CIS 5371, Fall 2024

Blockcipher

Viet Tung Hoang

The slides are loosely based on those of Prof. Mihir Bellare, UC San Diego.

Agenda

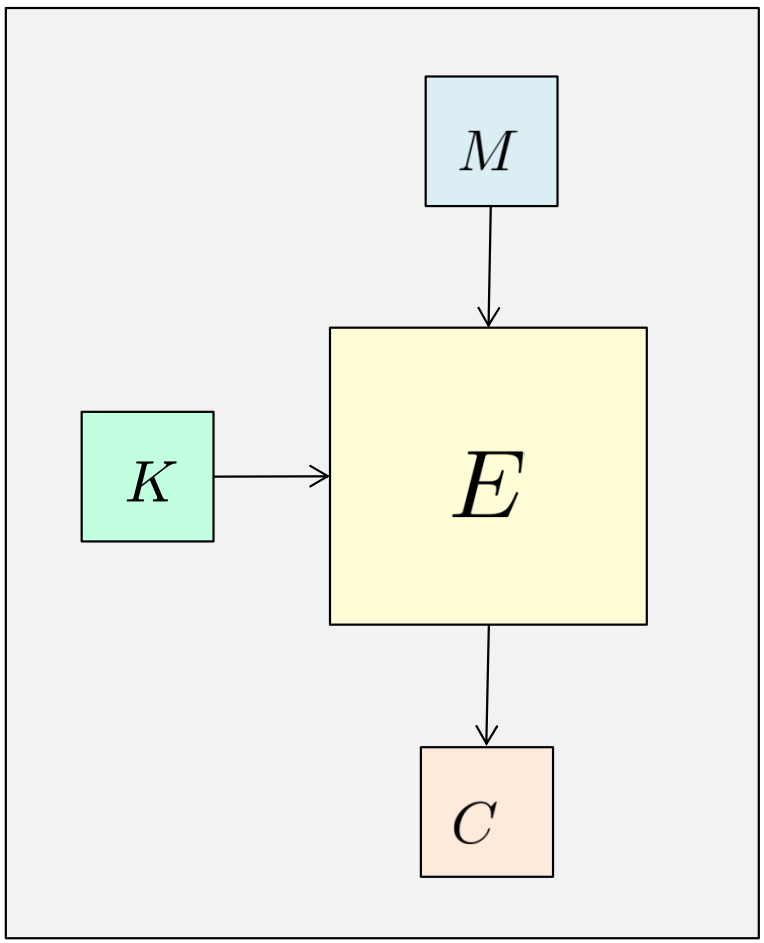
1. Blockcipher and Key Recovery

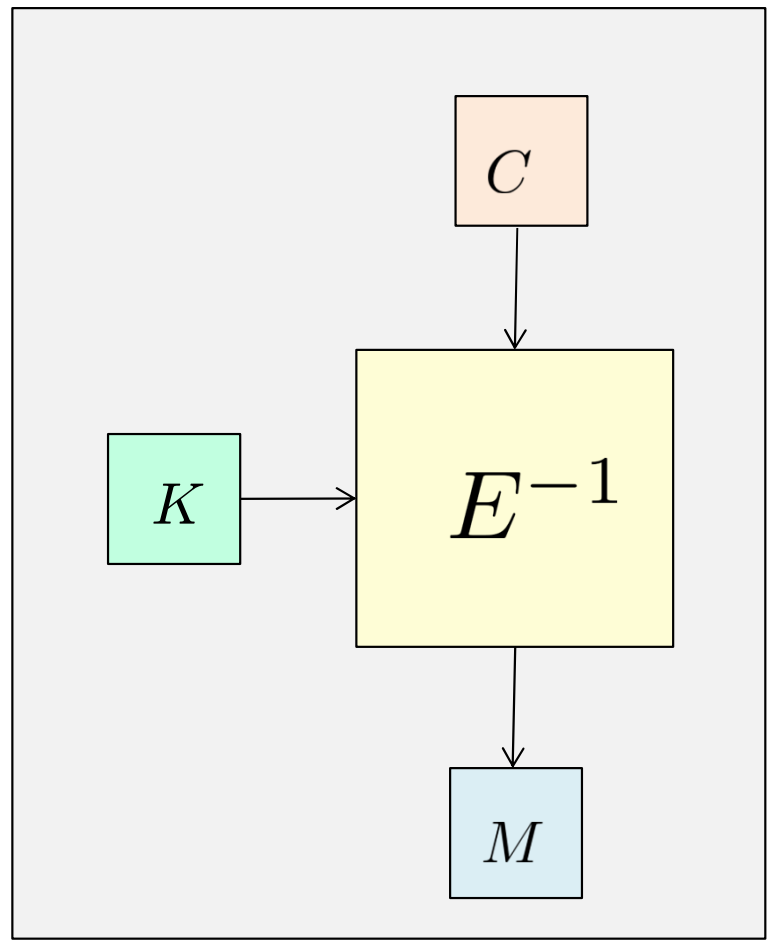
2. A Bird's-Eye View of Real Blockciphers

Blockcipher

$$E : \underbrace{\{0,1\}^k}_{\text{Key space}} \times \underbrace{\{0,1\}^n}_{\text{Domain}} \to \{0,1\}^n$$

efficiently invertible given the key





Blockcipher Usage

K

$C_1 \leftarrow E_K(M_1)$ ... $C_q \leftarrow E_K(M_q)$

K

Random key K is known to both parties, but not given to adversary A

Key-Recovery Attack: Scenario

K

$C_1 \leftarrow E_K(M_1)$ … $C_q \leftarrow E_K(M_q)$

K

Guess K

M_1 … M_q

Modeling Key-Recovery Attack

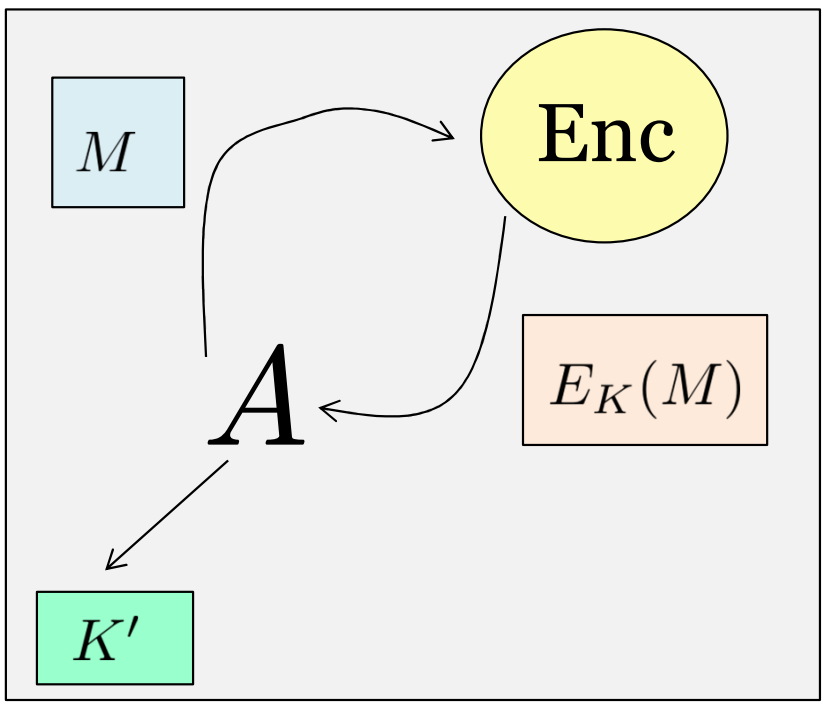
Multiple **chosen-plaintext** queries

Game KR_E

procedure Initialize()
$K \xleftarrow{\$} \mathcal{K}$

procedure Enc(M)
return $E_K(M)$

procedure Finalize(K')
return $(K' = K)$

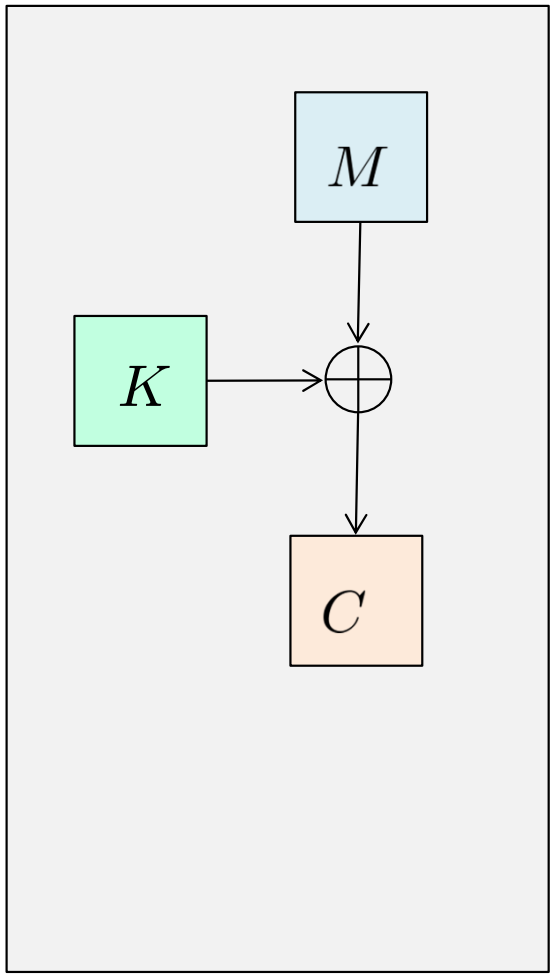


$$\mathbf{Adv}_E^{\mathrm{kr}}(A) = \Pr[\mathrm{KR}_E \Rightarrow 1]$$

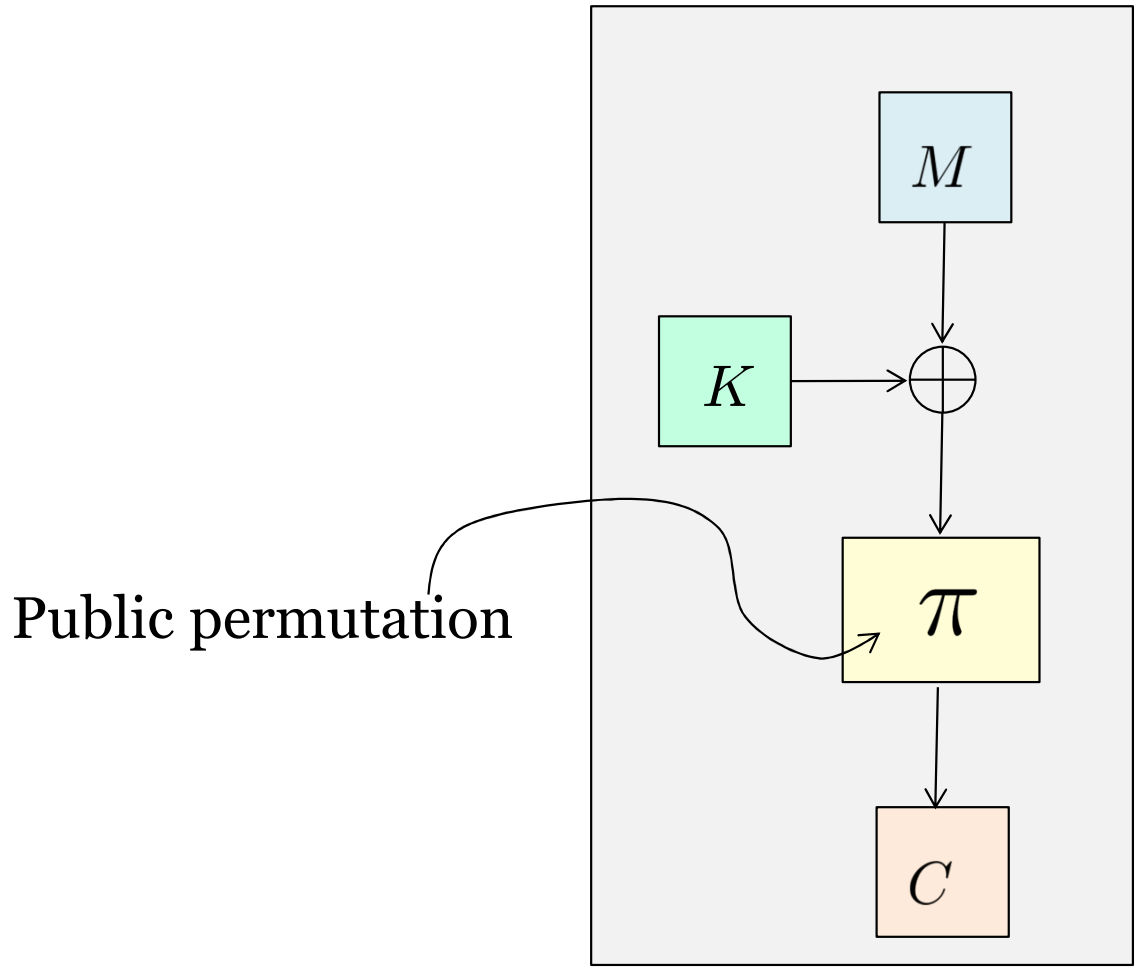
$\mathbf{Adv}_E^{\mathrm{kr}}(A) \approx 0$ means A is doing poorly

Practicing Key-Recovery Attack

$$E_K(M) = M \oplus K$$



$$E_K(M) = \pi(M \oplus K)$$



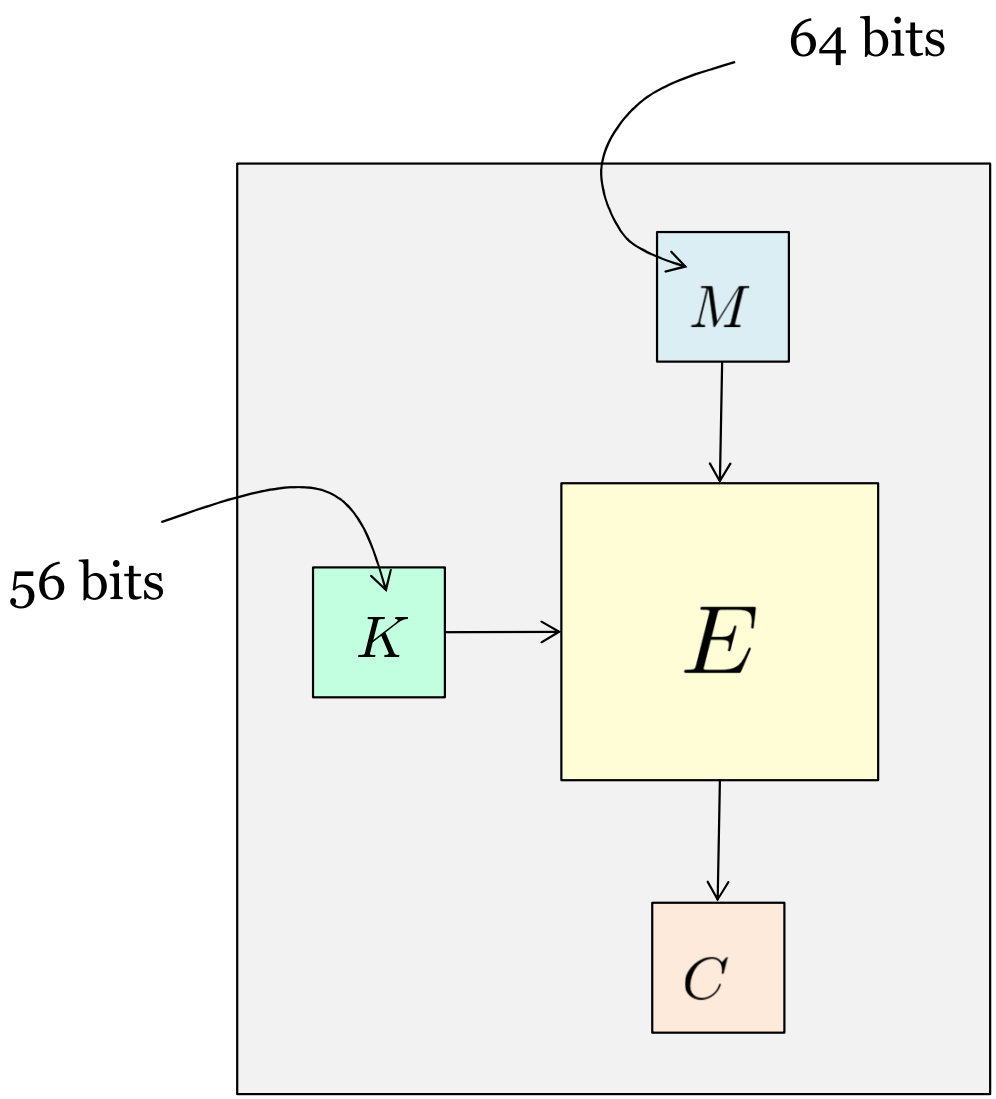
π, π^{-1} are public

Agenda

1. Blockcipher and Key Recovery

2. A Bird's-Eye View of Real Blockciphers

DES: Parameters and History



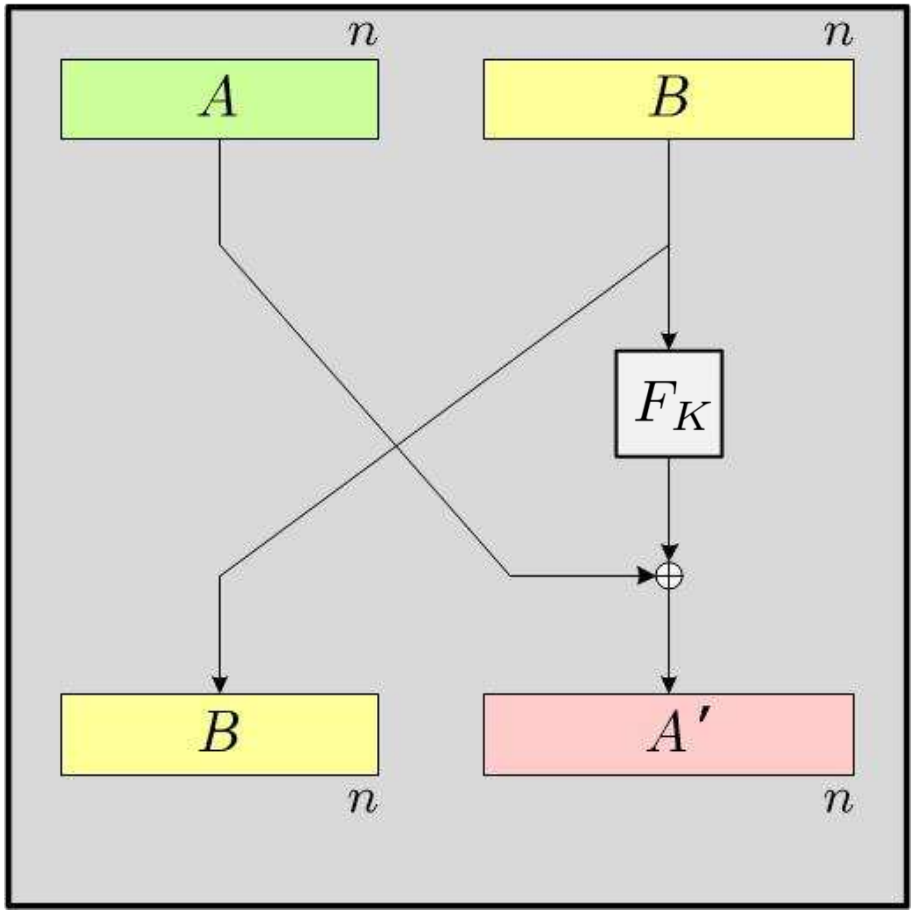
-Designed by IBM in 1974

-Used in ATM machines

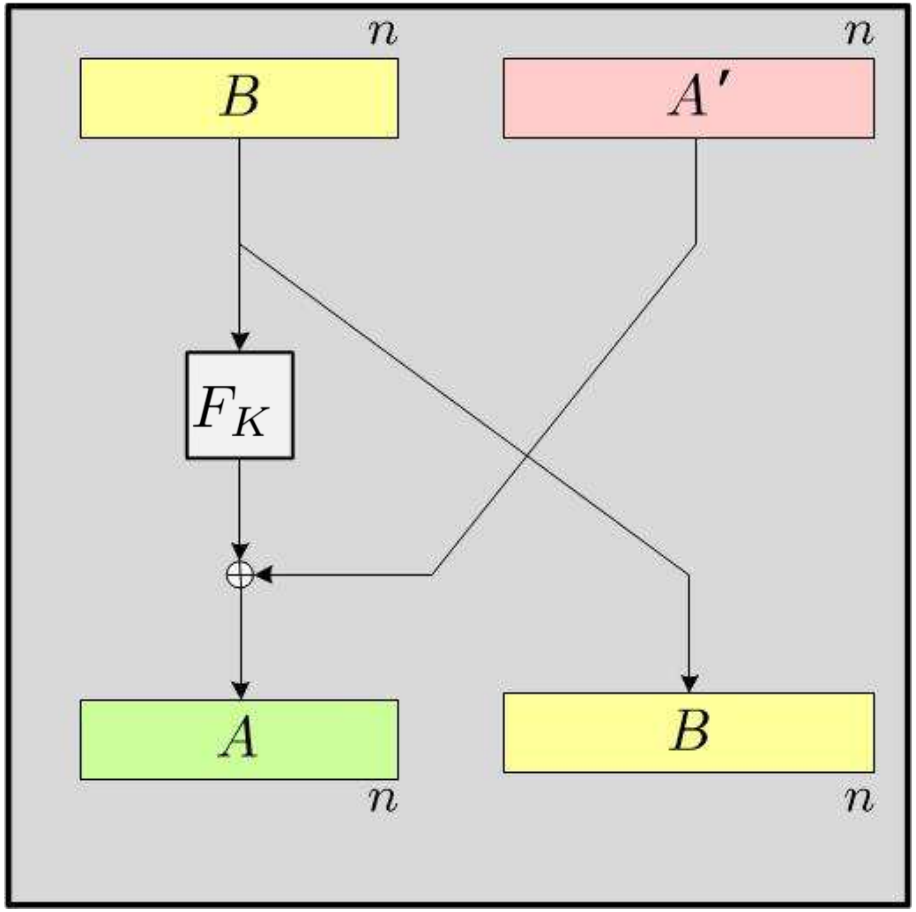
-Replaced in 2001

Design of DES: Feistel Network

(One-round) $\mathrm{Feistel}(K, \cdot)$

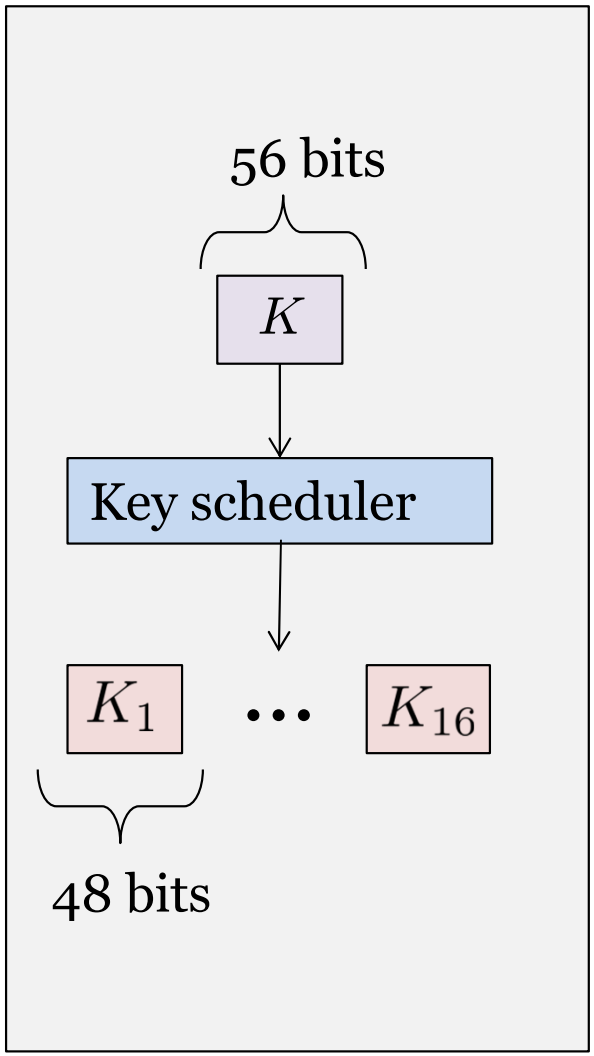


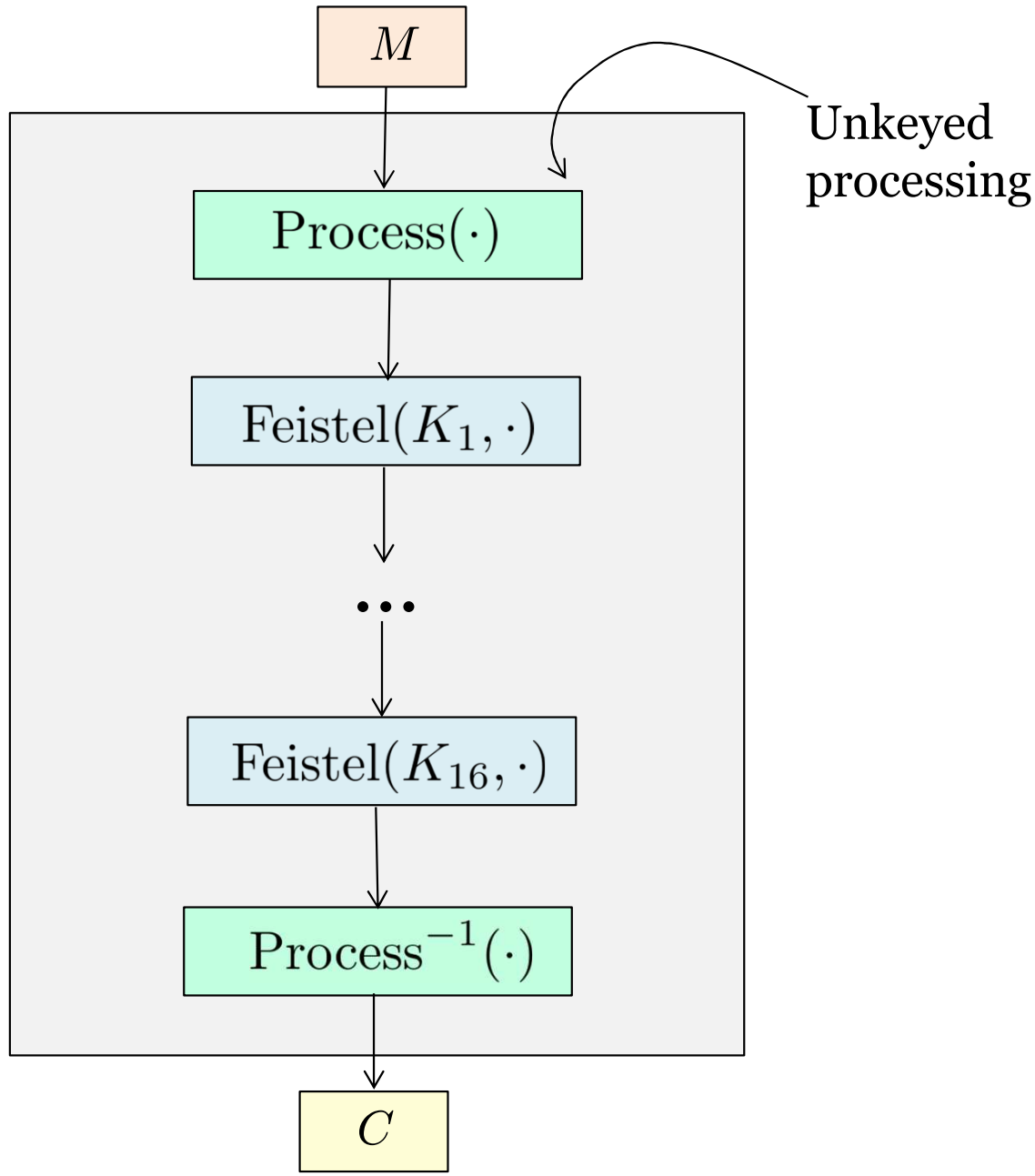
Inverse of Feistel



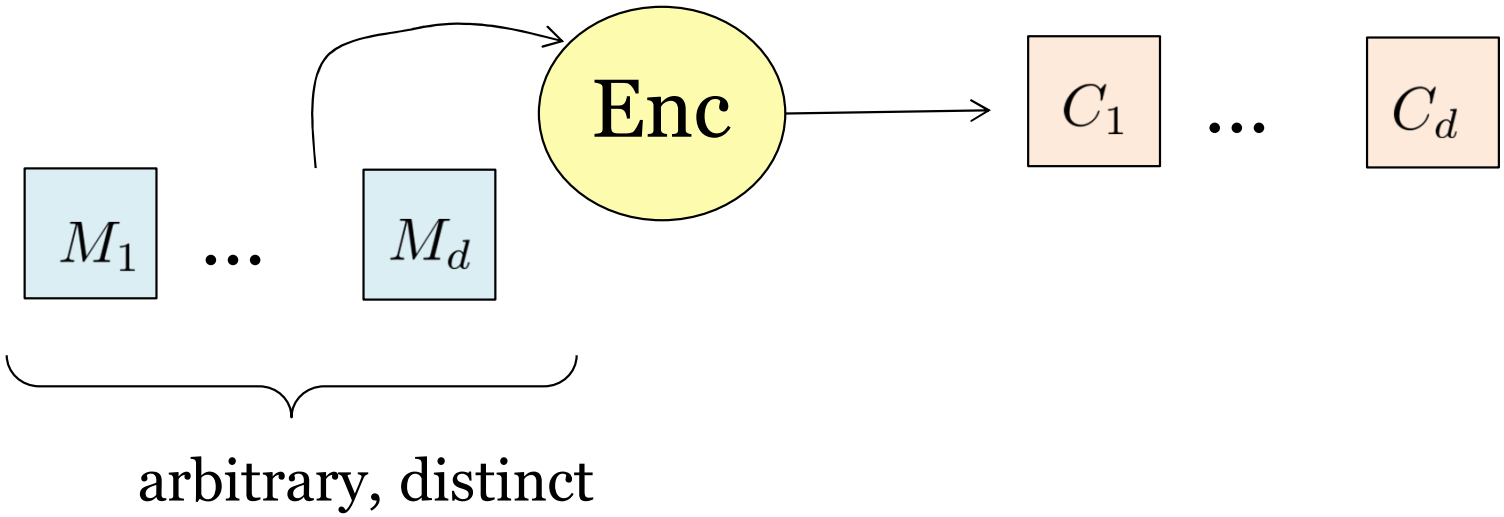
Question: How to invert?

Construction of DES





Exhaustive Key Search Attack



Fully parallelizable

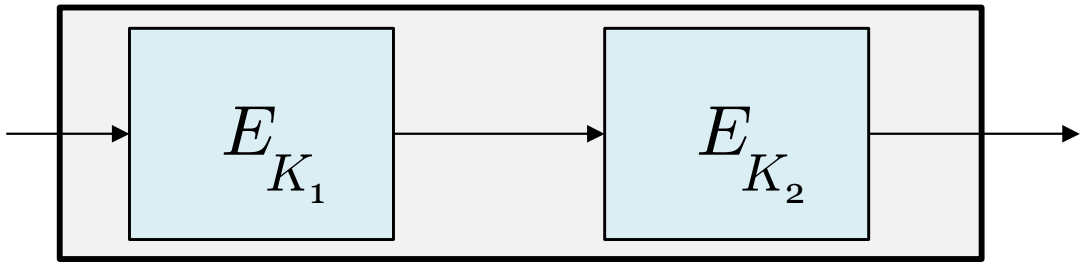
For $K \in \mathcal{K}$ do
  If $E_K(M_i) = C_i$ for every $i \in \{1, \ldots, d\}$ then return K

For $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$, if $d > k/n$ then $\mathbf{Adv}_E^{\mathrm{kr}}(A) \approx 1$

Exhaustive Key Search Attack on DES

Timeline	Source	Attack time
1997	DESCHALL	96 days
1998	Distributed.net	41 days
1998	EFF	56 hours
1998	Distributed.net + EFF	22 hours

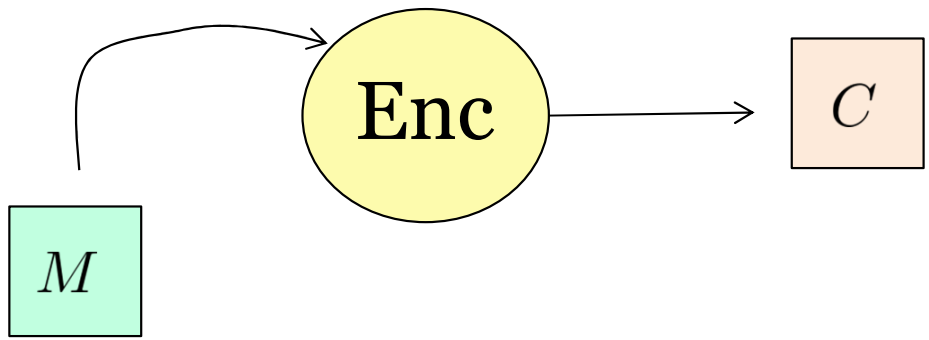
Incorrect Fix of DES: Double Encryption

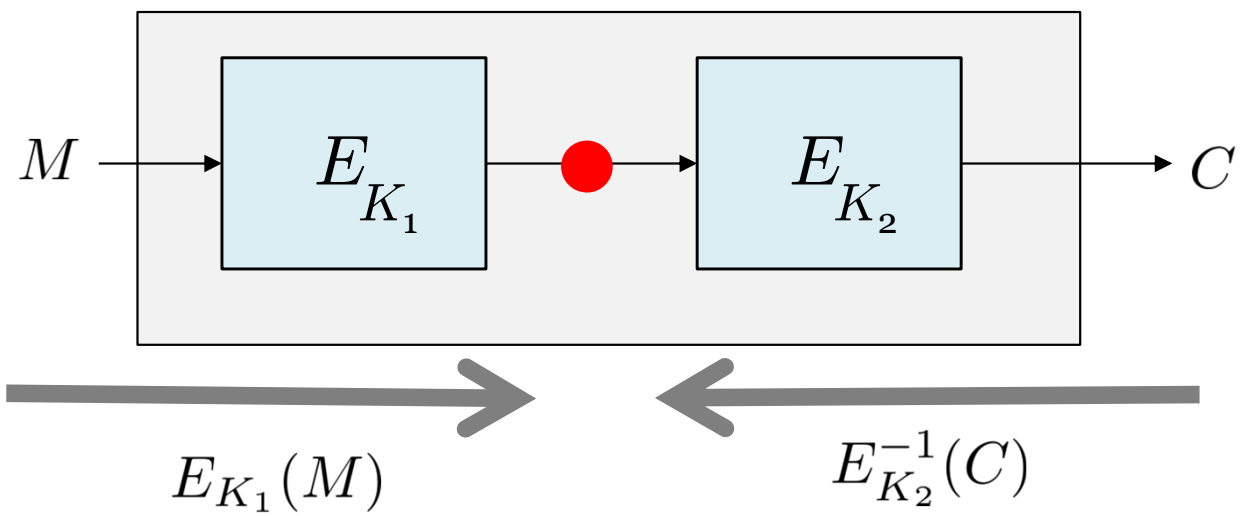


112-bit key → prohibitive for exhaustive key search

But there's a more clever attack!

Meet-in-the-Middle Attack: Intuition





Meet-in-the-Middle Attack

Let $L_1, \ldots, L_N$ be all possible DES keys

$N = 2^{56}$

	L_1	$E_{L_1}(M)$
$K_1 \longrightarrow$	L_i	$E_{L_i}(M)$
	L_N	$E_{L_N}(M)$

equal

$E^{-1}_{L_1}(C)$	L_1	
$E^{-1}_{L_j}(C)$	L_j	$\longleftarrow K_2$
$E^{-1}_{L_N}(C)$	L_N	

Find L_i, L_j such that $E_{L_i}(M) = E^{-1}_{L_j}(C)$

How to implement in
linear time?

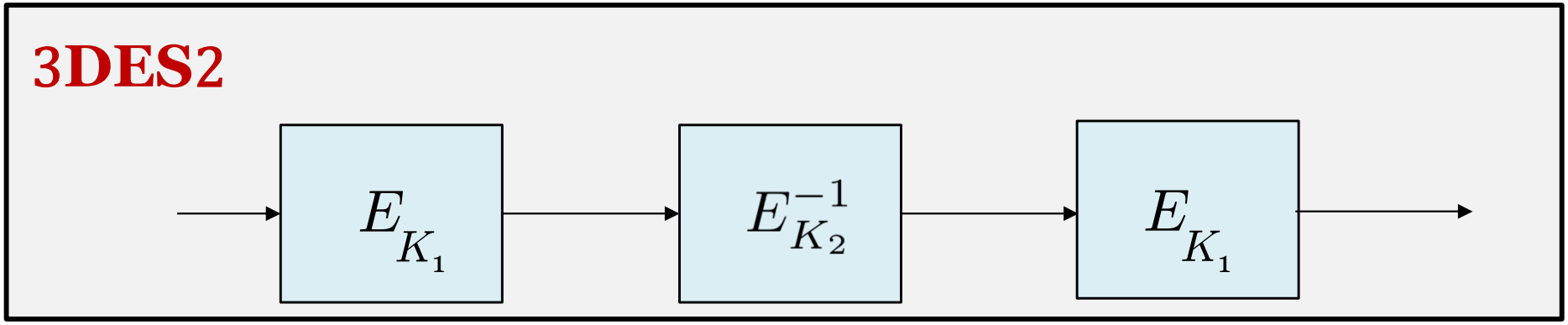
There are on average

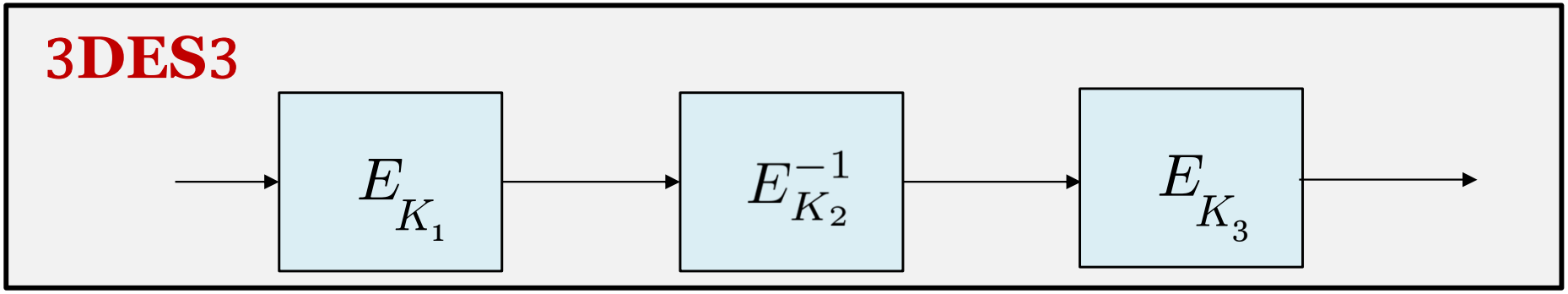
$$2^{48}$$

pairs of matching keys

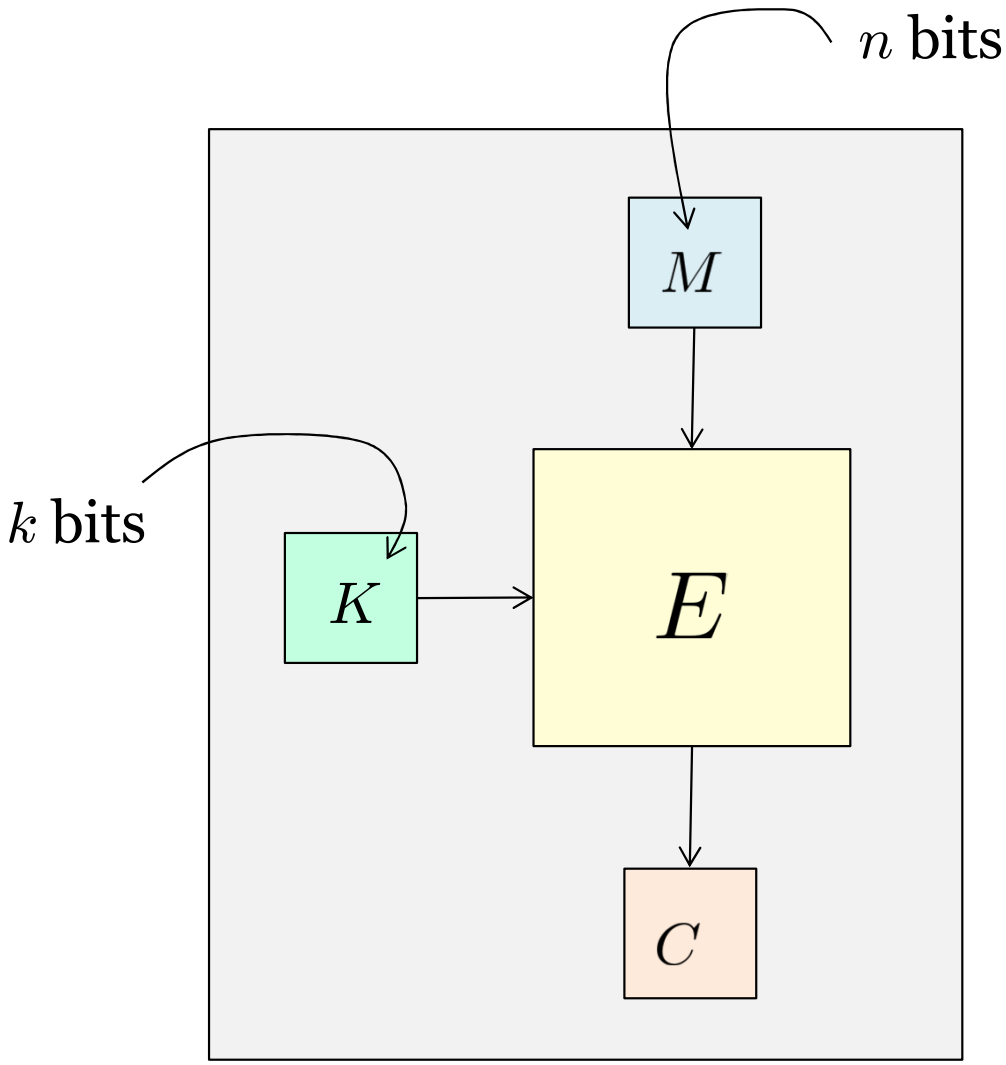
How to eliminate **false positives?**

The 3DES Constructions





Block Size Matters, Too

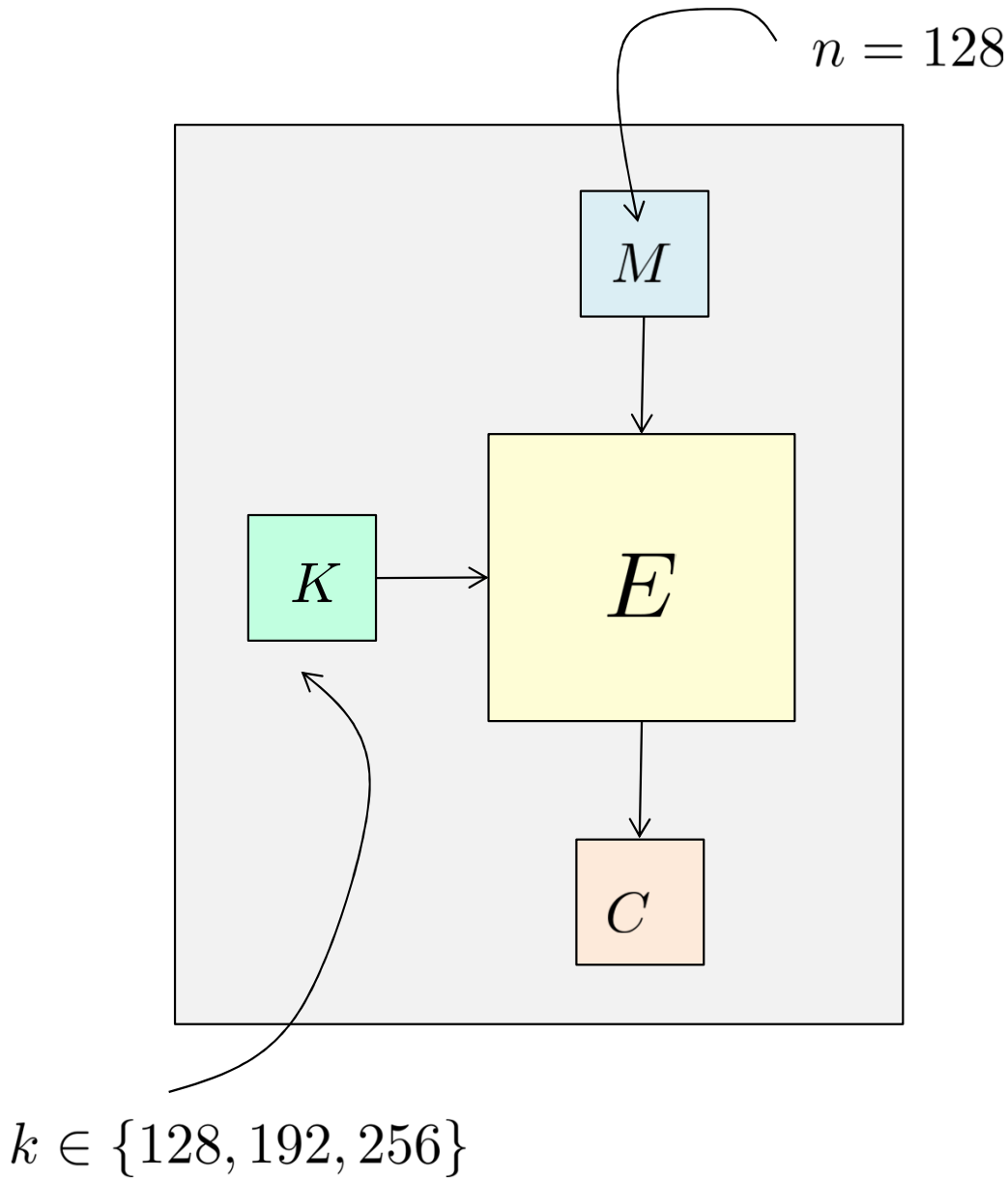


Distinguish outputs from random

Birthday attack: $O(2^{n/2})$ time

Practical for $\underbrace{\text{DES/2DES/3DES}}_{n = 64}$

State of the Art: AES



-NIST standard since 2001

-Best known key-recovery attack takes about 2^{126} time

Security Against Key Recovery Is Not Enough

A Bad Example: Consider the following $E : \{0,1\}^{128} \times \{0,1\}^{256} \to \{0,1\}^{256}$

$$E_K(M_1 M_2) = \mathbf{AES}_K(M_1) \| M_2$$

As secure against key recovery as AES

Send half of the message in the clear!

So What Is a Good Blockcipher?

Possible Properties	**Necessary**	**Sufficient**
Security against key recovery	Yes	No
Hard to find M given $C \leftarrow E_K(M)$	Yes	No
…		

Want: a single "master" property that is sufficient to ensure security of common usage of blockcipher.