Signature Scheme from Restricted Errors

Violetta Weger

CBCrypto 2023
International Workshop on Code-Based Cryptography

April 23, 2023

Marco Baldi, Sebastian Bitzer
Alessio Pavoni, Paolo Santini
Antonia Wachter-Zeh

Motivation

2016 NIST standardization call for post-quantum PKE/KEM and signatures

Motivation

2016 NIST standardization call for post-quantum PKE/KEM and signatures

- PKE/KEM: 1 lattice-based, round 4: 3 code-based
- Signature schemes: 1 hash-based and 2 based on ideal lattices

Motivation

2016 NIST standardization call for post-quantum PKE/KEM and signatures

- PKE/KEM: 1 lattice-based, round 4: 3 code-based
- Signature schemes: 1 hash-based and 2 based on ideal lattices

2022 NIST reopened standardization call for signature schemes

Idea of Signature Schemes

Signer | **Verifier**

Key Generation

Secret key $\mathcal{S}$, public key $\mathcal{P}$

$\xrightarrow{\mathcal{P}}$

Signing

Message m, signature σ

$\xrightarrow{m,\sigma}$

Verification

Verify σ

Idea of Signature Schemes

Signer		**Verifier**
Key Generation Secret key $\mathcal{S}$, public key $\mathcal{P}$	$\xrightarrow{\mathcal{P}}$	
Signing Message m, signature σ	$\xrightarrow{m,\sigma}$	**Verification** Verify σ

Two approaches to get a code-based signature scheme:

- Hash-and-sign
- Through ZK protocol

Idea of Signature Schemes

Signer | **Verifier**

Key Generation

Secret key $\mathcal{S}$, public key $\mathcal{P}$

$\xrightarrow{\mathcal{P}}$

Signing

Message m, signature σ

$\xrightarrow{m,\sigma}$

Verification

Verify σ

Two approaches to get a code-based signature scheme:

- Hash-and-sign

  $\rightarrow$ large public key sizes

- Through ZK protocol

  $\rightarrow$ large signature sizes

Idea of Signature Schemes

Signer | **Verifier**

Key Generation

Secret key $\mathcal{S}$, public key $\mathcal{P}$

$\xrightarrow{\mathcal{P}}$

Signing

Message m, signature σ

$\xrightarrow{m,\sigma}$

Verification

Verify σ

Two approaches to get a code-based signature scheme:

- Hash-and-sign
  - $\rightarrow$ large public key sizes
  - $\rightarrow$ Stefan's talk: FuLeeca
- Through ZK protocol
  - $\rightarrow$ large signature sizes
  - $\rightarrow$ this talk: restricted errors

Idea of Signature Schemes

Signer | **Verifier**

Key Generation

Secret key $\mathcal{S}$, public key $\mathcal{P}$

$\xrightarrow{\mathcal{P}}$

Signing

Message m, signature σ

$\xrightarrow{m,\sigma}$

Verification

Verify σ

Two approaches to get a code-based signature scheme:

- Hash-and-sign
  - → large public key sizes
  - → Stefan's talk: FuLeeca

- Through ZK protocol
  - → large signature sizes
  - → this talk: restricted errors

Idea of ZK Protocol

Prover		**Verifier**
$\mathcal{S}$: secret, $\mathcal{P}$: related public key c: commitments to secret r_b: response to challenge b	$\xrightarrow{\mathcal{P},c}$ $\xleftarrow{b}$ $\xrightarrow{r_b}$	b: challenge Recover c from r_b and $\mathcal{P}$

Idea of ZK Protocol

Prover		**Verifier**
$\mathcal{S}$: secret, $\mathcal{P}$: related public key c: commitments to secret r_b: response to challenge b	$\xrightarrow{\mathcal{P},c}$ $\xleftarrow{b}$ $\xrightarrow{r_b}$	b: challenge Recover c from r_b and $\mathcal{P}$

- *complete*: a honest prover gets accepted
- *zero-knowledge*: verifier does not gain information on $\mathcal{S}$
- *sound:* small probability of an impersonator getting accepted

Idea of ZK Protocol

Prover	Interaction	**Verifier**
$\mathcal{S}$: secret, $\mathcal{P}$: related public key c: commitments to secret r_b: response to challenge b	$\xrightarrow{\mathcal{P},c}$ $\xleftarrow{b}$ $\xrightarrow{r_b}$	b: challenge Recover c from r_b and $\mathcal{P}$

- *complete*: a honest prover gets accepted
- *zero-knowledge*: verifier does not gain information on $\mathcal{S}$
- *sound:* small probability of an impersonator getting accepted

Idea of ZK Protocol

Prover

$\mathcal{S}$: secret, $\mathcal{P}$: related public key
c: commitments to secret
b: Hash of message, c
r_b: response to challenge b

Fiat-Shamir

$\xrightarrow{\mathcal{P},(b,r_b)}$

Verifier

Recover c from r_b and $\mathcal{P}$
Verify $b = \text{Hash}(m, c)$

- *complete*: a honest prover gets accepted
- *zero-knowledge*: verifier does not gain information on $\mathcal{S}$
- *sound:* small probability of an impersonator getting accepted

Idea of ZK Protocol

Prover

N

$\mathcal{S}$: secret, $\mathcal{P}$: related public key
c: commitments to secret
b: Hash of message, c
r_b: response to challenge b

$\xrightarrow{\mathcal{P},(b,r_b)}$

Verifier

Recover c from r_b and $\mathcal{P}$
Verify $b = \text{Hash}(m, c)$

- *complete*: a honest prover gets accepted
- *zero-knowledge*: verifier does not gain information on $\mathcal{S}$
- *sound:* small probability of an impersonator getting accepted
- α cheating probability, λ bit security level
- *Rounds*: have to repeat ZK protocol N times: $2^\lambda < (1/\alpha)^N$

Code-based ZK Protocols

ZK protocol $\xrightarrow{\text{Fiat-Shamir}}$ Signature scheme

P.-L. Cayrel, P. Véron, S. El Yousfi Alaoui. "A zero-knowledge identification scheme based on the q-ary syndrome decoding problem", Selected Areas in Cryptography, 2011.

Syndrome Decoding Problem

Given parity-check matrix H, syndrome s, weight t, find e s.t. 1. $s = eH^\top$ 2. $\mathrm{wt}_H(e) \leq t$

Prover		**Verifier**
$\mathcal{S}$: e of weight t,		
$\mathcal{P}$: random H, $s = eH^\top$, t	$\xrightarrow{\mathcal{P}}$	
c_1: commitment to syndrome equation 1.	$\xleftarrow{b}$	$b \in \{1, 2\}$
c_2: commitment to weight 2.		
response: $r_1 = \varphi$, $r_2 = \varphi(e)$	$\xrightarrow{r_b}$	recover c_b from r_b and $\mathcal{P}$

Code-based ZK Protocols

ZK protocol $\xrightarrow{\text{Fiat-Shamir}}$ Signature scheme

P.-L. Cayrel, P. Véron, S. El Yousfi Alaoui. "A zero-knowledge identification scheme based on the q-ary syndrome decoding problem", Selected Areas in Cryptography, 2011.

Syndrome Decoding Problem

Given parity-check matrix H, syndrome s, weight t, find e s.t. 1. $s = eH^\top$ 2. $\mathrm{wt}_H(e) \leq t$

Prover **Verifier**

$\mathcal{S}$: e of weight t,

$\mathcal{P}$: random H, $s = eH^\top$

c_1: commitment to syn

c_2: commitment to weight 2. $\longleftarrow$

response: $r_1 = \varphi$, $r_2 = \varphi(e)$ $\xrightarrow{r_b}$ recover c_b from r_b and $\mathcal{P}$

Problem: large cheating probability $\rightarrow$ big signature sizes

Performance of Classical Approach

Example

- $\lambda = 128$ bit security level $\rightarrow N = 135$ $\quad\longrightarrow$ public key size: 832 b
- $q = 31, n = 256, k = 204$ $\quad\longrightarrow$ signature size: 43 kB

Performance of Classical Approach

Example

- $\lambda = 128$ bit security level $\rightarrow N = 135$ $\quad\rightarrow$ public key size: 832 b
- $q = 31, n = 256, k = 204$ $\quad\rightarrow$ signature size: 43 kB

for a long time not been considered practical

Performance of Classical Approach

Example

- $\lambda = 128$ bit security level $\rightarrow N = 135$ $\rightarrow$ public key size: 832 b
- $q = 31, n = 256, k = 204$ $\rightarrow$ signature size: 43 kB

for a long time not been considered practical

Recent improvements through in-the-head computations

$\rightarrow$ smaller signature sizes $\sim$ 10 kB

T. Feneuil, A. Joux, M. Rivain "Shared permutation for syndrome decoding: New zero-knowledge protocol and code-based signature", Designs, Codes and Cryptography, 2022.

T. Feneuil, A. Joux, M. Rivain "Syndrome decoding in the head: shorter signatures from zero-knowledge proofs", Crypto, 2022.

Performance of Classical Approach

Example

- $\lambda = 128$ bit security level $\rightarrow N = 135$ $\qquad\rightarrow$ public key size: 832 b
- $q = 31, n = 256, k = 204$ $\qquad\rightarrow$ signature size: 43 kB

for a long time not been considered practical

Recent improvements through in-the-head computations
$\rightarrow$ smaller signature sizes $\sim$ 10 kB

based on knowing we need many rounds

T. Feneuil, A. Joux, M. Rivai … zero-knowledge protocol and code-based signature", Design…

T. Feneuil, A. Joux, M. Rivain "Syndrome decoding in the head: shorter signatures from zero-knowledge proofs", Crypto, 2022.

Problem of Classical Approach

Classical CVE (1 round)

- public key size: seed of H, s; $\log_2(q)(n-k) < 0.1$ kB
- signature size: Hash(m, c) and response: transformation φ or $\varphi(e)$

Problem of Classical Approach

Classical CVE (1 round)

- public key size: seed of H, s; $\log_2(q)(n-k) < 0.1$ kB
- signature size: $\text{Hash}(m, c)$ and response: transformation φ or $\varphi(e)$

Which φ are allowed?

Problem of Classical Approach

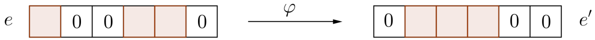
Classical CVE (1 round)

- public key size: seed of H, s; $\log_2(q)(n-k) < 0.1$ kB
- signature size: $\text{Hash}(m, c)$ and response: transformation φ or $\varphi(e)$

Which φ are allowed?

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, weight t, find $e \in \mathbb{F}_q^n$ such that $s = eH^\top$ and $\text{wt}_H(e) \leq t$.



Problem of Classical Approach

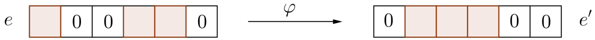
Classical CVE (1 round)

- public key size: seed of H, s; $\log_2(q)(n-k) < 0.1$ kB
- signature size: Hash(m, c) and response: transformation φ or $\varphi(e)$

Which φ are allowed?

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}, s \in \mathbb{F}_q^{n-k}$, weight t, find $e \in \mathbb{F}_q^n$ such that $s = eH^\top$ and $\mathrm{wt}_H(e) \leq t$.



$\rightarrow \varphi$: linear isometries of Hamming metric:
permutation + scalar multiplication

Problem of Classical Approach

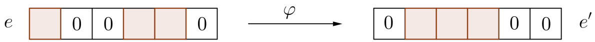
Classical CVE (1 round)

- public key size: seed of H, s; $\log_2(q)(n-k) < 0.1$ kB
- signature size: $\varphi(e) : t\log_2(q-1) + t\log_2(n)$ or $\varphi : n\log_2(q-1) + n\log_2(n)$

Which φ are allowed?

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, weight t, find $e \in \mathbb{F}_q^n$ such that $s = eH^\top$ and $\mathrm{wt}_H(e) \leq t$.

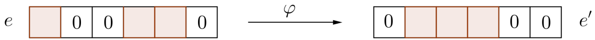


$\rightarrow \varphi$: linear isometries of Hamming metric:
permutation + scalar multiplication

Restricted Errors

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, weight t, find $e \in \mathbb{F}_q^n$ such that $s = eH^\top$ and $\mathrm{wt}(e) \leq t$.

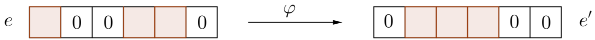


Can we avoid permutations - but keep the hardness of the problem?

Restricted Errors

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, weight t, find $e \in \mathbb{F}_q^n$ such that $s = eH^\top$ and $\mathrm{wt}(e) \le t$.



Can we avoid permutations - but keep the hardness of the problem?

↓

Restricted Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, syndrome $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ such that $s = eH^\top$.

Restricted Errors

M. Baldi, S. Bitzer, A. Pavoni, P. Santini, A. Wachter-Zeh, **V.W.** "Zero Knowledge Protocols and Signatures from the Restricted Syndrome Decoding Problem ", Preprint, 2023

Restricted Syndrome Decoding Problem

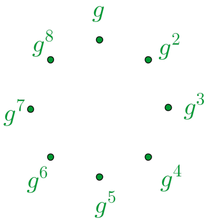
Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ such that $s = eH^\top$.

Restricted Errors

M. Baldi, S. Bitzer, A. Pavoni, P. Santini, A. Wachter-Zeh, **V.W.** "Zero Knowledge Protocols and Signatures from the Restricted Syndrome Decoding Problem ", Preprint, 2023

Restricted Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ such that $s = eH^\top$.



Idea

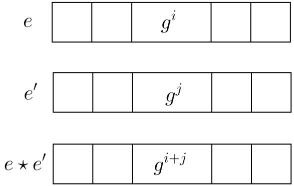
- $g \in \mathbb{F}_q^\star$ of order z,
  $\mathbb{E} = \{g^i \mid i \in \{1, \dots, z\}\}$

Restricted Errors

M. Baldi, S. Bitzer, A. Pavoni, P. Santini, A. Wachter-Zeh, **V.W.** "Zero Knowledge Protocols and Signatures from the Restricted Syndrome Decoding Problem ", Preprint, 2023

Restricted Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ such that $s = eH^\top$.



Idea

- $g \in \mathbb{F}_q^\star$ of order z,
  $\mathbb{E} = \{g^i \mid i \in \{1, \ldots, z\}\}$
- transf. $\varphi : \mathbb{E}^n \to \mathbb{E}^n$,
  $e \mapsto e \star e'$ for $e' \in \mathbb{E}^n$
- size of φ is $n \log_2(z)$
  (instead of $n \log_2((q-1)n)$)

Benefits of Restricted Errors

- Larger cost of solvers than for classical SDP $\rightarrow$ can choose smaller parameters

$\rightarrow$ Recall talk of Sebastian

- Size of φ and $\varphi(e)$ is smaller $\rightarrow$ smaller signature sizes
- Computations are easier (in $\mathbb{F}_z$ instead of $\mathbb{F}_q$) $\rightarrow$ smaller running times

Benefits of Restricted Errors

- Larger cost of solvers than for classical SDP → can choose smaller parameters

→ Recall talk of Sebastian

- Size of φ and $\varphi(e)$ is smaller → smaller signature sizes
- Computations are easier (in $\mathbb{F}_z$ instead of $\mathbb{F}_q$) → smaller running times

We can replace SDP with Restricted SDP in any code-based ZK protocol

Example GPS for $\lambda = 128$

$$q = 128, n = 220, k = 101, t = 90$$

→ signature size: 24.6 kB

Example Rest. GPS for $\lambda = 128$

$$q = 67, n = 147, k = 63, z = 11$$

→ signature size: 14.8 kB

S. Gueron, E. Persichetti, P. Santini. "Designing a practical code-based signature scheme from zero-knowledge proofs with trusted setup"

Benefits of Restricted Errors

- Larger cost of solvers than for classical SDP → can choose smaller parameters

→ Recall talk of Sebastian

- Size of φ and $\varphi(e)$ is smaller → smaller signature sizes
- Computations are easier (in $\mathbb{F}_z$ instead of $\mathbb{F}_q$) → smaller running times

We can replace SDP with Restricted SDP in any code-based ZK protocol

Example GPS for $\lambda = 128$

$$q = 128, n = 220, k = 101, t = 90$$

→ signature size: 24.6 kB

Example Rest. GPS for $\lambda = 128$

$$q = 67, n = 147, k = 63, z = 11$$

→ signature size: 14.8 kB

S. Gueron, E. Persichetti, P. Santini. "Designing a practical code-based signature scheme from zero-knowledge proofs with trusted setup"

But we can do even better: Restricted SDP in a subgroup G

Restricted-G SDP

$(\mathbb{E}^n, \star)$ is an abelian group isomorphic to $(\mathbb{F}_z^n, +)$

Restricted Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ s.t. $s = eH^\top$.

Restricted-G SDP

$(\mathbb{E}^n, \star)$ is an abelian group isomorphic to $(\mathbb{F}_z^n, +)$ $\rightarrow$ Subgroup $(G, \star) \leq (\mathbb{E}^n, \star)$

$$G = \langle x_1, \dots, x_m \rangle = \left\{ \prod_{i=1}^{m} x_i^{u_i} \mid u_i \in \{1, \dots, z\} \right\}$$

Restricted Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, find $e \in \mathbb{E}^n$ s.t. $s = eH^\top$.

Restricted-G SDP

$(\mathbb{E}^n, \star)$ is an abelian group isomorphic to $(\mathbb{F}_z^n, +)$ $\rightarrow$ Subgroup $(G, \star) \leq (\mathbb{E}^n, \star)$

$$G = \langle x_1, \ldots, x_m \rangle = \left\{ \prod_{i=1}^{m} x_i^{u_i} \mid u_i \in \{1, \ldots, z\} \right\}$$

Restricted-G Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^\star$, $G = \langle x_1, \ldots, x_m \rangle \leq \mathbb{E}^n$ find $e \in G$ s.t. $s = eH^\top$.

Restricted-G SDP

$(\mathbb{E}^n, \star)$ is an abelian group isomorphic to $(\mathbb{F}_z^n, +)$ $\rightarrow$ Subgroup $(G, \star) \leq (\mathbb{E}^n, \star)$

$$G = \langle x_1, \ldots, x_m \rangle = \left\{ \prod_{i=1}^{m} x_i^{u_i} \mid u_i \in \{1, \ldots, z\} \right\}$$

Restricted-G Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, $s \in \mathbb{F}_q^{n-k}$, $\mathbb{E} \subseteq \mathbb{F}_q^{\star}$, $G = \langle x_1, \ldots, x_m \rangle \leq \mathbb{E}^n$ find $e \in G$ s.t. $s = eH^\top$.

Classical		Rest.		Rest.-G
$n \log_2((q-1)n)$	$\rightarrow$	$n \log_2(z)$	$\rightarrow$	$m \log_2(z)$

Example

- $q = 13, n = 4, g = 3,$ $\rightarrow$ multiplicative order $z = 3$;

$$\mathbb{E} = \{g^0 = 1, g^1 = 3, g^2 = 9\}$$

- E.g. $e = (1, 9, 3, 3) \in \mathbb{E}^n$
- $m = 3$, generators

$$x_1 = (g^2, g^0, g^2, g^0),\ x_2 = (g^2, g^2, g^0, g^2, g^2),\ x_3 = (g^0, g^2, g^2, g^1).$$

- $G = \langle x_1, x_2, x_3 \rangle$
- E.g. $x_1^2 \star x_2^1 \star x_3^0 = (g^0, g^2, g^1, g^2) = (1, 9, 3, 9) \in G$, but $e = (1, 9, 3, 3) \notin G$
- $\mid G \mid = z^m = 9$, easy check:

$$M_G = \begin{pmatrix} 2 & 0 & 2 & 0 \\ 2 & 2 & 0 & 2 \\ 0 & 2 & 2 & 1 \end{pmatrix} \in \mathbb{F}_z^{m \times n}$$

Performance of Restricted SDP in G Signatures

Example GPS for $\lambda = 128$

- Classical GPS: $q = 128, n = 220, k = 101, t = 90$ $\longrightarrow$ signature size: 24.6 kB
- Restricted GPS: $q = 67, n = 147, k = 63, z = 11$ $\rightarrow$ signature size: 14.8 kB
- Restricted-G GPS: $q = 53, n = 82, k = 47, z = 13, m = 54$ $\rightarrow$ signature size: 12.7 kB

Performance of Restricted SDP in G Signatures

Example BG for $\lambda = 128$

- Classical BG: $q = 997, n = 61, k = 33, t = 31$ $\longrightarrow$ signature size: 8.9 kB
- Restricted BG: $q = 991, n = 77, k = 38, z = 33$ $\rightarrow$ signature size: 9.5 kB
- Restricted-G BG: $q = 1019, n = 40, k = 16, z = 509, m = 18$ $\rightarrow$ signature size: 7.2 kB

L. Bidoux, P. Gaborit. "Shorter Signatures from Proofs of Knowledge for the SD, MQ, PKP and RSD Problems "

Performance of Restricted SDP in G Signatures

Example BG for $\lambda = 128$

- Classical BG: $q = 997, n = 61, k = 33, t = 31$ → signature size: 8.9 kB
- Restricted BG: $q = 991, n = 77, k = 38, z = 33$ → signature size: 9.5 kB
- Restricted-G BG: $q = 1019, n = 40, k = 16, z = 509, m = 18$ → signature size: 7.2 kB

L. Bidoux, P. Gaborit. "Shorter Signatures from Proofs of Knowledge for the SD, MQ, PKP and RSD Problems "

Conclusion/Open Questions

- Can replace classical SDP with Restricted SDP/ Restricted-G SDP in any code-based ZK protocol.
- Achieve smaller signature sizes, smaller running times
- Can we exploit the commutativity of the restricted transformations?

Questions?

CROSS

Codes & Restricted Objects Signature Scheme

`http://cross-crypto.com/`

Thank you!

Running times

Running time given in kCycles, CROSS has only PoC, no optimization, parallelization

Scheme	**Key gen.**	**Signature gen.**	**Verification**
SPHINCS	1794	5802	6506
Dilitihium	49	140	61
CROSS	19	187	184

Solving Restricted SDP in subgroup G

- Recall Sebastian's talk: we want q, z such that $\mathbb{E}$ has no additive structure
- Publicly known: $x_1, \ldots, x_m$ generators of multiplicative group G
- $x_\ell = (g^{i_{1,\ell}}, \ldots, g^{i_{n,\ell}})$
- define $M_G \in \mathbb{F}_z^{m \times n}$ having rows $(i_{1,\ell}, \ldots, i_{n,\ell})$

$$M_G = \begin{bmatrix} i_{1,\ell} & \ldots & i_{n,\ell} \end{bmatrix}$$

J

$\hookrightarrow$ rank m'

$m' \geq \min \left\{ |\ J\ |, \frac{\lambda}{\log_2(z)} \right\} \rightarrow$ no improvement over enumerating all possible errors in these positions

Comparison

Scheme	Public Key size	Signature size	Total size	Variant
SPHINCS$^+$	<0.1	16.7	16.7	Fast
	<0.1	7.7	7.7	Short
Falcon	0.9	0.6	1.5	-
Dilitihium	1.3	2.4	3.7	-
CROSS	0.1	7.7	7.8	Fast
	0.1	7.2	7.3	Short
GPS	0.1	24.0	24.1	Fast
	0.1	19.8	19.9	Short
FJR	0.1	22.6	22.7	Fast
	0.1	16.0	16.1	Short
SDItH	0.1	11.5	11.6	Fast
	0.1	8.3	8.4	Short
Ret. of SDitH	0.1	12.1	12.1	Fast, V3
	0.1	5.7	5.8	Shortest, V3

Comparison

Scheme	Public Key size	Signature size	Total size	Variant
WAVE	3200	2.1	3202	-
Durandal	15.2	4.1	19.3	-
Ideal Rank BG	0.5	8.4	8.9	Fast
	0.5	6.1	6.6	Short
MinRank Fen	18.2	9.3	27.5	Fast
	18.2	7.1	25.3	Short
Rank SDP Fen	0.9	7.4	8.3	Fast
	0.9	5.9	6.8	Short
Beu	0.1	18.4	18.5	Fast
	0.1	12.1	12.2	Short
PKP BG	0.1	9.8	9.9	Fast
	0.1	8.8	8.9	Short
FuLeeca	0.4	0.3	0.7	-

Hash-and-Sign: CFS

PROVER		VERIFIER
KEY GENERATION		
$\mathcal{S} = H$ parity-check matrix $\mathcal{P} = (t, HP)$ permuted H		
SIGNING		
Choose message m $s = \text{Hash}(m)$ Find e: $s = eH^\top = eP(HP)^\top$, and $\text{wt}(e) \leq t$	$\xrightarrow{m, eP}$	
		VERIFICATION
		Check if $\text{wt}(eP) \leq t$ and $eP(HP)^\top = \text{Hash}(m)$

Hash-and-Sign: CFS

PROVER	VERIFIER
KEY GENERATION	
$\mathcal{S} = H$ parity-check matrix	
$\mathcal{P} = (t, HP)$ permuted H	
SIGNING	
Choose message m	
$s = \text{Hash}(m)$ Find e: $s = eH^\top = eP(HP)^\top$, and $\text{wt}(e) \leq t$	
$\xrightarrow{m, eP}$	
	VERIFICATION
	Check if $\text{wt}(eP) \leq t$ and $eP(HP)^\top = \text{Hash}(m)$

Problem: Distinguishability

Hash-and-Sign: CFS

PROVER	VERIFIER
KEY GENERATION	
$\mathcal{S} = H$ parity-check matrix	
$\mathcal{P} = (t, HP)$ permuted H	
SIGNING	
Choose message m	
$s = \text{Hash}(m)$	
Find e: $s = eH^\top = eP(HP)^\top$, and $\text{wt}(e) \leq t$	
$\xrightarrow{m, eP}$	
	VERIFICATION
	Check if $\text{wt}(eP) \leq t$ and $eP(HP)^\top = \text{Hash}(m)$

Not any s is syndrome of low weight e

ZKID

PROVER		VERIFIER
		VERIFICATION
commitments c_0, c_1	$\xrightarrow{c_0, c_1}$	
	$\xleftarrow{b}$	$b \in \{0, 1\}$
response r_b	$\xrightarrow{r_b}$	
		Verify c_b using $r_b, \mathcal{P}$

SIGNING		
Choose message m		
Construct signature s from $\mathcal{S}$, m		
	$\xrightarrow{m, s}$	
		VERIFICATION
		Verify signature s using $\mathcal{P}$, m

Signature Scheme

ZKID

PROVER		VERIFIER
		VERIFICATION
commitments c_0, c_1	$\xrightarrow{c_0, c_1}$	
	$\xleftarrow{b}$	$b \in \{0, 1\}$
response r_b	$\xrightarrow{r_b}$	
		Verify c_b using $r_b, \mathcal{P}$

Fiat-Shamir ↓

SIGNING		
Choose message m		
Construct signature s from $\mathcal{S}$, m		
	$\xrightarrow{m, s}$	
		VERIFICATION
		Verify signature s using $\mathcal{P}$, m

Signature Scheme

Fiat-Shamir

PROVER		VERIFIER
KEY GENERATION		
Given $\mathcal{P}, \mathcal{S}$ of some ZKID and message m		
SIGNING		
Choose commitment c $b = \text{Hash}(m, c)$ Compute response r_b Signature $s = (b, r_b)$		
	$\xrightarrow{m,s}$	
		VERIFICATION
		Using $r_b, \mathcal{P}$ construct c check if $b = \text{Hash}(m, c)$

CVE

PROVER		VERIFIER
KEY GENERATION		
Choose e with $\mathrm{wt}(e) \leq t$		
H parity-check matrix		
Compute $s = eH^\top$	$\xrightarrow{\mathcal{P}=(H,s,t)}$	
		VERIFICATION
Choose $u \in \mathbb{F}_q^n$, $\sigma \in \mathcal{S}_n$		
Set $c_1 = \mathrm{Hash}(\sigma, uH^\top)$		
Set $c_2 = \mathrm{Hash}(\sigma(u), \sigma(e))$	$\xrightarrow{c_1, c_2}$	
	$\xleftarrow{z}$	Choose $z \in \mathbb{F}_q^\times$
Set $y = \sigma(u + ze)$	$\xrightarrow{y}$	
$r_1 = \sigma$	$\xleftarrow{b}$	Choose $b \in \{1, 2\}$
$r_2 = \sigma(e)$	$\xrightarrow{r_b}$	$b = 1$: $c_1 = \mathrm{Hash}(\sigma, \sigma^{-1}(y)H^\top - zs)$
		$b = 2$: $\mathrm{wt}(\sigma(e)) = t$
		and $c_2 = \mathrm{Hash}(y - z\sigma(e), \sigma(e))$

CVE

PROVER		VERIFIER
KEY GENERATION		Recall SDP: (1) $s = eH^\top$ (2) $\mathrm{wt}(e) \leq t$
Choose e with $\mathrm{wt}(e) \leq t$		
H parity-check matrix		
Compute $s = eH^\top$	$\xrightarrow{\mathcal{P}=(H,s,t)}$	
		VERIFICATION
Choose $u \in \mathbb{F}_q^n$, $\sigma \in \mathcal{S}_n$		
Set $c_1 = \mathrm{Hash}(\sigma, uH^\top)$		
Set $c_2 = \mathrm{Hash}(\sigma(u), \sigma(e))$	$\xrightarrow{c_1, c_2}$	
	$\xleftarrow{z}$	Choose $z \in \mathbb{F}_q^\times$
Set $y = \sigma(u + ze)$	$\xrightarrow{y}$	
$r_1 = \sigma$	$\xleftarrow{b}$	Choose $b \in \{1, 2\}$
$r_2 = \sigma(e)$	$\xrightarrow{r_b}$	$b = 1$: $c_1 = \mathrm{Hash}(\sigma, \sigma^{-1}(y)H^\top - zs)$
		$b = 2$: $\mathrm{wt}(\sigma(e)) = t$
		and $c_2 = \mathrm{Hash}(y - z\sigma(e), \sigma(e))$

CVE

PROVER		VERIFIER
KEY GENERATION		
Choose e with $\mathrm{wt}(e) \leq t$		
H parity-check matrix		
Compute $s = eH^\top$	$\xrightarrow{\mathcal{P}=(H,s,t)}$	
		VERIFICATION
Choose $u \in \mathbb{F}_q^n$, $\sigma \in \mathcal{S}_n$		
Set $c_1 = \mathrm{Hash}(\sigma, uH^\top)$		
Set $c_2 = \mathrm{Hash}(\sigma(u), \sigma(e))$	$\xrightarrow{c_1, c_2}$	
	$\xleftarrow{z}$	Choose $z \in \mathbb{F}_q^\times$
Set $y = \sigma(u + ze)$	$\xrightarrow{y}$	
$r_1 = \sigma$	$\xleftarrow{b}$	Choose $b \in \{1,2\}$
$r_2 = \sigma(e)$	$\xrightarrow{r_b}$	$b = 1$: $c_1 = \mathrm{Hash}(\sigma, \sigma^{-1}(y)H^\top - zs)$
		$b = 2$: $\mathrm{wt}(\sigma(e)) = t$
		and $c_2 = \mathrm{Hash}(y - z\sigma(e), \sigma(e))$

Problem: big signature sizes

Cheating Probability

- Cheating probability = Probability of impersonator getting accepted
- For security level 2^λ want cheating probability $2^{-\lambda}$
- If cheating probability δ, with N rounds $\rightarrow$ cheating probability δ^N

Cheating Probability

- Cheating probability = Probability of impersonator getting accepted
- For security level 2^λ want cheating probability $2^{-\lambda}$
- If cheating probability δ, with N rounds $\rightarrow$ cheating probability δ^N
- might need many rounds: large communication cost

Cheating Probability

- Cheating probability = Probability of impersonator getting accepted
- For security level 2^λ want cheating probability $2^{-\lambda}$
- If cheating probability δ, with N rounds $\rightarrow$ cheating probability δ^N
- might need many rounds: large communication cost
- solution: compression technique
- do not send c_0^i, c_1^i in each round i
- before 1. round send $c = \text{Hash}(c_0^1, c_1^1, \dots, c_0^N, c_1^N)$
- ith round: receiving challenge b prover sends r_b^i, c_{1-b}^i
- end: verifier checks $c = \text{Hash}(c_0^1, c_1^1, \dots, c_0^N, c_1^N)$

C. Aguilar, P. Gaborit, J. Schrek. "A new zero-knowledge code based identification scheme with reduced communication", IEEE Information Theory Workshop, 2011.

Cheating Probability

- Cheating probability = Probability of impersonator getting accepted
- For security level 2^λ want cheating probability $2^{-\lambda}$
- If cheating probability δ, with N rounds $\rightarrow$ cheating probability δ^N
- might need many rounds: large communication cost
- other solution: MPC in the head
- third party: trusted helper sends commitments $\rightarrow \delta = 0$
- instead prover sends seeds of commitment: not ZK $\rightarrow$ cut and choose
- $x < N$ times send response, $N - x$ times send the seed of commitment
- to compress: use Merkle root or seed tree

T. Feneuil, A. Joux, M. Rivain. " Syndrome decoding in the head: Shorter signatures from zero-knowledge proofs", 2022.

Comparison

	ZKID	Hash-and-Sign
reduction to NP-hard		
low public key size		
low signature size		
fast verification		

Comparison

	ZKID	Hash-and-Sign
reduction to NP-hard	✓	×
low public key size		
low signature size		
fast verification		

Comparison

	ZKID	Hash-and-Sign
reduction to NP-hard	✓	×
low public key size	✓	×
low signature size		
fast verification		

Comparison

	ZKID	Hash-and-Sign	
reduction to NP-hard	✓	×	
low public key size	CVE: 70 B	WAVE: 3 MB	NIST: 3 KB
low signature size			
fast verification			

Comparison

	ZKID	Hash-and-Sign	
reduction to NP-hard	$\checkmark$	$\times$	
low public key size	CVE: 70 B	WAVE: 3 MB	NIST: 3 KB
low signature size	$\sim$	$\checkmark$	
fast verification			

Comparison

	ZKID	Hash-and-Sign	
reduction to NP-hard	✓	×	
low public key size	CVE: 70 B	WAVE: 3 MB	NIST: 3 KB
low signature size	CVE: 43 KB	WAVE: 1 KB	NIST: 2 KB
fast verification			

Comparison

	ZKID	Hash-and-Sign	
reduction to NP-hard	✓	×	
low public key size	CVE: 70 B	WAVE: 3 MB	NIST: 3 KB
low signature size	CVE: 43 KB	WAVE: 1 KB	NIST: 2 KB
fast verification	∼	✓	