Recent Advances and Challenges in Code-based Signatures

Violetta Weger

Technical University of Munich

CrossFyre 2022
October 7, 2022

Motivation

NIST announcement of re-opened standardization call

- Deadline March 1, 2023
- Want signatures not based on structured lattices
- Want short signature sizes and fast verification

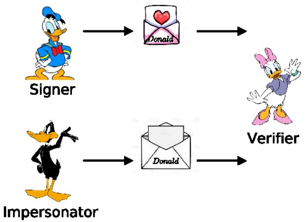
Motivation

NIST announcement of re-opened standardization call

- Deadline March 1, 2023
- Want signatures not based on structured lattices
- Want short signature sizes and fast verification

1. What is a signature scheme?
2. What is coding theory?
3. How to construct code-based signatures?
    - Hash-and-sign
    - Through ZKID
4. How do they compare?

Signature scheme



Goal

- No interest in security of message
- Want to verify identity of sender

Parties

- Prover: signs message, prove identity
- Verifier: receives message, verify identity
- Impersonator: wants to forge a signature

Performance

- Signature size
- Public and secret key size
- Verification time

Signature scheme

PROVER		VERIFIER
KEY GENERATION		
Construct secret key $\mathcal{S}$		
Construct public key $\mathcal{P}$		
	$\xrightarrow{\mathcal{P}}$	
SIGNING		
Choose message m		
Construct signature s from $\mathcal{S}$, m		
	$\xrightarrow{m,s}$	
		VERIFICATION
		Verify signature s using $\mathcal{P}$, m

Coding Theory

Set Up

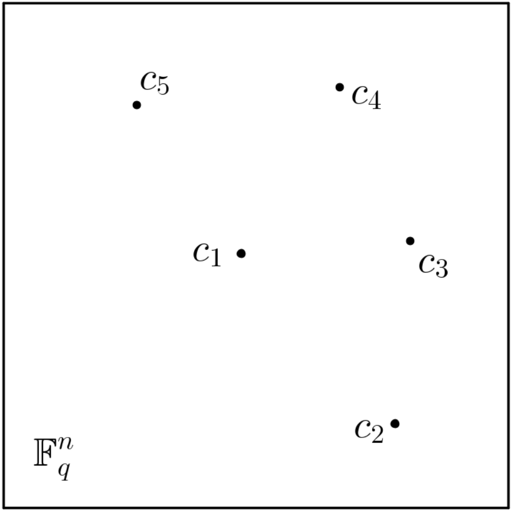
- $\mathbb{F}_q$: finite field with q elements
- $\mathcal{C}$ an $[n,k]$ linear code: $\mathcal{C} \subseteq \mathbb{F}_q^n$ linear subspace of dimension k
- $c \in \mathcal{C}$: codewords
- $G \in \mathbb{F}_q^{k \times n}$ generator matrix: $\mathcal{C} = \{xG \mid x \in \mathbb{F}_q^k\}$
- $H \in \mathbb{F}_q^{(n-k) \times n}$ parity-check matrix: $\mathcal{C} = \{c \in \mathbb{F}_q^n \mid cH^\top = 0\}$
- Syndrome: $s = eH^\top \in \mathbb{F}_q^{n-k}$
- Hamming metric: $x, y \in \mathbb{F}_q^n$

$$\mathrm{wt}(x) = \mid \{i \in \{1, \dots, n\} \mid x_i \neq 0\} \mid,$$
$$d(x,y) = \mathrm{wt}(x-y) = \mid \{i \in \{1, \dots, n\} \mid x_i \neq y_i\} \mid.$$

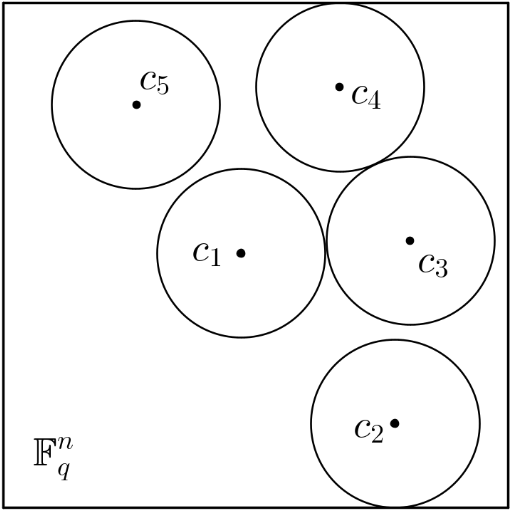
- Minimum Hamming distance of $\mathcal{C}$

$$d(\mathcal{C}) = \min\{\mathrm{wt}(x) \mid 0 \neq x \in \mathcal{C}\}.$$

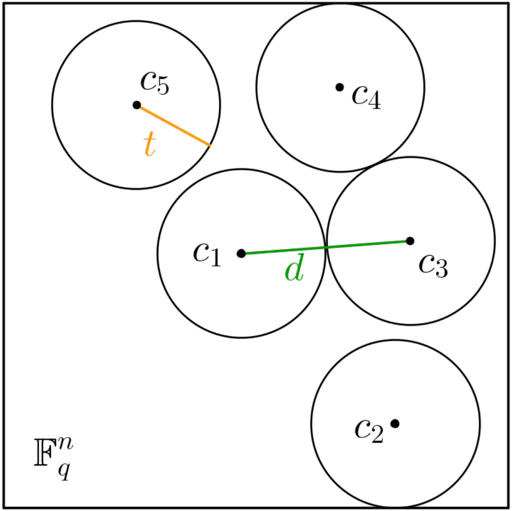
Coding Theory



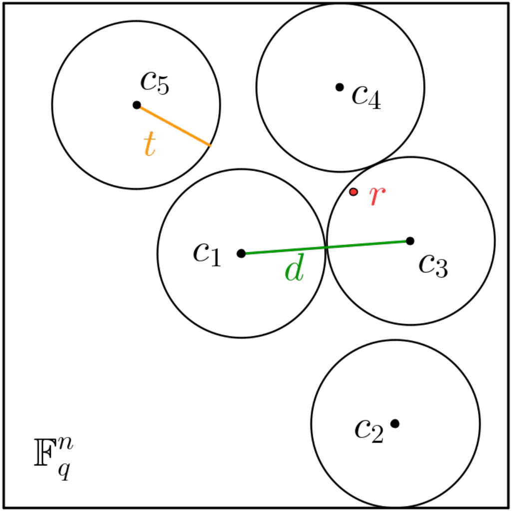
Coding Theory



Coding Theory



$t = \lfloor \frac{d-1}{2} \rfloor$



$t = \lfloor \frac{d-1}{2} \rfloor$ $r = c + e$

Coding Theory

- Can decode efficiently if algebraically structured

- Can decode efficiently if algebraically structured
- If random code: NP-complete problem!

Syndrome Decoding Problem

Given $H \in \mathbb{F}_q^{(n-k)\times n}$, syndrome $s \in \mathbb{F}_q^{n-k}$, target weight $t \in \mathbb{N}$, find $e \in \mathbb{F}_q^n$, such that

1. $\mathrm{wt}(e) \leq t$
2. $s = eH^\top$.

E. Berlekamp, R. McEliece, H. Van Tilborg. "On the inherent intractability of certain coding problems", IEEE Transactions on Information Theory, 1978.

Hash-and-Sign

N. Courtois, M. Finiasz, N. Sendrier. "How to achieve a McEliece-based digital signature scheme", ASIACRYPT, 2001.

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

N. Courtois, M. Finiasz, N. Sendrier. “How to achieve a McEliece-based digital signature scheme”, ASIACRYPT, 2001.

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

N. Courtois, M. Finiasz, N. Sendrier. "How to achieve a McEliece-based digital signature scheme", ASIACRYPT, 2001.

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

The story of Hash-and-Sign

1997 Random codes
large region of weak parameters

G. Kabatianskii, E. Krouk, B. Smeets. “A digital signature scheme based on random error-correcting codes”, IMA International Conference on Cryptography and Coding, 1997.

2001 High rate Goppa codes
distinguisher

N. Courtois, M. Finiasz, N. Sendrier. “How to achieve a McEliece-based digital signature scheme”, ASIACRYPT, 2001.

2013 LDGM codes
statistical attacks

M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, D. Schipani “Using LDGM codes and sparse syndromes to achieve digital signatures”, International Workshop on Post-Quantum Cryptography, 2013.

2018 $(u, u+v)$-construction,
large weights
large key sizes

T. Debris-Alazard, N. Sendrier, J.-P. Tillich. “Wave: A new family of trapdoor one-way preimage sampleable functions based on codes”, ASIACRYPT, 2019.

Through ZKID

- 2 Parties: Prover, Verifier
- 2 Stages: Key generation, Verification
- Prover wants to prove her knowledge of a secret to verifier, without revealing the secret

Through ZKID

- 2 Parties: Prover, Verifier
- 2 Stages: Key generation, Verification
- Prover wants to prove her knowledge of a secret to verifier, without revealing the secret

PROVER		VERIFIER
KEY GENERATION		
Construct secret key $\mathcal{S}$		
Construct public key $\mathcal{P}$	$\xrightarrow{\mathcal{P}}$	
		VERIFICATION
Construct commitments c_0, c_1		
	$\xrightarrow{c_0, c_1}$	
		Choose $b \in \{0, 1\}$
	$\xleftarrow{b}$	
Construct response r_b		
	$\xrightarrow{r_b}$	
		Verify c_b using $r_b, \mathcal{P}$

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

Through ZKID

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

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 $b = \text{Hash}(m, c)$

The story of code-based ZKID

1994 first code-based ZKID over $\mathbb{F}_2$

1997 better cheating probability

2011 generalization to $\mathbb{F}_q$

2011 quasi-cyclic structure over $\mathbb{F}_2$

J. Stern. "A new identification scheme based on syndrome decoding", Annual International Cryptology Conference, 1993.

P. Véron. "Improved identification schemes based on error-correcting codes", Applicable Algebra in Engineering, Communication and Computing, 1997.

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

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

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 $c_0 = \mathrm{Hash}(\sigma, uH^\top)$		
Set $c_1 = \mathrm{Hash}(\sigma(u), \sigma(e))$	$\xrightarrow{c_0, c_1}$	
	$\xleftarrow{z}$	Choose $z \in \mathbb{F}_q^\times$
Set $y = \sigma(u + ze)$	$\xrightarrow{y}$	
$r_0 = \sigma$	$\xleftarrow{b}$	Choose $b \in \{0, 1\}$
$r_1 = \sigma(e)$	$\xrightarrow{r_b}$	$b = 0$: $c_0 = \mathrm{Hash}(\sigma, \sigma^{-1}(y)H^\top - zs)$
		$b = 1$: $\mathrm{wt}(\sigma(e)) = t$
		and $c_1 = \mathrm{Hash}(y - z\sigma(e), \sigma(e))$

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 $c_0 = \mathrm{Hash}(\sigma, uH^\top)$		
Set $c_1 = \mathrm{Hash}(\sigma(u), \sigma(e))$	$\xrightarrow{c_0, c_1}$	
	$\xleftarrow{z}$	Choose $z \in \mathbb{F}_q^\times$
Set $y = \sigma(u + ze)$	$\xrightarrow{y}$	
$r_0 = \sigma$	$\xleftarrow{b}$	Choose $b \in \{0, 1\}$
$r_1 = \sigma(e)$	$\xrightarrow{r_b}$	$b = 0$: $c_0 = \mathrm{Hash}(\sigma, \sigma^{-1}(y)H^\top - zs)$
		$b = 1$: $\mathrm{wt}(\sigma(e)) = t$
		and $c_1 = \mathrm{Hash}(y - z\sigma(e), \sigma(e))$

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 $c = \mathrm{Hash}(c_0^1, c_1^1, \ldots, c_0^N, c_1^N)$
- ith round: receiving challenge b prover sends r_b^i, c_{1-b}^i
- end: verifier checks $c = \mathrm{Hash}(c_0^1, c_1^1, \ldots, c_0^N, c_1^N)$

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		

	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	✓	×
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	✓	×	
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			

	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	$\sim$	✓	

Questions?

Thank you!