Efficient and Optimally Secure Key-Length Extension for Block Ciphers via Randomized Cascading

Peter Gaži
ETH Zurich
Comenius University Bratislava

Stefano Tessaro
MIT

Eurocrypt 2012

Outline

Block Ciphers and Key-Length Extension

Existing Approaches

Our Generic Attacks

Our Construction

Outline

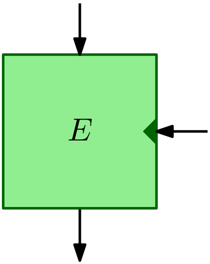
Block Ciphers and Key-Length Extension

Existing Approaches

Our Generic Attacks

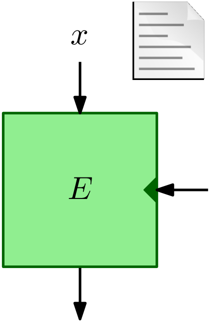
Our Construction

Block Ciphers



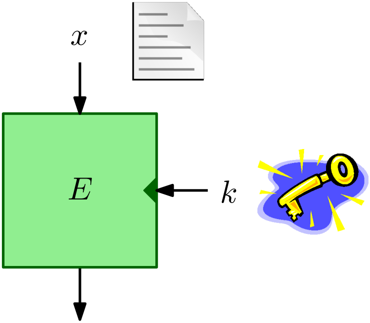
- e.g. DES, IDEA, AES

Block Ciphers



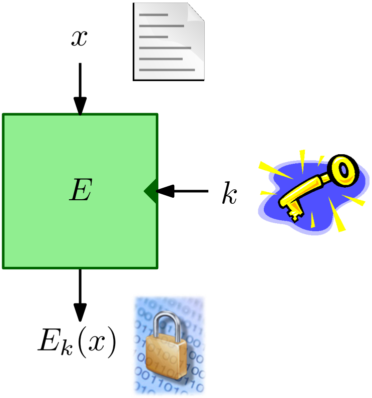
- e.g. DES, IDEA, AES
- $\{0,1\}^n$

Block Ciphers



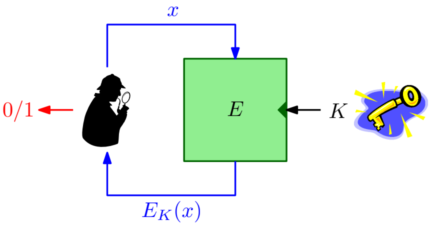
- e.g. DES, IDEA, AES
- $\{0,1\}^{\kappa} \times \{0,1\}^{n}$

Block Ciphers

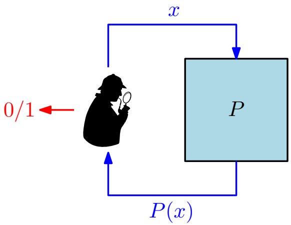


- e.g. DES, IDEA, AES
- $E: \{0,1\}^{\kappa} \times \{0,1\}^{n} \to \{0,1\}^{n}$

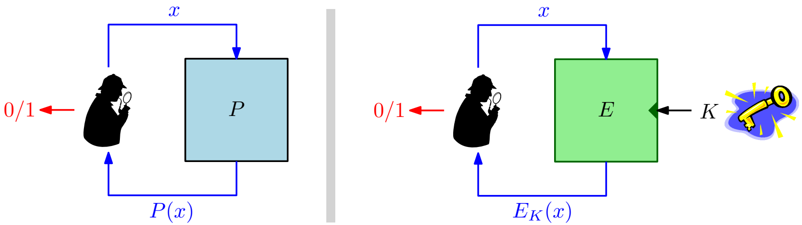
Block Cipher Security: Pseudo-Random Permutations



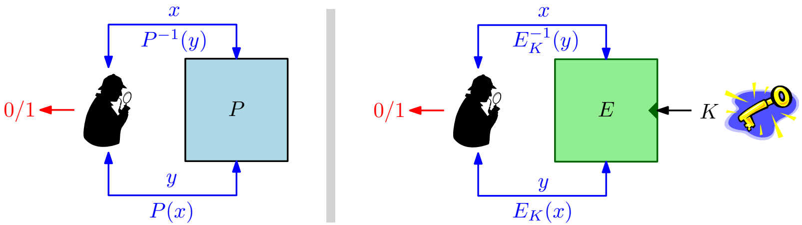
Block Cipher Security: Pseudo-Random Permutations



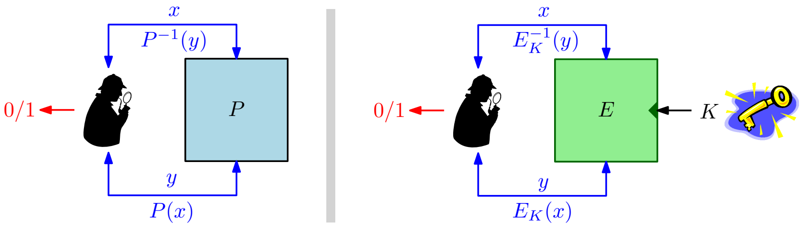
Block Cipher Security: Pseudo-Random Permutations



Block Cipher Security: Pseudo-Random Permutations



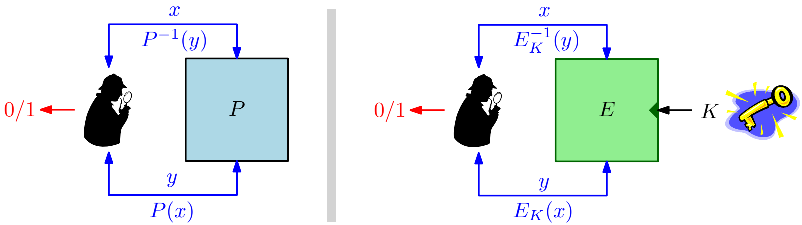
Block Cipher Security: Pseudo-Random Permutations



PRP advantage:

$$\Delta^D(P, E_K) := |\Pr[D(P) = 1] - \Pr[D(E_K) = 1]|$$

Block Cipher Security: Pseudo-Random Permutations

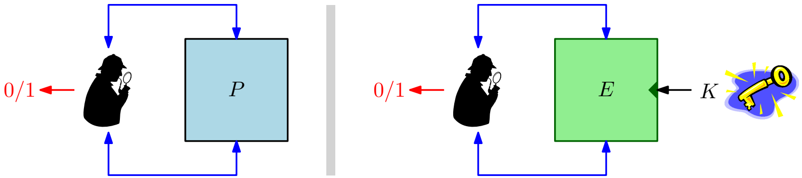


PRP advantage:
$$\Delta^D(P, E_K) := |\Pr[D(P) = 1] - \Pr[D(E_K) = 1]|$$

PRP security: What resources does D need to achieve

$$\Delta^D(P, E_K) \geq const \ ?$$

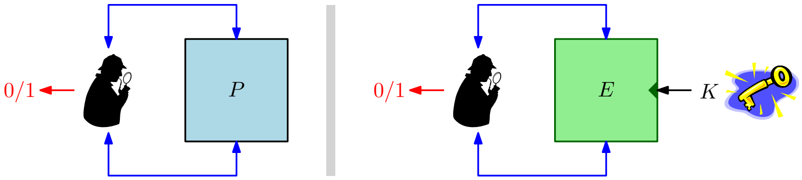
Sufficient Key Length Is Essential



- Key K recoverable in about 2^{κ} evaluations of E

Given $\mathcal{O} \in \{P, E_K\}$:

Sufficient Key Length Is Essential

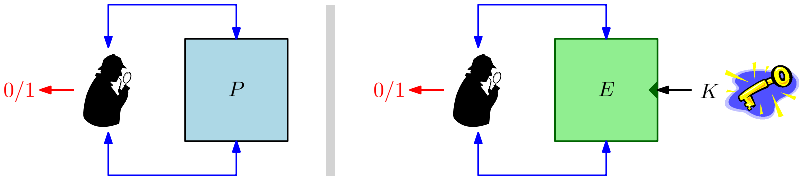


- Key K recoverable in about 2^{κ} evaluations of E

Given $\mathcal{O} \in \{P, E_K\}$:

1. $y \leftarrow \mathcal{O}(0^n)$

Sufficient Key Length Is Essential

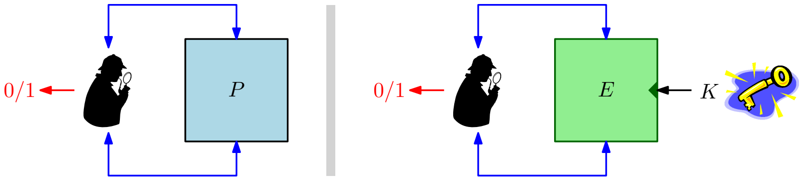


- Key K recoverable in about 2^{κ} evaluations of E

Given $\mathcal{O} \in \{P, E_K\}$:

1. $y \leftarrow \mathcal{O}(0^n)$
2. $\forall k \in \{0,1\}^{\kappa}: y^{(k)} \leftarrow E_k(0^n)$

Sufficient Key Length Is Essential

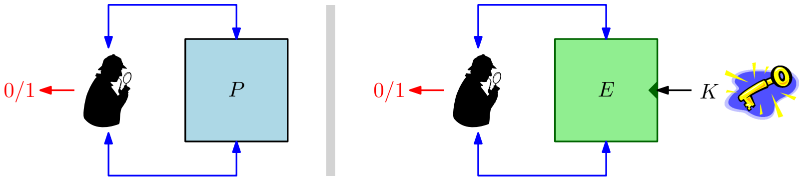


- Key K recoverable in about 2^κ evaluations of E

Given $\mathcal{O} \in \{P, E_K\}$:

1. $y \leftarrow \mathcal{O}(0^n)$
2. $\forall k \in \{0,1\}^\kappa : y^{(k)} \leftarrow E_k(0^n)$
3. if $y = y^{(k)}$, verify k

Sufficient Key Length Is Essential

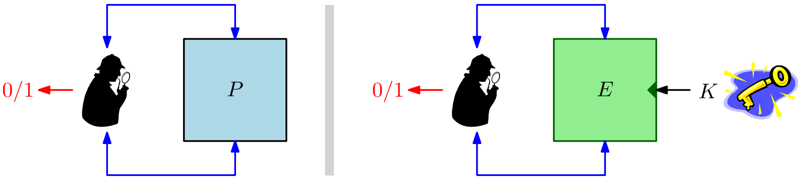


- Key K recoverable in about 2^κ evaluations of E

  Given $\mathcal{O} \in \{P, E_K\}$:
  1. $y \leftarrow \mathcal{O}(0^n)$
  2. $\forall k \in \{0,1\}^\kappa : y^{(k)} \leftarrow E_k(0^n)$
  3. if $y = y^{(k)}$, verify k

- Upper-bounds PRP security!
  - problem for e.g. DES

Sufficient Key Length Is Essential



- Key K recoverable in about 2^κ evaluations of E

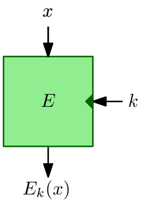
  Given $\mathcal{O} \in \{P, E_K\}$:

  1. $y \leftarrow \mathcal{O}(0^n)$
  2. $\forall k \in \{0,1\}^\kappa : y^{(k)} \leftarrow E_k(0^n)$
  3. if $y = y^{(k)}$, verify k

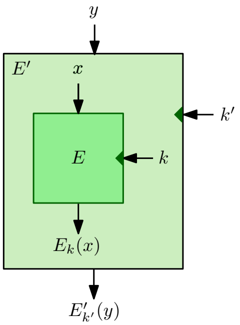
  generic attack!

- Upper-bounds PRP security!
  - problem for e.g. DES

This Paper: Key-Length Extension



This Paper: Key-Length Extension

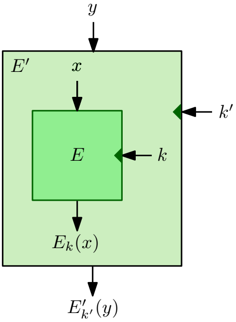


- Goal: construction

$$E'[E]\colon \{0,1\}^{\kappa'} \times \{0,1\}^n \to \{0,1\}^n$$

which is again a block cipher

This Paper: Key-Length Extension



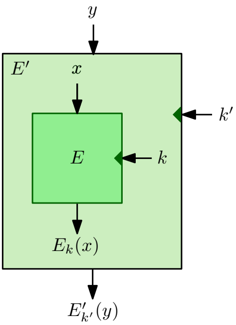
- Goal: construction

$$E'[E]\colon \{0,1\}^{\kappa'} \times \{0,1\}^n \to \{0,1\}^n$$

which is again a block cipher such that

- $\kappa' > \kappa$
- best generic attack requires $> 2^{\kappa}$ evaluations of E, E'

This Paper: Key-Length Extension



- Goal: construction

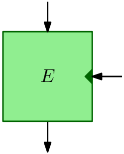
$$E'[E]\colon \{0,1\}^{\kappa'} \times \{0,1\}^n \to \{0,1\}^n$$

which is again a block cipher such that

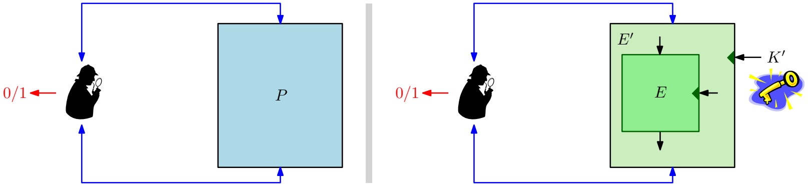
- $\kappa' > \kappa$
- best generic attack requires $> 2^{\kappa}$ evaluations of E, E'

Generic Security: Ideal Block Cipher Model

- $\forall k$: independent uniformly random permutation

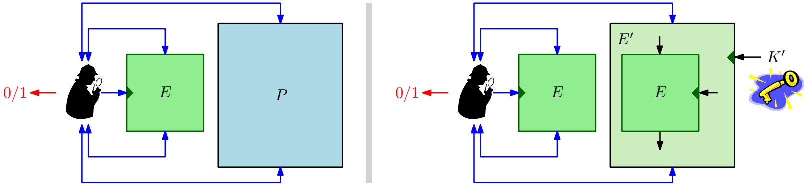


Key-Length Extension in ICM



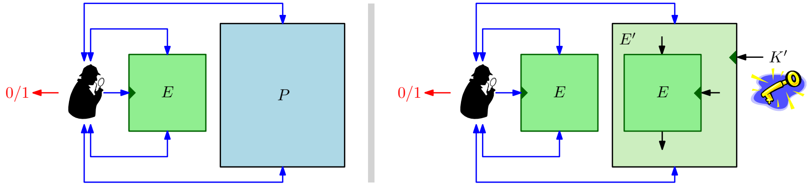
- queries to the construction (P or $E'_{K'}[E]$)

Key-Length Extension in ICM



- queries to the construction (P or $E'_{K'}[E]$)
- queries to $E(\cdot,\cdot)$

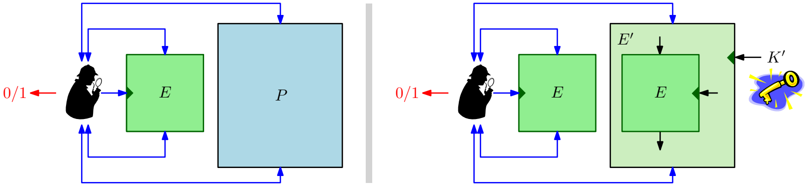
Key-Length Extension in ICM



- queries to the construction (P or $E'_{K'}[E]$)
- queries to $E(\cdot,\cdot)$

Complexity measure:
sum of queries

Key-Length Extension in ICM



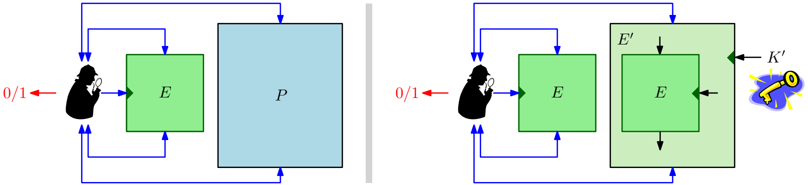
- queries to the construction (P or $E'_{K'}[E]$)
- queries to $E(\cdot,\cdot)$

Complexity measure:
sum of queries

PRP security: What resources does D need to achieve

$$\Delta^D\left((E,P),(E,E'_{K'}[E])\right) \geq const\ ?$$

Key-Length Extension in ICM



- queries to the construction (P or $E'_{K'}[E]$)
- queries to $E(\cdot,\cdot)$

Complexity measure:
sum of queries

PRP security: How many queries? does D need to achieve

$$\Delta^D\left((E,P),(E,E'_{K'}[E])\right) \geq const \ ?$$

Outline

Block Ciphers and Key-Length Extension

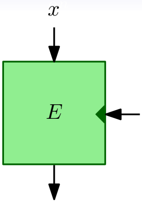
Existing Approaches

Our Generic Attacks

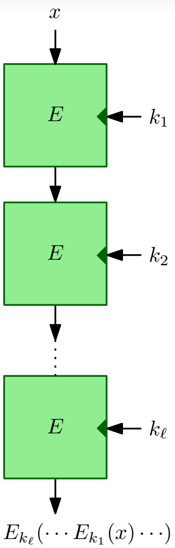
Our Construction

Outline

Approach I: Cascading

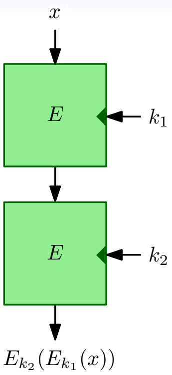


Approach I: Cascading



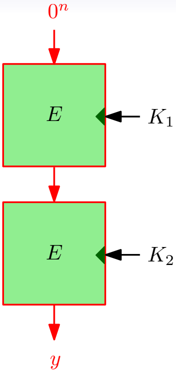
- Re-encrypt with independent keys

Approach I: Cascading



- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack
    (2^{κ} evaluations of E and E^{-1})

Approach I: Cascading

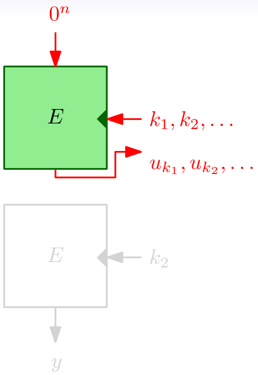


- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})

    Given $\mathcal{O} \in \{E_{K_2}(E_{K_1}(\cdot)), P(\cdot)\}$:

    1. $y \leftarrow \mathcal{O}(0^n)$

Approach I: Cascading

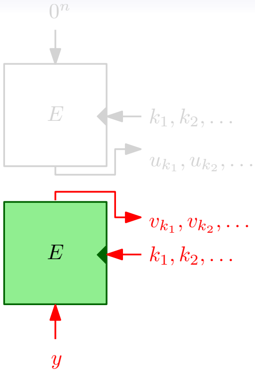


- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})

    Given $\mathcal{O} \in \{E_{K_2}(E_{K_1}(\cdot)), P(\cdot)\}$:

    1. $y \leftarrow \mathcal{O}(0^n)$
    2. $\forall k \in \{0,1\}^\kappa : u_k \leftarrow E_k(0^n)$

Approach I: Cascading

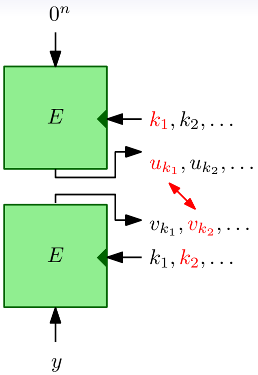


- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})

    Given $\mathcal{O} \in \{E_{K_2}(E_{K_1}(\cdot)), P(\cdot)\}$:

    1. $y \leftarrow \mathcal{O}(0^n)$
    2. $\forall k \in \{0,1\}^\kappa : u_k \leftarrow E_k(0^n)$
    3. $\forall k \in \{0,1\}^\kappa : v_k \leftarrow E_k^{-1}(y)$

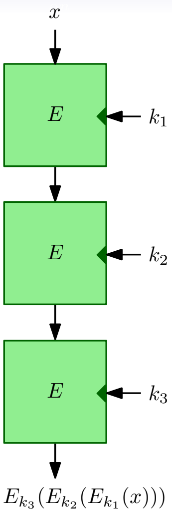
Approach I: Cascading



- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})

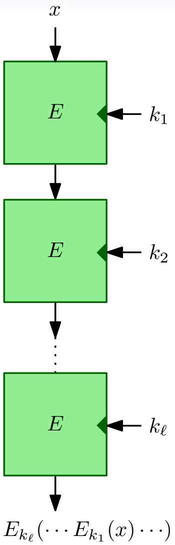
    Given $\mathcal{O} \in \{E_{K_2}(E_{K_1}(\cdot)), P(\cdot)\}$:
    1. $y \leftarrow \mathcal{O}(0^n)$
    2. $\forall k \in \{0,1\}^\kappa : u_k \leftarrow E_k(0^n)$
    3. $\forall k \in \{0,1\}^\kappa : v_k \leftarrow E_k^{-1}(y)$
    4. if $u_{k_i} = v_{k_j}$: verify (k_i, k_j)

Approach I: Cascading



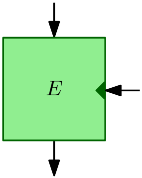
- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})
- Triple Encryption
  - Secure up to $2^{\kappa+\min\{n/2,\kappa/2\}}$ queries in ICM (Bellare and Rogaway, EC'06)
  - 3DES can be attacked in 2^{90} queries (Lucks, FSE'98)

Approach I: Cascading

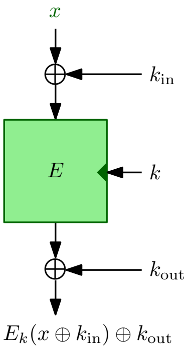


- Re-encrypt with independent keys
- Double Encryption
  - Meet-in-the-middle attack (2^κ evaluations of E and E^{-1})
- Triple Encryption
  - Secure up to $2^{\kappa+\min\{n/2,\kappa/2\}}$ queries in ICM (Bellare and Rogaway, EC'06)
  - 3DES can be attacked in 2^{90} queries (Lucks, FSE'98)
- Longer Cascades
  - Security improves for $\kappa < n$ in ICM (Gaži and Maurer, AC'09)

Approach II: Key Whitening

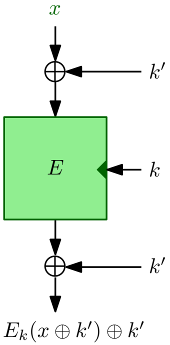


Approach II: Key Whitening



DESX [Rivest]

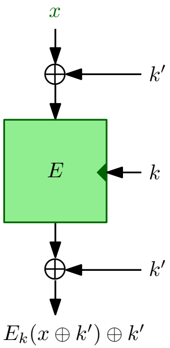
Approach II: Key Whitening



DESX [Rivest]

- Secure up to $2^{(\kappa+n)/2}$ queries in ICM (Kilian and Rogaway, Crypto'96)

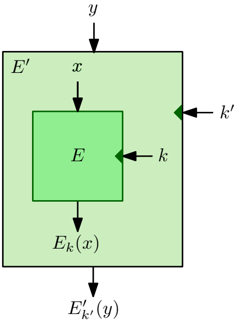
Approach II: Key Whitening



DESX [Rivest]

- Secure up to $2^{(\kappa+n)/2}$ queries in ICM (Kilian and Rogaway, Crypto'96)
- Also secure if $k_{\text{in}} = k_{\text{out}}$

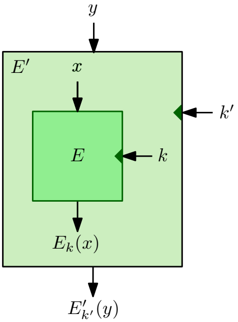
Can we do better?



So far ...

- no constructions secure beyond $2^{\kappa+\min\{\kappa/2,n/2\}}$ queries
- security beyond $2^{\max\{\kappa,n\}}$ requires 3 BC queries

Can we do better?



So far ...

- no constructions secure beyond $2^{\kappa+\min\{\kappa/2,n/2\}}$ queries
- security beyond $2^{\max\{\kappa,n\}}$ requires 3 BC queries

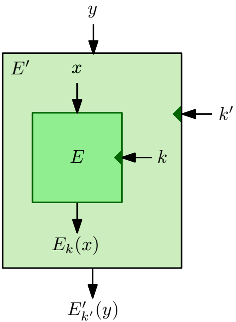
What can be achieved with at most 2 queries to E?

Outline

Outline

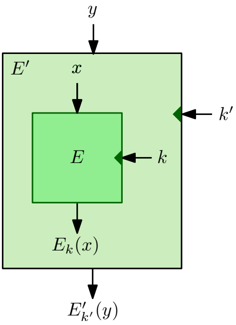
One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa, n\}}$ security!



One Query Is Not Enough

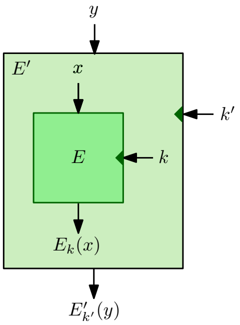
Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



- Assuming $\forall k' : y \neq y' \Rightarrow x \neq x'$

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!

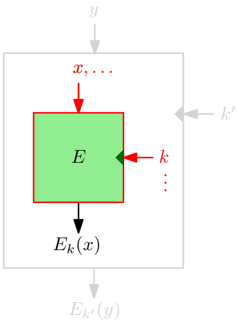


- Assuming $\forall k' \colon y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



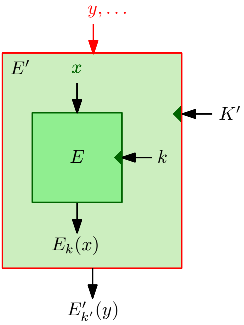
- Assuming $\forall k'\colon y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

  1. $2^{(n+\kappa)/2}$ random distinct queries (x_j, k_j) to E

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



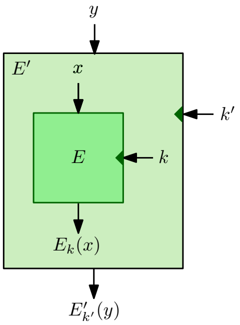
- Assuming $\forall k'$: $y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

  1. $2^{(n+\kappa)/2}$ random distinct queries (x_j, k_j) to E
  2. $2^{(n+\kappa)/2}$ distinct queries y_i to $\mathcal{O}$

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



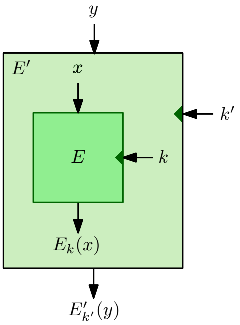
- Assuming $\forall k'\colon y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

  1. $2^{(n+\kappa)/2}$ random distinct queries (x_j, k_j) to E
  2. $2^{(n+\kappa)/2}$ distinct queries y_i to $\mathcal{O}$
  3. $\forall k'\colon z_i \leftarrow E'_{k'}[E](y_i)$ if E-value available

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



- Assuming $\forall k'$: $y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

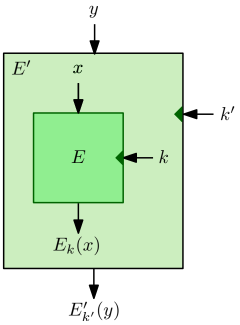
  1. $2^{(n+\kappa)/2}$ random distinct queries (x_j, k_j) to E
  2. $2^{(n+\kappa)/2}$ distinct queries y_i to $\mathcal{O}$
  3. $\forall k'$: $z_i \leftarrow E'_{k'}[E](y_i)$ if E-value available

     check $z_i \stackrel{?}{=} \mathcal{O}(y_i)$

     - if $\mathcal{O} = E'_{K'}$: succeeds for $k' = K'$
     - if $\mathcal{O} = P$: fails

One Query Is Not Enough

Any one-query construction can achieve at most $2^{\max\{\kappa,n\}}$ security!



- Assuming $\forall k' \colon y \neq y' \Rightarrow x \neq x'$

  Given $\mathcal{O} \in \{E'_{K'}[E](\cdot), P(\cdot)\}$:

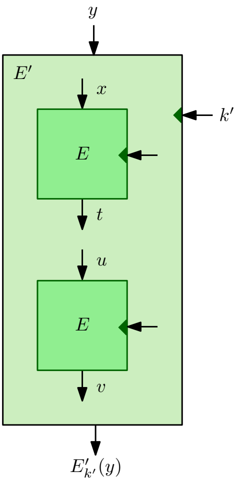
  1. $2^{(n+\kappa)/2}$ random distinct queries (x_j, k_j) to E
  2. $2^{(n+\kappa)/2}$ distinct queries y_i to $\mathcal{O}$
  3. $\forall k' \colon z_i \leftarrow E'_{k'}[E](y_i)$ if E-value available

     check $z_i \stackrel{?}{=} \mathcal{O}(y_i)$

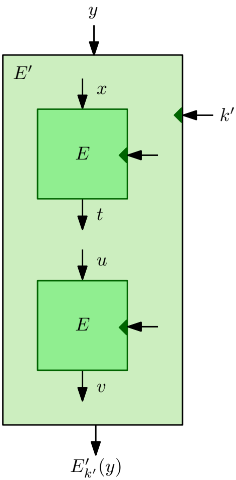
     - if $\mathcal{O} = E'_{K'}$: succeeds for $k' = K'$
     - if $\mathcal{O} = P$: fails

- Non-injective queries do no better

How About Two Queries?

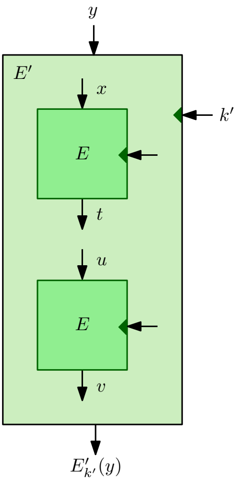


How About Two Queries?



A natural class of 2-query constructions can achieve at most $2^{\kappa+n/2}$ security.

How About Two Queries?

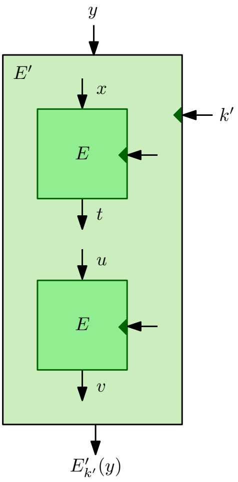


A natural class of 2-query constructions can achieve at most $2^{\kappa+n/2}$ security.

- Constructions with "injective queries":

$$\forall k': \quad y \neq y' \Rightarrow x \neq x'$$
$$\forall k': \quad t \neq t' \Rightarrow u \neq u'$$

How About Two Queries?



A natural class of 2-query constructions can achieve at most $2^{\kappa+n/2}$ security.

- Constructions with "injective queries":

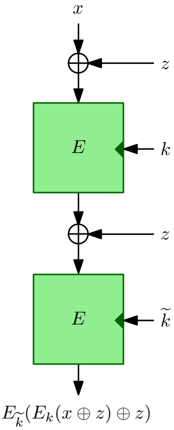
$$\forall k': \quad y \neq y' \Rightarrow x \neq x'$$
$$\forall k': \quad t \neq t' \Rightarrow u \neq u'$$

There is room for security increase, we achieve it!

Outline

Outline

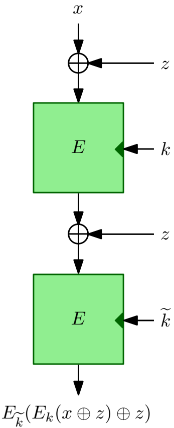
The Double XOR-Cascade



Definition

$$2XOR_{k,z}[E](x) := E_{\widetilde{k}}(E_k(x \oplus z) \oplus z)$$

The Double XOR-Cascade

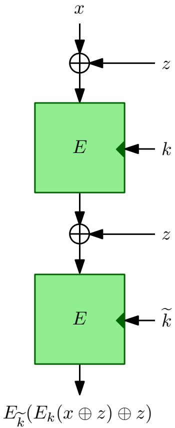


Definition

$$2XOR_{k,z}[E](x) := E_{\widetilde{k}}(E_k(x \oplus z) \oplus z)$$

- Same key z in both whitening steps

The Double XOR-Cascade

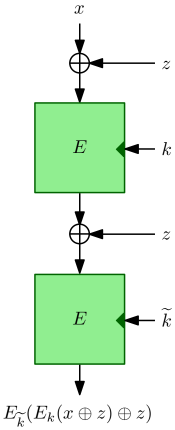


Definition

$$2XOR_{k,z}[E](x) := E_{\widetilde{k}}(E_k(x \oplus z) \oplus z)$$

- Same key z in both whitening steps
- $\widetilde{k}$ derived from k e.g. by a bit-flip

The Double XOR-Cascade



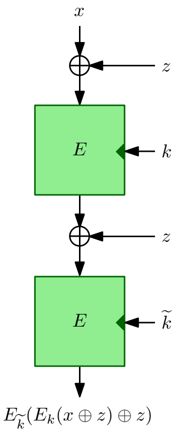
Definition

$$2XOR_{k,z}[E](x) := E_{\widetilde{k}}(E_k(x \oplus z) \oplus z)$$

- Same key z in both whitening steps
- $\widetilde{k}$ derived from k e.g. by a bit-flip

keylength $\kappa + n$

The Double XOR-Cascade



Definition

$$2XOR_{k,z}[E](x) := E_{\widetilde{k}}(E_k(x \oplus z) \oplus z)$$

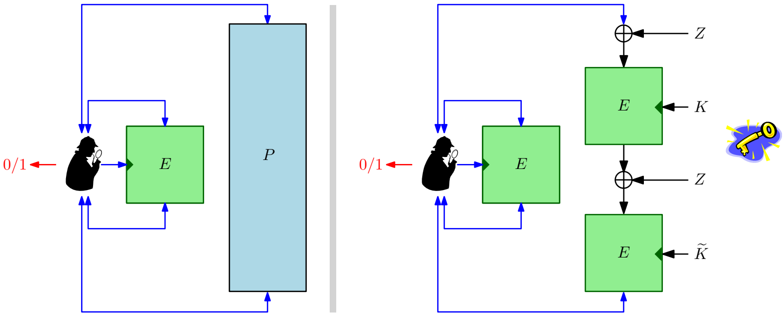
- Same key z in both whitening steps
- $\widetilde{k}$ derived from k e.g. by a bit-flip

keylength $\kappa + n$

Main Result

Double XOR-Cascade is secure up to $2^{\kappa + n/2}$ queries.

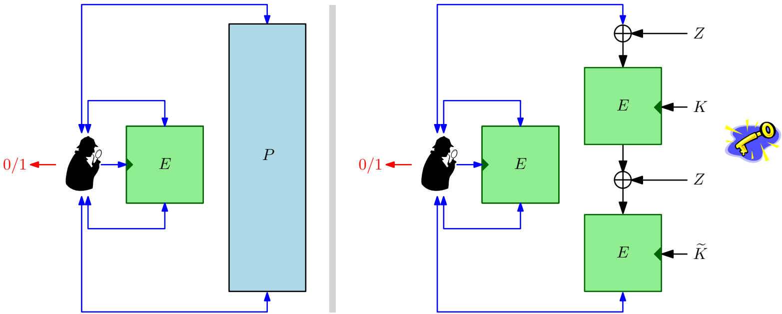
A Glimpse at the Proof



The Initial Setting

$$\Delta^D\left((E, P), (E, 2XOR_{K,Z}[E])\right)$$

A Glimpse at the Proof

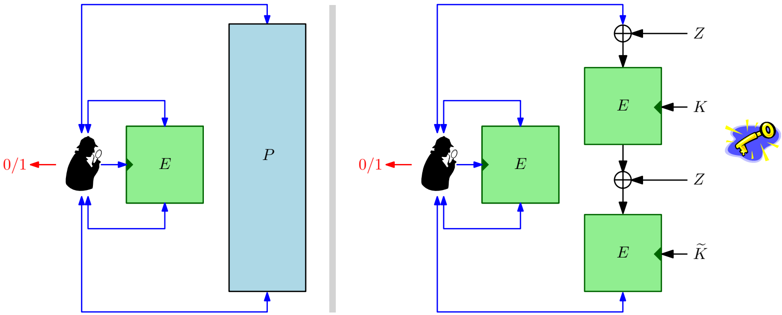


The Initial Setting

$$\Delta^D\big((E,P),(E,2XOR_{K,Z}[E])\big)$$

Goal: Δ^D small if $< 2^{\kappa+n/2}$ queries

A Glimpse at the Proof



The Initial Setting

$$\Delta^D\big((E, P), (E, 2XOR_{K,Z}[E])\big)$$

Goal: Δ^D small if $< 2^{\kappa+n/2}$ queries

Main Steps

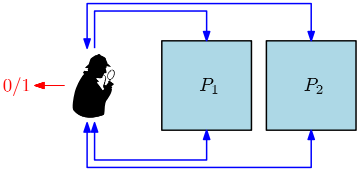
- Reduce to a simpler combinatorial problem
- Show it is hard

A Glimpse at the Proof (2)

The New Problem: Distinguishing Permutations

A Glimpse at the Proof (2)

The New Problem: Distinguishing Permutations

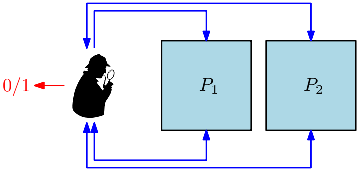


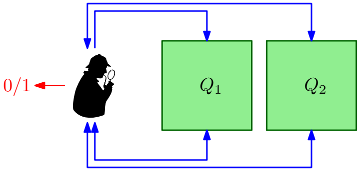
Independent

- P_1, P_2 independent uniformly random permutations

A Glimpse at the Proof (2)

The New Problem: Distinguishing Permutations





Independent

- P_1, P_2 independent uniformly random permutations

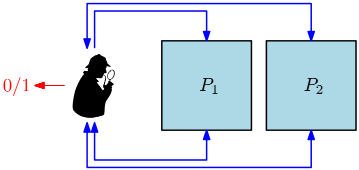
Correlated

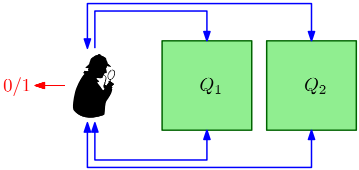
- Q_1, Q_2 random perms s.t. for a random secret Z

$$\forall x : Q_2(Q_1(x \oplus Z) \oplus Z) = x$$

A Glimpse at the Proof (2)

The New Problem: Distinguishing Permutations





Independent

- P_1, P_2 independent uniformly random permutations

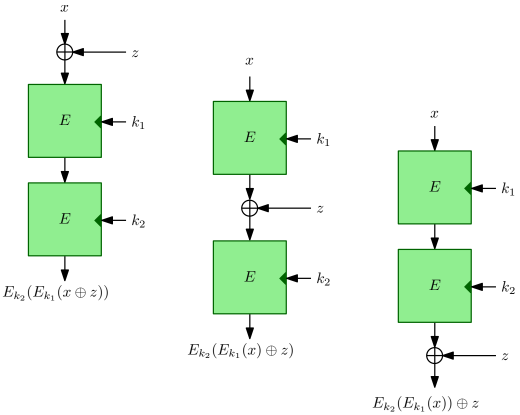
Correlated

- Q_1, Q_2 random perms s.t. for a random secret Z

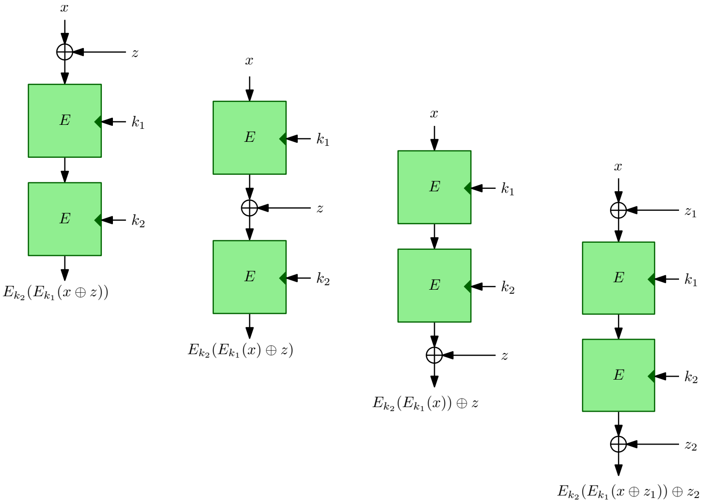
$$\forall x : Q_2(Q_1(x \oplus Z) \oplus Z) = x$$

Hard for $< 2^{n/2}$ queries!

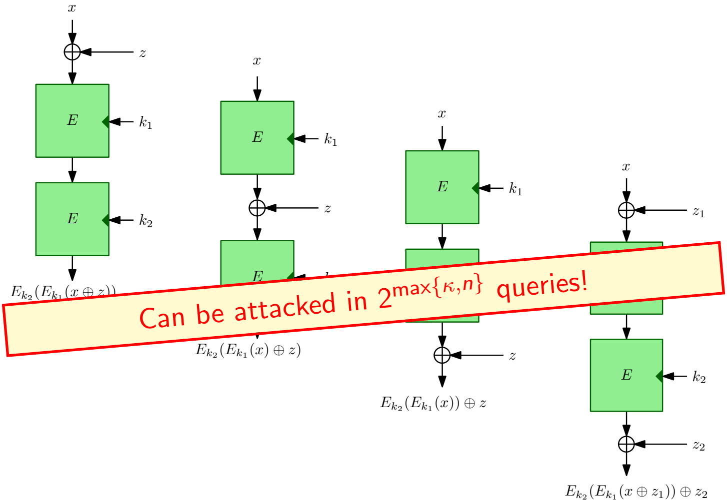
Details Are Important



Details Are Important

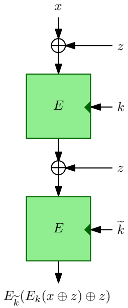


Details Are Important



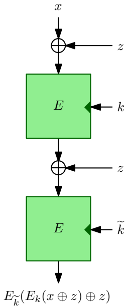
Summary

- New key-length extending construction for block ciphers
  - more efficient than triple encryption (2 BC queries per invocation)
  - more secure than triple encryption
    (Triple cascade: up to $2^{\kappa+\min\{\kappa/2,n/2\}}$)
    (Double XOR-cascade: up to $2^{\kappa+n/2}$)



Summary

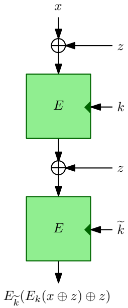
- New key-length extending construction for block ciphers
  - more efficient than triple encryption (2 BC queries per invocation)
  - more secure than triple encryption
    (Triple cascade: up to $2^{\kappa+\min\{\kappa/2,n/2\}}$)
    (Double XOR-cascade: up to $2^{\kappa+n/2}$)



- Generic attacks supporting optimality
  - one-query constructions insecure above $2^{\max\{\kappa,n\}}$
  - "injective" two-query constructions insecure above $2^{\kappa+n/2}$

Summary

- New key-length extending construction for block ciphers
  - more efficient than triple encryption (2 BC queries per invocation)
  - more secure than triple encryption
    (Triple cascade: up to $2^{\kappa+\min\{\kappa/2,n/2\}}$)
    (Double XOR-cascade: up to $2^{\kappa+n/2}$)



- Generic attacks supporting optimality
  - one-query constructions insecure above $2^{\max\{\kappa,n\}}$
  - "injective" two-query constructions insecure above $2^{\kappa+n/2}$

Thank you!