Cryptanalysis of Two Variants of the McEliece Cryptosystem

Ayoub Otmani [1]

Ayoub.Otmani@info.unicaen.fr

Léonard Dallot [1]

Leonard.Dallot@info.unicaen.fr

Jean-Pierre Tillich [2]

jean-pierre.tillich@inria.fr

[1] GREYC - Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen (UMR 6072)

[2] Équipe-projet Secret, INRIA-Rocquencourt

Séminaire ALI/SALSA. April 3, 2009.

I. Background

Introduction

- **Asymmetric cryptography concepts** introduced by Diffie & Hellman ('76)

- Rivest, Shamir & Adleman invented RSA ('77)
  - **First** asymmetric cryptosystem
  - Widely accepted for practical uses
  - **Extensively** studied that induces (too?) many security recommendations

- But, **alternative** cryptosystems exist ... such as McEliece cryptosystem

McEliece Cryptosystem

- Let $\mathfrak{F}_{n,k,t}$ be a family of codes of length n and dimension k **capable of correcting** $\leq t$ errors.
- Cryptosystem described by **three** algorithms:

1. $(PK, SK) \longleftarrow \texttt{Setup}(\mathbf{1}^{\lambda})$
2. $\mathbf{c} \in \mathbb{F}_2^n \longleftarrow \texttt{Encrypt}(\mathbf{m} \in \mathbb{F}_2^k)$
3. $\mathbf{m}' \in \mathbb{F}_2^k \longleftarrow \texttt{Decrypt}(\mathbf{c}' \in \mathbb{F}_2^n)$

McEliece.Setup

$(PK, SK) \leftarrow \texttt{Setup}(\mathbf{1}^{\lambda})$

1. Take n, k, t according to λ
2. *Randomly* choose a *generator matrix* $G' \in \mathfrak{F}_{n,k,t}$
3. *Randomly* pick:
   - $n \times n$ *permutation* matrix P
   - $k \times k$ *invertible* matrix S
4. Set $G = S \times G' \times P$ and $\gamma : \mathbb{F}_2^n \mapsto \mathbb{F}_2^k$ as the decoding algorithm associated with G'
5. Output

$$PK = (G, t) \quad \text{and} \quad SK = (S, P, \gamma)$$

McEliece.Encrypt

$\mathbf{c} \in \mathbb{F}_2^n \leftarrow \texttt{Encrypt}(\mathbf{m} \in \mathbb{F}_2^k)$

1. Pick a *random* vector $\mathbf{e} \in \mathbb{F}_2^n$ of *weight* $\leq t$
2. Output $\mathbf{c} = \mathbf{m} \times G \oplus \mathbf{e}$

McEliece.Decrypt

$\mathbf{m}' \in \mathbb{F}_2^k \leftarrow \texttt{Decrypt}(\mathbf{c}' \in \mathbb{F}_2^n)$

1. Calculate $\mathbf{z} = \mathbf{c}' \times P^{-1}$ // $\mathbf{z} = \mathbf{m} \times (S \times G') \oplus (\mathbf{e} \times P^{-1})$
2. Compute $\mathbf{y} = \gamma(\mathbf{z})$ // $\mathbf{y} = \mathbf{m} \times S$
3. Output $\mathbf{m}' = \mathbf{y} \times S^{-1}$ // $\mathbf{m}' = \mathbf{m}$

McEliece Cryptosystem – Security Assumptions

- **One-Wayness under Chosen Plaintext Attack (OW-CPA)**

  Difficult to invert `Encrypt` (*decoding attack*)

- **Private key recovery**

  Difficult to extract secret matrices or an *equivalent* secret matrix having an *efficient* decoding algorithm from the public matrix (*structural attack*)

Remark. Public code and secret code are *permutation equivalent*

McEliece Cryptosystem Security – OW-CPA

1. Decoding **random** linear codes is **NP-Hard**

E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. **On the intractability of certain coding problems**. *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

2. **Best practical** algorithms operate **exponentially** with the length and the rate

D.J. Bernstein, T. Lange, and C. Peters. Attacking and defending the mceliece cryptosystem. In *PQCrypto*, pages 31–46, 2008.

3. Permuted Goppa codes **look like** random linear codes

McEliece Cryptosystem – Private Key Recovery

- Hardness does **not** come from the *problem of permutation equivalence* because in practise *Support Splitting Algorithm* **easily** solves it

  N. Sendrier. **Finding the permutation between equivalent codes: the support splitting algorithm**. IEEE Transactions on Information Theory, vol. 46, no. 4, pages 1193-1203, July 2000.

- But rather from the **huge sizes** of $\mathfrak{F}_{n,k,t}$ and the symmetric group of order n

Remark.

Original McEliece scheme **is still unbroken** unlike many other variants...

McEliece Cryptosystem Variants

Replacing Goppa codes

1. Reed-Solomon codes (Niederreiter '86)
2. Concatenated codes
3. Reed-Muller codes (Sidelnikov '94)

Insecure McEliece Cryptosystem Variants

- Reed-Solomon codes

V.M. Sidelnikov and S.O. Shestakov. **On the insecurity of cryptosystems based on generalized Reed-Solomon codes**. *Discrete Mathematics and Applications*, 1(4):439–444, 1992.

- Concatenated codes

N. Sendrier. **On the Structure of Randomly Permuted Concatenated Code.** Rapport de recherche de l'INRIA - Rocquencourt. Janvier 1995

- Reed-Muller codes.

L. Minder and A. Shokrollahi. **Cryptanalysis of the Sidelnikov cryptosystem**. In *Eurocrypt 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 347–360, Barcelona, Spain, 2007.

McEliece Cryptosystem

- **Three advantages**
  - Fast encryption/decryption algorithms
  - Original scheme still secure
  - Alternative solution to RSA for quantum computers!

- **Main drawback: huge public key**

  For instance, parameters proposed in '78 (now outdated)

  * Goppa codes with $n = 1024$, $k = 524$
  * Private key $\simeq$ 300 Kbits
  * Public key $\simeq$ 500 Kbits

Reducing Key Sizes

1. Sparse matrices

A. Shokrollahi C. Monico, J. Rosenthal. **Using low density parity check codes in the McEliece cryptosystem**. In *IEEE International Symposium on Information Theory (ISIT 2000)*, page 215, Sorrento, Italy, 2000.

2. Quasi-cyclic matrices

P. Gaborit. **Shorter keys for code based cryptography**. In *Proceedings of the 2005 International Workshop on Coding and Cryptography (WCC 2005)*, pages 81–91, Bergen, Norway, March 2005.

3. Sparse quasi-cyclic matrices

M. Baldi, G. F. Chiaraluce. **Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes**. In *IEEE International Symposium on Information Theory*, pages 2591–2595, Nice, France, March 2007.

Low Density Parity Check Codes

Some facts.

- Invented by Gallager ('68) and rediscovered by Mackay ('98)
- Linear codes defined by very sparse parity check matrices
- Iteratively decoded through Belief Propagation algorithm
- For any cryptographic use, one **has to hide the sparsity** of matrices

Notation.

$\mathfrak{L}_{n,k,t}$: family of LDPC codes of length n, dimension k and correcting capability of t errors.

LDPC Codes in the McEliece Cryptosystem

$\texttt{Setup}(1^\lambda)$

1. Randomly choose a parity check matrix $H' \in \mathfrak{L}_{n,k,t}$
2. Randomly pick sparse invertible $(n-k) \times (n-k)$ matrix T and $k \times k$ matrix S
3. Set $H = T \times H'$
4. Output $SK = (H', T)$ and $PK = (H, S, t)$

Remark.

H and H' define the same code $\mathscr{C}$.

LDPC Codes in the McEliece Cryptosystem

`Encrypt(`$\mathbf{m}$`)`

1. Compute a generator matrix G in *row reduced echelon form* from H.
2. Set $\tilde{G} = S^{-1} \times G$
3. Output $\mathbf{c} = \mathbf{m} \times \tilde{G} \oplus \mathbf{e}$

`Decrypt(`$\mathbf{c}$`)`

1. Decode $\mathbf{c}$ with H' // G and $\tilde{G}$ define the same code $\mathscr{C}$
2. Extract $\mathbf{m} \times S^{-1}$ from $\mathbf{m} \times \tilde{G}$
3. Output $\mathbf{m}$

LDPC Codes in the McEliece Cryptosystem – Security Assumption

- **Dual** of the public code **must not** have codewords of **small weight**
- It should be hard to devise a **sparse** parity check matrix $\tilde{H}$ equivalent to H'
- It turns out **not** to be the case

A. Shokrollahi, C. Monico, J. Rosenthal. **Using low density parity check codes in the McEliece cryptosystem**. In *IEEE International Symposium on Information Theory (ISIT 2000)*, page 215, Sorrento, Italy, 2000.

LDPC Codes in the McEliece Cryptosystem – Structural Attack

Notation.

- Let v_i be the ith row of a matrix V
- Let $v_i \cap v_j$ be the *intersection vector* of v_i and v_j

Basic observation. T and H' are (very) sparse matrices

With *non-negligible* probability, for **many** ℓ, there exist i, j such that

$$h'_\ell = h_i \cap h_j$$

Secret Parity Check Matrix Recovery

1. for any i, j do compute $v = h_i \cap h_j$
2. if $v \in \mathscr{C}$ then $\mathcal{B} = \mathcal{B} \cup \{v\}$
3. for any ℓ do
4. if $wt(h_\ell \oplus v) < wt(h_\ell)$ then
5. $h_\ell = h_\ell \oplus v$
6. end if
7. end for
8. *Goto 1*
9. end if
10. end for;
11. Output $\mathcal{B}$

II. Quasi-Cyclic Codes

Circulant Matrix

Definition.

- M is a *circulant* $p \times p$ matrix if

$$M = \begin{pmatrix} m_0 & m_1 & \cdots & m_{p-1} \\ m_{p-1} & m_0 & \cdots & m_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ m_1 & m_2 & \cdots & m_0 \end{pmatrix}$$

- *Weight* of M is the weight of $\mathbf{m} = (m_0, \ldots, m_{p-1})$

Notation.

$$M \quad \longmapsto \quad \mathbf{m}(x) = m_0 + m_1 x + \cdots m_{p-1} x^{p-1}$$

Circulant Matrix

Properties. Let M and N be circulant $p \times p$ matrices

- $M + N$ is circulant

$$M + N \longmapsto \mathbf{m}(x) + \mathbf{n}(x)$$

- $M \times N$ is circulant

$$M \times N \longmapsto \mathbf{m}(x) \cdot \mathbf{n}(x) \mod (x^p - 1)$$

- M^T is circulant

$$M^T \longmapsto \mathbf{m}(\frac{1}{x}) \cdot x^p$$

- M is **invertible** iff $\mathbf{m}(x)$ is **coprime** with $x^p - 1$

Circulant-by-Block Matrix

Definition. $M = [M_{i,j}]$ is *circulant-by-block* if $M_{i,j}$ is a circulant $p \times p$ matrix

$$M \longmapsto \mathbf{M}(x) = [\mathbf{m}_{i,j}(x)]$$

Properties. Let M and N be circulant-by-block matrices

- $M + N$, $M \times N$, M^T are also circulant-by-block matrices
- M is invertible iff $\det(\mathbf{M})(x)$ is coprime with $(x^p - 1)$
- M^{-1} is a circulant-by-block matrix

Quasi-Cylic Codes

- Let $n = pn_0$ and $r = pr_0$ with p, n_0 and r_0 positive integers
- Let H be an $r \times n$ *parity check matrix* of a code $\mathscr{C}$

Definition.

$\mathscr{C}$ is **quasi-cyclic** if $H = [H_{i,j}]$ with each $H_{i,j}$ is a *circulant* $p \times p$ matrix

$\mathscr{C}$ is a **quasi-cyclic low density parity check** code if each $H_{i,j}$ is *sparse*

Notation.

$$H \longmapsto \mathbf{H}(x) = [\mathbf{h}_{i,j}(x)]$$

III. Cryptanalysis of a McEliece Cryptosystem Based on Quasi-Cyclic Subcodes of BCH Codes

McEliece Cryptosystem Based on Quasi-Cyclic Subcodes of BCH Codes ('05)

P. Gaborit. **Shorter keys for code based cryptography**. *Proceedings of Workshop on Codes and Cryptography*, Bergen, (2005), page 81-90.

In a nutshell.

- Let $\mathscr{C}_0$ be a *cyclic* code of length $n = pn_0$, dimension $K = pK_0$ and capable of correcting t errors
- Let $\mathfrak{L}_{n,k,t}$ be the family of subcodes of $\mathscr{C}_0$ of dimension $k = K - p$
- Public code is a *quasi-cyclic* code *equivalent* to a code of $\mathfrak{L}_{n,k,t}$

Remark. The number of subcodes is $\geq 2^{K-p}$

McEliece Cryptosystem Based on Quasi-Cyclic Subcodes of BCH Codes

$\texttt{Setup}(1^\lambda)$

1. Choose a parity check matrix H_0 of $\mathscr{C}_0$
2. Randomly pick a vector $\mathbf{v} \notin \mathscr{C}_0^\perp$
3. Randomly pick a quasi-circulant generator matrix G of the code defined by the parity check matrix $\left(\begin{array}{c} H_0 \\ \hline \mathbf{v} \end{array}\right)$
4. Randomly pick an $n_0 \times n_0$ permutation matrix P
5. Calculate G in *row reduced echelon form* from H
6. Compute $G' = S^{-1} \times G \times P^{-1}$
7. Output $PK = (G', t)$ and $SK = (S, H, P)$

Cryptanalysis

Principle.

- Find an $n_0 \times n_0$ matrix X such that

$$H_0 \times (G \times X)^T = 0$$

- Secret permutation P satisfies this linear equation
- Number of unknowns is n_0^2 and number of equations is $(k-p)(n-K) = p^2(k_0-1)(n_0-K_0)$
- For the proposed parameters, we always have $p > n_0 \rightsquigarrow P$ is the unique solution!

Example.

- Parameters A: $p = 91$, $n_0 = 45$ and $k_0 = 43$
- Parameters B: $p = 89$, $n_0 = 23$ and $k_0 = 21$

IV. Cryptanalysis of a McEliece Cryptosystem Based on Quasi-Cyclic LDPC Codes

McEliece Cryptosystem Based on Quasi-Cyclic LDPC Codes ('07)

Description.

- Assume $r_0 = 1$
- Let $\mathscr{C}$ be a QC-LDPC code defined by

$$H = [H_1 \ \cdots \ H_{n_0}]$$

  where H_i is a *sparse circulant* $p \times p$ matrix of *column weight* d_v
- $\mathscr{C}$ is *able to decode* up to t' errors
- H_{n_0} has *full rank* and *dimension* of $\mathscr{C}$ is $k = p(n_0 - 1)$

McEliece Cryptosystem Based on Quasi-Cyclic LDPC Codes

$\texttt{Setup}(1^\lambda)$

1. Choose integers s, m such that $m << p$ and $t = t'/m$
2. Randomly pick *invertible* matrix
   - $S = [S_{i,j}]$ where $S_{i,j}$ is *sparse circulant* $p \times p$ matrix of *weight* s
   - $Q = [Q_{i,j}]$ where $Q_{i,j}$ is *sparse circulant* $p \times p$ matrix of *weight* m
3. Calculate a generator matrix G in *row reduced echelon form* from H
4. Compute $G' = S^{-1} \times G \times Q^{-1}$
5. Output $PK = (G', t)$ and $SK = (S, H, Q)$

McEliece Cryptosystem Based on Quasi-Cyclic LDPC Codes

`Encrypt(x)`

1. Randomly choose an error $\mathbf{e} \in \mathbb{F}_2^n$ of *weight* t
2. Calculate $\mathbf{y} = \mathbf{x} \cdot G' \oplus \mathbf{e}$

`Decrypt(y)`

1. Calculate $\mathbf{z} = \mathbf{y} \cdot Q$ // $\mathbf{z} = (\mathbf{x} \cdot S^{-1} \times G) \oplus \mathbf{e} \cdot Q$
2. Decode $\mathbf{z}$ into $\mathbf{x}'$ // $\mathbf{x}' = \mathbf{x} \cdot S^{-1}$
3. Output $\mathbf{x}' \cdot S$

Remark.

$\mathbf{e}' = \mathbf{e} \cdot Q$ is of weight $\leq mt = t'$

McEliece Cryptosystem Based on Quasi-Cyclic LDPC Codes

Proposed parameters.

- Q is chosen in *diagonal form*

$$Q = \begin{pmatrix} Q_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & Q_{n_0} \end{pmatrix}$$

- Q_i's are *invertible*

Suggested values.

- $n_0 = 4$, $p = 4032$, $d_v = 13$, $t' = 190$ and $t = 27$
- $s = m = 190/27 = 7$

Cryptosystem Analysis

Preliminaries.

- Since $H = [H_1 \ \cdots \ H_{n_0}]$ with H_{n_0} *invertible*

$$G = \left(\begin{array}{c|c} & (H_{n_0}^{-1} H_1)^T \\ I_k & \vdots \\ & (H_{n_0}^{-1} H_{n_0-1})^T \end{array} \right)$$

- This implies that k **first columns of public matrix** G' is equal to

$$G'_{\leq k} = S^{-1} \times \begin{pmatrix} Q_1^{-1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & Q_{n_0-1}^{-1} \end{pmatrix}$$

Cryptosystem Analysis

Or, equivalently by **inverting** $G_{\leq k}$ and **adopting a polynomial approach**

$$\left(\mathbf{G}'_{\leq k}\right)^{-1}(x) = \begin{pmatrix} \mathbf{q}_1(x)\cdot\mathbf{s}_{1,1}(x) & \cdots & \mathbf{q}_1(x)\cdot\mathbf{s}_{1,n_0-1}(x) \\ \vdots & & \vdots \\ \mathbf{q}_i(x)\cdot\mathbf{s}_{i,1}(x) & \cdots & \mathbf{q}_i(x)\cdot\mathbf{s}_{i,n_0-1}(x) \\ \vdots & & \vdots \\ \mathbf{q}_{n_0-1}(x)\cdot\mathbf{s}_{n_0-1,1}(x) & \cdots & \mathbf{q}_{n_0-1}(x)\cdot\mathbf{s}_{n_0-1,n_0-1}(x) \end{pmatrix}$$

where $\mathbf{q}_i(x)$ and $\mathbf{s}_{i,j}(x)$ are **sparse polynomials**: they are both of weight m and degree $< p$

Cryptosystem Analysis

Cryptanalysis principle

Given a polynomial $\mathbf{g}(x)$ of degree $< p$, **find** two polynomials $\mathbf{q}(x)$ and $\mathbf{s}(x)$ of weight $m << p$ such that

$$\mathbf{g}(x) \quad = \quad \mathbf{q}(x) \cdot \mathbf{s}(x) \mod (x^p - 1)$$

Remark.

- With high probability the weight of $\mathbf{g}(x)$ is m^2
- More precisely, with high probability there exists ℓ such that

$$\left(x^\ell \cdot \mathbf{q}(x)\right) \cap \mathbf{g}(x) = x^\ell \cdot \mathbf{q}(x)$$

Cryptosystem Analysis

Lemma.

- Let $\mathbf{q}(x)$ be a polynomial of degree $< p$ and weight m
- Let $\ell_1, \ldots, \ell_j$ be different integers $< p$
- **Randomly** pick $0 \leq \ell \leq p-1$ **different** from $\ell_1, \ldots, \ell_j$

$$\text{Prob}\left\{ \left(x^{\ell_1} + \cdots + x^{\ell_j}\right) \cdot \mathbf{q}(x) \ \cap \ x^{\ell} \cdot \mathbf{q}(x) \neq 0 \right\} \quad \leq \quad j\frac{m(m-1)}{p-j}$$

Cryptosystem Analysis

Proof.

- Set first $\mathbf{q}(x) = x^{e_1} + \cdots x^{e_m}$ and $\mathbf{r}(x) = \left(x^{\ell_1} + \cdots + x^{\ell_j}\right) \cdot \mathbf{q}(x)$
- By the union bound

$$\text{Prob}\left\{\mathbf{r}(x) \cap x^{\ell} \cdot \mathbf{q}(x) \neq 0\right\} \leq \sum_{a \in \{\ell_1, \ldots, \ell_j\}} \text{Prob}\left\{x^{a} \cdot \mathbf{q}(x) \cap x^{\ell} \cdot \mathbf{q}(x) \neq 0\right\}$$

- $\text{Prob}\left\{x^{a} \cdot \mathbf{q}_i(x) \cap x^{\ell} \cdot \mathbf{q}_i(x) \neq 0\right\}$ is **at most** the fraction of ℓ different from $\ell_1, \ldots, \ell_j$ such that there exist e_b and e_c with

$$a + e_b = \ell + e_c \mod p$$

- Thus,

$$\text{Prob}\left\{x^{a} \cdot \mathbf{q}_i(x) \cap x^{\ell} \cdot \mathbf{q}_i(x) \neq 0\right\} \leq \frac{m(m-1)}{p-j}$$

Cryptosystem Analysis

Probabilistic model.

- Let $\mathbf{q}(x)$ be a *fixed* polynomial of weight m and degree $< p$
- Let $\ell_1, \ldots, \ell_m$ be *different chosen integers uniformly and independently*
- Set $\mathbf{s}(x) = x^{\ell_1} + \cdots + x^{\ell_m}$ and $\mathbf{g}(x) = \mathbf{q}(x) \cdot \mathbf{s}(x) \mod (x^p - 1)$

Proposition.

Let ℓ be an **arbitrary** element in $\{\ell_1, \ldots, \ell_m\}$

The probability that $\mathbf{g}(x)$ contains **exactly** $x^\ell \cdot \mathbf{q}(x)$ verifies

$$\text{Prob}\left\{ x^\ell \cdot \mathbf{q}(x) \cap \mathbf{g}(x) = x^\ell \cdot \mathbf{q}(x) \right\} \geq \left(1 - \frac{m(m-1)}{p-1}\right)^{m-1}$$

Cryptosystem Analysis

Proof.

- Set $L = \{\ell_1, \ldots, \ell_m\} - \{\ell\}$
- By the independence in the choice of the $(m-1)$ integers different from ℓ

$$\text{Prob}\left\{ x^\ell \cdot \mathbf{q}(x) \cap \sum_{a \in L} x^a \cdot \mathbf{q}(x) = \emptyset \right\} = \prod_{a \in L} \text{Prob}\left\{ x^\ell \cdot \mathbf{q}(x) \cap x^a \cdot \mathbf{q}(x) = \emptyset \right\}$$

- Apply Lemma with $j = 1$ for any $a \in L$

$$\text{Prob}\left\{ x^\ell \cdot \mathbf{q}(x) \cap x^a \cdot \mathbf{q}(x) = \emptyset \right\} \quad \geq \quad 1 - \frac{m(m-1)}{p-1}$$

Numerical results. For $p = 4032$ and $m = 7$ then

$$\left(1 - \frac{m(m-1)}{p-1}\right)^{m-1} \geq 0.99$$

Cryptanalysis - First Strategy

Input. $\mathbf{g}(x)$ of weight $\leq m^2$ and degree $< p$

Output. $\mathbf{q}(x)$ and $\mathbf{s}(x)$ of weight m and degree $< p$ such that

$$\mathbf{q}(x) \cdot \mathbf{s}(x) = \mathbf{g}(x) \mod (x^p - 1)$$

---

1. Enumerate all the m-tuples $(e_1, \ldots, e_m)$ of the support of $\mathbf{g}(x)$
2. Calculate $\mathbf{q}(x) = x^{e_1} + \cdots + x^{e_m}$
3. If $\mathbf{q}(x)$ *is coprime with* $x^p - 1$ then
4. Calculate $\mathbf{s} = \mathbf{q}^{-1}(x) \cdot \mathbf{g}(x) \mod (x^p - 1)$
5. If $wt(\mathbf{s}) = m$ then
6. Return $\mathbf{q}(x)$ and $\mathbf{s}(x)$
7. end if
8. end if

Cryptanalysis - First Strategy

- **Time complexity.**

$$O\left(\binom{m^2}{m} p^2\right)$$

- **Numerical results.** For $p = 4032$ and $m = 7$, we obtain $2^{50.3}$ operations

- **Probability of success.** $\geq 99\%$

But we can do faster...

Cryptanalysis - Second Strategy

1. For each $1 \leq d \leq p-1$ do
2. $\quad \mathbf{g}_d(x) = x^d \cdot \mathbf{g}(x) \mod (x^p - 1)$
3. $\quad \mathbf{q}(x) = \mathbf{g}_d(x) \cap \mathbf{g}(x)$
4. $\quad$ If *($wt(\mathbf{q}) = m$)* and *($\mathbf{q}(x)$ coprime with $x^p - 1$)* then
5. $\quad\quad \mathbf{s}(x) = \mathbf{q}^{-1}(x) \cdot \mathbf{g}(x) \mod (x^p - 1)$
6. $\quad\quad$ If *$wt(\mathbf{s}) = m$* then
7. $\quad\quad\quad$ Return $\mathbf{q}(x)$ and $\mathbf{s}(x)$
8. $\quad\quad$ End if
9. $\quad$ End if
10. End for

Cryptanalysis - Second Strategy

- **Time complexity.**

$$O\left(p^3\right)$$

- **Numerical results.** For $p = 4032$, we obtain 2^{36} operations

- **Probability of success.** Difficult to evaluate but experimentally $\simeq 69\%$

Cryptanalysis - Second Strategy

Probabilistic model.

- Fix an integer $1 \leq d \leq p-1$
- Randomly pick $m-2$ different positive integers $\ell_1, \ldots, \ell_{m-2} \leq p-1$
- Randomly pick m different integers $e_1, \ldots, e_m \leq p-1$
- Define the polynomials

$$\mathbf{s}(x) = 1 + x^d + x^{\ell_1} + \cdots + x^{\ell_{m-2}} \quad \text{and} \quad \mathbf{q}(x) = x^{e_1} + \cdots + x^{e_m}$$

Cryptanalysis - Second Strategy

Proposition.

Let $\mathbf{g}_d(x) = x^d \cdot \mathbf{g}(x) \mod (x^p - 1)$

Then Prob$\left\{ \mathbf{g}_d(x) \cap \mathbf{g}(x) = x^d \cdot \mathbf{q}(x) \right\} \geq q$ where

$$q = \prod_{a=1}^{m-2} \left(1 - 3(a+1)\frac{m(m-1)}{p-a-1}\right) \prod_{b=1}^{m-1} \left(1 - \frac{3b}{p-b}\right)$$

Numerical values.

When $m = 7$ and $p = 4032$ then $q > 0.50$.

Secret Parity Check Matrix Extraction

- Once secret matrices S and $Q_1, \ldots, Q_{n_0-1}$ are found, calculate matrix

$$\tilde{G} = S \times G' \times \begin{pmatrix} Q_1 & & & \mathbf{0} \\ & \ddots & & \\ & & Q_{n_0-1} & \\ \mathbf{0} & & & I_p \end{pmatrix} = \left(\begin{array}{c|c} & (H_{n_0}^{-1} H_1)^T \times Q_{n_0}^{-1} \\ I_k & \vdots \\ & (H_{n_0}^{-1} H_{n_0-1})^T \times Q_{n_0}^{-1} \end{array} \right)$$

- Note that we **still need to discover** $H_1, \ldots, H_{n_0}$ and Q_{n_0}

Secret Parity Check Matrix Extraction

- Define $A_i = H_i \times H_{n_0}^{-1} \times (Q_{n_0}^{-1})^T$ and $B_{i,j} = A_i \times A_j^{-1}$
- Note that we also have:

$$B_{i,j} = H_i \times H_j^{-1}$$

- Define the code $\mathscr{C}_1$ spanned by the generator matrix G_1

$$G_1 = \begin{pmatrix} I_p & B_{2,1} & \cdots & B_{n_0-1,1} \end{pmatrix}$$

- Then we have $H_1 \times G_1 = \begin{pmatrix} H_1 & H_2 & \cdots & H_{n_0-1} \end{pmatrix}$.
- $\mathscr{C}_1$ contains codewords of small weight $(n_0 - 1)d_v = 39$.

Secret Parity Check Matrix Extraction

- Applying dedicated algorithms like CANTEAUT-CHABEAU (Time complexity is about $2^{46,75}$)

- Final step:
  1. Compute $H_i^{-1} \times A_i = H_{n_0}^{-1} \times (Q_{n_0}^{-1})^T$
  2. Apply strategy 1 or 2 to find H_{n_0} and Q_{n_0}

Conclusion

- **Key reduction** is a **crucial** issue when considering McEliece cryptosystems
- **Hiding structure** is also a **main security** issue
- **Successfully** combining these two aspects represents a **big challenge**