Code-Based Cryptography

McEliece Cryptosystem

I. Márquez-Corbella

informatiques mathématiques

2. McEliece Cryptosystem

1. Formal Definition
2. Security-Reduction Proof
3. McEliece Assumptions
4. **Notions of Security**
5. Critical Attacks - Semantic Secure Conversions
6. Reducing the Key Size
7. Reducing the Key Size - LDPC codes
8. Reducing the Key Size - MDPC codes
9. Implementation

One-Wayness property

Let Π be a cryptosystem.

Π is **One-Wayness** $\iff$ The probability of success of any adversary **running in polynomial time** is negligible

One-Wayness property

Let Π be a cryptosystem.

Π is **One-Wayness** $\iff$ The probability of success of any adversary **running in polynomial time** is negligible

Without the private key it is **computationally impossible** to recover the plaintext

One-Wayness property

Let Π be a cryptosystem.

Π is **One-Wayness** $\iff$ The probability of success of any adversary **running in polynomial time** is negligible

Without the private key it is **computationally impossible** to recover the plaintext

If we assume that:

1. Decoding a random linear code is HARD.
2. Goppa codes are pseudorandom

One-Wayness property

Let Π be a cryptosystem.

Π is **One-Wayness** $\iff$ The probability of success of any adversary **running in polynomial time** is negligible

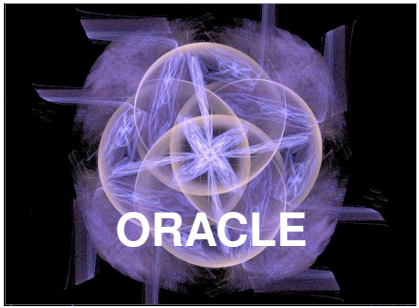
Without the private key it is **computationally impossible** to recover the plaintext

If we assume that:

1. Decoding a random linear code is HARD.
2. Goppa codes are pseudorandom

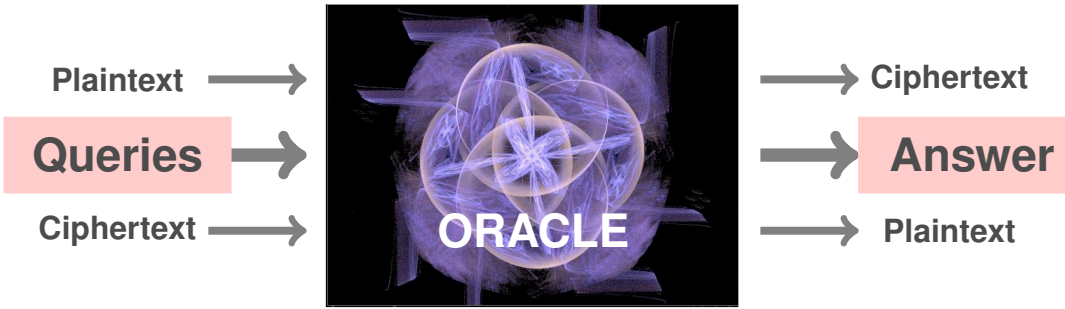
$\Longrightarrow$ **McEliece is a OW scheme**

Oracle

Queries → 

→ Answer

Oracle


Plaintext
Queries
Ciphertext
ORACLE
Ciphertext
Answer
Plaintext

Goal 1: Non-malleability

Given: $\mathbf{y}_1 = \texttt{Encrypt}\left(\mathbf{m}_1, \mathrm{K}_p\right)$

Goal: Find $\mathbf{y}_2 = \texttt{Encrypt}\left(\mathbf{m}_2, \mathrm{K}_p\right)$

such that a relationship exists between $\mathbf{m}_1$ and $\mathbf{m}_2$

D. Dolve, C. Dwork and M. Naor.
Non-Malleable Cryptography.
In Proc. of the 23rd STOC, 1991.

McEliece does not satisfy Non-Malleability

1. The adversary intercept a ciphertext

$$\mathbf{y} = \mathbf{m}G + \mathbf{e}$$

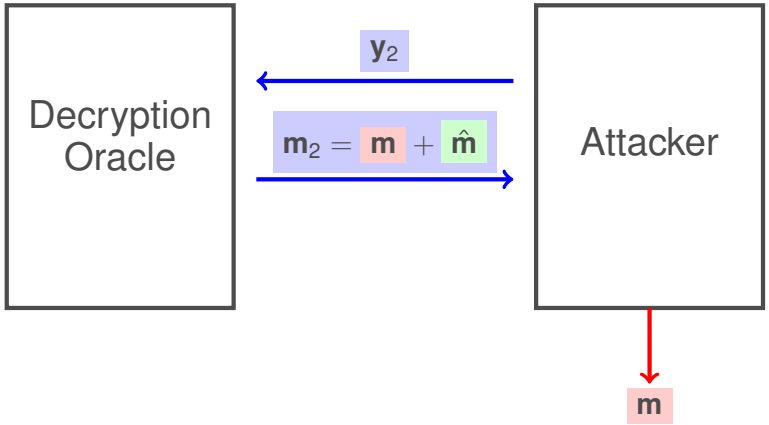
2. With the **public-key** G_{Pub} he can choose a codeword: $\hat{\mathbf{c}} = \hat{\mathbf{m}}G_{\text{Pub}}$
3. Now, the adversary can generate a new ciphertext:

$$\mathbf{y}_2 = \mathbf{y} + \hat{\mathbf{c}} = \underbrace{\left(\mathbf{m} + \hat{\mathbf{m}}\right)}_{\mathbf{m}_2} G_{\text{Pub}} + e$$

The plaintext of the new ciphertext is: $\mathbf{m}_2 = \mathbf{m} + \hat{\mathbf{m}}$

McEliece does not satisfy Non-Malleability

Suppose that the adversary has acces to a decryption oracle



Goal 2: Indistinguishability - Semantic Security

Given: $\mathbf{y}_1 = \texttt{Encrypt}\left(\mathbf{m}_1, K_p\right)$

Goal (Indistinguishability): Learn something about $\mathbf{m}_1$

S. Goldwasser and S. Micali.
Probabilistic encryption.
Journal of Computer and System Sciences, 270-299, 1984.

Goal 2: Indistinguishability - Semantic Security

Given: $y_1 = \texttt{Encrypt}\left(m_1, K_p\right)$

Goal (Indistinguishability): Learn something about m_1

Goal (Non-Malleability): Find $y_2 = \texttt{Encrypt}\left(m_2, K_p\right)$

such that a relationship exists between m_1 and m_2

S. Goldwasser and S. Micali.
Probabilistic encryption.
Journal of Computer and System Sciences, 270-299, 1984.

Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

Attack Models 1 - CPA

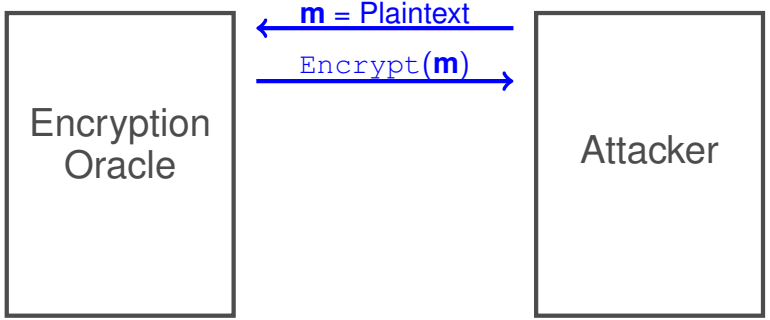
Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

This is inevitable in **Public-Key Schemes**

Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

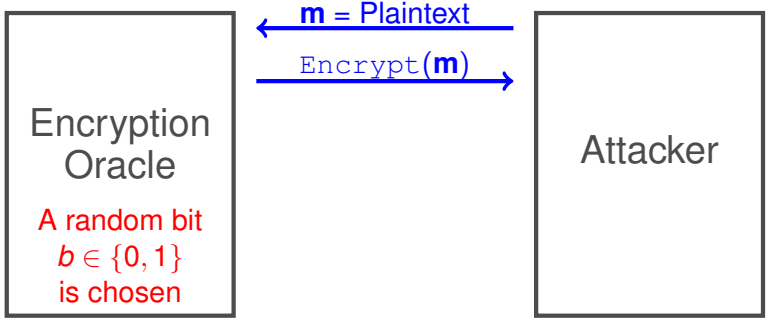
This is inevitable in **Public-Key Schemes**



Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

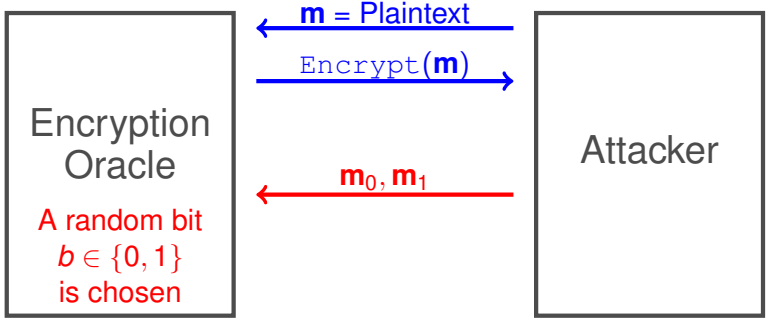
This is inevitable in **Public-Key Schemes**



Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

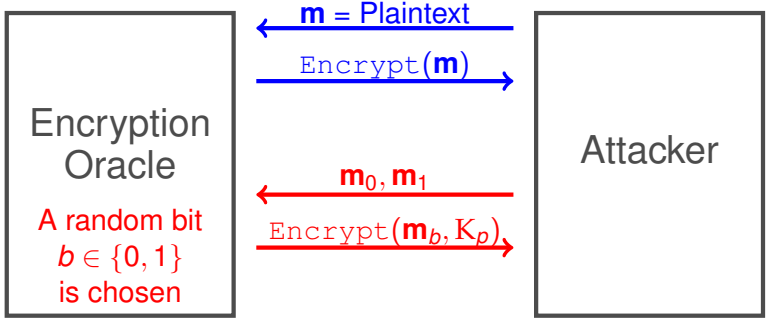
This is inevitable in **Public-Key Schemes**



Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

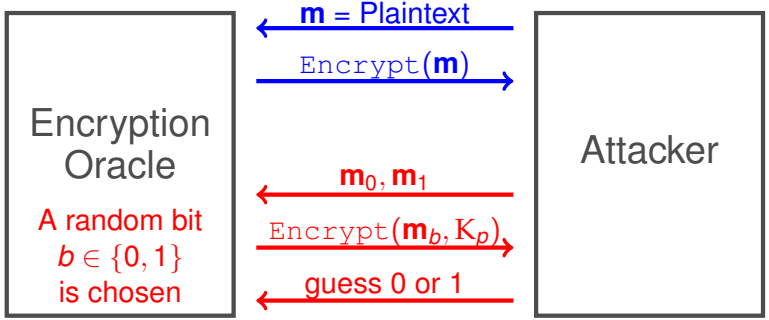
This is inevitable in **Public-Key Schemes**



Attack Models 1 - CPA

Chosen Plaintext Attack (CPA): The adversary can encrypt any message of his choice.

This is inevitable in **Public-Key Schemes**



Attack Models 2 - CCA1 and CCA2

Chosen Ciphertext Attack (CCA): The adversary gets acces to an oracle for the decryption function.

Attack Models 2 - CCA1 and CCA2

Chosen Ciphertext Attack (CCA): The adversary gets acces to an oracle for the decryption function.

- **Chosen Ciphertext Attack (CCA1):**

  The adversary can use this oracle before it gets the challenge ciphertext. The queries cannot depend on the ciphertext $\mathcal{C}$.

Attack Models 2 - CCA1 and CCA2

Chosen Ciphertext Attack (CCA): The adversary gets acces to an oracle for the decryption function.

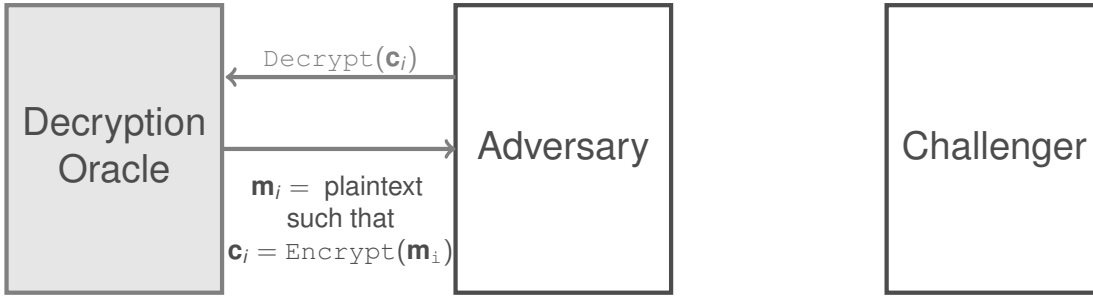
- **Chosen Ciphertext Attack (CCA1):**

  The adversary can use this oracle before it gets the challenge ciphertext. The queries cannot depend on the ciphertext $\mathcal{C}$.

- **Adaptative Chosen Ciphertext Attack (CCA2):**

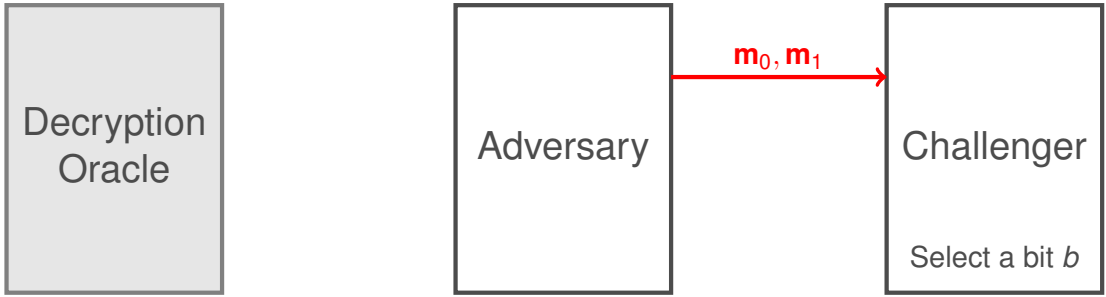
  The adversary gets acces to a decryption oracle without restrictions.

Attack Models 2 - CCA1 and CCA2

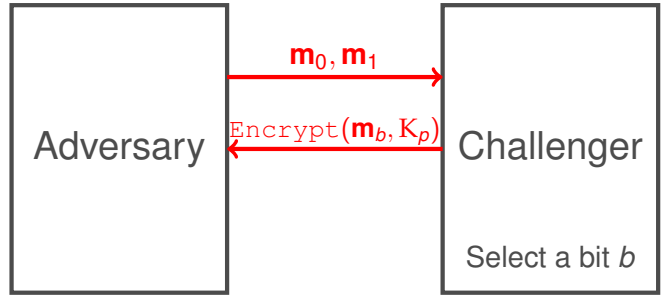


Attack Models 2 - CCA1 and CCA2

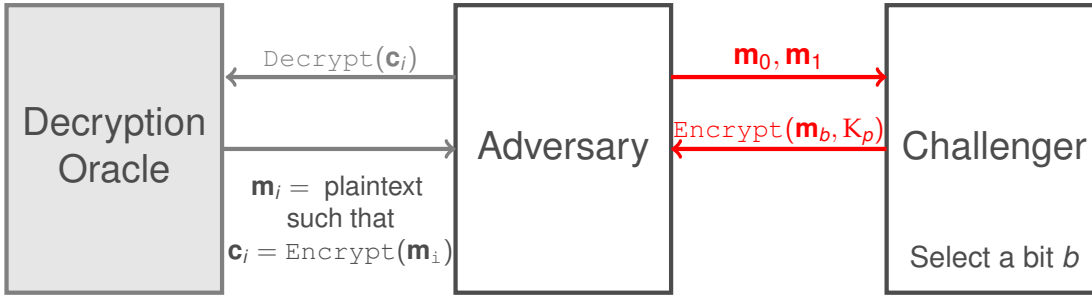
Attack Models 2 - CCA1 and CCA2



Attack Models 2 - CCA1 and CCA2

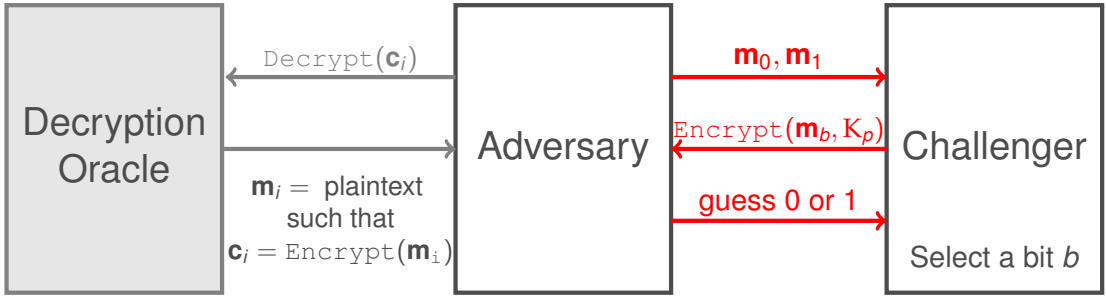


Attack Models 2 - CCA1 and CCA2



CCA2

Attack Models 2 - CCA1 and CCA2



CCA2

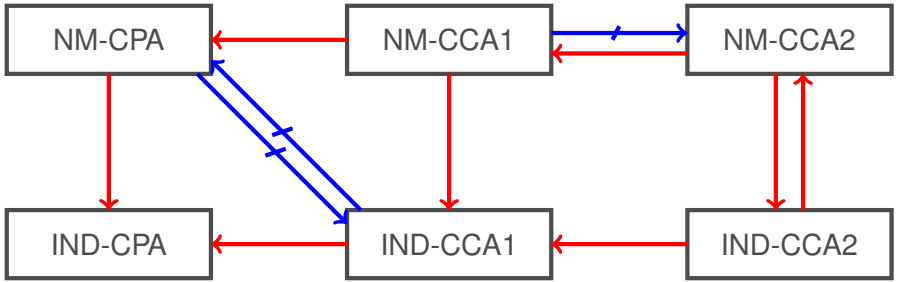
Implications and Separations

One can *m*ix-and-match the **goals** and the **attacks**: $\left\{ \begin{array}{lll} IND-CPA, & IND-CCA1, & IND-CCA2, \\ NM-CPA, & NM-CCA1, & NM-CCA2 \end{array} \right\}$

M. Bellare, A. Desai, D. Pointcheval and P. Rogaway.
Relations Among Notions of Security for Public-Key Encryption Schemes.
Crypto 98. Lecture Notes in Computer Science. Vol 1462.

Implications and Separations

One can *mix*-and-match the **goals** and the **attacks**: $\left\{ \begin{array}{lll} IND-CPA, & IND-CCA1, & IND-CCA2, \\ NM-CPA, & NM-CCA1, & NM-CCA2 \end{array} \right\}$

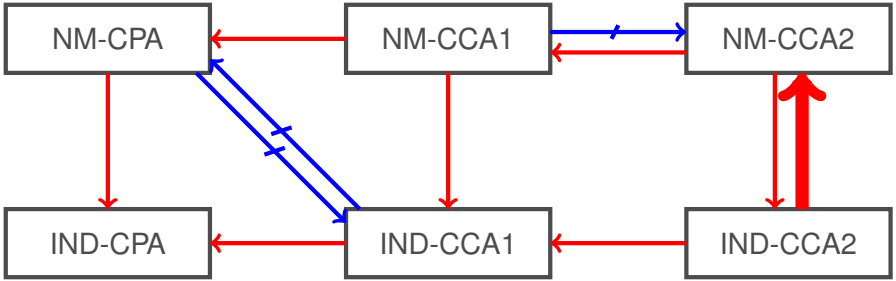


Implications: $A \rightarrow B$: B provides stronger notion of security compared to A

Separations: $A \not\rightarrow B$: There exists an encryption scheme which is secure in the sense of A but which is not secure in the sence of B

Implications and Separations

One can *m*ix-and-match the **goals** and the **attacks**: $\left\{ \begin{array}{lll} IND-CPA, & IND-CCA1, & IND-CCA2, \\ NM-CPA, & NM-CCA1, & NM-CCA2 \end{array} \right\}$

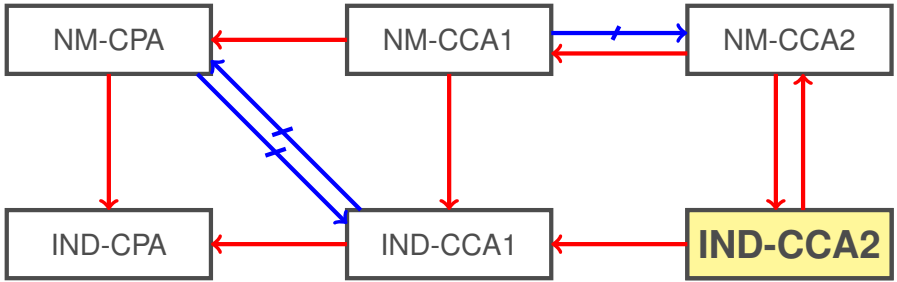


Implications: $A \rightarrow B$: B provides stronger notion of security compared to A

Separations: $A \not\rightarrow B$: There exists an encryption scheme which is secure in the sense of A but which is not secure in the sence of B

Implications and Separations

One can *m*ix-and-match the **goals** and the **attacks**: $\left\{ \begin{array}{lll} IND-CPA, & IND-CCA1, & IND-CCA2, \\ NM-CPA, & NM-CCA1, & NM-CCA2 \end{array} \right\}$

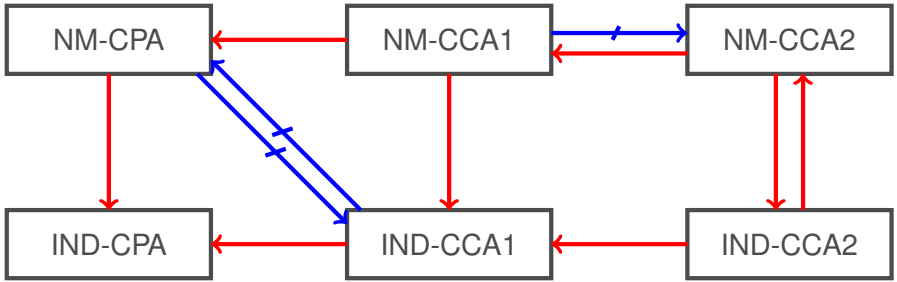


Implications: $A \rightarrow B$: B provides stronger notion of security compared to A

Separations: $A \not\rightarrow B$: There exists an encryption scheme which is secure in the sense of A but which is not secure in the sence of B

Implications and Separations

One can *mix*-and-match the **goals** and the **attacks**: $\left\{ \begin{array}{lll} IND-CPA, & IND-CCA1, & IND-CCA2, \\ NM-CPA, & NM-CCA1, & NM-CCA2 \end{array} \right\}$



Implications: $A \rightarrow B$: B provides stronger notion of security compared to A

Separations: $A \not\rightarrow B$: There exists an encryption scheme which is secure in the sense of A but which is not secure in the sence of B

2. McEliece Cryptosystem

1. Formal Definition
2. Security-Reduction Proof
3. McEliece Assumptions
4. Notions of Security
5. **Critical Attacks - Semantic Secure Conversions**
6. Reducing the Key Size
7. Reducing the Key Size - LDPC codes
8. Reducing the Key Size - MDPC codes
9. Implementation