6	178.1/75.6	128
RainHash+MAYO-128$_{s/f}$	UOV/WMQ	0.486	16.5/27.8	114.4/49.8	128
One-More-MAYO-128$_{s/f}$	UOV/One-More-WMQ	0.469	6.7/10.4	42/40	128

[a] May be improved upon using more recent lattice NIZKs. [b] Bounded number of signatures, 1.3MB public key. [c] Implements a version of [BLNS23b].
[d] Uses a generic NIZK scheme. [e] Non-interactive construction with pseudorandom messages.

Competitiveness

Scheme	Assumptions	Signing (in KB)	Showing (in KB)	Runtime Showing (in ms)	Security (in bits)
Agrawal et al. [AKSY22]	One-More-ISIS	1.4	45	–	109
del Pino[a] & Katsumata [dPK22]	MSIS/MLWE/NTRU	851	102.6	–	128
Lyubashevsky[b] et al. [LNP22]	MSIS/MLWE	$\gg 1000$	150	–	128
Bootle et al. [BLNS23b]	ISIS-f/NTRU	> 100	> 100	–	128
Policharla et al. [PWFW23]	MSIS/MLWE+Poseidon	2.4	$85 - 173$	$304 - 4822$	115
Lyubashevsky[c]et al. [LSS24]	ISIS-f/NTRU	1.3	29	< 200	128
Argo et al. [AGJ+24] (implementing [JRS23])	MSIS/MLWE	45	79.6	< 500	128
Beullens[d] et al. [BLNS23a]	MSIS/MLWE/NTRU	60	22	–	128
Jeudy and Sanders [JS24]	MSIS/MLWE	60	41	–	128
Baldimtsi[e] et al. [BCGY24]	One-More-ISIS	1	68	–	128
SHAKE256-deg16+MAYO-$128_{s/f}$	UOV/WMQ	0.486	24.3/41.6	178.1/75.6	128
RainHash+MAYO-$128_{s/f}$	UOV/WMQ	0.486	16.5/27.8	114.4/49.8	128
One-More-MAYO-$128_{s/f}$	UOV/One-More-WMQ	0.469	6.7/10.4	42/40	128

[a] May be improved upon using more recent lattice NIZKs. [b] Bounded number of signatures, 1.3MB public key. [c] Implements a version of [BLNS23b].
[d] Uses a generic NIZK scheme. [e] Non-interactive construction with pseudorandom messages.

Thanks for listening! Paper on eprint: 2026/109

References I

- Sven Argo, Tim Güneysu, Corentin Jeudy, Georg Land, Adeline Roux-Langlois, and Olivier Sanders.
  Practical post-quantum signatures for privacy.
  pages 1523–1537, 2024.
- Shweta Agrawal, Elena Kirshanova, Damien Stehlé, and Anshu Yadav.
  Practical, round-optimal lattice-based blind signatures.
  pages 39–53, 2022.
- Foteini Baldimtsi, Jiaqi Cheng, Rishab Goyal, and Aayush Yadav.
  Non-interactive blind signatures: Post-quantum and stronger security.
  pages 70–104, 2024.

References II

- Ward Beullens, Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler.
  Lattice-based blind signatures: Short, efficient, and round-optimal.
  pages 16–29, 2023.
- Jonathan Bootle, Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Alessandro Sorniotti.
  A framework for practical anonymous credentials from lattices.
  pages 384–417, 2023.
- Rafaël del Pino and Shuichi Katsumata.
  A new framework for more efficient round-optimal lattice-based (partially) blind signature via trapdoor sampling.
  pages 306–336, 2022.

References III


Corentin Jeudy, Adeline Roux-Langlois, and Olivier Sanders.
Lattice signature with efficient protocols, application to anonymous credentials.
pages 351–383, 2023.


Corentin Jeudy and Olivier Sanders.
Improved lattice blind signatures from recycled entropy.
Cryptology ePrint Archive, Report 2024/1289, 2024.


Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Maxime Plançon.
Efficient lattice-based blind signatures via gaussian one-time signatures.
pages 498–527, 2022.

References IV


Vadim Lyubashevsky, Gregor Seiler, and Patrick Steuer.
The LaZer library: Lattice-based zero knowledge and succinct proofs for quantum-safe privacy.
pages 3125–3137, 2024.


Guru-Vamsi Policharla, Bas Westerbaan, Armando Faz-Hernández, and Christopher A. Wood.
Post-quantum privacy pass via post-quantum anonymous credentials.
IACR Cryptol. ePrint Arch., page 414, 2023.

Extra Slides

Removing The Hash Calls

- Achieve the hiding not through a commitment scheme, but inherent randomness from the proof generation:

$$t = H(\mu, \pi_1) \oplus r$$

  where $(\pi_1, r) \leftarrow \mathcal{P}_1$. This only adds small overhead to the proof itself.
- Use a new assumption: One-More-MAYO, that allows querying the MAYO map for chosen targets

▷ Less conservative option, but it significantly improves runtime and size. (Primarily impacts for smaller security parameters)