Strengthening the Known-Key Security Notion for Block Ciphers

Benoît Cogliati[1] Yannick Seurin[2]

[1]Versailles University, France

[2]ANSSI, France

March 23, 2016 — FSE 2016

In a Nutshell

- we reconsider the formalization of known-key attacks against block ciphers
- the first rigorous formalization (Known-Key-indifferentiability) by Andreeva, Bogdanov and Mennink (ABM) at FSE 2013 only considered a single known key
- we extend this notion to multiple known keys and prove separation results from the ABM single-key notion
- we explore the security of the Iterated Even-Mansour construction under this new security definition

In a Nutshell

- we reconsider the formalization of known-key attacks against block ciphers
- the first rigorous formalization (Known-Key-indifferentiability) by Andreeva, Bogdanov and Mennink (ABM) at FSE 2013 only considered a single known key
- we extend this notion to multiple known keys and prove separation results from the ABM single-key notion
- we explore the security of the Iterated Even-Mansour construction under this new security definition

In a Nutshell

- we reconsider the formalization of known-key attacks against block ciphers
- the first rigorous formalization (Known-Key-indifferentiability) by Andreeva, Bogdanov and Mennink (ABM) at FSE 2013 only considered a single known key
- we extend this notion to multiple known keys and prove separation results from the ABM single-key notion
- we explore the security of the Iterated Even-Mansour construction under this new security definition

In a Nutshell

- we reconsider the formalization of known-key attacks against block ciphers
- the first rigorous formalization (Known-Key-indifferentiability) by Andreeva, Bogdanov and Mennink (ABM) at FSE 2013 only considered a single known key
- we extend this notion to multiple known keys and prove separation results from the ABM single-key notion
- we explore the security of the Iterated Even-Mansour construction under this new security definition

Outline

Background on Known-Key Attacks

Formalizing Multiple Known-Key Security

Multiple Known-Key Security of the Iterated Even-Mansour Construction

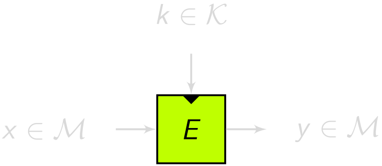
Outline

Background on Known-Key Attacks

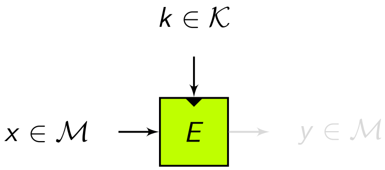
Formalizing Multiple Known-Key Security

Multiple Known-Key Security of the Iterated Even-Mansour Construction

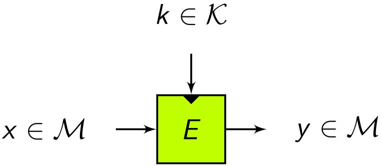
Block Ciphers



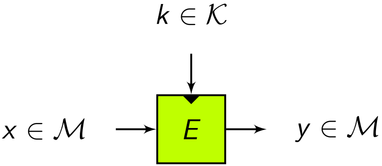
Block Ciphers



Block Ciphers



Block Ciphers



Usual security notion: pseudorandomness

No attacker should be able to distinguish:

- E_k for a random key $k \leftarrow_{\$} \mathcal{K}$
- a uniformly random permutation of the message space $\mathcal{M}$

Known-Key Attacks

Introduced by Knudsen and Rijmen at AC 2007 [KR07].

Definition (Known-key attack, informally)

Given a random key k, find a "property" of permutation E_k more efficiently than for a random, black-box permutation.

Example 1: unary relation

Given $k \in \mathcal{K}$, find $x, y \in \mathcal{M}$ such that the $n/2$ first bits of x and y are 0 and $E_k(x) = y$ in time less than $\sim 2^{n/2}$ evaluations of E.

Example 2: binary relation

Given $k \in \mathcal{K}$, find $x_1, y_1, x_2, y_2 \in \mathcal{M}$ such that $E_k(x_i) = y_i$, $i = 1, 2$, and $x_1 \oplus y_1 = x_2 \oplus y_2$ in time less than $\sim 2^{n/2}$ evaluations of E.

Known-Key Attacks

Introduced by Knudsen and Rijmen at AC 2007 [KR07].

Definition (Known-key attack, informally)

Given a random key k, find a "property" of permutation E_k more efficiently than for a random, black-box permutation.

Example 1: unary relation

Given $k \in \mathcal{K}$, find $x, y \in \mathcal{M}$ such that the $n/2$ first bits of x and y are 0 and $E_k(x) = y$ in time less than $\sim 2^{n/2}$ evaluations of E.

Example 2: binary relation

Given $k \in \mathcal{K}$, find $x_1, y_1, x_2, y_2 \in \mathcal{M}$ such that $E_k(x_i) = y_i$, $i = 1, 2$, and $x_1 \oplus y_1 = x_2 \oplus y_2$ in time less than $\sim 2^{n/2}$ evaluations of E.

Known-Key Attacks

Introduced by Knudsen and Rijmen at AC 2007 [KR07].

Definition (Known-key attack, informally)

Given a random key k, find a "property" of permutation E_k more efficiently than for a random, black-box permutation.

Example 1: unary relation

Given $k \in \mathcal{K}$, find $x, y \in \mathcal{M}$ such that the $n/2$ first bits of x and y are 0 and $E_k(x) = y$ in time less than $\sim 2^{n/2}$ evaluations of E.

Example 2: binary relation

Given $k \in \mathcal{K}$, find $x_1, y_1, x_2, y_2 \in \mathcal{M}$ such that $E_k(x_i) = y_i$, $i = 1, 2$, and $x_1 \oplus y_1 = x_2 \oplus y_2$ in time less than $\sim 2^{n/2}$ evaluations of E.

A "Generic" Known-Key Attack

Assume $\mathcal{K} = \mathcal{M}$ for simplicity. Consider the set of pairs

$$\mathcal{R}_{\text{diag}} = \{(k, E_k(k)) : k \in \mathcal{K}\} \subset \mathcal{M} \times \mathcal{M}.$$

Then:

- given a random key k, it is easy to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $E_k(x) = y$ (simply take $x = k$ and $y = E_k(k)$)
- given a random permutation P, it is hard to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $P(x) = y$.

$\Rightarrow$ impossible to formalize KK attacks for a single block cipher E

A "Generic" Known-Key Attack

Assume $\mathcal{K} = \mathcal{M}$ for simplicity. Consider the set of pairs

$$\mathcal{R}_{\text{diag}} = \{(k, E_k(k)) : k \in \mathcal{K}\} \subset \mathcal{M} \times \mathcal{M}.$$

Then:

- given a random key k, it is easy to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $E_k(x) = y$ (simply take $x = k$ and $y = E_k(k)$)
- given a random permutation P, it is hard to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $P(x) = y$.

$\Rightarrow$ impossible to formalize KK attacks for a single block cipher E

A "Generic" Known-Key Attack

Assume $\mathcal{K} = \mathcal{M}$ for simplicity. Consider the set of pairs

$$\mathcal{R}_{\text{diag}} = \{(k, E_k(k)) : k \in \mathcal{K}\} \subset \mathcal{M} \times \mathcal{M}.$$

Then:

- given a random key k, it is easy to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $E_k(x) = y$ (simply take $x = k$ and $y = E_k(k)$)
- given a random permutation P, it is hard to find $(x, y) \in \mathcal{R}_{\text{diag}}$ such that $P(x) = y$.

$\Rightarrow$ impossible to formalize KK attacks for a single block cipher E

Formalizing Known-Key Security

- first formalization of KK-security by Andreeva, Bogdanov, and Mennink at FSE 2013 [ABM13]
- circumvents impossibility results by considering a class of block ciphers based on some ideal primitive $\mathcal{F}$ (e.g. random function(s), random permutation(s), etc.)
- uses the indifferentiability notion [MRH04]
- informally, the ABM security notion ensures that for a random key k, $E_k^{\mathcal{F}}$ "behaves" as a random permutation even when k is known to the attacker (assuming $\mathcal{F}$ is ideal)

Formalizing Known-Key Security

- first formalization of KK-security by Andreeva, Bogdanov, and Mennink at FSE 2013 [ABM13]
- circumvents impossibility results by considering a class of block ciphers based on some ideal primitive $\mathcal{F}$ (e.g. random function(s), random permutation(s), etc.)
- uses the indifferentiability notion [MRH04]
- informally, the ABM security notion ensures that for a random key k, $E_k^{\mathcal{F}}$ "behaves" as a random permutation even when k is known to the attacker (assuming $\mathcal{F}$ is ideal)

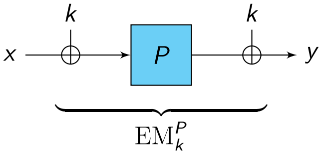
Formalizing Known-Key Security

- first formalization of KK-security by Andreeva, Bogdanov, and Mennink at FSE 2013 [ABM13]
- circumvents impossibility results by considering a class of block ciphers based on some ideal primitive $\mathcal{F}$ (e.g. random function(s), random permutation(s), etc.)
- uses the indifferentiability notion [MRH04]
- informally, the ABM security notion ensures that for a random key k, $E_k^{\mathcal{F}}$ "behaves" as a random permutation even when k is known to the attacker (assuming $\mathcal{F}$ is ideal)

Formalizing Known-Key Security

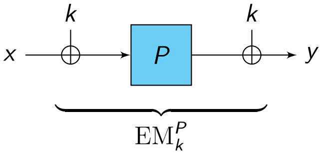
- first formalization of KK-security by Andreeva, Bogdanov, and Mennink at FSE 2013 [ABM13]
- circumvents impossibility results by considering a class of block ciphers based on some ideal primitive $\mathcal{F}$ (e.g. random function(s), random permutation(s), etc.)
- uses the indifferentiability notion [MRH04]
- informally, the ABM security notion ensures that for a random key k, $E_k^{\mathcal{F}}$ "behaves" as a random permutation even when k is known to the attacker (assuming $\mathcal{F}$ is ideal)

Example: The 1-Round Even-Mansour Construction



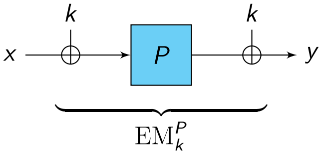
- based on a public permutation P modeled as ideal (uniformly random)
- provably secure in the secret key model (pseudorandomness) [EM97]
- provably secure against (the ABM notion of) known-key attacks: for any key k, EM_k^P "behaves" as a random permutation (assuming P is a random permutation)

Example: The 1-Round Even-Mansour Construction



- based on a public permutation P modeled as ideal (uniformly random)
- provably secure in the secret key model (pseudorandomness) [EM97]
- provably secure against (the ABM notion of) known-key attacks: for any key k, EM_k^P "behaves" as a random permutation (assuming P is a random permutation)

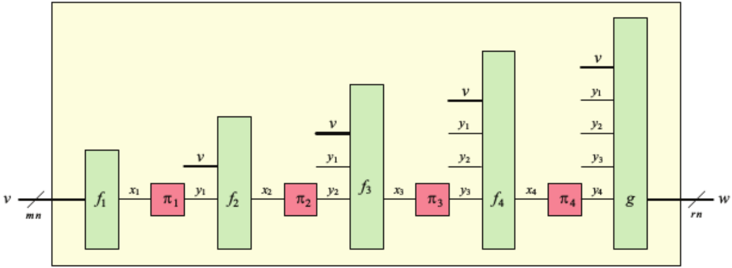
Example: The 1-Round Even-Mansour Construction



- based on a public permutation P modeled as ideal (uniformly random)
- provably secure in the secret key model (pseudorandomness) [EM97]
- provably secure against (the ABM notion of) known-key attacks: for any key k, EM_k^P "behaves" as a random permutation (assuming P is a random permutation)

Limitation of ABM Notion: A Motivating Example

- Rogaway-Steinberger compression functions [RS08a]: defined from a few public permutations $\pi_1, \ldots, \pi_\mu$
- provably secure in the Random Permutation Model



Source: [RS08b]

Limitation of ABM Notion: A Motivating Example

- natural idea: instantiate the π_i's using a block cipher E:

$$\pi_1 = E_{k_1}, \ldots, \pi_\mu = E_{k_\mu}$$

  with $k_1, \ldots, k_\mu$ public, independently drawn keys
- under which security assumption on E does the construction remain secure?
- resistance to chosen-key attacks: too strong
- ABM known-key security notion: too weak because it considers a single key
- here, the attacker is given multiple known keys
  $\Rightarrow$ we need to extend the KK security notion

Limitation of ABM Notion: A Motivating Example

- natural idea: instantiate the π_i's using a block cipher E:

$$\pi_1 = E_{k_1}, \ldots, \pi_\mu = E_{k_\mu}$$

  with $k_1, \ldots, k_\mu$ public, independently drawn keys
- under which security assumption on E does the construction remain secure?
- resistance to chosen-key attacks: too strong
- ABM known-key security notion: too weak because it considers a single key
- here, the attacker is given multiple known keys
  $\Rightarrow$ we need to extend the KK security notion

Limitation of ABM Notion: A Motivating Example

- natural idea: instantiate the π_i's using a block cipher E:

$$\pi_1 = E_{k_1}, \ldots, \pi_\mu = E_{k_\mu}$$

  with $k_1, \ldots, k_\mu$ public, independently drawn keys
- under which security assumption on E does the construction remain secure?
- resistance to chosen-key attacks: too strong
- ABM known-key security notion: too weak because it considers a single key
- here, the attacker is given multiple known keys
  $\Rightarrow$ we need to extend the KK security notion

Limitation of ABM Notion: A Motivating Example

- natural idea: instantiate the π_i's using a block cipher E:

$$\pi_1 = E_{k_1}, \ldots, \pi_\mu = E_{k_\mu}$$

  with $k_1, \ldots, k_\mu$ public, independently drawn keys
- under which security assumption on E does the construction remain secure?
- resistance to chosen-key attacks: too strong
- ABM known-key security notion: too weak because it considers a single key
- here, the attacker is given multiple known keys
  $\Rightarrow$ we need to extend the KK security notion

Limitation of ABM Notion: A Motivating Example

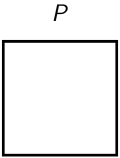
- natural idea: instantiate the π_i's using a block cipher E:

$$\pi_1 = E_{k_1}, \ldots, \pi_\mu = E_{k_\mu}$$

  with $k_1, \ldots, k_\mu$ public, independently drawn keys
- under which security assumption on E does the construction remain secure?
- resistance to chosen-key attacks: too strong
- ABM known-key security notion: too weak because it considers a single key
- here, the attacker is given multiple known keys
  $\Rightarrow$ we need to extend the KK security notion

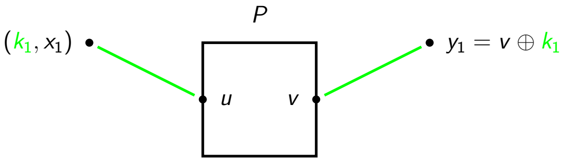
Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):



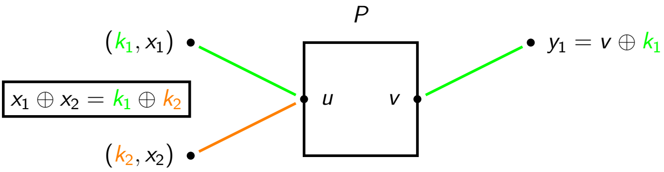
Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):



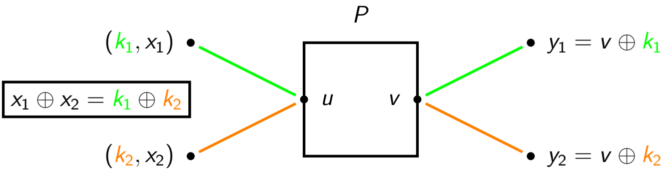
Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):



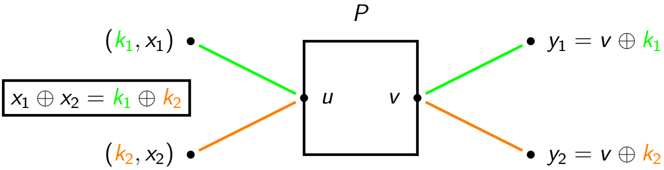
Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):



Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):

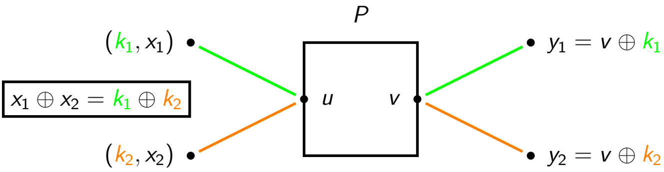


Then (x_1, y_1) and (x_2, y_2) satisfy $y_1 = \mathrm{EM}^P_{k_1}(x_1)$, $y_2 = \mathrm{EM}^P_{k_2}(x_2)$, and

$$x_1 \oplus x_2 = y_1 \oplus y_2 \tag{1}$$

Multiple KK-Attack against 1-round EM

The attacker is given a pair of keys (k_1, k_2):



Then (x_1, y_1) and (x_2, y_2) satisfy $y_1 = \mathrm{EM}^P_{k_1}(x_1)$, $y_2 = \mathrm{EM}^P_{k_2}(x_2)$, and

$$x_1 \oplus x_2 = y_1 \oplus y_2 \tag{1}$$

But, given oracle access to two random permutations P_1 and P_2, finding (x_1, y_1) and (x_2, y_2) satisfying $y_1 = P_1(x_1)$, $y_2 = P_2(x_2)$ and Eq. (1) requires $\sim 2^{n/2}$ queries.

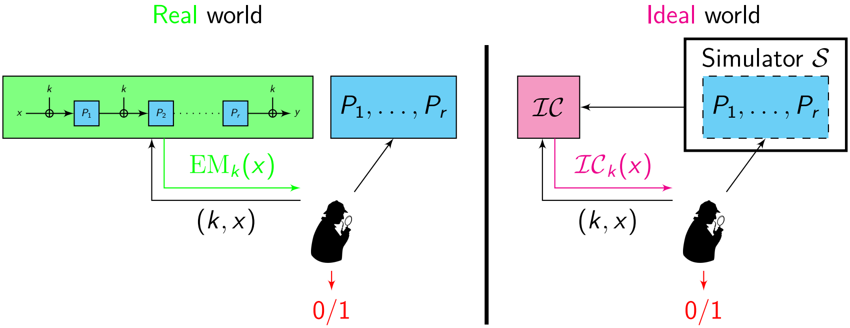
Outline

Background on Known-Key Attacks

Formalizing Multiple Known-Key Security

Multiple Known-Key Security of the Iterated Even-Mansour Construction

Indifferentiability (Standard Notion)

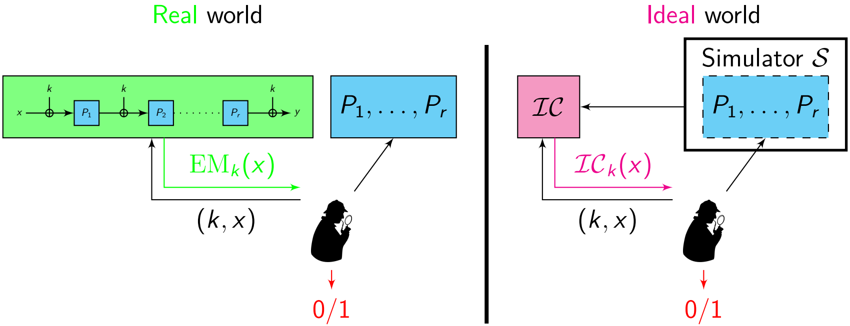


The attacker $\mathcal{D}$ must distinguish:

- the real world: construction + random permutations $P_1, \ldots, P_r$
- the ideal world: ideal cipher $\mathcal{IC}$ + simulator $\mathcal{S}$

NB: no hidden secret in the real world
(but $\mathcal{D}$ can only make a limited number of queries)

Indifferentiability (Standard Notion)



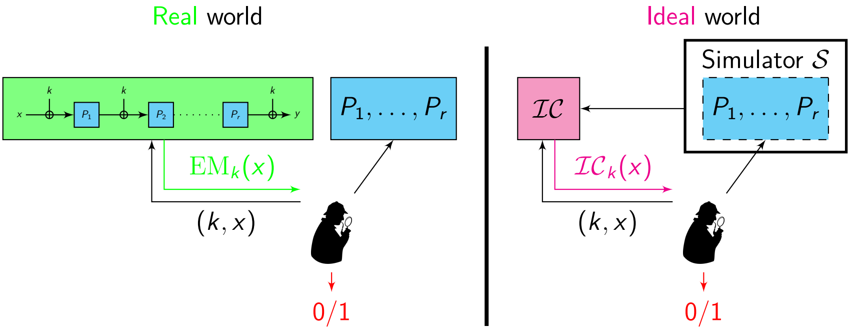
The attacker $\mathcal{D}$ must distinguish:

- the real world: construction + random permutations $P_1, \ldots, P_r$
- the ideal world: ideal cipher $\mathcal{IC}$ + simulator $\mathcal{S}$

NB: no hidden secret in the real world

(but $\mathcal{D}$ can only make a limited number of queries)

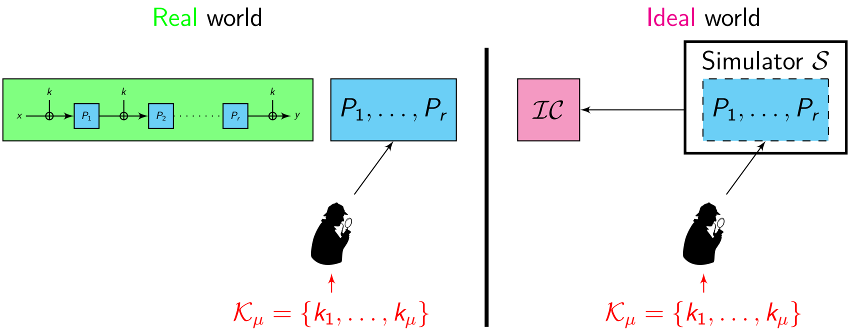
Indifferentiability (Standard Notion)



Definition (Indifferentiability [MRH04])

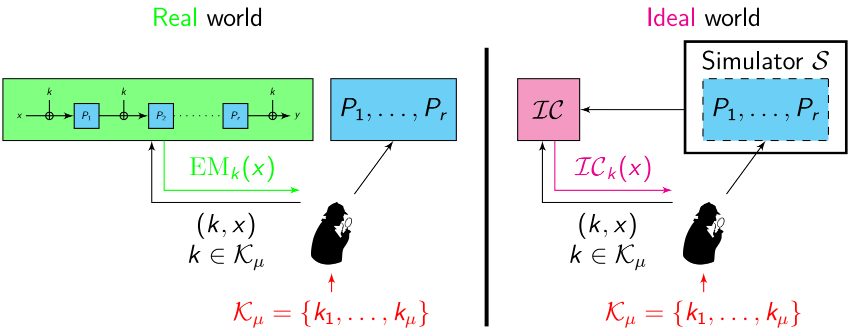
A block cipher construction is said (q_d, q_s, ε)-indifferentiable from an ideal cipher if there exists a simulator $\mathcal{S}$ such that for any distinguisher $\mathcal{D}$ making at most q_d queries in total, $\mathcal{S}$ makes at most q_s ideal cipher queries and $\mathcal{D}$ distinguishes the two worlds with adv. at most ε

Multiple Known-Key (μ-KK) Indifferentiability



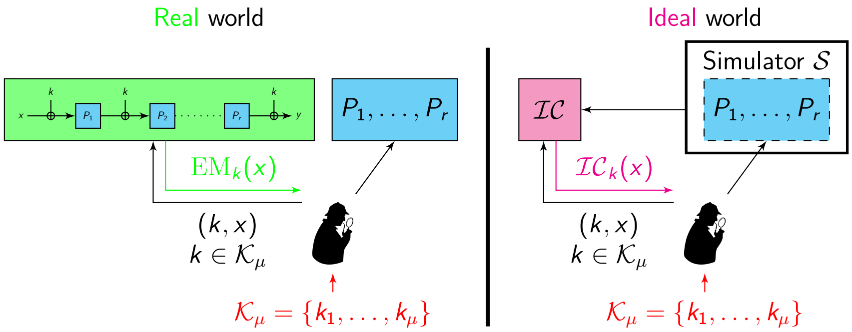
- the attacker is given a set of μ keys $\mathcal{K}_\mu = \{k_1, \ldots, k_\mu\}$
- it can query the construction/$\mathcal{IC}$ oracle only with these keys
- $\mu = 1 \Rightarrow$ one recovers the ABM known-key notion
- $\mu =$ full key space $\Rightarrow$ standard indifferentiability ("chosen" key)

Multiple Known-Key (μ-KK) Indifferentiability



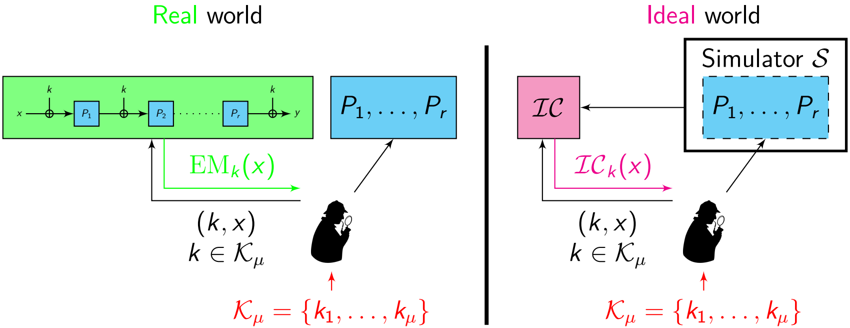
- the attacker is given a set of μ keys $\mathcal{K}_\mu = \{k_1, \ldots, k_\mu\}$
- it can query the construction/$\mathcal{IC}$ oracle only with these keys
- $\mu = 1 \Rightarrow$ one recovers the ABM known-key notion
- $\mu =$ full key space $\Rightarrow$ standard indifferentiability ("chosen" key)

Multiple Known-Key (μ-KK) Indifferentiability



- the attacker is given a set of μ keys $\mathcal{K}_\mu = \{k_1, \ldots, k_\mu\}$
- it can query the construction/$\mathcal{IC}$ oracle only with these keys
- $\mu = 1 \Rightarrow$ one recovers the ABM known-key notion
- $\mu =$ full key space $\Rightarrow$ standard indifferentiability ("chosen" key)

Multiple Known-Key (μ-KK) Indifferentiability



- the attacker is given a set of μ keys $\mathcal{K}_\mu = \{k_1, \ldots, k_\mu\}$
- it can query the construction/$\mathcal{IC}$ oracle only with these keys
- $\mu = 1 \Rightarrow$ one recovers the ABM known-key notion
- $\mu =$ full key space $\Rightarrow$ standard indifferentiability ("chosen" key)

Composition Theorems

Indifferentiability allows to "compose security proofs"

Theorem (Composition for μ-KK-indiff. [MRH04])

Let Γ be a cryptosystem based on a block cipher E.
Let $\mathcal{C}^{\mathcal{F}}$ be a block cipher construction based on some ideal primitive $\mathcal{F}$. If

1. *Γ is secure when $E = \mathcal{IC}$ is an ideal cipher*
2. *construction $\mathcal{C}^{\mathcal{F}}$ is μ-KK-indifferentiable from an ideal cipher*
3. *cryptosystem Γ only calls E with keys in $\{k_1, \ldots, k_\mu\}$*

then Γ remains secure when instantiated with $E = \mathcal{C}^{\mathcal{F}}$.

Composition Theorems

Indifferentiability allows to "compose security proofs"

Theorem (Composition for μ-KK-indiff. [MRH04])

Let Γ be a cryptosystem based on a block cipher E.
Let $\mathcal{C}^{\mathcal{F}}$ be a block cipher construction based on some ideal primitive $\mathcal{F}$. If

1. *Γ is secure when $E = \mathcal{IC}$ is an ideal cipher*
2. *construction $\mathcal{C}^{\mathcal{F}}$ is μ-KK-indifferentiable from an ideal cipher*
3. *cryptosystem Γ only calls E with keys in $\{k_1, \ldots, k_\mu\}$*

then Γ remains secure when instantiated with $E = \mathcal{C}^{\mathcal{F}}$.

Composition Theorems

Indifferentiability allows to "compose security proofs"

Theorem (Composition for μ-KK-indiff. [MRH04])

Let Γ be a cryptosystem based on a block cipher E.
Let $\mathcal{C}^{\mathcal{F}}$ be a block cipher construction based on some ideal primitive $\mathcal{F}$. If

1. *Γ is secure when $E = \mathcal{IC}$ is an ideal cipher*
2. *construction $\mathcal{C}^{\mathcal{F}}$ is μ-KK-indifferentiable from an ideal cipher*
3. *cryptosystem Γ only calls E with keys in $\{k_1, \ldots, k_\mu\}$*

then Γ remains secure when instantiated with $E = \mathcal{C}^{\mathcal{F}}$.

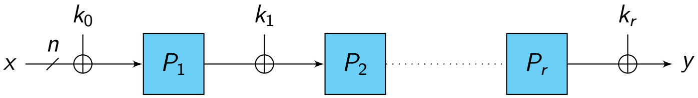
Outline

Background on Known-Key Attacks

Formalizing Multiple Known-Key Security

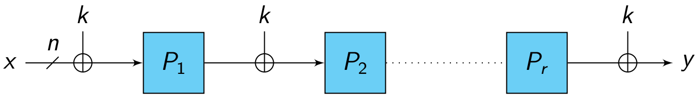
Multiple Known-Key Security of the Iterated Even-Mansour Construction

The Iterated Even-Mansour Construction



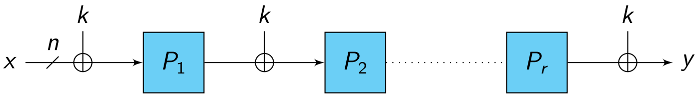
- public permutations P_i's are modeled as ideal (uniformly random and independent)
- we focus on the trivial key-schedule: round keys are equal
- previous indifferentiability results:
  - (fully) indifferentiable from an IC for 12 rounds [LS13]
  - 1-KK-indifferentiable from an IC for 1 round [ABM13]

The Iterated Even-Mansour Construction



- public permutations P_i's are modeled as ideal (uniformly random and independent)
- we focus on the trivial key-schedule: round keys are equal
- previous indifferentiability results:
  - (fully) indifferentiable from an IC for 12 rounds [LS13]
  - 1-KK-indifferentiable from an IC for 1 round [ABM13]

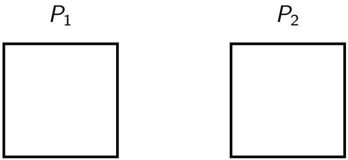
The Iterated Even-Mansour Construction



- public permutations P_i's are modeled as ideal (uniformly random and independent)
- we focus on the trivial key-schedule: round keys are equal
- previous indifferentiability results:
  - (fully) indifferentiable from an IC for 12 rounds [LS13]
  - 1-KK-indifferentiable from an IC for 1 round [ABM13]

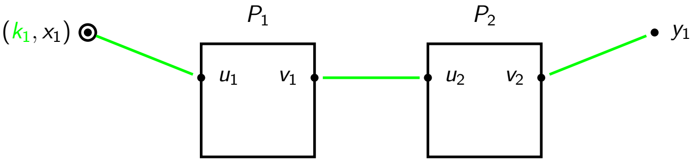
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



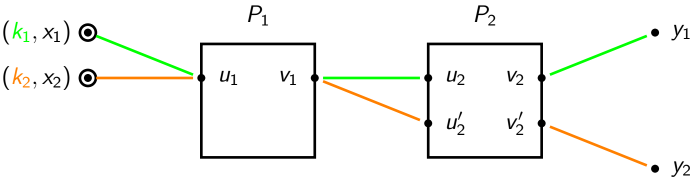
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



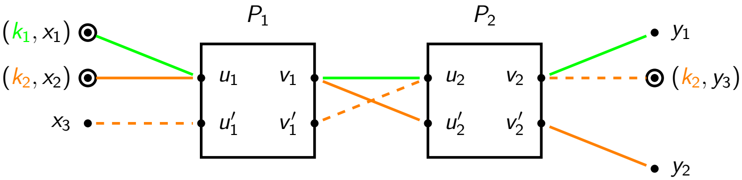
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



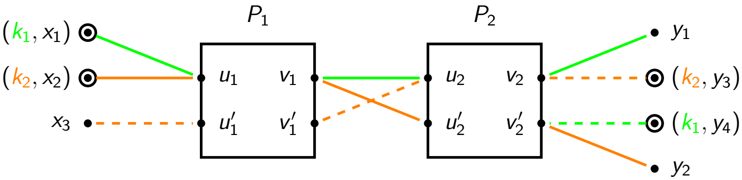
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



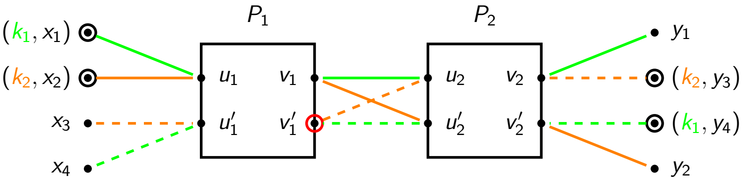
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



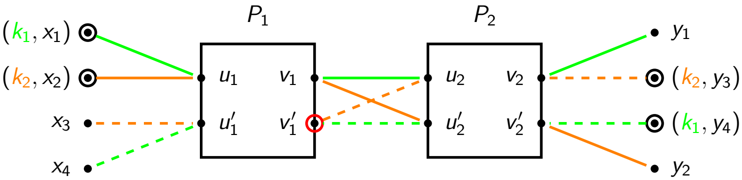
Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):

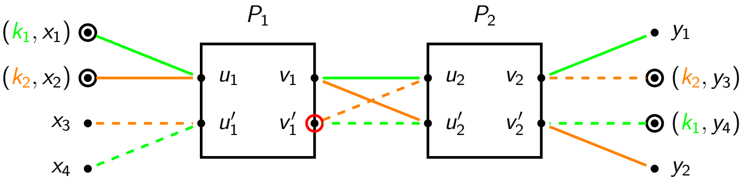


Then

$$\begin{cases} x_1 \oplus x_2 \oplus x_3 \oplus x_4 = 0 \\ y_1 \oplus y_2 \oplus y_3 \oplus y_4 = 0 \end{cases}$$

Multiple KK-Attack against 2-round EM

The attacker is given a pair of keys (k_1, k_2):



Then

$$\begin{cases} x_1 \oplus x_2 \oplus x_3 \oplus x_4 = 0 \\ y_1 \oplus y_2 \oplus y_3 \oplus y_4 = 0 \end{cases}$$

But, given (k_1, k_2) and oracle access to an ideal cipher E, it is hard to find such input/output pairs.

Positive Results

Theorem (μ-KK-indifferentiability)

The 9-round IEM construction is μ-KK-indifferentiable from an ideal cipher.

NB1: full indifferentiability requires $4 \leq r \leq 12$ rounds

NB2: actually not fully proved in the paper, only roughly sketched
(# of rounds very unlikely to be tight)

Theorem (μ-KK-sequential indifferentiability)

The 3-round IEM construction is μ-KK-sequentially indifferentiable from an ideal cipher.

NB: full sequential indifferentiability requires exactly 4 rounds [CS15]

Positive Results

Theorem (μ-KK-indifferentiability)

The 9-round IEM construction is μ-KK-indifferentiable from an ideal cipher.

NB1: full indifferentiability requires $4 \leq r \leq 12$ rounds

NB2: actually not fully proved in the paper, only roughly sketched (# of rounds very unlikely to be tight)

Theorem (μ-KK-sequential indifferentiability)

The 3-round IEM construction is μ-KK-sequentially indifferentiable from an ideal cipher.

NB: full sequential indifferentiability requires exactly 4 rounds [CS15]

Positive Results

Theorem (μ-KK-indifferentiability)

The 9-round IEM construction is μ-KK-indifferentiable from an ideal cipher.

NB1: full indifferentiability requires $4 \leq r \leq 12$ rounds

NB2: actually not fully proved in the paper, only roughly sketched 😇
(# of rounds very unlikely to be tight)

Theorem (μ-KK-sequential indifferentiability)

The 3-round IEM construction is μ-KK-sequentially indifferentiable from an ideal cipher.

NB: full sequential indifferentiability requires exactly 4 rounds [CS15]

Positive Results

Theorem (μ-KK-indifferentiability)

The 9-round IEM construction is μ-KK-indifferentiable from an ideal cipher.

NB1: full indifferentiability requires $4 \leq r \leq 12$ rounds

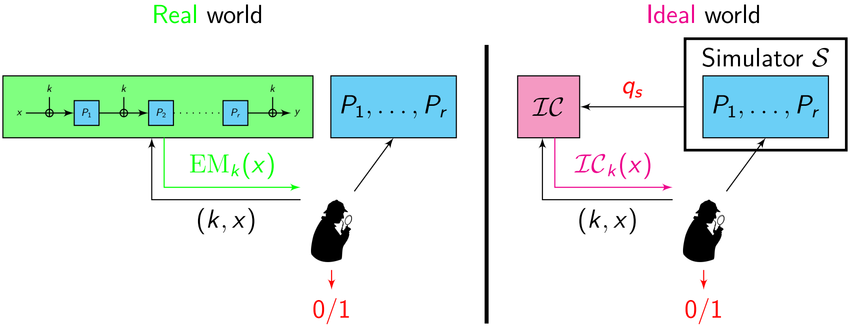
NB2: actually not fully proved in the paper, only roughly sketched (# of rounds very unlikely to be tight)

Theorem (μ-KK-sequential indifferentiability)

The 3-round IEM construction is μ-KK-sequentially indifferentiable from an ideal cipher.

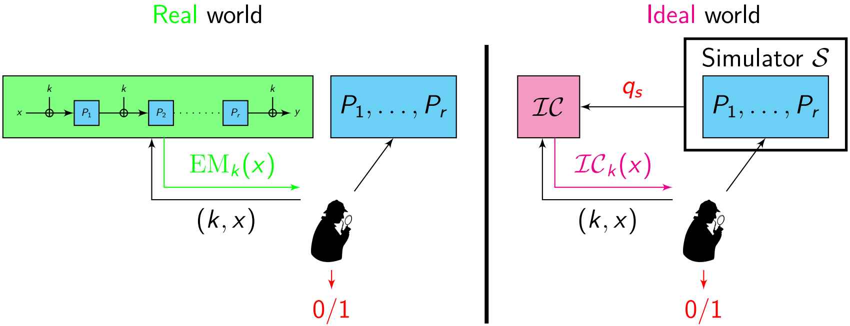
NB: full sequential indifferentiability requires exactly 4 rounds [CS15]

Full vs. Sequential Indifferentiability



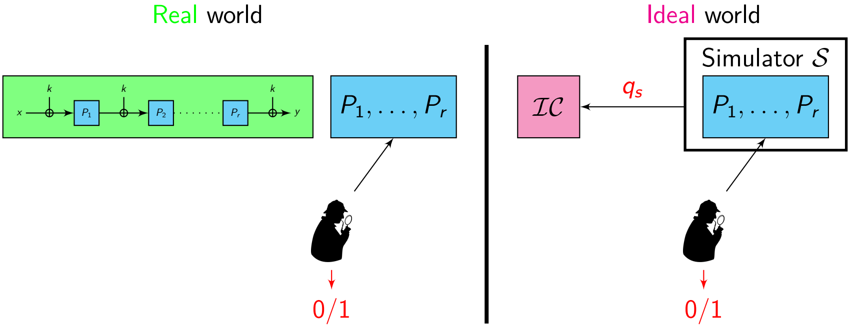
- full indifferentiability: $\mathcal{D}$ can queries its oracle as it wishes
- sequential indifferentiability: two query phases
  1. $\mathcal{D}$ first queries only P_i's/$\mathcal{S}$
  2. and then only construction/$\mathcal{IC}$
- full indiff. $\Rightarrow$ sequential indiff.

Full vs. Sequential Indifferentiability



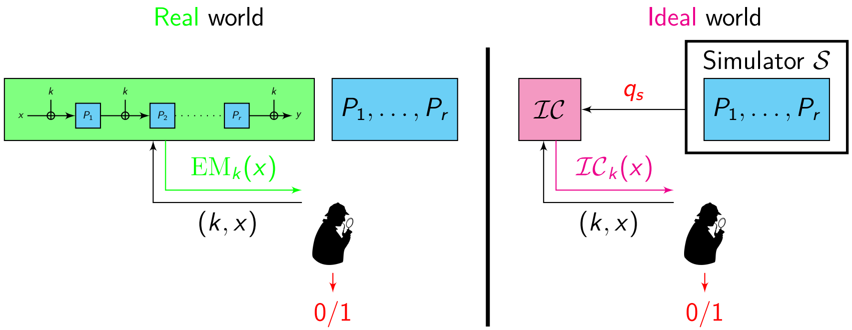
- full indifferentiability: $\mathcal{D}$ can queries its oracle as it wishes
- sequential indifferentiability: two query phases
  1. $\mathcal{D}$ first queries only P_i's/$\mathcal{S}$
  2. and then only construction/$\mathcal{IC}$
- full indiff. $\Rightarrow$ sequential indiff.

Full vs. Sequential Indifferentiability



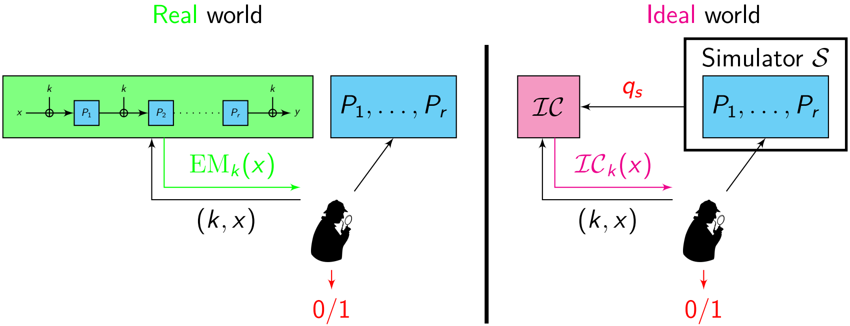
- full indifferentiability: $\mathcal{D}$ can queries its oracle as it wishes
- sequential indifferentiability: two query phases
  1. $\mathcal{D}$ first queries only P_i's/$\mathcal{S}$
  2. and then only construction/$\mathcal{IC}$
- full indiff. $\Rightarrow$ sequential indiff.

Full vs. Sequential Indifferentiability



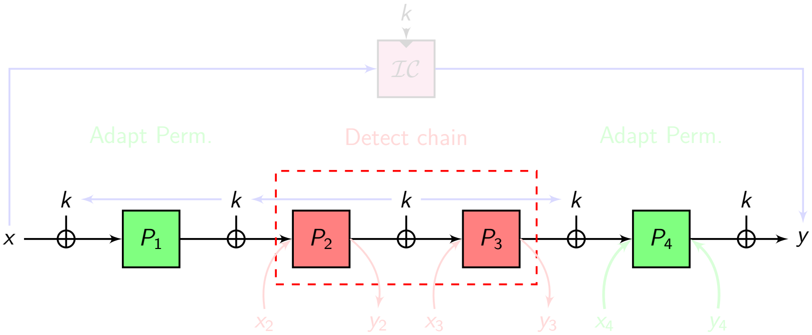
- full indifferentiability: $\mathcal{D}$ can queries its oracle as it wishes
- sequential indifferentiability: two query phases
  1. $\mathcal{D}$ first queries only P_i's/$\mathcal{S}$
  2. and then only construction/$\mathcal{IC}$
- full indiff. $\Rightarrow$ sequential indiff.

Full vs. Sequential Indifferentiability



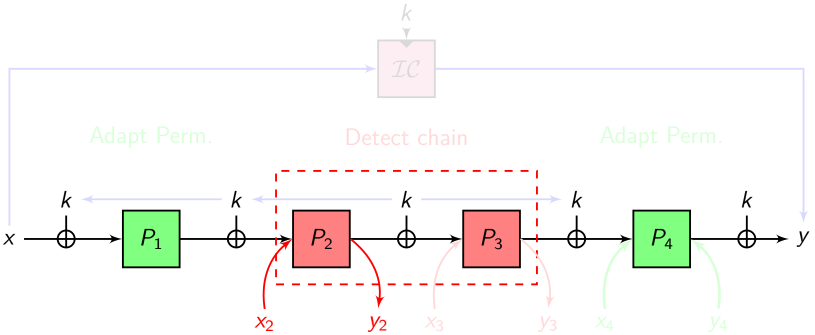
- full indifferentiability: $\mathcal{D}$ can queries its oracle as it wishes
- sequential indifferentiability: two query phases
  1. $\mathcal{D}$ first queries only P_i's/$\mathcal{S}$
  2. and then only construction/$\mathcal{IC}$
- full indiff. $\Rightarrow$ sequential indiff.

Sequential Indifferentiability for 4 Rounds: Simulator



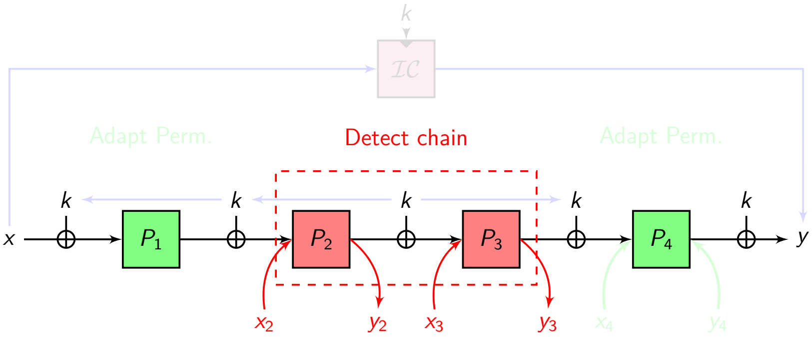
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



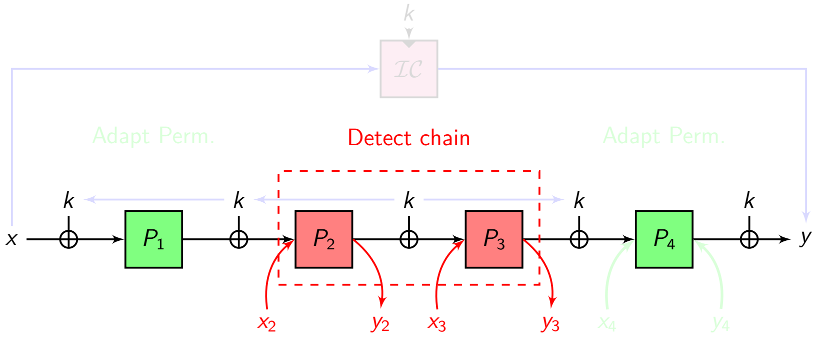
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



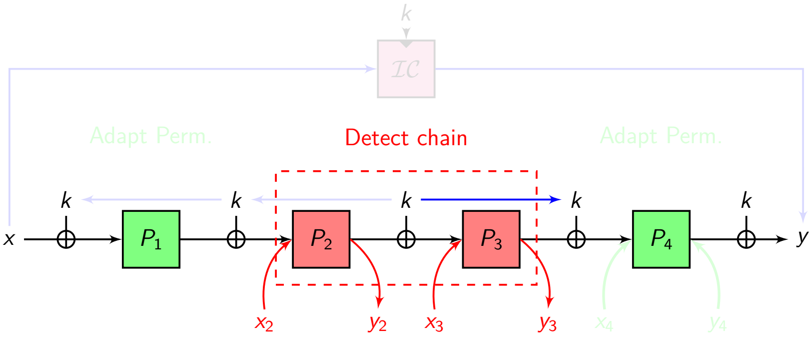
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



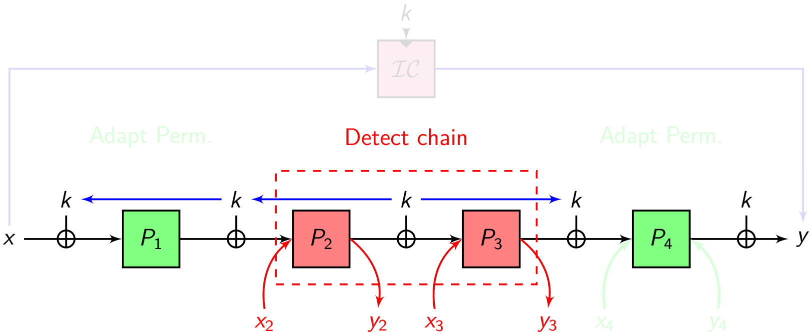
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



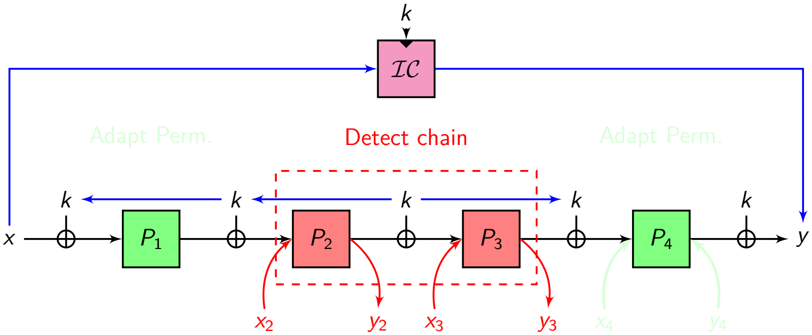
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



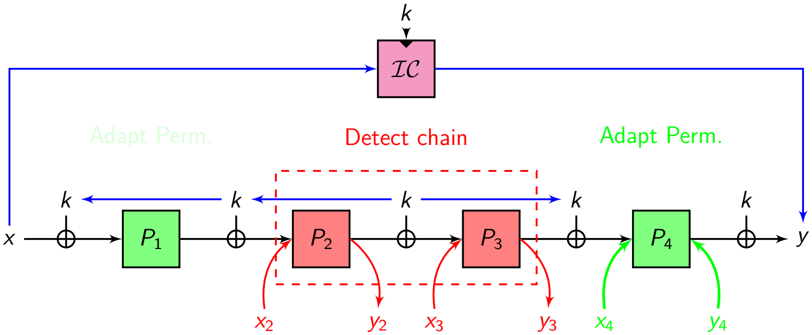
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



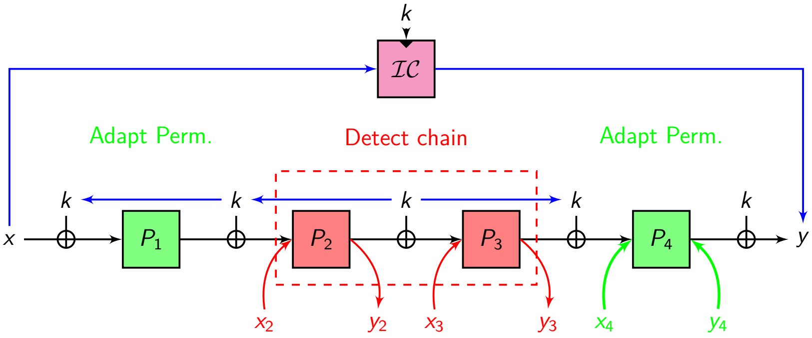
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3$
- $y_4 = \mathcal{IC}(k, x) \oplus k$

Sequential Indifferentiability for 4 Rounds: Simulator



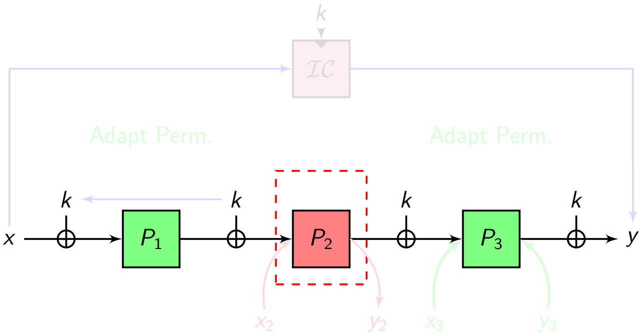
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3 \sim$ random
- $y_4 = \mathcal{IC}(k, x) \oplus k \sim$ random

Sequential Indifferentiability for 4 Rounds: Simulator



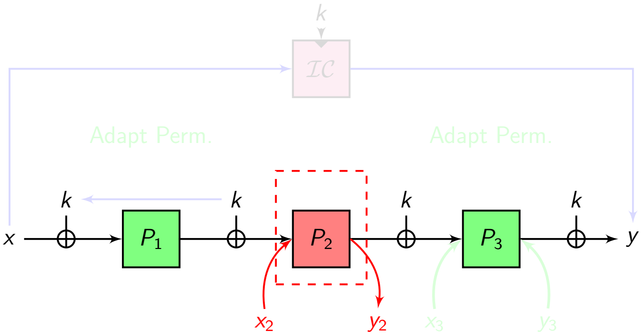
- two queries needed to deduce the key: $k = y_2 \oplus x_3$
- $x_4 = y_3 \oplus k = y_2 \oplus x_3 \oplus y_3 \sim$ random
- $y_4 = \mathcal{IC}(k, x) \oplus k \sim$ random

μ-KK Sequential Indifferentiability for 3 Rounds



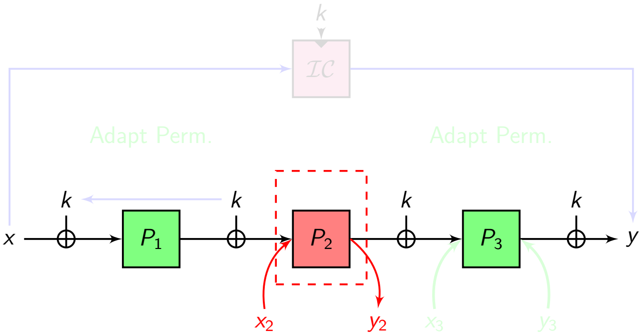
- the simulator can complete chains for each key $k \in \{k_1, \dots, k_\mu\}$
- $x_3 = y_2 \oplus k$
- $y_3 = \mathcal{IC}(k, x) \oplus k$

μ-KK Sequential Indifferentiability for 3 Rounds



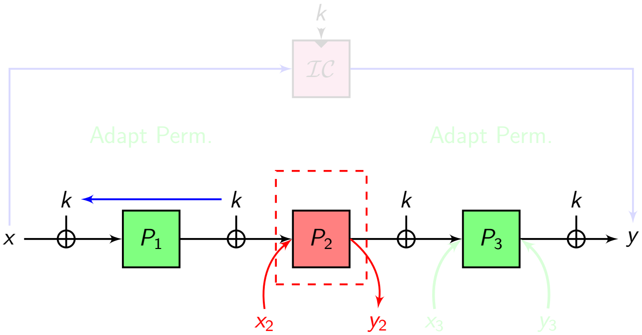
- the simulator can complete chains for each key $k \in \{k_1, \ldots, k_\mu\}$
- $x_3 = y_2 \oplus k$
- $y_3 = \mathcal{IC}(k, x) \oplus k$

μ-KK Sequential Indifferentiability for 3 Rounds



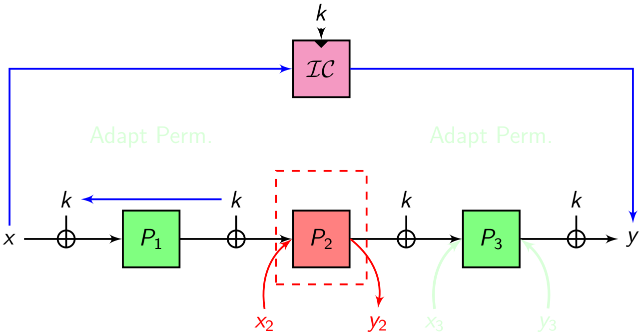
- the simulator can complete chains for each key $k \in \{k_1, \dots, k_\mu\}$
- $x_3 = y_2 \oplus k$
- $y_3 = \mathcal{IC}(k, x) \oplus k$

μ-KK Sequential Indifferentiability for 3 Rounds



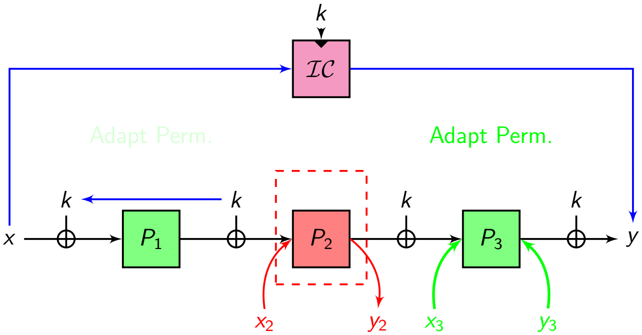
- the simulator can complete chains for each key $k \in \{k_1, \ldots, k_\mu\}$
- $x_3 = y_2 \oplus k$
- $y_3 = \mathcal{IC}(k, x) \oplus k$

μ-KK Sequential Indifferentiability for 3 Rounds



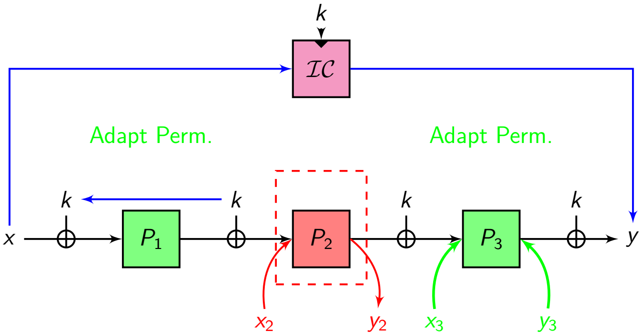
- the simulator can complete chains for each key $k \in \{k_1, \ldots, k_\mu\}$
- $x_3 = y_2 \oplus k$
- $y_3 = \mathcal{IC}(k, x) \oplus k$

μ-KK Sequential Indifferentiability for 3 Rounds



- the simulator can complete chains for each key $k \in \{k_1, \ldots, k_\mu\}$
- $x_3 = y_2 \oplus k \sim$ random
- $y_3 = \mathcal{IC}(k, x) \oplus k \sim$ random

μ-KK Sequential Indifferentiability for 3 Rounds



- the simulator can complete chains for each key $k \in \{k_1, \ldots, k_\mu\}$
- $x_3 = y_2 \oplus k \sim$ random
- $y_3 = \mathcal{IC}(k, x) \oplus k \sim$ random

Conclusion

Summary of known results on the iterated Even-Mansour construction (trivial key schedule $(k, k, \ldots, k)$)

Security notion	# of rounds	Security bound	Simul. (q_S / t_S)	Ref.
Secret key (pseudorandomness)	1	$q^2/2^n$	—	[EM97, DKS12]
	2	$q^{3/2}/2^n$	—	[CLL+14]
XOR Related-Key	3	$q^2/2^n$	—	[CS15, FP15]
1-KK-indiff.	1*	0	q / q	[ABM13]
μ-KK-Seq-indiff., $\mu > 1$	3*	$\mu^2 q^2/2^n$	μq / μq	this paper
Full Seq-indiff.	4*	$q^4/2^n$	q^2 / q^2	[CS15]
μ-KK-indiff., $\mu > 1$	9	$\mu^6 q^6/2^n$	$\mu^2 q$ / $\mu^2 q$	this paper
Full indiff.	12	$q^{12}/2^n$	q^4 / q^6	[LS13]

* tight

The end...

Thanks for your attention!

Comments or questions?

References I

Elena Andreeva, Andrey Bogdanov, and Bart Mennink. Towards Understanding the Known-Key Security of Block Ciphers. In Shiho Moriai, editor, *Fast Software Encryption - FSE 2013*, volume 8424 of *LNCS*, pages 348–366. Springer, 2013.

Shan Chen, Rodolphe Lampe, Jooyoung Lee, Yannick Seurin, and John P. Steinberger. Minimizing the Two-Round Even-Mansour Cipher. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 (Proceedings, Part I)*, volume 8616 of *LNCS*, pages 39–56. Springer, 2014. Full version available at `http://eprint.iacr.org/2014/443`.

Benoît Cogliati and Yannick Seurin. On the Provable Security of the Iterated Even-Mansour Cipher against Related-Key and Chosen-Key Attacks. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 (Proceedings, Part I)*, volume 9056 of *LNCS*, pages 584–613. Springer, 2015. Full version available at `http://eprint.iacr.org/2015/069`.

References II

Orr Dunkelman, Nathan Keller, and Adi Shamir. Minimalism in Cryptography: The Even-Mansour Scheme Revisited. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 336–354. Springer, 2012.

Shimon Even and Yishay Mansour. A Construction of a Cipher from a Single Pseudorandom Permutation. *Journal of Cryptology*, 10(3):151–162, 1997.

Pooya Farshim and Gordon Procter. The Related-Key Security of Iterated Even-Mansour Ciphers. In Gregor Leander, editor, *Fast Software Encryption - FSE 2015*, volume 9054 of *LNCS*, pages 342–363. Springer, 2015. Full version available at http://eprint.iacr.org/2014/953.

Lars R. Knudsen and Vincent Rijmen. Known-Key Distinguishers for Some Block Ciphers. In Kaoru Kurosawa, editor, *Advances in Cryptology - ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 315–324. Springer, 2007.

References III

Rodolphe Lampe and Yannick Seurin. How to Construct an Ideal Cipher from a Small Set of Public Permutations. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013 (Proceedings, Part I)*, volume 8269 of *LNCS*, pages 444–463. Springer, 2013. Full version available at `http://eprint.iacr.org/2013/255`.

Ueli M. Maurer, Renato Renner, and Clemens Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In Moni Naor, editor, *Theory of Cryptography Conference-TCC 2004*, volume 2951 of *LNCS*, pages 21–39. Springer, 2004.

Phillip Rogaway and John P. Steinberger. Constructing Cryptographic Hash Functions from Fixed-Key Blockciphers. In David Wagner, editor, *Advances in Cryptology - CRYPTO 2008*, volume 5157 of *LNCS*, pages 433–450. Springer, 2008.

Phillip Rogaway and John P. Steinberger. Security/Efficiency Tradeoffs for Permutation-Based Hashing. In Nigel P. Smart, editor, *Advances in Cryptology - EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 220–236. Springer, 2008.