Enhancing Code Based Zero-knowledge Proofs

using Rank Metric

16/12/2020

Emanuele Bellini[1] Philippe Gaborit[2] Alexandros Hasikos[13] Victor Mateu[1]

[1]TII Cryptography Research Centre [2]University of Limogés [3]Universitat Pompeu Fabra

Contents

Contributions

Contributions of this work

Cryptography Research Centre

- Adapt Jain et al. (2012) work and design a perfectly binding and computationally hiding commitment scheme based on the Rank Syndrome Decoding (RSD) Problem.

Contributions of this work

- Adapt Jain et al. (2012) work and design a perfectly binding and computationally hiding commitment scheme based on the Rank Syndrome Decoding (RSD) Problem.
- Design interactive protocols for:
  - Knowledge of valid opening.
  - Proving linear relations.
  - Proving multiplicative (or any bitwise) relations.

Contributions of this work

Cryptography Research Centre

- Adapt Jain et al. (2012) work and design a perfectly binding and computationally hiding commitment scheme based on the Rank Syndrome Decoding (RSD) Problem.
- Design interactive protocols for:
  - Knowledge of valid opening.
  - Proving linear relations.
  - Proving multiplicative (or any bitwise) relations.
- Compute secure parameters for both LPN and RSD variants of the protocols.

Contributions of this work

Cryptography Research Centre

- Adapt Jain et al. (2012) work and design a perfectly binding and computationally hiding commitment scheme based on the Rank Syndrome Decoding (RSD) Problem.
- Design interactive protocols for:
  - Knowledge of valid opening.
  - Proving linear relations.
  - Proving multiplicative (or any bitwise) relations.
- Compute secure parameters for both LPN and RSD variants of the protocols.
- Implement and compare (performance and efficiency) of both LPN and RSD variants with suggested parameters for 128 bits of security.

Highlights

Cryptography Research Centre

- Our work is the first zero-knowlegde protocol for arbitrary circuits whose security relies on the Rank Syndrome Decoding problem.

Highlights

Cryptography Research Centre

- Our work is the first zero-knowlegde protocol for arbitrary circuits whose security relies on the Rank Syndrome Decoding problem.
- Our proposal (RSD) generates proofs that are 60% smaller that the LPN variant for the same security level.

Highlights

Cryptography Research Centre

- Our work is the first zero-knowlegde protocol for arbitrary circuits whose security relies on the Rank Syndrome Decoding problem.
- Our proposal (RSD) generates proofs that are 60% smaller that the LPN variant for the same security level.
- Public parameters of the RSD variant are only 1% of respective parameters for the LPN variant.

Preliminaries

Definitions

Cryptography Research Centre

Definition (Linear $(n, k)_q$-code)

A linear $(n, k)_q-$ code C is a vector subspace of $(\mathbb{F}_q)^n$ of dimension k, where k and n are positive integers such that $k < n$, q is a prime power, and $\mathbb{F}_q$ is the finite field with q elements. Elements of the vector space are called **vectors** or **words**, while elements of the code are called **codewords**.

Definitions

Cryptography Research Centre

Definition (Linear $(n, k)_q$-code)

A linear $(n, k)_q-$ code C is a vector subspace of $(\mathbb{F}_q)^n$ of dimension k, where k and n are positive integers such that $k < n$, q is a prime power, and $\mathbb{F}_q$ is the finite field with q elements. Elements of the vector space are called **vectors** or **words**, while elements of the code are called **codewords**.

Definition (Generator and Parity Check Matrices)

A matrix $G \in \mathcal{M}^*_{k,n}(\mathbb{F}_q)$ is called a generator matrix of C if its rows form a basis of C, i.e. $C = \left\{x \cdot G : x \in (\mathbb{F}_q)^k\right\}$. A matrix $H \in \mathcal{M}^*_{n-k,n}(\mathbb{F}_q)$ is called a parity-check matrix of C if $C = \left\{x \in (\mathbb{F}_q)^n : H \cdot x^T = 0\right\}$

Definitions

Cryptography Research Centre

Definition (Hamming weight $w_H(v)$ of a vector)

The hamming weight $w_H(v)$ of a vector v is the number of its non-zero bits.

Definitions

Cryptography Research Centre

Definition (Hamming weight $w_H(v)$ of a vector)

The hamming weight $w_H(v)$ of a vector v is the number of its non-zero bits.

Definition (Rank weight $w_R(v)$ of a vector)

The rank weight $w_R(v)$ of a vector v is the rank of its matrix representation (number of linearly independent vectors).

Definitions

Cryptography Research Centre

Definition (Rank preserving transformation function $\Pi_{P,Q}(v)$)

Let $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$ be a q-ary matrix of size $m \times m$, $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ be a q-ary matrix of size $n \times n$, and $v = (v_1, \ldots, v_n) \in (\mathbb{F}_{q^m})^n$. We define the function $\Pi_{P,Q}$ such that $(\pi_1, \ldots, \pi_n) = \Pi_{P,Q}(v) = \phi^{-1}(Q \cdot \phi(v)\, P) \in (\mathbb{F}_{q^m})^n$, where for $h = 1, \ldots, n$, $\pi_h := \beta_1 \sum_{i=1}^m \sum_{j=1}^n Q_{1,i} v_{i,j} P_{j,h} + \ldots + \beta_m \sum_{i=1}^m \sum_{j=1}^n Q_{m,i} v_{i,j} P_{j,h}$, with $\beta = \{\beta_1, \ldots, \beta_m\}$ a basis of $(\mathbb{F}_q)^m$

Definitions

Cryptography Research Centre

Definition (Rank preserving transformation function $\Pi_{P,Q}(v)$)

Let $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$ be a q-ary matrix of size $m \times m$, $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ be a q-ary matrix of size $n \times n$, and $v = (v_1, \ldots, v_n) \in (\mathbb{F}_{q^m})^n$. We define the function $\Pi_{P,Q}$ such that $(\pi_1, \ldots, \pi_n) = \Pi_{P,Q}(v) = \phi^{-1}(Q \cdot \phi(v)\, P) \in (\mathbb{F}_{q^m})^n$, where for $h = 1, \ldots, n$, $\pi_h := \beta_1 \sum_{i=1}^m \sum_{j=1}^n Q_{1,i} v_{i,j} P_{j,h} + \ldots + \beta_m \sum_{i=1}^m \sum_{j=1}^n Q_{m,i} v_{i,j} P_{j,h}$, with $\beta = \{\beta_1, \ldots, \beta_m\}$ a basis of $(\mathbb{F}_q)^m$

Gaborit et al. (2011) proved that:

- For any $x, y \in (\mathbb{F}_{q^m})^n$ any full rank $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ and $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$
- $\Pi_{P,Q}$ has the rank preserving property $\mathrm{w_R}(\Pi_{P,Q}(x)) = \mathrm{w_R}(x)$ and is a linear mapping $a\Pi_{P,Q}(x) + b\Pi_{P,Q}(y) = \Pi_{P,Q}(ax + by)$.
- $\Pi_{P,Q}$ is invertible if P and Q are.

Decoding Problem

The decoding problem for random linear codes, consists of searching for the closest codeword to a given vector:

Decoding Problem

Cryptography Research Centre

The decoding problem for random linear codes, consists of searching for the closest codeword to a given vector:

Decoding Problem

Given $G, y = xG + e$, and the weight w, find the pair (x, e), where the weight of e is w.

Decoding Problem

The decoding problem for random linear codes, consists of searching for the closest codeword to a given vector:

Decoding Problem

Given $G, y = xG + e$, and the weight w, find the pair (x, e), where the weight of e is w.

In the case of random linear codes, the decoding problem is equivalent to the syndrome decoding problem:

Decoding Problem

The decoding problem for random linear codes, consists of searching for the closest codeword to a given vector:

Decoding Problem

Given G, $y = xG + e$, and the weight w, find the pair (x, e), where the weight of e is w.

In the case of random linear codes, the decoding problem is equivalent to the syndrome decoding problem:

Syndrome Decoding Problem

Given H, $s = Hy$, and the weight w, find y, where the hamming weight of y is w.

Decoding Problem

The decoding problem for random linear codes, consists of searching for the closest codeword to a given vector:

Decoding Problem

Given $G, y = xG + e$, and the weight w, find the pair (x, e), where the weight of e is w.

In the case of random linear codes, the decoding problem is equivalent to the syndrome decoding problem:

Syndrome Decoding Problem

Given $H, s = Hy$, and the weight w, find y, where the hamming weight of y is w.

The Rank Syndrome Decoding problem is the same as the Syndrome Decoding problem however the metric used for the weight of the error is the **rank** instead of the Hamming weight.

Commitment Schemes

Cryptography Research Centre

Definition (Commitment Schemes)

A triple of algorithms (**Setup**, **Com**, **Ver**) is called a commitment scheme if it satisfied the following:

- On input 1^ℓ the setup algorithm **Setup** outputs the public commitment parameters **pp**.
- The commitment algorithm **Com** takes as input a message **m** from a message space **M** and the public commitment parameters **pp**, and outputs a commitment/opening pair **(c, d)**.
- The verification algorithm **Ver** take the parameters **pp**, a message **m**, a commitment **c** and an opening **d** and outputs **true** or **false**.

Properties of commitment schemes

Cryptography Research Centre

The commitment scheme we will describe satisfies the following security properties:

- *Correctness*: **Ver** evaluates to **true** if the inputs are honestly computed.

Properties of commitment schemes

The commitment scheme we will describe satisfies the following security properties:

- *Correctness*: **Ver** evaluates to **true** if the inputs are honestly computed.
- *Perfect Binding*: With overwhelming probability over the choice of the public commitment parameters, no commitment can be opened in two different ways.

Properties of commitment schemes

The commitment scheme we will describe satisfies the following security properties:

- *Correctness*: **Ver** evaluates to **true** if the inputs are honestly computed.
- *Perfect Binding*: With overwhelming probability over the choice of the public commitment parameters, no commitment can be opened in two different ways.
- *Computational Hiding*: A commitment, computationally hides the committed message if the commitments are computationally indistinguishable.

Commitment Scheme

Commitment scheme in the rank metric

Cryptography Research Centre

Let q be the prime characteristic, m the degree of the q-ary extension field $\mathbb{F}_{q^m}$, the bit length μ of a message $\mathrm{m} \in \mathbb{F}_q^{\mu}$, the bit length π of the randomness $\mathrm{s} \in \mathbb{F}_q^{\pi}$, the length n of the linear code C, and the rank weight ρ of an error $\mathrm{e} \in \mathbb{F}_{q^m}^{n}$.

Commitment scheme in the rank metric

Cryptography Research Centre

Let q be the prime characteristic, m the degree of the q-ary extension field $\mathbb{F}_{q^m}$, the bit length μ of a message $\mathrm{m} \in \mathbb{F}_q^{\mu}$, the bit length π of the randomness $\mathrm{s} \in \mathbb{F}_q^{\pi}$, the length n of the linear code C, and the rank weight ρ of an error $\mathrm{e} \in \mathbb{F}_{q^m}^{n}$.

Setup(1^{ℓ})

$G_{\mathrm{m}} \leftarrow \mathcal{M}^{*}_{\frac{\mu}{m},n}(\mathbb{F}_{q^m})$

$G_{\mathrm{s}} \leftarrow \mathcal{M}^{*}_{\frac{\pi}{m},n}(\mathbb{F}_{q^m})$

return $G = \left(G_{\mathrm{s}}^{\mathrm{T}} \| G_{\mathrm{m}}^{\mathrm{T}}\right)^{\mathrm{T}}$

Com$_G(\mathrm{m})$

$\mathrm{s} \leftarrow_{\$} \mathrm{F}_2^{\pi}$

$\mathrm{e} \leftarrow_{\$} \mathbb{F}_{q^m}^{n}$, s.t. $\mathrm{w_R}(\mathrm{e}) = \rho$

$c = (s \| \mathrm{m}) \cdot G + e$

return c, s

Ver$_G(\mathrm{c}, \mathrm{m}', \mathrm{s}')$

$\mathrm{e}' = \mathrm{c} + (\mathrm{s}' \| \mathrm{m}') \cdot G$

if $\mathrm{w_R}(\mathrm{e}') = \rho$ return True

else return False

Parameters

In order to compare our scheme (RSD) with the Hamming weight variant (LPN) we had to compute parameters for both.

Parameters

Cryptography Research Centre

In order to compare our scheme (RSD) with the Hamming weight variant (LPN) we had to compute parameters for both.

For a quantum security level of 128 bits:

		Parameters	$\lvert$Secret$\rvert$	$\lvert$Public Param.$\rvert$	Average Communication
Hamming	Formula	(n, k, w)	$k + n$	$n + kn + \log_2(w)$	$5n + \lceil 2/3(n \log_2(n)) \rceil + 2\lambda$
	Bits	(2640,1320,284)	3960	3487449	33461
Rank (this work)	Formula	(q, m, n, k, ρ)	$mk + mn$	$mn + mkn + \log_2(\rho)$	$5mn + \lceil 2/3(m^2 + n^2) \rceil + 2\lambda$
	Bits	(2,43,38,17,13)	2365	29416	10622

Table 1: Communication cost and parameters bit sizes of the Σ-protocol of knowledge of valid opening

Parameters

In order to compare our scheme (RSD) with the Hamming weight variant (LPN) we had to compute parameters for both.

For a quantum security level of 128 bits:

		Parameters	|Secret|	|Public Param.|	Average Communication
Hamming	Formula	(n, k, w)	$k + n$	$n + kn + \log_2(w)$	$5n + \lceil 2/3(n \log_2(n)) \rceil + 2\lambda$
	Bits	(2640,1320,284)	3960	3487449	33461
Rank (this work)	Formula	(q, m, n, k, ρ)	$mk + mn$	$mn + mkn + \log_2(\rho)$	$5mn + \lceil 2/3(m^2 + n^2) \rceil + 2\lambda$
	Bits	(2,43,38,17,13)	2365	29416	10622

Table 1: Communication cost and parameters bit sizes of the Σ-protocol of knowledge of valid opening

Average communication cost is about 60% lower while the public parameters size is two orders of magnitude lower. The size of the secret in ZKP is 40% lower.

Performance

Implementation

We have implemented both the work from Jain et al. (2012) and our variant with the parameters shown in previous slide.

- Implemented in C++ using the NTL library from Victor Shoup.
- Benchmarks conducted on 2.9GHz Quad-Core Intel Core i7 with 16GB of LPDD3 RAM at 2133MHz.
- You can access the code https://github.com/Crypto-TII/2020-CANS-rank_commitments

Commitment Scheme

Commitment Scheme					
Jain et. al.			This work		
Routine	**Subroutine**	**Time [ms]**	**Routine**	**Subroutine**	**Time [ms]**
Setup	Generate matrix A	1.303	Setup	Generate matrix G	0.030
Commitment	Generate random vector r	negl.	Commitment	Generate random vector π	negl.
	Generate error vector e	0.168		Generate error vector e	1.800
	Compute commitment c	0.029		Compute commitment c	0.025
	Total	0.197		**Total**	1.825
Verification	Recover error vector e	0.029	Verification	Recover error vector e	0.0250
	Compute hamming weight of e	0.001		Compute rank of e	0.0160
	Total	0.030		**Total**	0.041

Table 2: Commitment scheme performance comparison.

Knowledge of Valid Opening

Cryptography Research Centre

Knowledge of Valid Opening							
Jain et.al.				This work			
Routine	**Subroutine**		**Time [ms]**	**Routine**	**Subroutine**		**Time[ms]**
Proof gen.	Generate π		0.552	Proof gen.	Generate $\Pi_{P,Q}$		0.135
	Generate random vectors		negl.		Generate random vectors		negl.
	Comm. 0	t_0	0.032		Comm. 0	r_0	0.020
		$\mathsf{E}(t_\pi, t_0)$	0.400			$\mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0)$	0.035
		$\mathsf{Com}(\mathsf{E}(t_\pi, t_0))$	0.200			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0))$	1.860
	Comm. 1	t_1	0.038		Comm. 1	r_1	0.044
		$\mathsf{E}(t_1)$	0.391			$\mathsf{E}(\mathsf{r}_1)$	0.019
		$\mathsf{Com}(\mathsf{E}(t_1))$	0.203			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_1))$	1.809
	Comm. 2	t_2	0.040		Comm. 2	r_2	0.044
		$\mathsf{E}(t_2)$	0.396			$\mathsf{E}(\mathsf{r}_2)$	0.018
		$\mathsf{Com}(\mathsf{E}(t_2))$	0.197			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_2))$	1.736
	Total		1.897		**Total**		5.585
Proof ver.	Verif. 0	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(t_\pi, t_0), \mathsf{s}_0))$	0.423	Proof ver.	Verif. 0	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0))$	0.077
		$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(t_1), \mathsf{s}_1)$	0.426			$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1)$	0.064
		$t_0 + \pi^{-1}(t_1) \in \mathsf{Img}(A)$	170.888			$\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_1) \in \mathsf{Img}(G)$	2.559
	Verif. 1	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(t_\pi, t_0), \mathsf{s}_0))$	0.424		Verif. 1	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0))$	0.080
		$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(t_2), \mathsf{s}_2)$	0.444			$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2)$	0.066
		$t_0 + \pi^{-1}(t_2) + y \in \mathsf{Img}(A)$	175.526			$\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_2) + \mathsf{y} \in \mathsf{Img}(G)$	2.47
	Verif. 2	$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(t_1), \mathsf{s}_1)$	0.459		Verif. 2	$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1)$	0.064
		$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(t_2), \mathsf{s}_2)$	0.446			$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2)$	0.064
		$\mathsf{w_H}(t_1 + t_2)$	0.001			$\mathsf{w_R}(\mathsf{r}_1 + \mathsf{r}_2)$	0.018
	Total		349.037		**Total**		5.462

Notable Observations

Cryptography Research Centre

- The generation of the commitment is slower in the rank metric because the algorithm that generates an error of certain rank slow.
- The verification of the commitment is slower in the rank metric because computing the rank of a matrix is slower that computing the Hamming weight of a vector.
- The generation of matrix A (Hamming metric) is slower than G (Rank metric) because of their difference in the dimensions.
- Verification time of Zero-Knowlegde proofs in the rank metric is around 70-100 times faster than the Hamming metric.

Conclusions and Future Work

Conclusions

Cryptography Research Centre

- We showed that quantum-resistant commitments and zero-knowledge proofs can be built upon the Rank Syndrome Decoding problem.

Conclusions

Cryptography Research Centre

- We showed that quantum-resistant commitments and zero-knowledge proofs can be built upon the Rank Syndrome Decoding problem.
- Our protocol is quasi-linear in the size of the circuit and has soundness 2/3.

Conclusions

- We showed that quantum-resistant commitments and zero-knowledge proofs can be built upon the Rank Syndrome Decoding problem.
- Our protocol is quasi-linear in the size of the circuit and has soundness 2/3.
- Provide implementations of both variants (Hamming and Rank) with parameters achieving 128 bits of security.

Future work

Cryptography Research Centre

- Use structured codes (quasi-cyclic) to further improve efficiency and performance.
- Look for a better proof construction than iterative challenge response.
- Design and implementation of the 5-pass version.

References I

Gaborit, P., Schrek, J., and Zémor, G. (2011). Full cryptanalysis of the chen identification protocol. In Yang, B.-Y., editor, *Post-Quantum Cryptography*, pages 35–50, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jain, A., Krenn, S., Pietrzak, K., and Tentes, A. (2012). Commitments and efficient zero-knowledge proofs from learning parity with noise. In Wang, X. and Sako, K., editors, *Advances in Cryptology – ASIACRYPT 2012*, pages 663–680, Berlin, Heidelberg. Springer Berlin Heidelberg.

Technology Innovation Institute

Cryptography Research Centre

tii.ae