Revisiting (R)CCA Security and Replay Protection

Christian Badertscher, Ueli Maurer, Christopher Portmann, **Guilherme Rito**

Public Key Cryptography 2021
May 2021

Introduction

Motivation Security notions are designed with concrete applications in mind.

Problem However, they often do not match the exact requirements of their applications.

What is their purpose then?

Introduction

Motivation Security notions are designed with concrete applications in mind.

Problem However, they often do not match the exact requirements of their applications.

What is their purpose then?

Introduction

Motivation Security notions are designed with concrete applications in mind.

Problem However, they often do not match the exact requirements of their applications.

What is their purpose then?

Introduction

Motivation Security notions are designed with concrete applications in mind.

Problem However, they often do not match the exact requirements of their applications.

What is their purpose then?

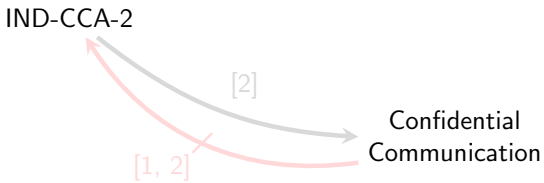
Introduction - Problem

What is the purpose of CCA-2 security for PKE schemes?

[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

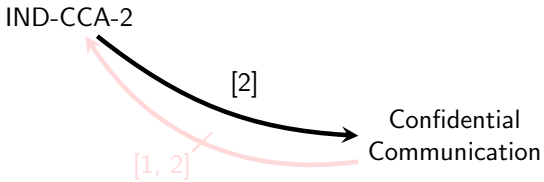
What is the purpose of CCA-2 security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

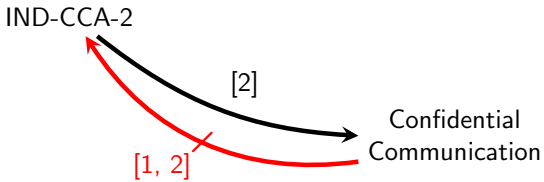
What is the purpose of CCA-2 security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

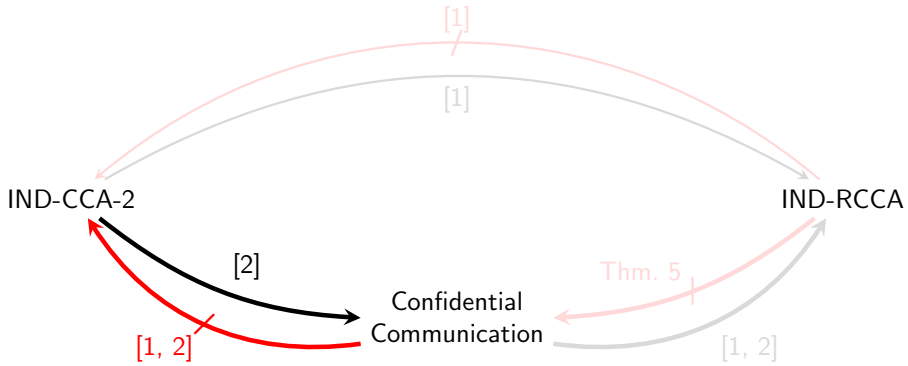
What is the purpose of CCA-2 security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

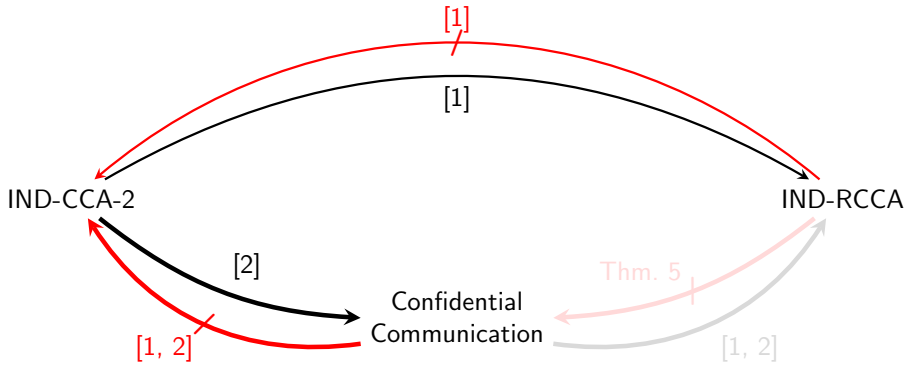
What is the purpose of RCCA security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

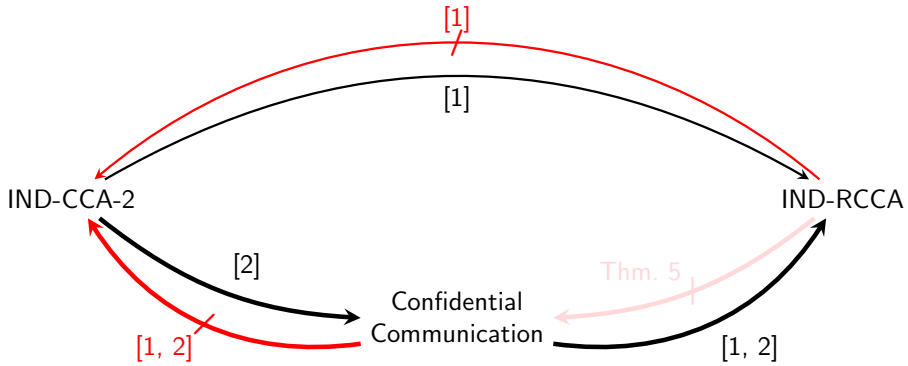
What is the purpose of RCCA security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

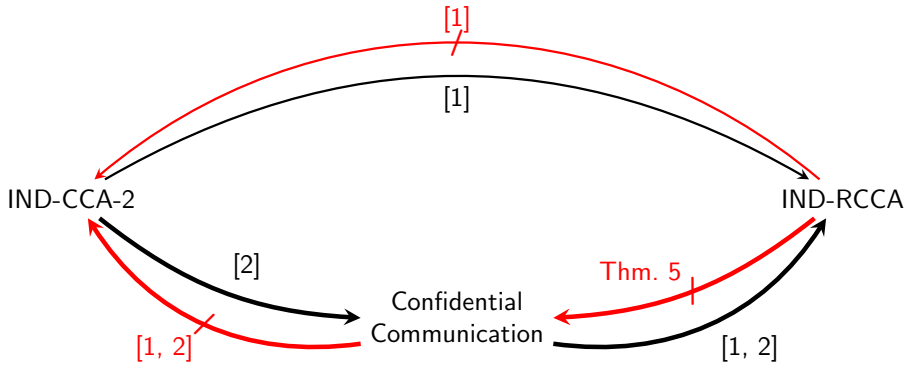
What is the purpose of RCCA security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Introduction - Problem

What is the purpose of RCCA security for PKE schemes?



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions I

Identify technical inconsistencies with intermediate notions (IND-pd-RCCA, IND-sd-RCCA):

- Not implied by IND-CCA-2 for probabilistic decryption
  (contradicting a claim made in [1]);
- No (known) operational meaning.

[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

Contributions I

Identify technical inconsistencies with intermediate notions (IND-pd-RCCA, IND-sd-RCCA):

- Not implied by IND-CCA-2 for probabilistic decryption (contradicting a claim made in [1]);
- No (known) operational meaning.

[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

Contributions I

Identify technical inconsistencies with intermediate notions (IND-pd-RCCA, IND-sd-RCCA):

- Not implied by IND-CCA-2 for probabilistic decryption
  (contradicting a claim made in [1]);
- No (known) operational meaning.

[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

Contributions I

Identify technical inconsistencies with intermediate notions (IND-pd-RCCA, IND-sd-RCCA):

- Not implied by IND-CCA-2 for probabilistic decryption
  (contradicting a claim made in [1]);
- No (known) operational meaning.

[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

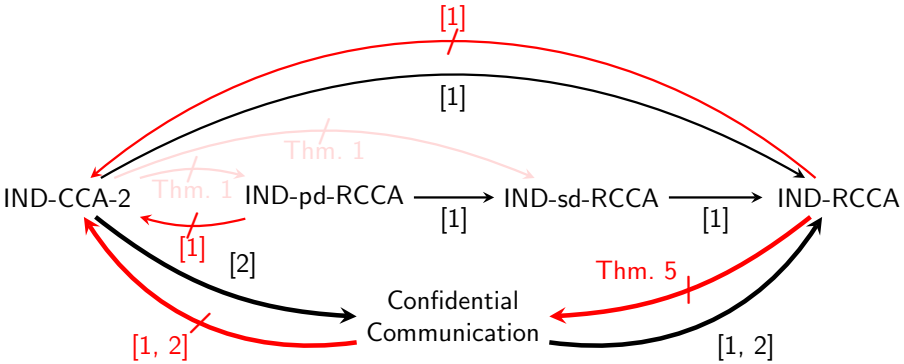
Contributions I

Identify technical inconsistencies with intermediate notions (IND-pd-RCCA, IND-sd-RCCA):

- Not implied by IND-CCA-2 for probabilistic decryption
  (contradicting a claim made in [1]);
- No (known) operational meaning.

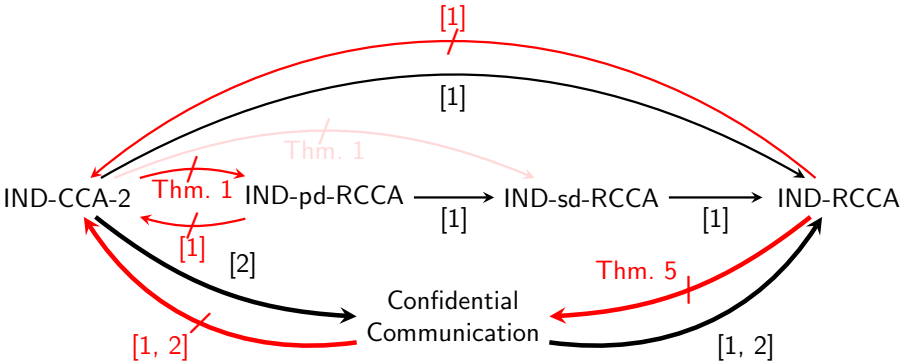
[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

Contributions I



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).
[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

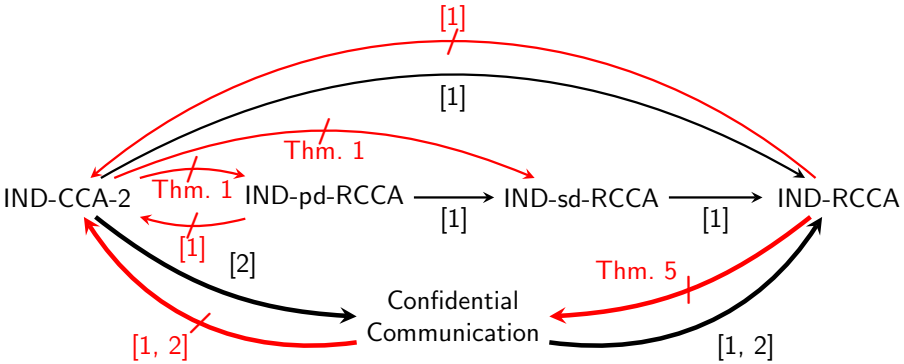
Contributions I



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions I



[1] Canetti, Krawczyk and Nielsen (CRYPTO '03).

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

- Systematic approach for characterizing security using a composable framework:
  - Define applications of PKE schemes as benchmarks (i.e. composable notions):
    - Benchmark 1:
      Confidential communication (introduced in [2])
    - Benchmark 2:
      Confidential communication with secret replay protection
    - Benchmark 3:
      Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

Systematic approach for characterizing security using a composable framework:

Define applications of PKE schemes as benchmarks (i.e. composable notions):

Benchmark 1:
Confidential communication (introduced in [2])

Benchmark 2:
Confidential communication with secret replay protection

Benchmark 3:
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

Systematic approach for characterizing security using a composable framework:

Define applications of PKE schemes as benchmarks (i.e. composable notions):

Benchmark 1:
Confidential communication (introduced in [2])
Benchmark 2:
Confidential communication with secret replay protection
Benchmark 3:
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

- Systematic approach for characterizing security using a composable framework:
  - Define applications of PKE schemes as benchmarks (i.e. composable notions):
    - Benchmark 1:
      Confidential communication (introduced in [2])
    - Benchmark 2:
      Confidential communication with secret replay protection
    - Benchmark 3:
      Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

- Systematic approach for characterizing security using a composable framework:
  - Define applications of PKE schemes as benchmarks (i.e. composable notions):
    - Benchmark 1:
      Confidential communication (introduced in [2])
    - Benchmark 2:
      Confidential communication with secret replay protection
    - Benchmark 3:
      Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Clean up the space of game-based notions for confidential communication:

- Systematic approach for characterizing security using a composable framework:
  - Define applications of PKE schemes as benchmarks (i.e. composable notions):
    - Benchmark 1:
      Confidential communication (introduced in [2])
    - Benchmark 2:
      Confidential communication with secret replay protection
    - Benchmark 3:
      Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II - New Benchmarks

Figure: New security notions are marked with *.

IND-CCA-2 IND-RCCA

Benchmark 3* Benchmark 2* Benchmark 1

Contributions II - New Benchmarks

Figure: New security notions are marked with *.

IND-CCA-2

IND-RCCA

Benchmark 3* Benchmark 2* Benchmark 1

Contributions II - New Benchmarks

Figure: New security notions are marked with *.

IND-CCA-2

IND-RCCA

Benchmark 3*

Benchmark 2*

Benchmark 1

Contributions II - New Benchmarks

Figure: New security notions are marked with *.

IND-CCA-2

IND-RCCA

Benchmark 3* Benchmark 2* Benchmark 1

Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1:
Confidential communication (introduced in [2])

Benchmark 2:
Confidential communication with secret replay protection

Benchmark 3:
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1: IND-cl-RCCA
Confidential communication (introduced in [2])

Benchmark 2:
Confidential communication with secret replay protection

Benchmark 3:
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1: IND-cl-RCCA
Confidential communication (introduced in [2])

Benchmark 2: IND-srp-RCCA
Confidential communication with secret replay protection

Benchmark 3:
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1: IND-cl-RCCA
Confidential communication (introduced in [2])

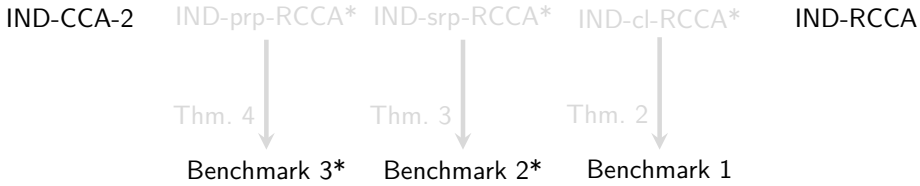
Benchmark 2: IND-srp-RCCA
Confidential communication with secret replay protection

Benchmark 3: IND-prp-RCCA
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

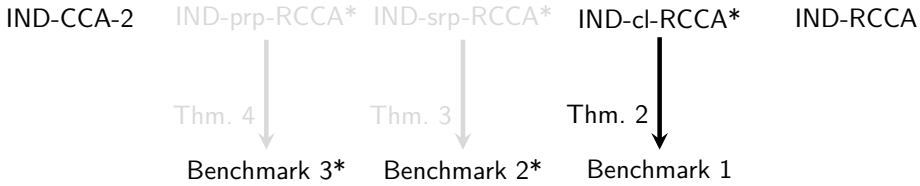
Contributions II

Figure: New security notions are marked with *.



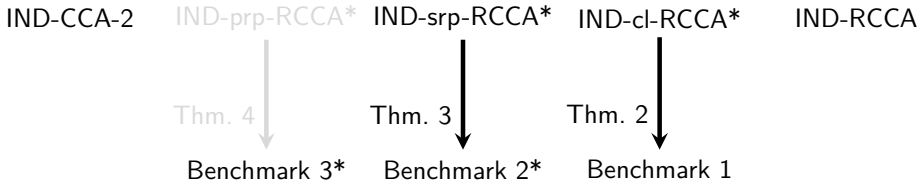
Contributions II

Figure: New security notions are marked with *.



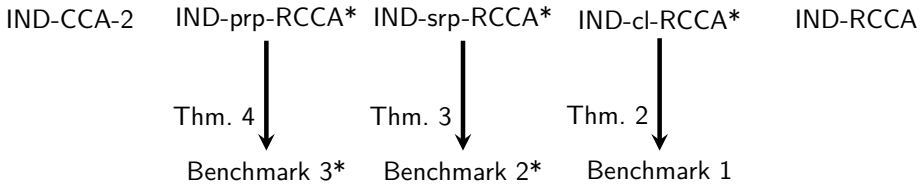
Contributions II

Figure: New security notions are marked with *.



Contributions II

Figure: New security notions are marked with *.



Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1: IND-cl-RCCA
Confidential communication (introduced in [2])

Benchmark 2: IND-srp-RCCA
Confidential communication with secret replay protection

Benchmark 3: IND-prp-RCCA
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Propose new game-based notions capturing the applications:

Benchmark 1: IND-cl-RCCA
Confidential communication (introduced in [2])

Benchmark 2: IND-srp-RCCA
Confidential communication with secret replay protection

Benchmark 3: IND-prp-RCCA
Confidential communication with public replay protection

[2] Coretti, Maurer and Tackmann (ASIACRYPT '13).

Contributions II

Fully characterize our game-based notions:

- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.

Contributions II

Fully characterize our game-based notions:

- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.

Contributions II

Fully characterize our game-based notions:

- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.

Contributions II

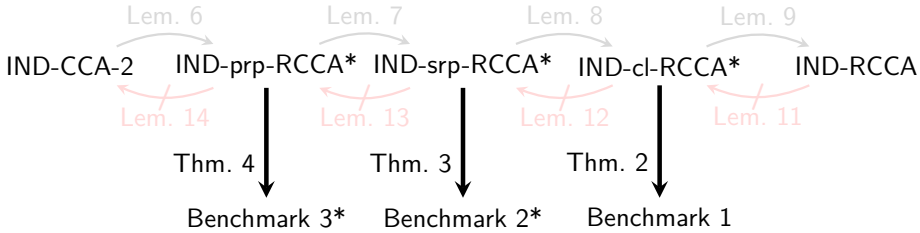
Fully characterize our game-based notions:

- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.

Contributions II

Fully characterize our game-based notions:

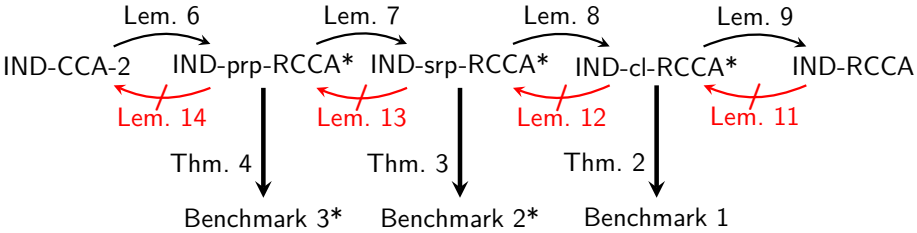
- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.



Contributions II

Fully characterize our game-based notions:

- Prove they are placed between CCA-2 and RCCA;
- Give all relations between them.



Preliminaries

Preliminaries - PKE

Definition (PKE scheme; Correctness)

A Public Key Encryption (PKE) scheme Π is a triple $\Pi = (G, E, D)$ of Probabilistic Polynomial-Time Algorithms (PPTs);

Correctness For any PPT adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{c|c} (\mathrm{pk}, \mathrm{sk}) \leftarrow G(1^k) & D_{\mathrm{sk}}(E_{\mathrm{pk}}(m)) \neq m \\ m \leftarrow \mathbf{A}(1^k, \mathrm{pk}) & \end{array}\right]$$

is negligible in k.

Preliminaries - PKE

Definition (PKE scheme; Correctness)

A Public Key Encryption (PKE) scheme Π is a triple $\Pi = (G, E, D)$ of Probabilistic Polynomial-Time Algorithms (PPTs);

Correctness For any PPT adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{c|c} (\mathrm{pk}, \mathrm{sk}) \leftarrow G(1^k) & D_{\mathrm{sk}}(E_{\mathrm{pk}}(m)) \neq m \\ m \leftarrow \mathbf{A}(1^k, \mathrm{pk}) & \end{array}\right]$$

is negligible in k.

Preliminaries - PKE

Definition (PKE scheme; Correctness)

A Public Key Encryption (PKE) scheme Π is a triple $\Pi = (G, E, D)$ of Probabilistic Polynomial-Time Algorithms (PPTs);

Correctness For any PPT adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{c|c} (\mathrm{pk}, \mathrm{sk}) \leftarrow G(1^k) & D_{\mathrm{sk}}(E_{\mathrm{pk}}(m)) \neq m \\ m \leftarrow \mathbf{A}(1^k, \mathrm{pk}) & \end{array}\right]$$

is negligible in k.

Preliminaries - PKE

Definition (PKE scheme; Correctness)

A Public Key Encryption (PKE) scheme Π is a triple $\Pi = (G, E, D)$ of Probabilistic Polynomial-Time Algorithms (PPTs);

Correctness For any PPT adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{c|c} (\mathrm{pk}, \mathrm{sk}) \leftarrow G(1^k) & D_{\mathrm{sk}}(E_{\mathrm{pk}}(m)) \neq m \\ m \leftarrow \mathbf{A}(1^k, \mathrm{pk}) & \end{array}\right]$$

is negligible in k.

Preliminaries - PKE

Definition (PKE scheme; Correctness)

A Public Key Encryption (PKE) scheme Π is a triple $\Pi = (G, E, D)$ of Probabilistic Polynomial-Time Algorithms (PPTs);

Correctness For any PPT adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{c|c} (\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k) & D_{\mathtt{sk}}(E_{\mathtt{pk}}(m)) \neq m \\ m \leftarrow \mathbf{A}(1^k, \mathtt{pk}) & \end{array}\right]$$

is negligible in k.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

PKE scheme $\Pi = (G, E, D)$ is IND-X secure if no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-X}}$ and $\mathbf{G}_1^{\Pi\text{-IND-X}}$ (specified ahead) with non-negligible advantage (in the security parameter k).

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

PKE scheme $\Pi = (G, E, D)$ is IND-X secure if no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-X}}$ and $\mathbf{G}_1^{\Pi\text{-IND-X}}$ (specified ahead) with non-negligible advantage (in the security parameter k).

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

PKE scheme $\Pi = (G, E, D)$ is IND-X secure if no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-X}}$ and $\mathbf{G}_1^{\Pi\text{-IND-X}}$ (specified ahead) with non-negligible advantage (in the security parameter k).

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

PKE scheme $\Pi = (G, E, D)$ is IND-X secure if no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-X}}$ and $\mathbf{G}_1^{\Pi\text{-IND-X}}$ (specified ahead) with non-negligible advantage (in the security parameter k).

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Definition (Generic Game-Based Security Notion X)

For $b \in \{0,1\}$, $\mathbf{G}_b^{\Pi\text{-IND-X}}$ is as follows:

Initialization: $(\mathtt{pk}, \mathtt{sk}) \leftarrow G(1^k)$; send $\mathtt{pk}$ to $\mathbf{D}$.

First decryption stage: On input $(\mathtt{ciphertext}, c)$, send $D_{\mathtt{sk}}(c)$ to $\mathbf{D}$.

Challenge stage: On input $(\mathtt{test\ messages}, m_0, m_1)$, send $c^* = E_{\mathtt{pk}}(m_b)$ to $\mathbf{D}$.

Second decryption stage: On input $(\mathtt{ciphertext}, c)$, send $\mathbf{D}$:

$$\text{reply} = \begin{cases} \mathtt{test}, & \text{if } P_X(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = 1 \\ m = D_{\mathtt{sk}}(c), & \text{otherwise,} \end{cases}$$

where P_X is a predicate defined by X.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot, \cdot, \cdot, \cdot, c^*, c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot, \cdot, \cdot, \cdot, c^*, c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot, \cdot, \cdot, \cdot, c^*, c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot,\cdot,\cdot,\cdot,c^*,c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot,\mathtt{sk},(m_0,m_1),\cdot,\cdot,c) := (D_{\mathtt{sk}}(c) \in \{m_0,m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot, \cdot, \cdot, \cdot, c^*, c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\text{CCA-2}}(\cdot, \cdot, \cdot, \cdot, c^*, c) := (c^* = c)$;

IND-RCCA: $P_{\text{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\mathsf{CCA\text{-}2}}(\cdot,\cdot,\cdot,\cdot,c^*,c) := (c^* = c)$;

IND-RCCA: $P_{\mathsf{RCCA}}(\cdot,\mathtt{sk},(m_0,m_1),\cdot,\cdot,c) := (D_{\mathtt{sk}}(c) \in \{m_0,m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\mathsf{CCA\text{-}2}}(\cdot,\cdot,\cdot,\cdot,c^*,c) := (c^* = c)$;

IND-RCCA: $P_{\mathsf{RCCA}}(\cdot,\mathtt{sk},(m_0,m_1),\cdot,\cdot,c) := (D_{\mathtt{sk}}(c) \in \{m_0,m_1\})$.

For security notion X:

P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

Preliminaries - PKE

Examples of P_X predicates:

IND-CCA-2: $P_{\mathsf{CCA\text{-}2}}(\cdot,\cdot,\cdot,\cdot,c^*,c) := (c^* = c)$;

IND-RCCA: $P_{\mathsf{RCCA}}(\cdot,\mathtt{sk},(m_0,m_1),\cdot,\cdot,c) := (D_{\mathtt{sk}}(c) \in \{m_0,m_1\})$.

For security notion X:

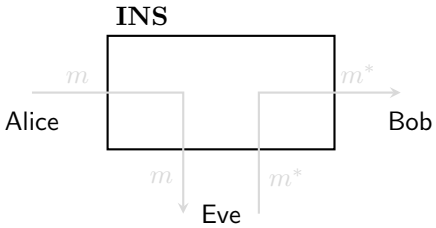
P_X can be interpreted as *"defining what are replays"* of the challenge ciphertext c^*;

$\mathbf{D}$ not allowed to query the decryption of replays of c^*.

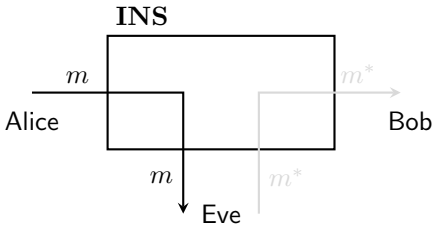
Preliminaries

Preliminaries - Channels

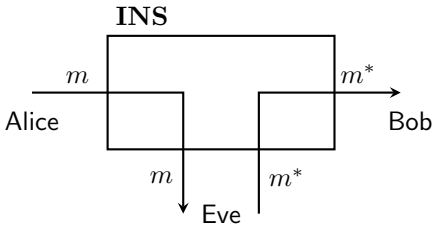
Preliminaries - Channels



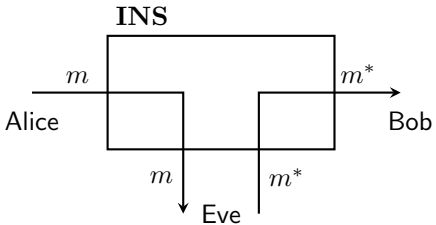
Preliminaries - Channels

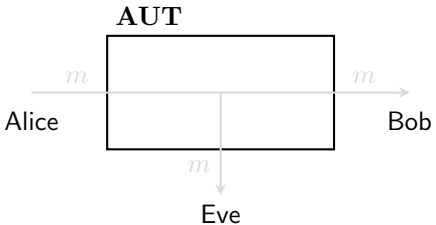


Preliminaries - Channels

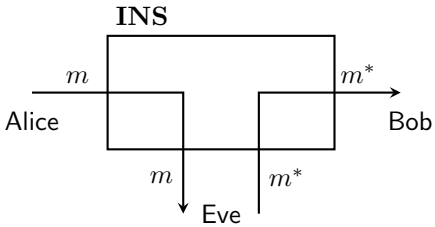


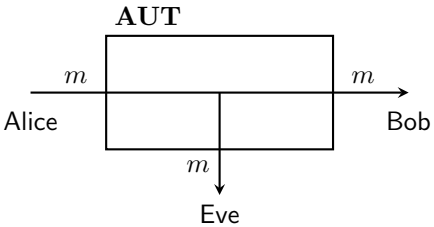
Preliminaries - Channels



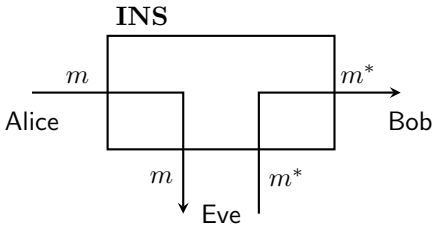


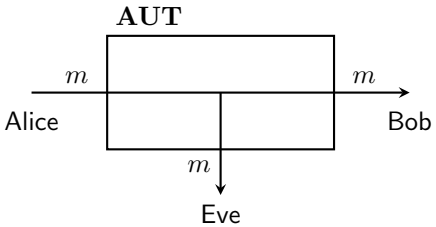
Preliminaries - Channels

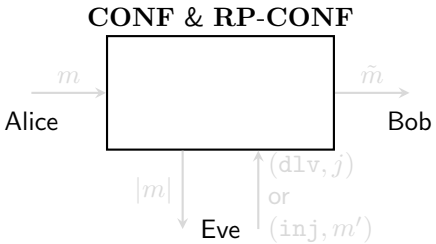




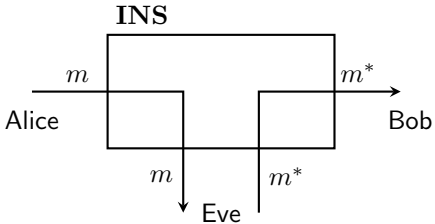
Preliminaries - Channels

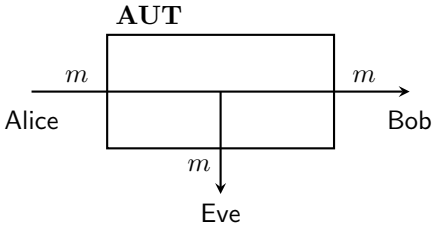


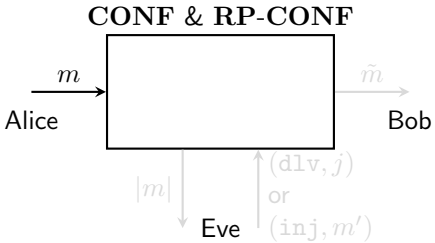




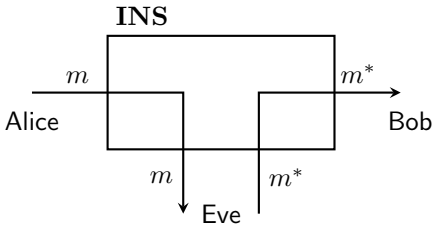
Preliminaries - Channels

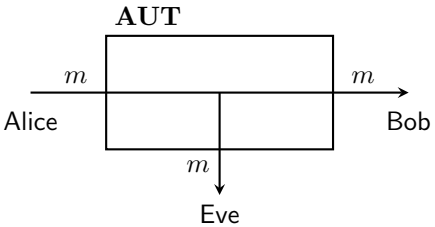


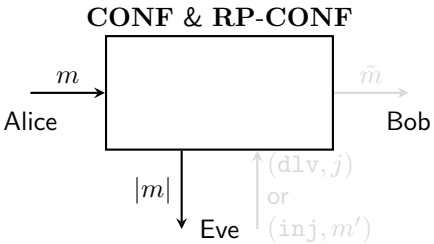




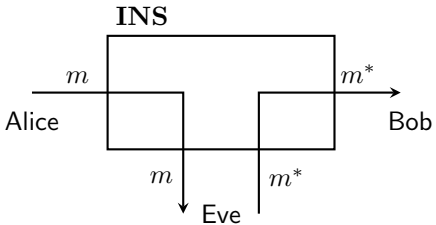
Preliminaries - Channels

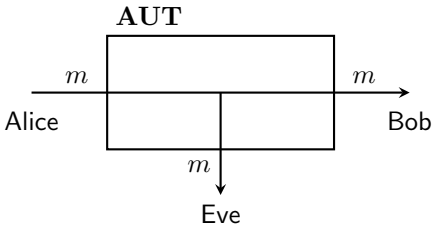


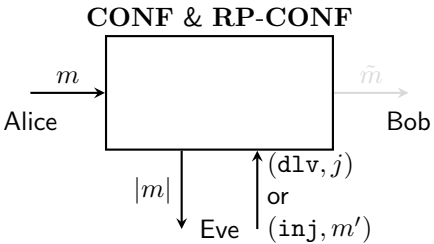




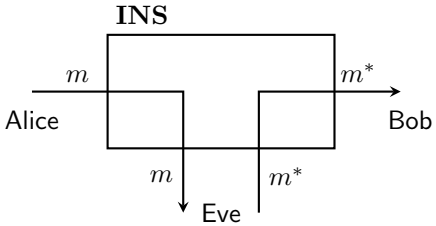
Preliminaries - Channels

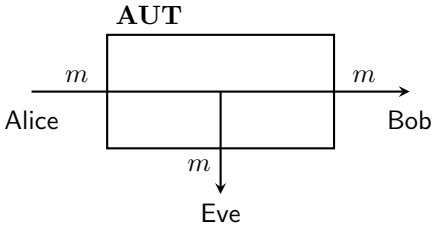


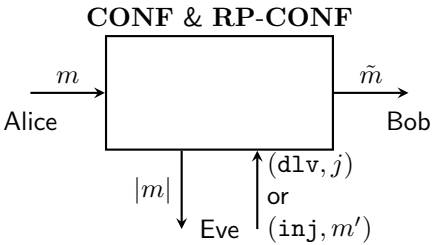




Preliminaries - Channels







Preliminaries - Channels

CONF *vs.* **RP-CONF**:

- **CONF** channels allow for replays:
  - Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
- **RP-CONF** channels do not allow for replays:
  - For each j:
    - First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
    - Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

- **CONF** channels allow for replays:
  - Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
- **RP-CONF** channels do not allow for replays:
  - For each j:
    - First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
    - Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

Later queries $(\texttt{dlv}, j)$ are ignored.

Preliminaries - Channels

CONF *vs.* **RP-CONF**:

CONF channels allow for replays:

Each $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;

RP-CONF channels do not allow for replays:

For each j:

First $(\texttt{dlv}, j)$ query delivers to Bob the j-th message sent by Alice;
Later queries $(\texttt{dlv}, j)$ are ignored.

Benchmarks

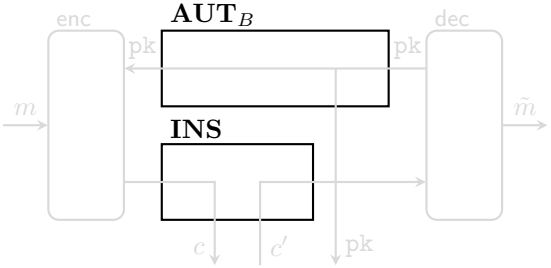
Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

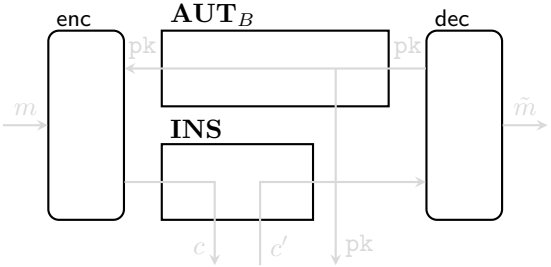
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

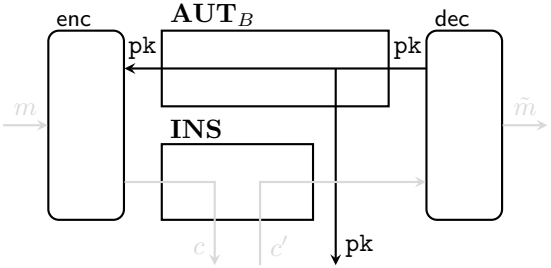
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

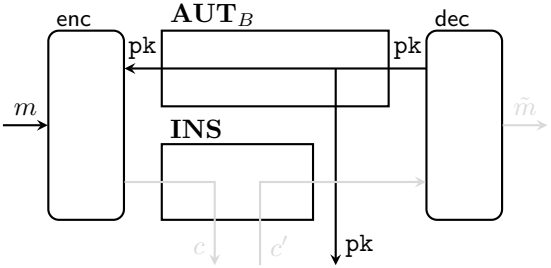
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

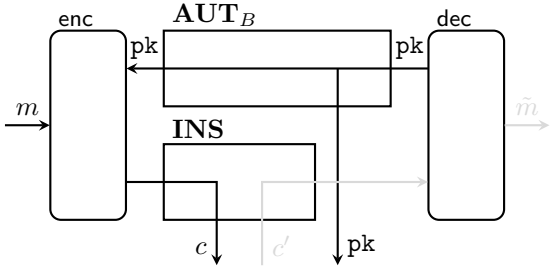
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

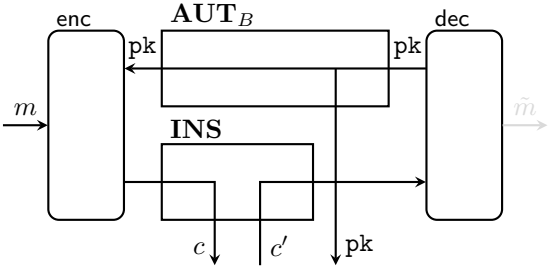
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

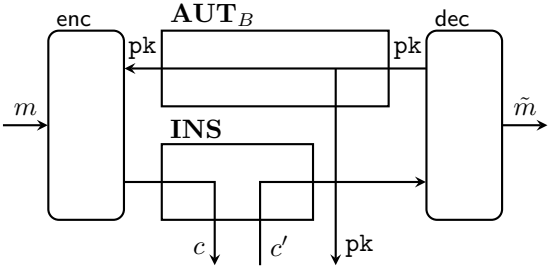
Figure: Real and ideal systems for construction of confidential channel.



Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

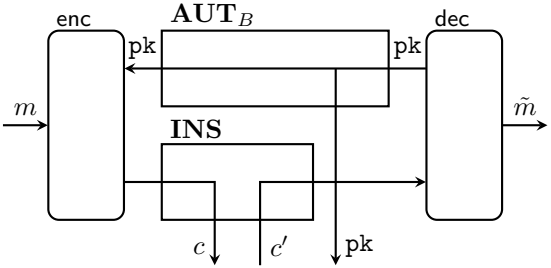
Figure: Real and ideal systems for construction of confidential channel.

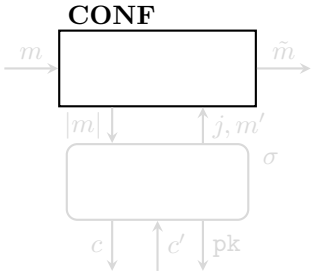


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

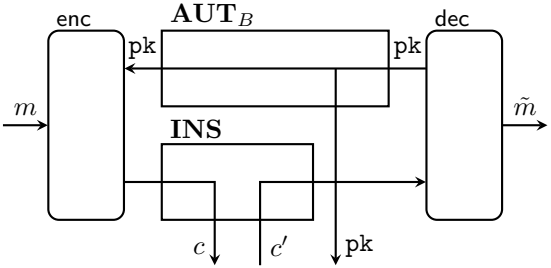


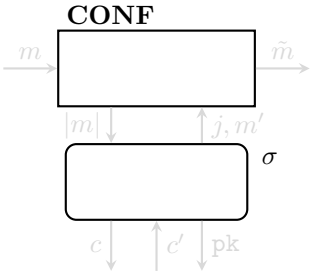


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

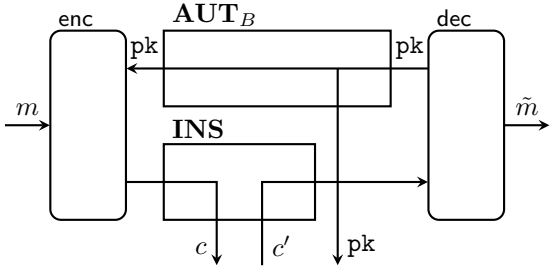


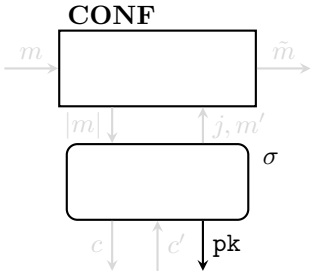


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

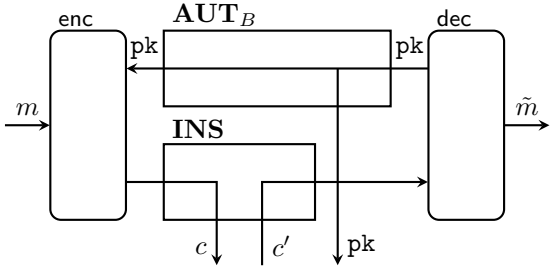


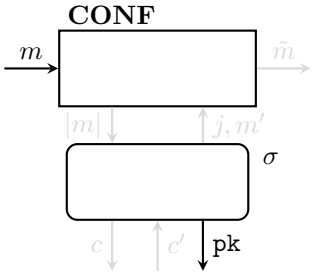


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

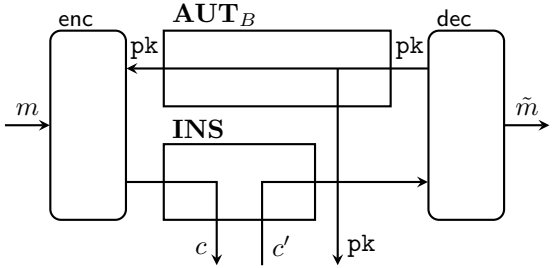


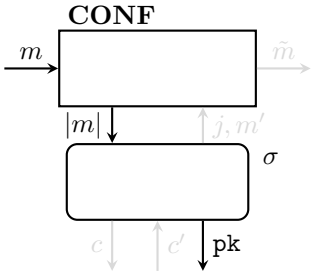


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

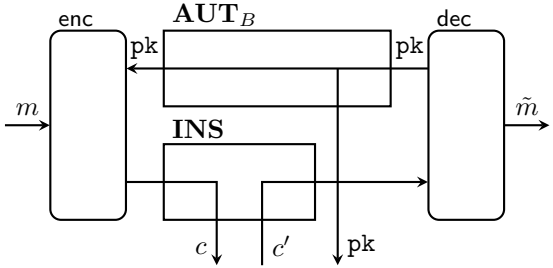


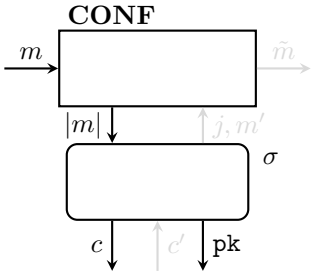


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

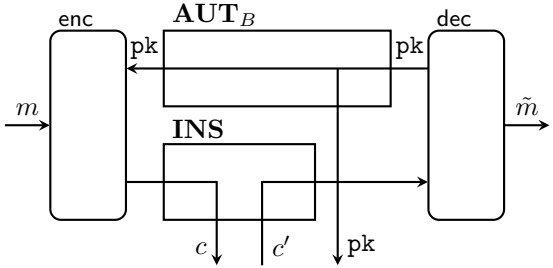


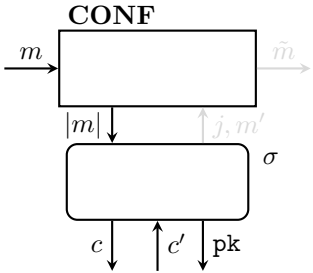


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

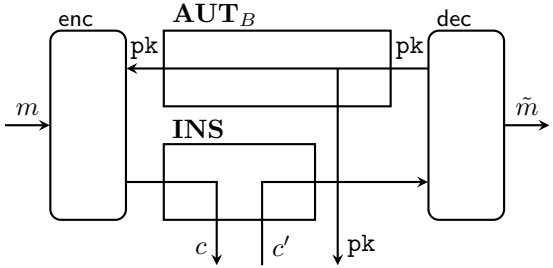


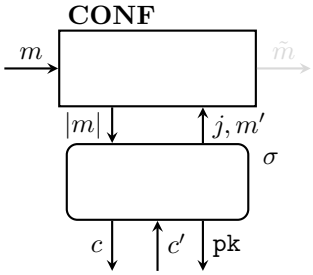


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

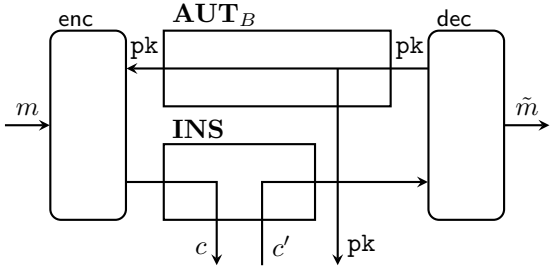


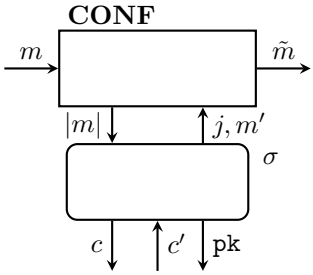


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

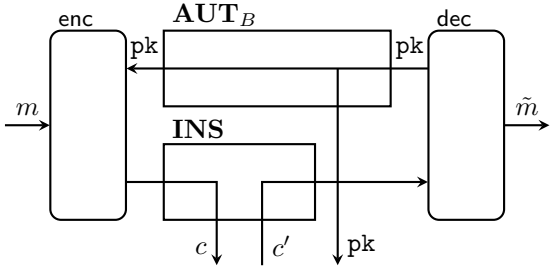


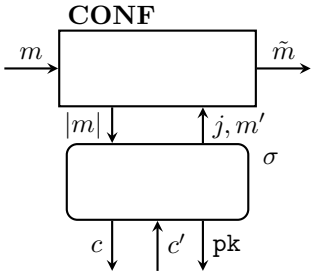


Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 1

Figure: Real and ideal systems for construction of confidential channel.

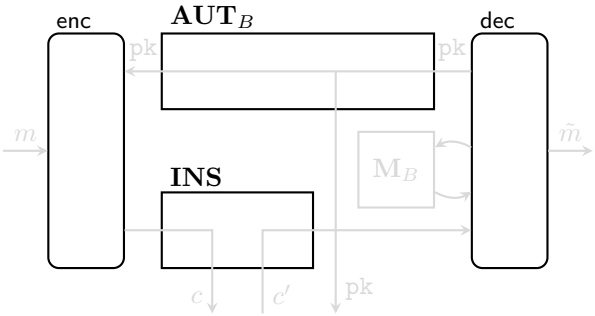




Benchmark 1 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

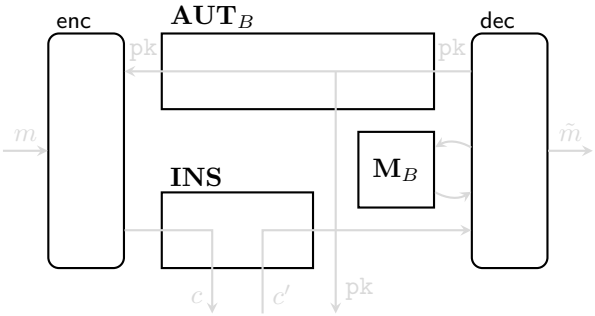
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

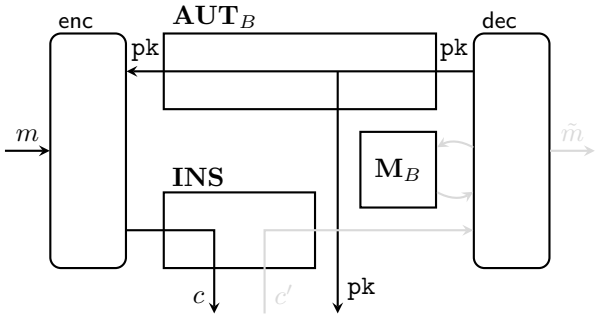
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

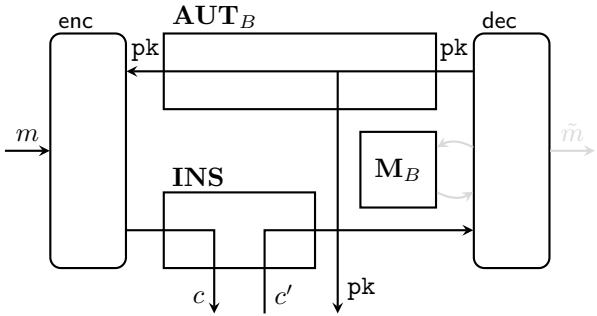
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

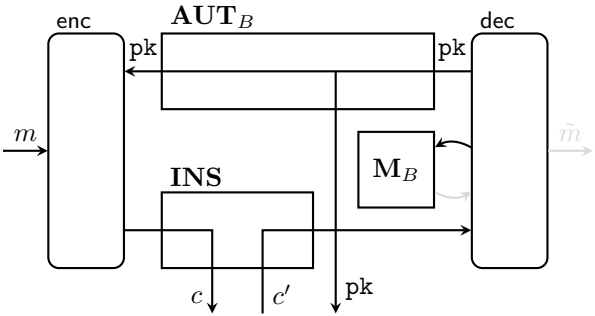
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

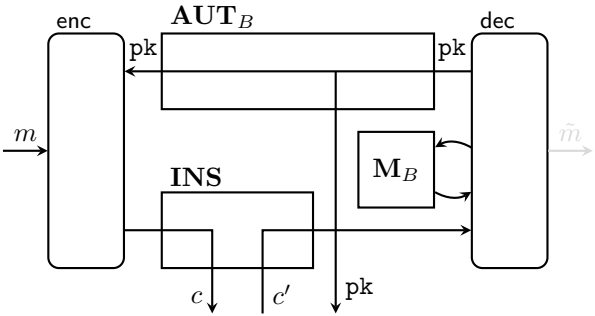
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

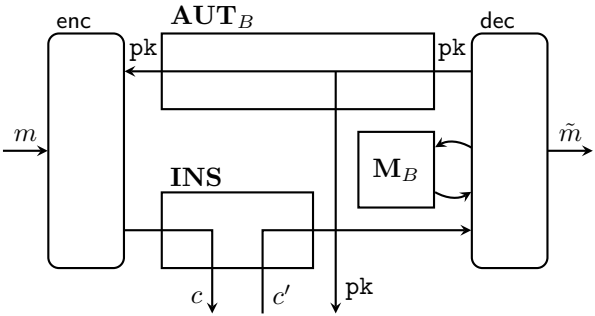
Figure: Real and ideal systems for construction of replay protected confidential channel.



Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

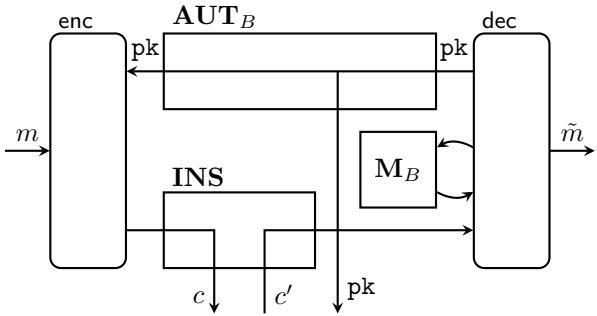
Figure: Real and ideal systems for construction of replay protected confidential channel.

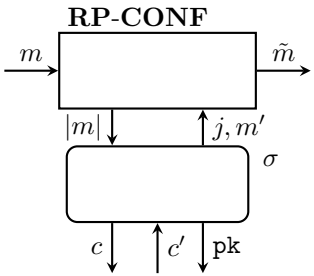


Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

Figure: Real and ideal systems for construction of replay protected confidential channel.

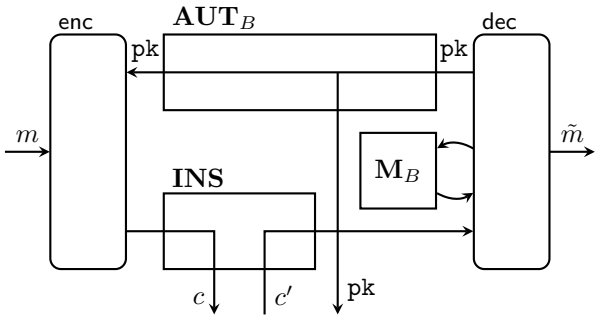


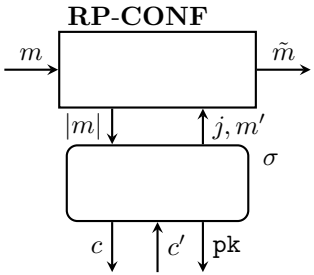


Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmarks - Benchmark 2

Figure: Real and ideal systems for construction of replay protected confidential channel.





Benchmark 2 achieved if real and ideal worlds are indistinguishable.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

- $\mathbf{CONF}$ channel requires non-malleability
- Eve can only
  - Learn message lengths $|m|$;
  - Forward messages to receiver $(\mathtt{dlv}, j)$;
  - Send new messages to receiver $(\mathtt{inj}, m')$.
- Malleable PKE scheme can be RCCA secure:
  - Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
  - Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

- **CONF** channel requires non-malleability
- Eve can only
  - Learn message lengths $|m|$;
  - Forward messages to receiver $(\mathtt{dlv}, j)$;
  - Send new messages to receiver $(\mathtt{inj}, m')$.
- Malleable PKE scheme can be RCCA secure:
  - Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
  - Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Benchmark 1: IND-RCCA not sufficient

Recall: $P_{\mathsf{RCCA}}(\cdot, \mathtt{sk}, (m_0, m_1), \cdot, \cdot, c) := (D_{\mathtt{sk}}(c) \in \{m_0, m_1\})$.

CONF channel requires non-malleability

Eve can only

- Learn message lengths $|m|$;
- Forward messages to receiver $(\mathtt{dlv}, j)$;
- Send new messages to receiver $(\mathtt{inj}, m')$.

Malleable PKE scheme can be RCCA secure:

- Binary message space $\Longrightarrow$ second decryption stage queries are "useless";
- Mauling ciphertexts does not help in winning the security game.

Achieving Benchmark 1

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-cl-RCCA secure if **there is an efficiently computable predicate** $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ such that no PPT $\mathbf{D}$ distinguishes systems $\mathbf{G}_0^{\Pi\text{-IND-cl-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-cl-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-cl-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) = \begin{cases} 1, & v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1 \ \wedge m_b = D_{\mathtt{sk}}(c) \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{D}$ has **no oracle access to** v.

v disallows strategies that could win the game, but would not break confidentiality;

Checking $m_b = D_{\mathtt{sk}}(c)$ guarantees that no meaningful mauling attack is disallowed.

Achieving Benchmark 1: IND-cl-RCCA

IND-cl-RCCA achieves Benchmark 1 **for a single message**;

$[n]$IND-cl-RCCA (the n-challenge version) achieves Benchmark 1 for n messages.

Achieving Benchmark 1: IND-cl-RCCA

IND-cl-RCCA achieves Benchmark 1 **for a single message**;

$[n]$IND-cl-RCCA (the n-challenge version) achieves Benchmark 1 for n messages.

Achieving Benchmark 1: IND-cl-RCCA

IND-cl-RCCA achieves Benchmark 1 **for a single message**;

$[n]$IND-cl-RCCA (the n-challenge version) achieves Benchmark 1 for n messages.

Achieving Benchmark 1: IND-cl-RCCA

Does IND-cl-RCCA capture replay-protection?

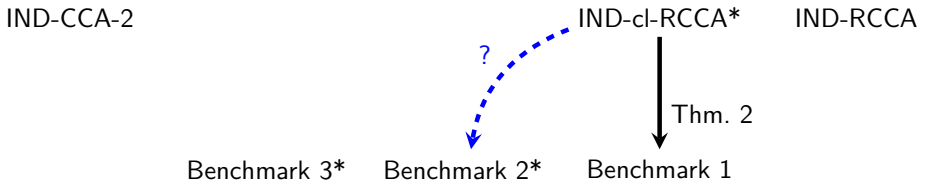
Achieving Benchmark 1: IND-cl-RCCA

Does IND-cl-RCCA capture replay-protection?

Achieving Benchmark 1: IND-cl-RCCA

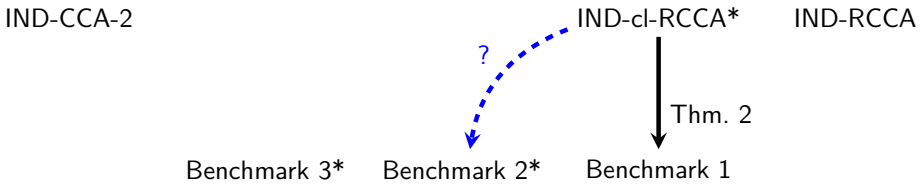
Does IND-cl-RCCA capture replay-protection?

Achieving Benchmark 1: IND-cl-RCCA



Does IND-cl-RCCA capture replay-protection?

Achieving Benchmark 2



Does IND-cl-RCCA capture replay-protection?

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1. v takes as input the secret key;
   The game does not give oracle access to v;
   How can a reduction check if two (arbitrary) ciphertexts are replays of one another?
2. v might not compute an equivalence relation;
   Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1. v takes as input the secret key;
   The game does not give oracle access to v;
   How can a reduction check if two (arbitrary) ciphertexts are replays of one another?
2. v might not compute an equivalence relation;
   Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1. v takes as input the secret key;
   - The game does not give oracle access to v;
   - How can a reduction check if two (arbitrary) ciphertexts are replays of one another?
2. v might not compute an equivalence relation;
   - Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1. v takes as input the secret key;
   The game does not give oracle access to v;
   How can a reduction check if two (arbitrary) ciphertexts are replays of one another?
2. v might not compute an equivalence relation;
   Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1. v takes as input the secret key;
   The game does not give oracle access to v;
   How can a reduction check if two (arbitrary) ciphertexts are replays of one another?
2. v might not compute an equivalence relation;
   Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1 v takes as input the secret key;
The game does not give oracle access to v;
How can a reduction check if two (arbitrary) ciphertexts are replays of one another?

2 v might not compute an equivalence relation;
Not clear if v could detect certain replays.

Achieving Benchmark 2

Could not prove that IND-cl-RCCA guarantees Benchmark 2.

1 v takes as input the secret key;
The game does not give oracle access to v;
How can a reduction check if two (arbitrary) ciphertexts are replays of one another?

2 v might not compute an equivalence relation;
Not clear if v could detect certain replays.

Achieving Benchmark 2

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathsf{pk}, \mathsf{sk}, c^*, c)$ **satisfying:**

1. for each $(\mathsf{pk}, \mathsf{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';
2. for each $(\mathsf{pk}, \mathsf{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
   if $v(\mathsf{pk}, \mathsf{sk}, c, c') = 1$, then $\delta(D_{\mathsf{sk}}(c), D_{\mathsf{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathsf{pk}, \mathsf{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathsf{pk}, \mathsf{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathsf{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';
2. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
   if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';
2. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
   if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';
2. for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
   if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':

if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathrm{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, with oracle access to v (on any pair of ciphertexts) throughout the entire game, distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\text{IND-srp-RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA

PKE scheme $\Pi = (G, E, D)$ is IND-srp-RCCA secure if *there is an efficiently computable predicate* $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ **satisfying:**

1 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, v computes an equivalence relation over ciphertexts c, c';

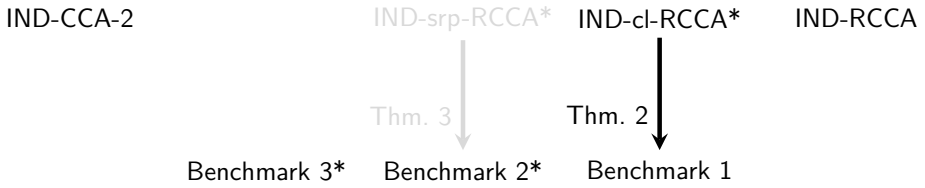
2 for each $(\mathtt{pk}, \mathtt{sk}) \in \mathrm{supp}(G)$, and each ciphertext pair c, c':
if $v(\mathtt{pk}, \mathtt{sk}, c, c') = 1$, then $\delta(D_{\mathtt{sk}}(c), D_{\mathtt{sk}}(c')) \leq \mathsf{negl}(k)$ [1],

such that no PPT $\mathbf{D}$, **with oracle access to** v **(on any pair of ciphertexts) throughout the entire game,** distinguishes $\mathbf{G}_0^{\Pi\text{-IND-srp-RCCA}}$ and $\mathbf{G}_1^{\Pi\text{-IND-srp-RCCA}}$ with non-negligible advantage.

$$P_{\mathsf{IND\text{-}srp\text{-}RCCA}}(\mathtt{pk}, \mathtt{sk}, (m_0, m_1), b, c^*, c) \ := (v(\mathtt{pk}, \mathtt{sk}, c^*, c) = 1).$$

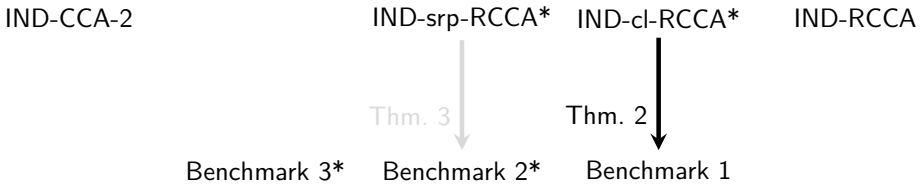
[1] Over $D_{\mathtt{sk}}(\cdot)$'s internal randomness.

Achieving Benchmark 2: IND-srp-RCCA



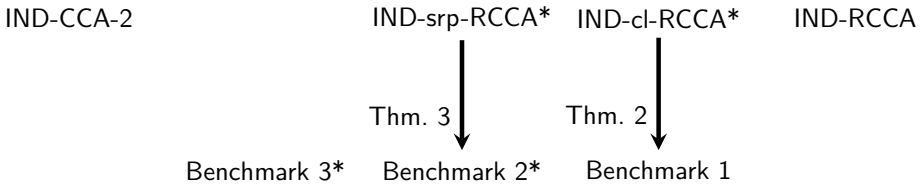
Predicate $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ is used by dec for replay detection.

Achieving Benchmark 2: IND-srp-RCCA



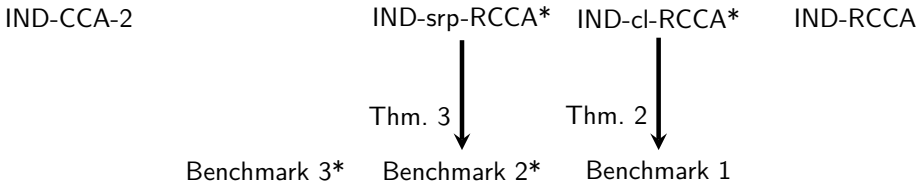
Predicate $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ is used by dec for replay detection.

Achieving Benchmark 2: IND-srp-RCCA



Predicate $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ is used by dec for replay detection.

Achieving Benchmark 2: IND-srp-RCCA



Predicate $v(\mathtt{pk}, \mathtt{sk}, c^*, c)$ is used by dec for replay detection.

Achieving Benchmark 3

Achieving Benchmark 3: IND-prp-RCCA

IND-prp-RCCA defined similarly to IND-srp-RCCA.

Main difference: predicate v is not given the secret key.

Achieving Benchmark 3: IND-prp-RCCA

IND-prp-RCCA defined similarly to IND-srp-RCCA.

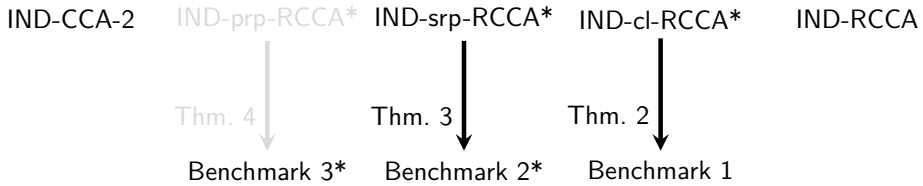
Main difference: predicate v is not given the secret key.

Achieving Benchmark 3: IND-prp-RCCA

IND-prp-RCCA defined similarly to IND-srp-RCCA.

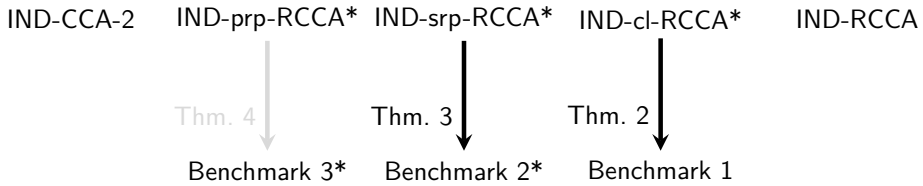
Main difference: predicate v is not given the secret key.

Achieving Benchmark 3: IND-prp-RCCA



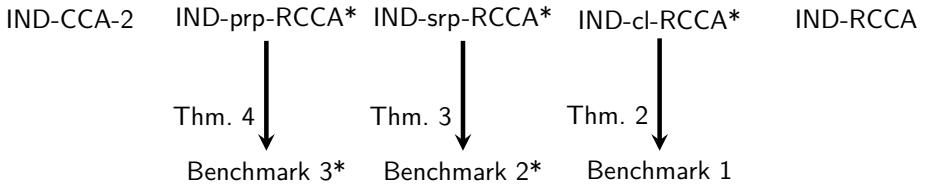
Predicate v is used by rp for replay detection.

Achieving Benchmark 3: IND-prp-RCCA



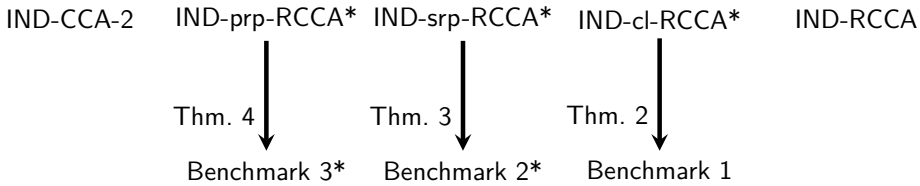
Predicate v is used by rp for replay detection.

Achieving Benchmark 3: IND-prp-RCCA



Predicate v is used by rp for replay detection.

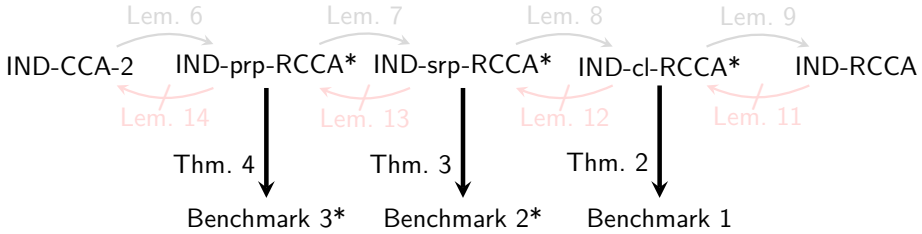
Achieving Benchmark 3: IND-prp-RCCA



Predicate v is used by rp for replay detection.

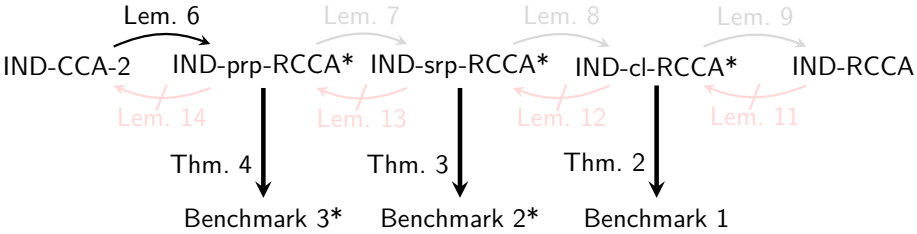
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



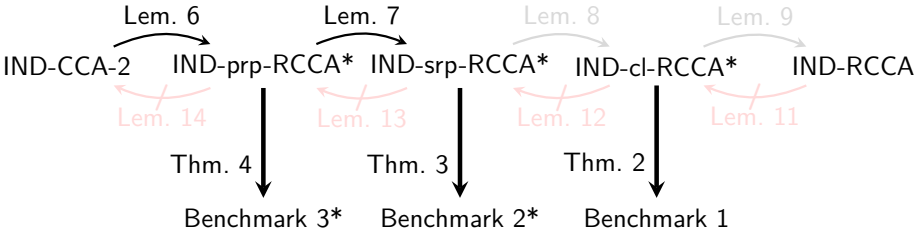
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



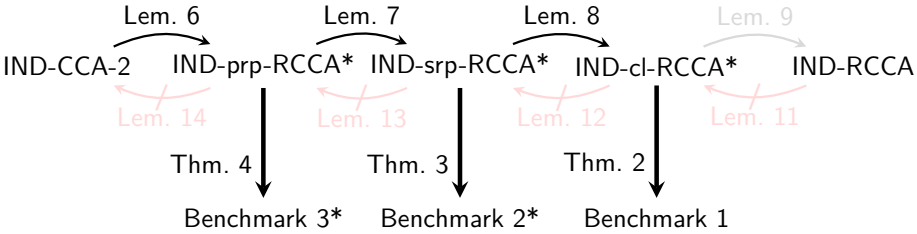
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



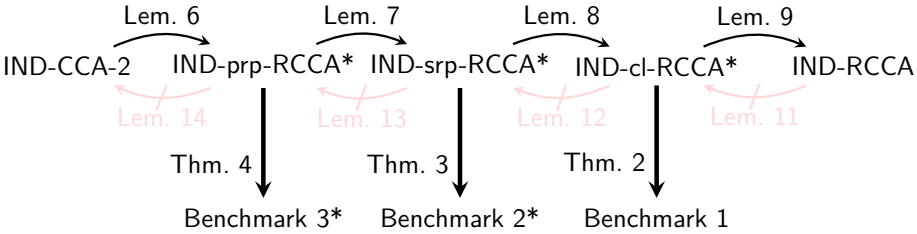
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



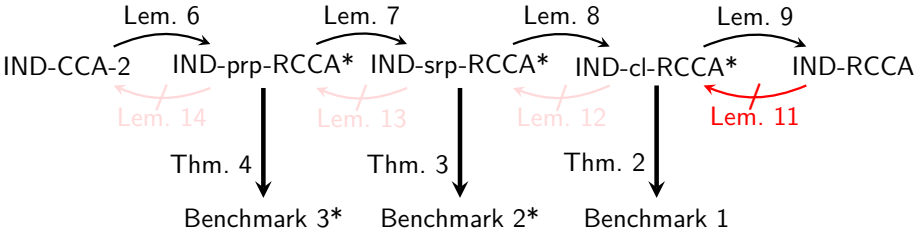
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



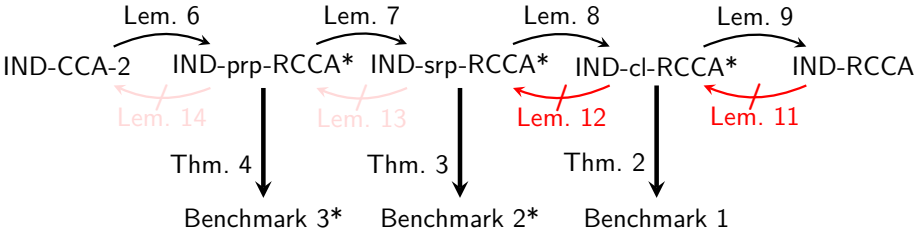
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



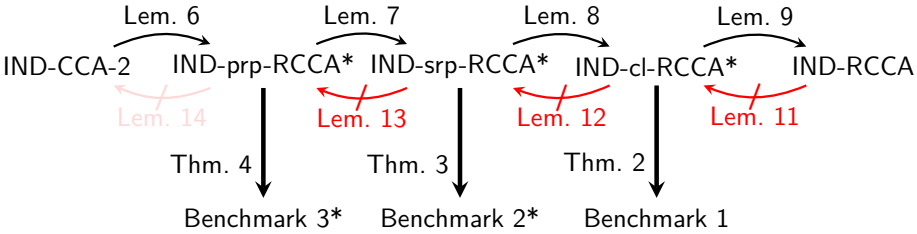
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



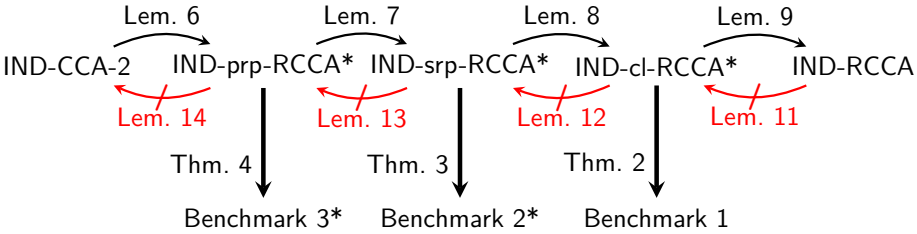
Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



Relations Between Notions

Figure: Relations between security notions. Security notions introduced in this paper are marked with *.



Thank You!

Ran Canetti, Hugo Krawczyk, and Jesper Buus Nielsen.
Relaxing chosen-ciphertext security.
In Dan Boneh, editor, *Advances in Cryptology – CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 565–582, Santa Barbara, CA, USA, August 17–21, 2003. Springer, Heidelberg, Germany.

Sandro Coretti, Ueli Maurer, and Björn Tackmann.
Constructing confidential channels from authenticated channels - public-key encryption revisited.
In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology – ASIACRYPT 2013, Part I*, volume 8269 of *Lecture Notes in Computer Science*, pages 134–153, Bengalore, India, December 1–5, 2013. Springer, Heidelberg, Germany.