A Reaction Attack on LEDApkc

Tomas Fabsic, Viliam Hromada, Pavol Zajac

Slovak University of Technology in Bratislava, Slovakia

CBC 2018

Contents

Contents

Contents

Contents

Contents

LEDApkc

- It is a public-key cryptosystem submitted to NIST's Post-Quantum Cryptography Standardization Process.
- Authors: Baldi, Barenghi, Chiaraluce, Pelosi, Santini
- It is similar to the QC-LDPC McEliece cryptosystem.

LEDApkc

- It is a public-key cryptosystem submitted to NIST's Post-Quantum Cryptography Standardization Process.
- Authors: Baldi, Barenghi, Chiaraluce, Pelosi, Santini
- It is similar to the QC-LDPC McEliece cryptosystem.

LEDApkc

- It is a public-key cryptosystem submitted to NIST's Post-Quantum Cryptography Standardization Process.
- Authors: Baldi, Barenghi, Chiaraluce, Pelosi, Santini
- It is similar to the QC-LDPC McEliece cryptosystem.

Circulant matrices - definition

Definition

An $n \times n$ matrix C is circulant if it is of the form:

$$C = \begin{pmatrix} c_0 & c_1 & c_2 & \dots & c_{n-1} \\ c_{n-1} & c_0 & c_1 & \dots & c_{n-2} \\ c_{n-2} & c_{n-1} & c_0 & \dots & c_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & c_3 & \dots & c_0 \end{pmatrix}$$

Private key in LEDApkc

- Consists of two matrices: H and Q
- Both H and Q are composed of circulant blocks.

Private key in LEDApkc

- Consists of two matrices: H and Q
- Both H and Q are composed of circulant blocks.

Private key in LEDApkc - matrix H

- H is a sparse parity-check matrix for an LDPC code which is able to correct t errors.

$$H = (H_0|H_1|\dots|H_{n_0-1}),$$

where each H_i is a circulant matrix (i.e. H is quasi-cyclic (QC)).

Private key in LEDApkc - matrix Q

- Q is a sparse invertible matrix.

$$Q = \begin{pmatrix} Q_{00} & \cdots & Q_{0,n_0-1} \\ \vdots & \ddots & \vdots \\ Q_{n_0-1,0} & \cdots & Q_{n_0-1,n_0-1} \end{pmatrix},$$

where each Q_{ij} is a sparse circulant matrix.

LEDApkc - public key

- It is formed by a matrix G' such that

$$G'(HQ)^T = 0. \tag{1}$$

- G' is in the systematic form.
- G' is composed of circulant blocks.

LEDApkc - public key

- It is formed by a matrix G' such that

$$G'(HQ)^T = 0. \tag{1}$$

- G' is in the systematic form.
- G' is composed of circulant blocks.

LEDApkc - public key

- It is formed by a matrix G' such that

$$G'(HQ)^T = 0. \tag{1}$$

- G' is in the systematic form.
- G' is composed of circulant blocks.

LEDApkc - encryption

- A plaintext u encrypts to a ciphertext x as follows:

$$x = uG' + e, \tag{2}$$

where e is a randomly generated error vector with Hamming weight t.

LEDApkc - decryption

- The receiver uses a bitflipping algorithm which employes the matrices H and Q.
- The decryption fails with some probability - this probability is referred to as the decoding failure rate (DFR).

LEDApkc - decryption

- The receiver uses a bitflipping algorithm which employes the matrices H and Q.
- The decryption fails with some probability - this probability is referred to as the decoding failure rate (DFR).

Contents

Previous reaction attack

- The authors of LEDApkc recognize that the FHS+ attack presented in
  Fabsic, Hromada, Stankovski, Zajac, Guo, Johansson: A Reaction Attack on the QC-LDPC McEliece Cryptosystem, PQCrypto 2017
  is applicable to LEDApkc.

Idea of the FHS+ attack

- The attack is based on the GJS attack of Guo, Johansson and Stankovski on QC-MDPC McEliece.

Distances

Definition

We say that a distance d is present in a vector v of length p if there exist two 1s in v in positions p_1 and p_2 such that

$$d = \min\{p_1 - p_2 \mod p, \quad p_2 - p_1 \mod p\}.$$

E.g., the distance between the 1s in

$$(0, 1, 0, 0, 0, 0, 0, 1, 0)$$

is 3.

Definition

We say that a distance d is present in a $p \times p$ circulant matrix C if the distance d is present in the first row of C.

Distances

Definition

We say that a distance d is present in a vector v of length p if there exist two 1s in v in positions p_1 and p_2 such that

$$d = \min\{p_1 - p_2 \mod p, \quad p_2 - p_1 \mod p\}.$$

E.g., the distance between the 1s in

$$(0, 1, 0, 0, 0, 0, 0, 1, 0)$$

is 3.

Definition

We say that a distance d is present in a $p \times p$ circulant matrix C if the distance d is present in the first row of C.

Distances

Definition

We say that a distance d is present in a vector v of length p if there exist two 1s in v in positions p_1 and p_2 such that

$$d = \min\{p_1 - p_2 \mod p, \quad p_2 - p_1 \mod p\}.$$

E.g., the distance between the 1s in

$$(0, 1, 0, 0, 0, 0, 0, 1, 0)$$

is 3.

Definition

We say that a distance d is present in a $p \times p$ circulant matrix C if the distance d is present in the first row of C.

Distances

Definition

We say that a distance d is present in a vector v of length p if there exist two 1s in v in positions p_1 and p_2 such that

$$d = \min\{p_1 - p_2 \mod p, \quad p_2 - p_1 \mod p\}.$$

E.g., the distance between the 1s in

$$(0, 1, 0, 0, 0, 0, 0, 1, 0)$$

is 3.

Definition

We say that a distance d is present in a $p \times p$ circulant matrix C if the distance d is present in the first row of C.

Observation 1

- Suppose that the circulant blocks in H and in Q are of size $p \times p$.
- Let e be the error vector added to a message during the encryption.
- Let $e = (e^0, e^1, \ldots, e^{n/p-1})$, where each e^i has length p.

Observation

Suppose that e^i contains a distance d. If the distance d is present in some block H_j in H, then the probability that decryption fails is lower!

Observation 1

- Suppose that the circulant blocks in H and in Q are of size $p \times p$.
- Let e be the error vector added to a message during the encryption.
- Let $e = (e^0, e^1, \ldots, e^{n/p-1})$, where each e^i has length p.

Observation

Suppose that e^i contains a distance d. If the distance d is present in some block H_j in H, then the probability that decryption fails is lower!

Observation 1

- Suppose that the circulant blocks in H and in Q are of size $p \times p$.
- Let e be the error vector added to a message during the encryption.
- Let $e = (e^0, e^1, \ldots, e^{n/p-1})$, where each e^i has length p.

Observation

Suppose that e^i contains a distance d. If the distance d is present in some block H_j in H, then the probability that decryption fails is lower!

Observation 1

- Suppose that the circulant blocks in H and in Q are of size $p \times p$.
- Let e be the error vector added to a message during the encryption.
- Let $e = (e^0, e^1, \ldots, e^{n/p-1})$, where each e^i has length p.

Observation

Suppose that e^i contains a distance d. If the distance d is present in some block H_j in H, then the probability that decryption fails is lower!

Observation 2

Observation

- *If a distance d is present in e^i and at the same time it is present in one of the blocks $Q_{i,0}, \ldots, Q_{i,n_0-1}$ in the i-th block-row of Q, then v $(v = eQ)$ has smaller hamming weight then normal.*
- *Smaller hamming weight of $v \Rightarrow$ lower probability of the decoding error.*

Observation 2

Observation

- *If a distance d is present in e^i and at the same time it is present in one of the blocks $Q_{i,0}, \ldots, Q_{i,n_0-1}$ in the i-th block-row of Q, then v $(v = eQ)$ has smaller hamming weight then normal.*
- *Smaller hamming weight of $v \Rightarrow$ lower probability of the decoding error.*

Observation 2

Observation

- *If a distance d is present in e^i and at the same time it is present in one of the blocks $Q_{i,0}, \ldots, Q_{i,n_0-1}$ in the i-th block-row of Q, then v $(v = eQ)$ has smaller hamming weight then normal.*
- *Smaller hamming weight of $v \Rightarrow$ lower probability of the decoding error.*

The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

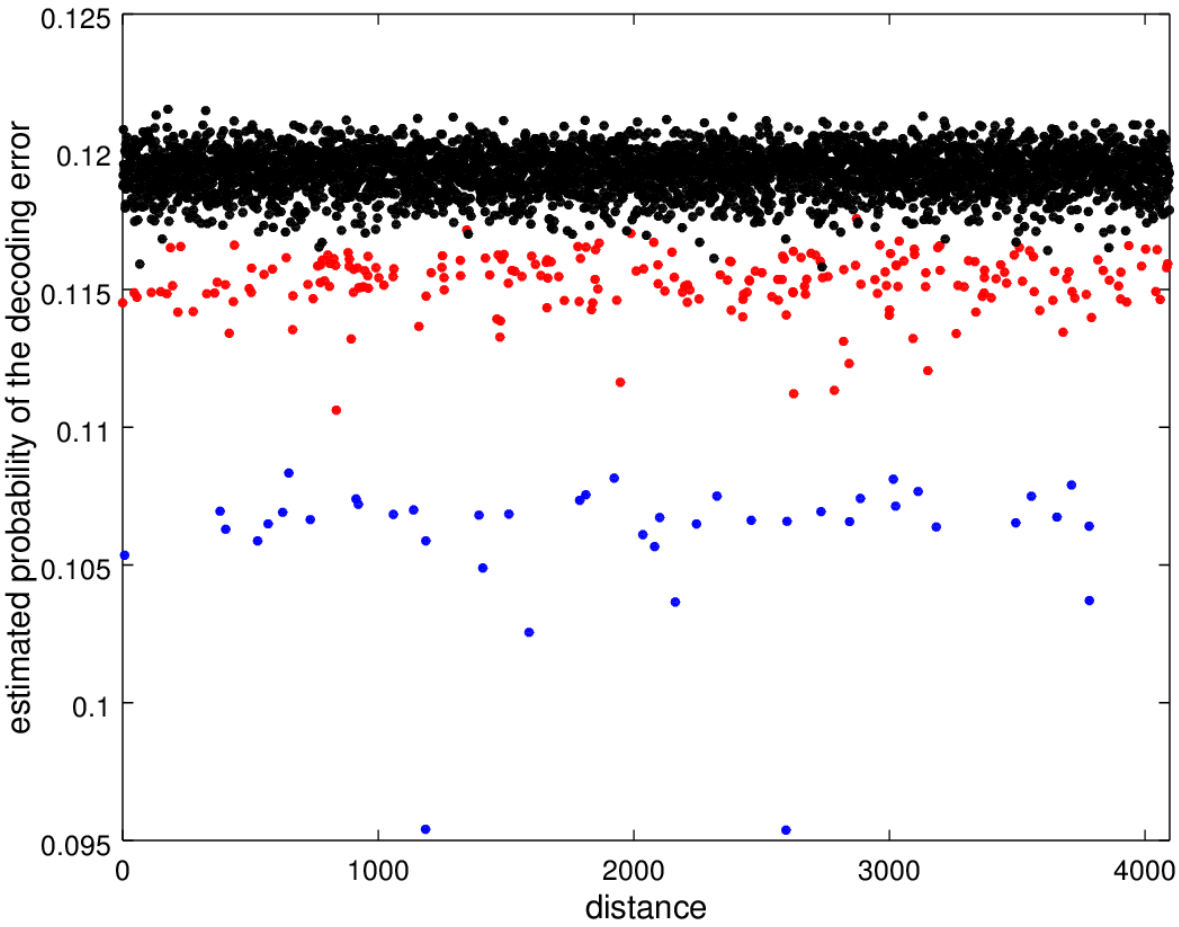
The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

The attack

1. Send a large number of encrypted messages with a randomly generated error vector e.
2. Observe when the recipient requests a message to be resend. (This means that the recipient experienced a decoding error.)
3. Group the encrypted messages into groups Σ_d according to the rule: A message belongs to Σ_d if its error vector contains the distance d in some e^i.
4. For each Σ_d estimate the probability of the decoding error.
5. Select the distances with low estimates of the probability of the decoding error. (These are the distances present in blocks of H and in blocks of Q.)
6. Reconstruct candidates for matrices H and Q.

Experiment results



Contents

Lifetime of keys in LEDApkc

- The authors of LEDApkc propose to use a key pair for a limited lifetime to avoid the FHS+ attack.
- They estimate that it is safe to use a key pair for $10^4 \times \mathrm{DFR}^{-1}$ decryptions (low number of decryptions makes it harder for the attacker to distinguish which distances are in H and Q).

Lifetime of keys in LEDApkc

- The authors of LEDApkc propose to use a key pair for a limited lifetime to avoid the FHS+ attack.
- They estimate that it is safe to use a key pair for $10^4 \times \mathrm{DFR}^{-1}$ decryptions (low number of decryptions makes it harder for the attacker to distinguish which distances are in H and Q).

Contents

Main question

- Can an attacker break LEDApkc provided a key pair is used only for $10^4 \times \mathrm{DFR}^{-1}$ decryptions?

Idea

- Use only distances in Q.
- In FHS+ attack the distances in Q were easier to distinguish than distances in H.
- Maybe the distances in Q can still be distinguished even if the attacker has information from no more than $10^4 \times \mathrm{DFR}^{-1}$ decryptions.

Idea

- Use only distances in Q.
- In FHS+ attack the distances in Q were easier to distinguish than distances in H.
- Maybe the distances in Q can still be distinguished even if the attacker has information from no more than $10^4 \times \mathrm{DFR}^{-1}$ decryptions.

Idea

- Use only distances in Q.
- In FHS+ attack the distances in Q were easier to distinguish than distances in H.
- Maybe the distances in Q can still be distinguished even if the attacker has information from no more than $10^4 \times \mathrm{DFR}^{-1}$ decryptions.

Experiment

- We ran the same experiment as in FHS+ attack.
- We restricted the number of decryptions to $10^4 \times \mathrm{DFR}^{-1}$.
- We used the recommended parameters for LEDApkc, but we increased the number of errors in the vector e (to make the experiment less computationally expensive).

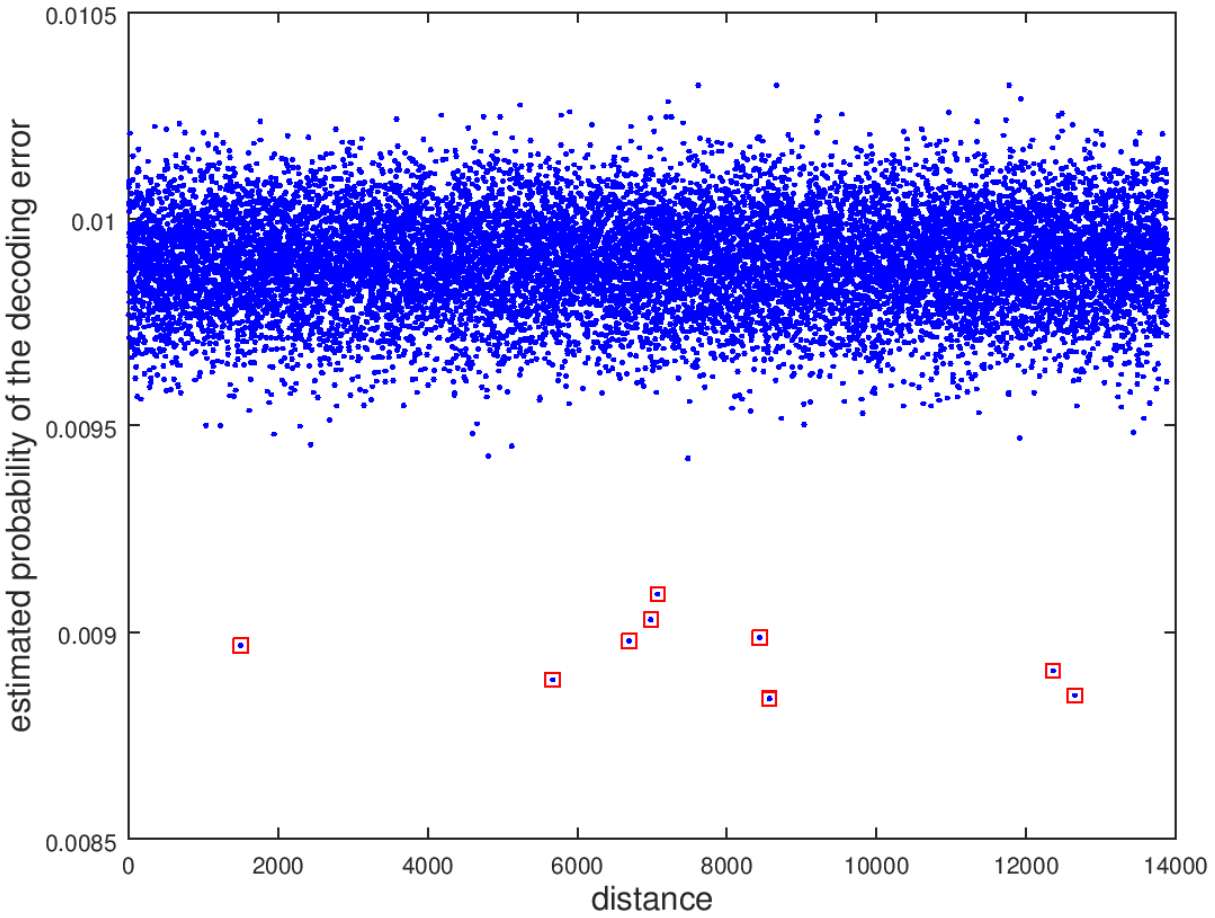
Experiment

- We ran the same experiment as in FHS+ attack.
- We restricted the number of decryptions to $10^4 \times \mathrm{DFR}^{-1}$.
- We used the recommended parameters for LEDApkc, but we increased the number of errors in the vector e (to make the experiment less computationally expensive).

Experiment

- We ran the same experiment as in FHS+ attack.
- We restricted the number of decryptions to $10^4 \times \mathrm{DFR}^{-1}$.
- We used the recommended parameters for LEDApkc, but we increased the number of errors in the vector e (to make the experiment less computationally expensive).

Experiment results



Attack

1. Send at most $10^4 \times \mathrm{DFR}^{-1}$ ciphertexts and always observes whether the ciphertext was successfully decrypted or not.
2. Learn distances in Q.
3. Build a set of candidates for Q.
4. Build candidates for the matrix $G = G'(Q^T)$.
   (Matrix G is a generator matrix of the secret LDPC code with parity-check matrix H.)
5. Apply Stern's algorithm on candidates for the matrix G to find a low-weight codeword in the dual of the secret LDPC code and thus recover the matrix H.

Attack

1. Send at most $10^4 \times \mathrm{DFR}^{-1}$ ciphertexts and always observes whether the ciphertext was successfully decrypted or not.
2. Learn distances in Q.
3. Build a set of candidates for Q.
4. Build candidates for the matrix $G = G'(Q^T)$.
   (Matrix G is a generator matrix of the secret LDPC code with parity-check matrix H.)
5. Apply Stern's algorithm on candidates for the matrix G to find a low-weight codeword in the dual of the secret LDPC code and thus recover the matrix H.

Attack

1. Send at most $10^4 \times \mathrm{DFR}^{-1}$ ciphertexts and always observes whether the ciphertext was successfully decrypted or not.
2. Learn distances in Q.
3. Build a set of candidates for Q.
4. Build candidates for the matrix $G = G'(Q^T)$.
   (Matrix G is a generator matrix of the secret LDPC code with parity-check matrix H.)
5. Apply Stern's algorithm on candidates for the matrix G to find a low-weight codeword in the dual of the secret LDPC code and thus recover the matrix H.

Attack

1. Send at most $10^4 \times \mathrm{DFR}^{-1}$ ciphertexts and always observes whether the ciphertext was successfully decrypted or not.
2. Learn distances in Q.
3. Build a set of candidates for Q.
4. Build candidates for the matrix $G = G'(Q^T)$.
   (Matrix G is a generator matrix of the secret LDPC code with parity-check matrix H.)
5. Apply Stern's algorithm on candidates for the matrix G to find a low-weight codeword in the dual of the secret LDPC code and thus recover the matrix H.

Attack

1. Send at most $10^4 \times \mathrm{DFR}^{-1}$ ciphertexts and always observes whether the ciphertext was successfully decrypted or not.
2. Learn distances in Q.
3. Build a set of candidates for Q.
4. Build candidates for the matrix $G = G'(Q^T)$.
   (Matrix G is a generator matrix of the secret LDPC code with parity-check matrix H.)
5. Apply Stern's algorithm on candidates for the matrix G to find a low-weight codeword in the dual of the secret LDPC code and thus recover the matrix H.

Performance of the attack

- Provided the adversary can learn distances in Q, the complexity of the attack is below 2^{99} for a parameter set for 128-bit security.

Limitations of the attack

- We only verified that the attacker can learn distances in Q in experiments where the number of errors in the vector e was artificially increased.
- The complexity of the attack rises significantly if the number of circulant blocks in Q increases.
  (To achieve a complexity below the security level we therefore focused on the recommended parameter set with the minimum number of blocks in Q.)

Limitations of the attack

- We only verified that the attacker can learn distances in Q in experiments where the number of errors in the vector e was artificially increased.
- The complexity of the attack rises significantly if the number of circulant blocks in Q increases.
  (To achieve a complexity below the security level we therefore focused on the recommended parameter set with the minimum number of blocks in Q.)

The End

Thank you for your attention!