

# Crafting an Attack on the Full-Round Craft

---

Eran Lambooi

2026 – Rennes

Inria, Paris

The Inria logo, featuring the word "Inria" in a stylized, red, cursive script.

“A cipher is broken if an attacker can encrypt an unseen plaintext into its corresponding ciphertext.”

“A cipher is broken if an attacker can encrypt an unseen plaintext into its corresponding ciphertext.”

Faster than the best generic attack, inside the security model the designers claimed.

Lightweight (Tweakable) Block Ciphers

Meet-in-the-Middle Attacks

Dissection Attacks

The 25-Round MitM Attack on CRAFT [LN26]

A Dissection Attack on Full CRAFT

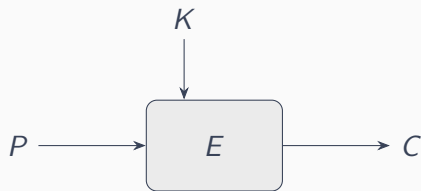
# Why Lightweight Ciphers?

- **NOT** Lightweight security!
- Constrained devices (RFID, sensors, implants): area, energy, latency
- Attacker has *physical access*: side channels, faults
- Countermeasures are cheaper when part of the design

# Why Lightweight Ciphers?

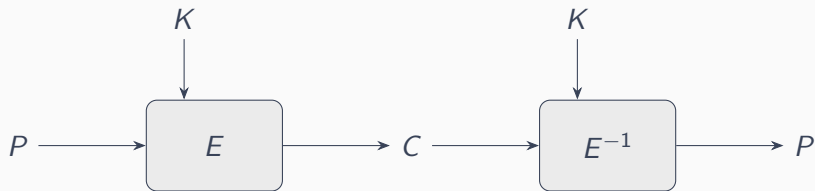
- **NOT** Lightweight security!
- Constrained devices (RFID, sensors, implants): area, energy, latency
- Attacker has *physical access*: side channels, faults
- Countermeasures are cheaper when part of the design
- **Design with built in countermeasures  $\Rightarrow$  CRAFT**

# Block Ciphers



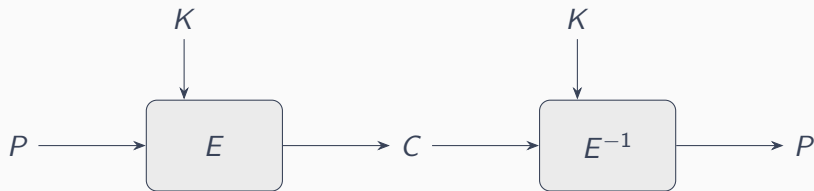
Secret **key**  $K$ : one permutation of  $\mathbb{F}_2^n$  per key

# Block Ciphers



Secret **key**  $K$ : one permutation of  $\mathbb{F}_2^n$  per key

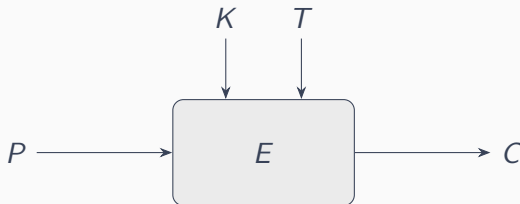
# Block Ciphers



Secret **key**  $K$ : one permutation of  $\mathbb{F}_2^n$  per key

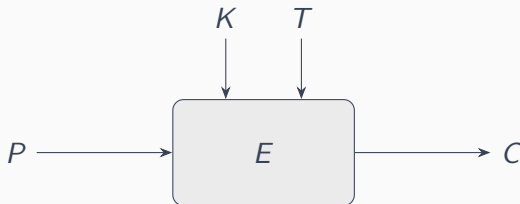
Want: the family looks like  $2^\kappa$  random permutations

# Tweakable Block Ciphers



Public **tweak**  $T$ : every  $T$  gives a new family of permutations of  $\mathbb{F}_2^n$

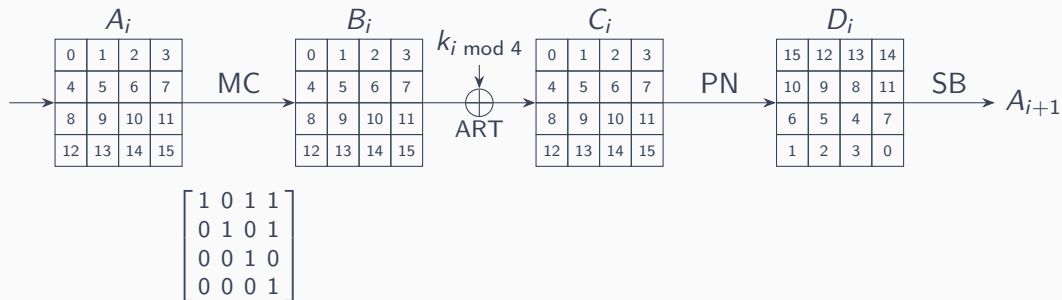
# Tweakable Block Ciphers



Public **tweak**  $T$ : every  $T$  gives a new family of permutations of  $\mathbb{F}_2^n$

Changing  $T$  must be cheap and the attacker controls it

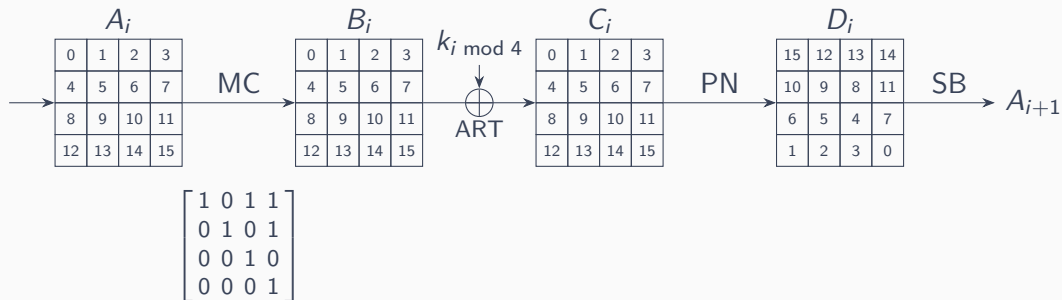
# Craft [BLMR19] at a Glance



- 64-bit state,  $4 \times 4$  nibbles
- 128-bit key  $K_0 \| K_1$ , 64-bit tweak  $T$
- 32 rounds, involutive components

$$\begin{aligned} k_0 &= K_0 \oplus T & k_2 &= K_0 \oplus Q(T) \\ k_1 &= K_1 \oplus T & k_3 &= K_1 \oplus Q(T) \end{aligned}$$

# Craft [BLMR19] at a Glance



- 64-bit state,  $4 \times 4$  nibbles
- 128-bit key  $K_0 \| K_1$ , 64-bit tweak  $T$
- 32 rounds, involutive components

$$k_0 = K_0 \oplus T \quad k_2 = K_0 \oplus Q(T)$$

$$k_1 = K_1 \oplus T \quad k_3 = K_1 \oplus Q(T)$$

For  $T = 0$ :  $K_0, K_1, K_0, K_1, \dots$

No mixing of key nibbles at all

# Attacks on Craft (Single Key)

| Rounds | Time         | Memory    | Data        | Attack  | Setting | Reference |
|--------|--------------|-----------|-------------|---------|---------|-----------|
| 21     | $2^{106.53}$ | $2^{100}$ | $2^{60.99}$ | ID      | CP      | [HSE23]   |
| 23     | $2^{111.46}$ | $2^{120}$ | $2^{60.99}$ | D       | CP      | [SYCHW24] |
|        | $2^{109}$    | $2^{36}$  | $2^{58}$    | TD-MitM | CC      | [MNN25]   |
| 24     | $2^{110}$    | $2^{34}$  | $2^{60}$    | TD-MitM | CC      | [MNN25]   |
| 25     | $2^{117.58}$ | $2^{48}$  | $2^{60}$    | TD-MitM | CC      | [MNN25]   |
|        | $2^{125}$    | $2^{100}$ | $2^1$       | MitM    | KP      | [LN26]    |
| 26     | $2^{118}$    | $2^{34}$  | $2^{64}$    | TD-MitM | CC      | [MNN25]   |

|       |                             |         |                             |
|-------|-----------------------------|---------|-----------------------------|
| D     | Differential                | MitM    | Meet-in-the-Middle          |
| ID    | Impossible Differential     | TD-MitM | Truncated-Differential MitM |
| CP/CC | Chosen Plaintext/Ciphertext | KP      | Known Plaintext             |

# Attacks on Craft (Single Key)

| Rounds | Time         | Memory    | Data        | Attack     | Setting | Reference |
|--------|--------------|-----------|-------------|------------|---------|-----------|
| 21     | $2^{106.53}$ | $2^{100}$ | $2^{60.99}$ | ID         | CP      | [HSE23]   |
| 23     | $2^{111.46}$ | $2^{120}$ | $2^{60.99}$ | D          | CP      | [SYCHW24] |
|        | $2^{109}$    | $2^{36}$  | $2^{58}$    | TD-MitM    | CC      | [MNN25]   |
| 24     | $2^{110}$    | $2^{34}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
| 25     | $2^{117.58}$ | $2^{48}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
|        | $2^{125}$    | $2^{100}$ | $2^1$       | MitM       | KP      | [LN26]    |
| 26     | $2^{118}$    | $2^{34}$  | $2^{64}$    | TD-MitM    | CC      | [MNN25]   |
| 32     | $2^{121}$    | $2^{120}$ | 3           | Dissection | KP      | this work |
| 36     | $2^{126}$    | $2^{124}$ | 3           | Dissection | KP      | this work |

D

Differential

MitM

Meet-in-the-Middle

ID

Impossible Differential

TD-MitM

Truncated-Differential MitM

CP/CC

Chosen Plaintext/Ciphertext

KP

Known Plaintext

Lightweight (Tweakable) Block Ciphers

Meet-in-the-Middle Attacks

Dissection Attacks

The 25-Round MitM Attack on CRAFT [LN26]

A Dissection Attack on Full CRAFT

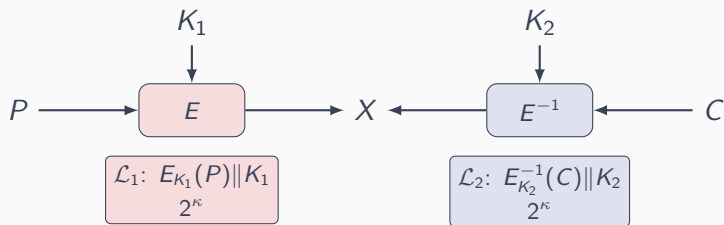
## Meet-in-the-Middle: The Classic Example [DH77]

**Double encryption:**  $n$ -bit block, two independent  $\kappa$ -bit keys    Brute force:  $2^{2\kappa}$



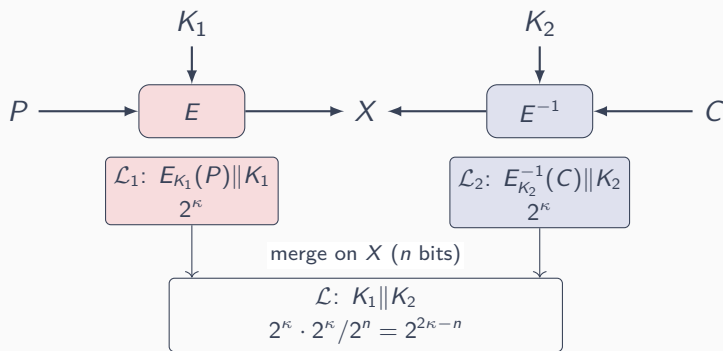
## Meet-in-the-Middle: The Classic Example [DH77]

**Double encryption:**  $n$ -bit block, two independent  $\kappa$ -bit keys    Brute force:  $2^{2\kappa}$



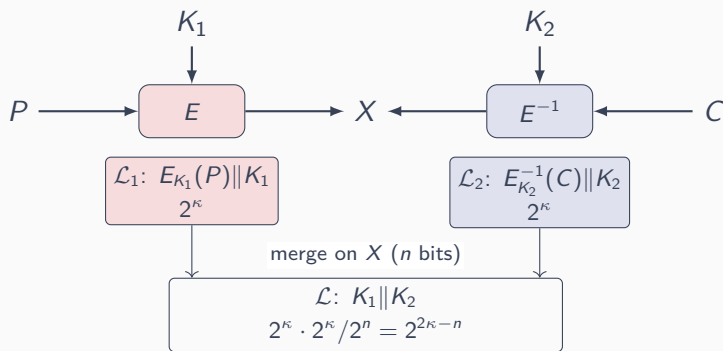
# Meet-in-the-Middle: The Classic Example [DH77]

**Double encryption:**  $n$ -bit block, two independent  $\kappa$ -bit keys    Brute force:  $2^{2\kappa}$



# Meet-in-the-Middle: The Classic Example [DH77]

**Double encryption:**  $n$ -bit block, two independent  $\kappa$ -bit keys    Brute force:  $2^{2\kappa}$

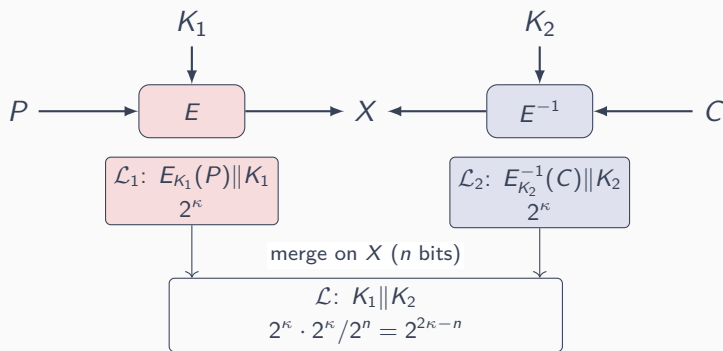


Time  $2^\kappa + 2^\kappa + 2^{2\kappa - n}$

Memory  $2^\kappa$

# Meet-in-the-Middle: The Classic Example [DH77]

**Double encryption:**  $n$ -bit block, two independent  $\kappa$ -bit keys    Brute force:  $2^{2\kappa}$

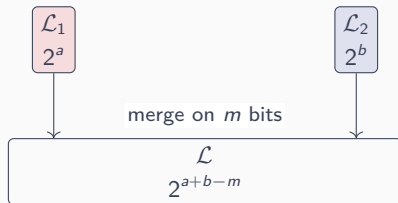


Time  $2^\kappa + 2^\kappa + 2^{2\kappa - n}$

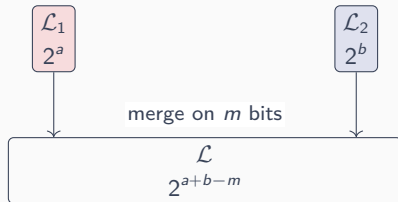
Memory  $2^\kappa$

Independent guesses  $\Rightarrow$  cost is **additive**, not multiplicative

# Merging Lists

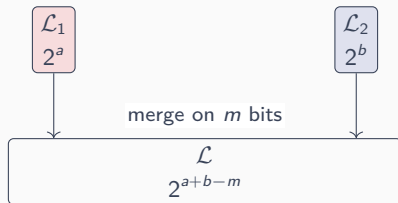


# Merging Lists



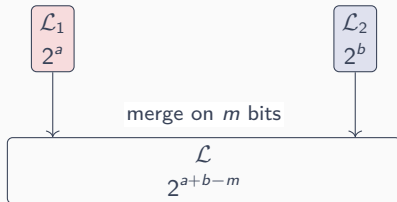
- Sort (or hash) on the  $m$  matched bits: time  $2^a + 2^b + 2^{a+b-m}$

# Merging Lists



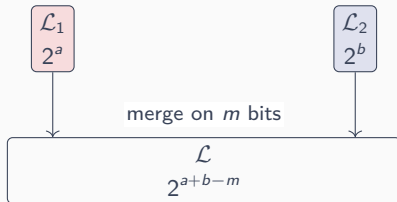
- Sort (or hash) on the  $m$  matched bits: time  $2^a + 2^b + 2^{a+b-m}$
- **Not enough filter?** Use  $\ell$  pairs, merge on  $X_1 \parallel \dots \parallel X_\ell$ :  $2^{2\kappa - \ell n}$

# Merging Lists



- Sort (or hash) on the  $m$  matched bits: time  $2^a + 2^b + 2^{a+b-m}$
- **Not enough filter?** Use  $\ell$  pairs, merge on  $X_1 \parallel \dots \parallel X_\ell$ :  $2^{2\kappa - \ell n}$
- **Or:** use few pairs, test the remaining candidates by trial encryption

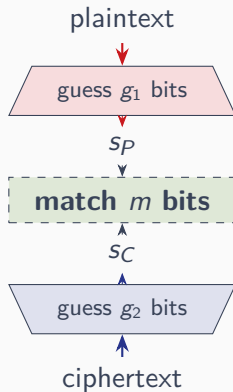
# Merging Lists



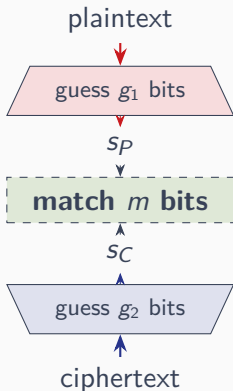
- Sort (or hash) on the  $m$  matched bits: time  $2^a + 2^b + 2^{a+b-m}$
- **Not enough filter?** Use  $\ell$  pairs, merge on  $X_1 \parallel \dots \parallel X_\ell$ :  $2^{2\kappa - \ell n}$
- **Or:** use few pairs, test the remaining candidates by trial encryption

Every attack in this talk is a tree of such merges

# MitM Applied to Block Ciphers

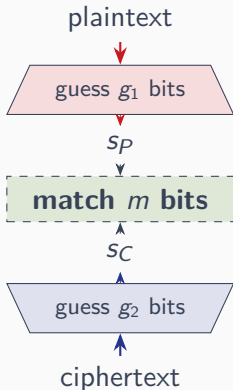


# MitM Applied to Block Ciphers



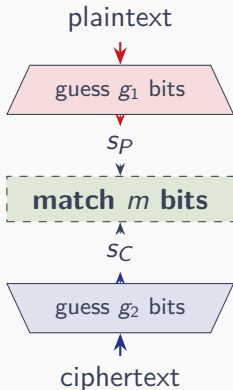
- Partial states: compute only part of the state [CE85]

# MitM Applied to Block Ciphers



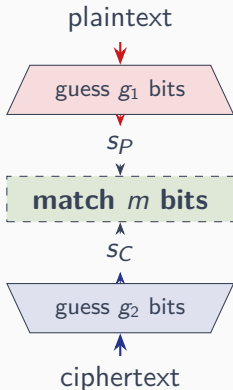
- Partial states: compute only part of the state [CE85]
- $\mathcal{L}_1: s_P \parallel g_1$ ,  $\mathcal{L}_2: s_C \parallel g_2$ , merge on  $m$  bits: time  $2^{g_1} + 2^{g_2} + 2^{g_1+g_2-m}$

# MitM Applied to Block Ciphers



- Partial states: compute only part of the state [CE85]
- $\mathcal{L}_1: s_P \parallel g_1$ ,  $\mathcal{L}_2: s_C \parallel g_2$ , merge on  $m$  bits: time  $2^{g_1} + 2^{g_2} + 2^{g_1+g_2-m}$
- Guesses need not be key bits: **guess state** values instead [DSP07]

# MitM Applied to Block Ciphers



- Partial states: compute only part of the state [CE85]
- $\mathcal{L}_1: s_P \parallel g_1$ ,  $\mathcal{L}_2: s_C \parallel g_2$ , merge on  $m$  bits: time  $2^{g_1} + 2^{g_2} + 2^{g_1+g_2-m}$
- Guesses need not be key bits: **guess state** values instead [DSP07]
- Minimal data: often 1 or 2 known plaintexts

Lightweight (Tweakable) Block Ciphers

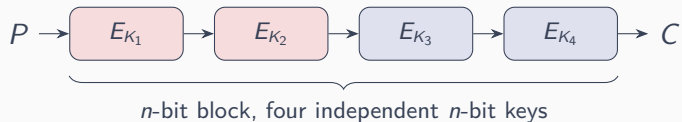
Meet-in-the-Middle Attacks

Dissection Attacks

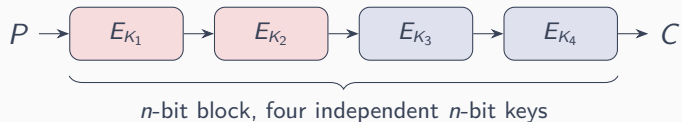
The 25-Round MitM Attack on CRAFT [LN26]

A Dissection Attack on Full CRAFT

## Multiple Encryption [MH81]

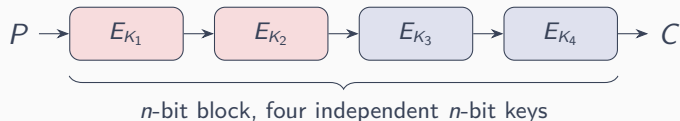


## Multiple Encryption [MH81]



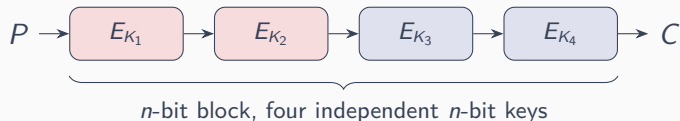
- Need  $\ell = 4$  plaintext-ciphertext pairs to determine  $4n$  key bits

## Multiple Encryption [MH81]



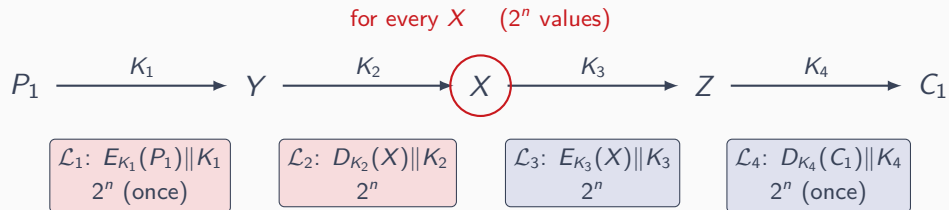
- Need  $\ell = 4$  plaintext-ciphertext pairs to determine  $4n$  key bits
- Plain MitM (guess  $K_1, K_2$  vs.  $K_3, K_4$ ): time  $2^{2n}$ , memory  $2^{2n}$

## Multiple Encryption [MH81]

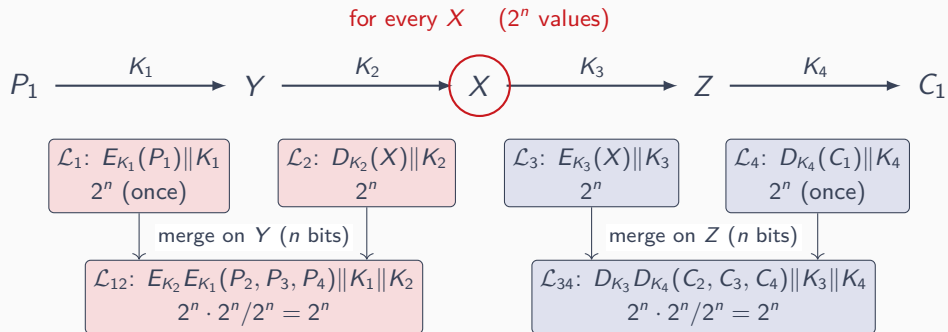


- Need  $\ell = 4$  plaintext-ciphertext pairs to determine  $4n$  key bits
- Plain MitM (guess  $K_1, K_2$  vs.  $K_3, K_4$ ): time  $2^{2n}$ , memory  $2^{2n}$
- Can we do better? **Yes:** *dissect* the problem [DDKS19]

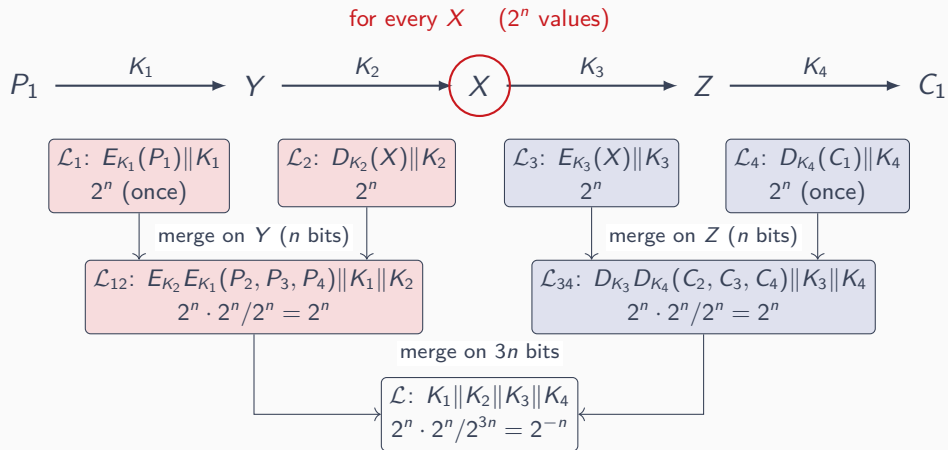
# Guess in the Middle



# Guess in the Middle



# Guess in the Middle



Over all  $X$ :  $\approx 1$  key

Time  $2^n \times 2^n = 2^{2n}$

Memory  $2^n$

## Dissection: The Take-Away

- Guess an *internal state*, split the problem in two *independent* halves

## Dissection: The Take-Away

- Guess an *internal state*, split the problem in two *independent* halves
- Solve each half with a MitM, merge the resulting lists

## Dissection: The Take-Away

- Guess an *internal state*, split the problem in two *independent* halves
- Solve each half with a MitM, merge the resulting lists
- Generalizes to  $r$ -fold encryption, trade-offs between time and memory

## Dissection: The Take-Away

- Guess an *internal state*, split the problem in two *independent* halves
- Solve each half with a MitM, merge the resulting lists
- Generalizes to  $r$ -fold encryption, trade-offs between time and memory
- Typically: dissection **saves memory**, not time

## Dissection: The Take-Away

- Guess an *internal state*, split the problem in two *independent* halves
- Solve each half with a MitM, merge the resulting lists
- Generalizes to  $r$ -fold encryption, trade-offs between time and memory
- Typically: dissection **saves memory**, not time

In this talk: dissection that **saves time**

Lightweight (Tweakable) Block Ciphers

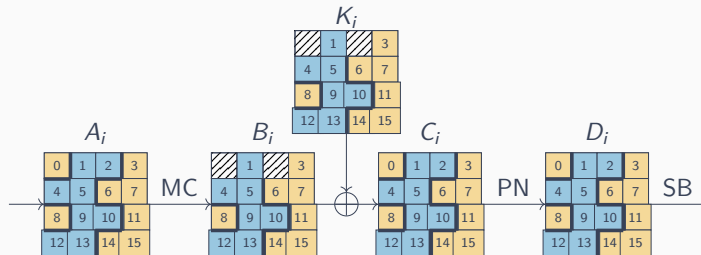
Meet-in-the-Middle Attacks

Dissection Attacks

The 25-Round MitM Attack on CRAFT [LN26]

A Dissection Attack on Full CRAFT

# Decomposing Craft



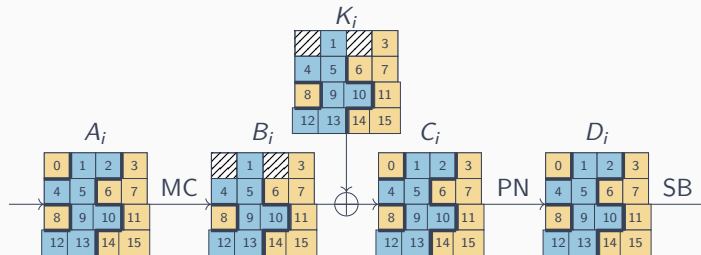
$$L = \{1, 2, 4, 5, 9, 10, 12, 13\}$$

$$R = \{0, 3, 6, 7, 8, 11, 14, 15\}$$

$$MC = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Upper triangular, aligned with PN

# Decomposing Craft



$$L = \{1, 2, 4, 5, 9, 10, 12, 13\}$$

$$R = \{0, 3, 6, 7, 8, 11, 14, 15\}$$

$$MC = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Upper triangular, aligned with PN

Coupled through two nibbles per round:

$$B_i[0] = A_i[0] + A_i[8] + A_i[12]$$

$$B_i[2] = A_i[2] + A_i[10] + A_i[14]$$

## Guess After the Key Addition, Match Two Rounds Apart

- In  $R$  guess  $C_i[0] = B_i[0] + k_i[0]$ , in  $L$  guess  $C_i[2] = B_i[2] + k_i[2]$
- Guessing after the key addition: each half needs only 56 key bits
- Key schedule  $k_i = k_{i+2}$ : the key cancels

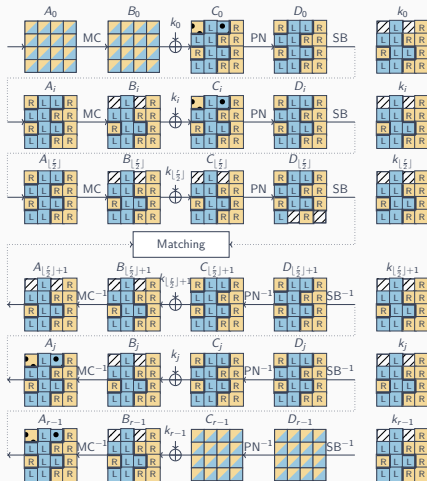
$$S_i = C_i[0] + C_{i+2}[0] + \underbrace{A_i[0] + A_{i+2}[0] + A_i[8] + A_{i+2}[8]}_{\text{known in } R}$$

$$S'_i = C_i[2] + C_{i+2}[2] + \underbrace{A_i[2] + A_{i+2}[2] + A_i[10] + A_{i+2}[10]}_{\text{known in } L}$$

- Coupling nibbles  $S_i = A_i[12] + A_{i+2}[12]$ ,  $S'_i = A_i[14] + A_{i+2}[14]$
- Match on  $S_i$  and  $S'_i$ :  $8(r-2)$  bits of filter

For  $T \neq 0$ :  $k_i + k_{i+2} = T + Q(T)$  is known.

# A Meet-in-the-Middle per Half

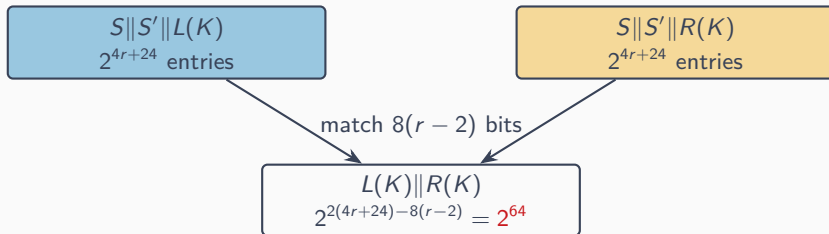


For one known pair  $(P, C)$  and  $L$ :

- Guess  $L(K)$ : 56 bits
- Guess one nibble per round, except next to the match:  $4(r - 2)$  bits
- Match in the middle (before/after SB): 24 bits
- List of  $(S \parallel S', L(K))$ :

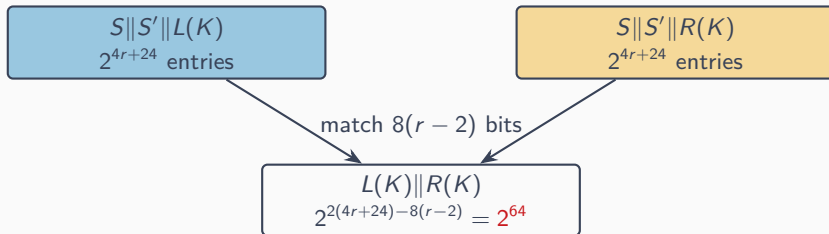
$$2^{56+4(r-2)-24} = 2^{4r+24}$$

## Merging the Halves



- Add the 16 unguessed key bits:  $2^{80}$  keys, test with 2 more pairs
- Time:  $2 \cdot (2^{56+4\lfloor r/2 \rfloor} + 2^{4r+24})$   $r = 25$ :  $2^{125}$
- Lists grow by  $2^4$  per round

## Merging the Halves



- Add the 16 unguessed key bits:  $2^{80}$  keys, test with 2 more pairs
- Time:  $2 \cdot \left( 2^{56+4\lfloor r/2 \rfloor} + 2^{4r+24} \right)$   $r = 25$ :  $2^{125}$
- Lists grow by  $2^4$  per round

**Idea:** merge more often  $\Rightarrow$  Dissection!

Lightweight (Tweakable) Block Ciphers

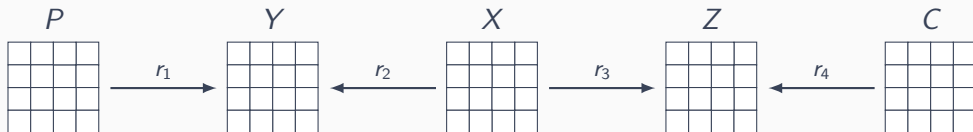
Meet-in-the-Middle Attacks

Dissection Attacks

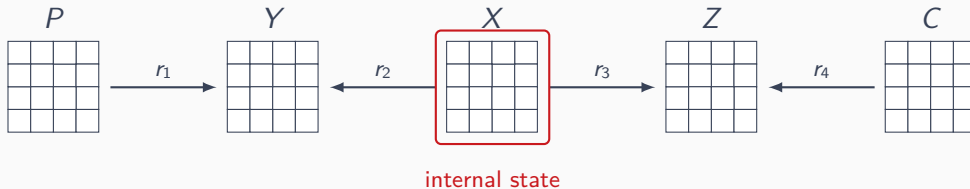
The 25-Round MitM Attack on CRAFT [LN26]

A Dissection Attack on Full CRAFT

# Dissecting Craft

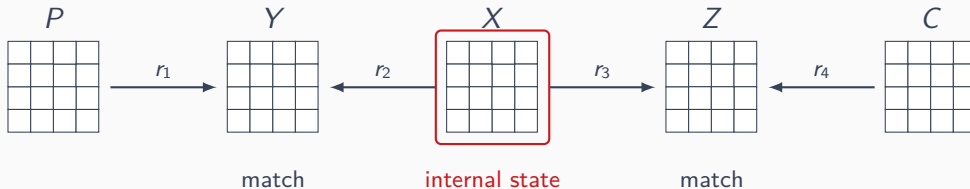


# Dissecting Craft



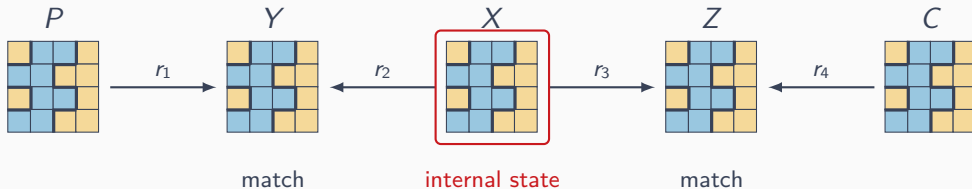
- $X$ : state after  $r_1 + r_2$  rounds

# Dissecting Craft



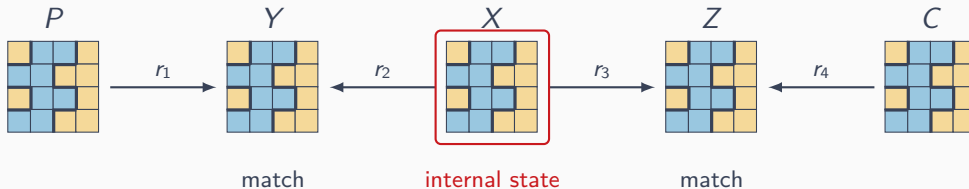
- $X$ : state after  $r_1 + r_2$  rounds
- Two MitM attacks of *half the length*:  $P \rightarrow X$  and  $C \rightarrow X$

# Dissecting Craft



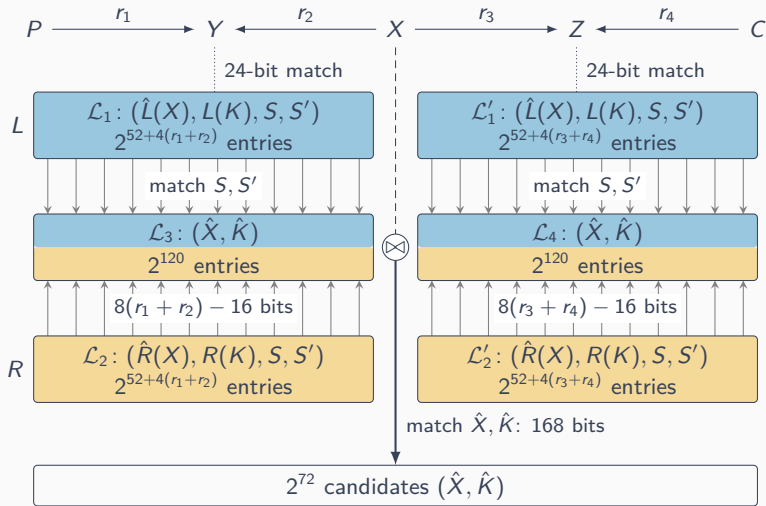
- $X$ : state after  $r_1 + r_2$  rounds
- Two MitM attacks of *half the length*:  $P \rightarrow X$  and  $C \rightarrow X$
- Each MitM itself split into  $L$  and  $R$ , merged using  $S, S'$

# Dissecting Craft



- $X$ : state after  $r_1 + r_2$  rounds
- Two MitM attacks of *half the length*:  $P \rightarrow X$  and  $C \rightarrow X$
- Each MitM itself split into  $L$  and  $R$ , merged using  $S, S'$
- Merge the two sides on  $(X, K)$

# Overview



## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$
- State guesses:  $4(r_1 - 1) + 4(r_2 - 1)$  bits

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$
- State guesses:  $4(r_1 - 1) + 4(r_2 - 1)$  bits
- Filter in  $Y$ : 24 bits

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$
- State guesses:  $4(r_1 - 1) + 4(r_2 - 1)$  bits
- Filter in  $Y$ : 24 bits

$$\mathcal{L}_1: \left( \hat{L}(X), L(K), S, S' \right)$$

$$|\mathcal{L}_1| = 2^{28+56+4(r_1+r_2)-8-24} = 2^{52+4(r_1+r_2)}$$

$$\text{Time: } 2^{80+4 \max\{r_1, r_2\}} + 2^{52+4(r_1+r_2)}$$

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$
- State guesses:  $4(r_1 - 1) + 4(r_2 - 1)$  bits
- Filter in  $Y$ : 24 bits

$$\mathcal{L}_1: (\hat{L}(X), L(K), S, S')$$

$$|\mathcal{L}_1| = 2^{28+56+4(r_1+r_2)-8-24} = 2^{52+4(r_1+r_2)}$$

$$\text{Time: } 2^{80+4 \max\{r_1, r_2\}} + 2^{52+4(r_1+r_2)}$$

Same for  $R$ :  $\mathcal{L}_2$

## Step 1: Meet-in-the-Middle per Half

For every  $\hat{L}(X) \in \{0, 1\}^{28}$  and  $L(K) \in \{0, 1\}^{56}$ :

- MitM over  $r_1$  and  $r_2$  rounds, meeting in  $Y$
- State guesses:  $4(r_1 - 1) + 4(r_2 - 1)$  bits
- Filter in  $Y$ : 24 bits

$$\mathcal{L}_1: (\hat{L}(X), L(K), S, S')$$

$$|\mathcal{L}_1| = 2^{28+56+4(r_1+r_2)-8-24} = 2^{52+4(r_1+r_2)}$$

$$\text{Time: } 2^{80+4 \max\{r_1, r_2\}} + 2^{52+4(r_1+r_2)}$$

Same for  $R$ :  $\mathcal{L}_2$

$\hat{L}(X)$ : 28 bits, one nibble of  $L(X)$  is itself a state guess and gives no filter

## Step 2: Merging $L$ and $R$

Match  $\mathcal{L}_1$  and  $\mathcal{L}_2$  on  $S\|S'$ :  $8(r_1 + r_2) - 16$  bits

## Step 2: Merging $L$ and $R$

Match  $\mathcal{L}_1$  and  $\mathcal{L}_2$  on  $S\|S'$ :  $8(r_1 + r_2) - 16$  bits

- Entries per bin, per list:

$$2^{52+4(r_1+r_2)-(8(r_1+r_2)-16)} = 2^{68-4(r_1+r_2)}$$

## Step 2: Merging $L$ and $R$

Match  $\mathcal{L}_1$  and  $\mathcal{L}_2$  on  $S \parallel S'$ :  $8(r_1 + r_2) - 16$  bits

- Entries per bin, per list:

$$2^{52+4(r_1+r_2)-(8(r_1+r_2)-16)} = 2^{68-4(r_1+r_2)}$$

- Total number of merged entries:

$$\underbrace{2^{8(r_1+r_2)-16}}_{\text{bins}} \cdot \left(2^{68-4(r_1+r_2)}\right)^2 = 2^{120}$$

## Step 2: Merging $L$ and $R$

Match  $\mathcal{L}_1$  and  $\mathcal{L}_2$  on  $S \parallel S'$ :  $8(r_1 + r_2) - 16$  bits

- Entries per bin, per list:

$$2^{52+4(r_1+r_2)-(8(r_1+r_2)-16)} = 2^{68-4(r_1+r_2)}$$

- Total number of merged entries:

$$\underbrace{2^{8(r_1+r_2)-16}}_{\text{bins}} \cdot \left(2^{68-4(r_1+r_2)}\right)^2 = 2^{120}$$

$\mathcal{L}_3$ :  $2^{120}$  candidates  $(\hat{X}, \hat{K})$ , **independent of**  $r_1 + r_2$

## Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries

## Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries
- Both lists contain  $\hat{X}$  (56 bits) and  $\hat{K}$  (112 bits)

## Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries
- Both lists contain  $\hat{X}$  (56 bits) and  $\hat{K}$  (112 bits)
- Intersect on 168 bits:

$$2^{120+120-168} = 2^{72} \text{ candidates } (\hat{X}, \hat{K})$$

## Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries
- Both lists contain  $\hat{X}$  (56 bits) and  $\hat{K}$  (112 bits)
- Intersect on 168 bits:

$$2^{120+120-168} = 2^{72} \text{ candidates } (\hat{X}, \hat{K})$$

- Add the 16 unguessed key bits:  $2^{88}$  keys

## Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries
- Both lists contain  $\hat{X}$  (56 bits) and  $\hat{K}$  (112 bits)
- Intersect on 168 bits:

$$2^{120+120-168} = 2^{72} \text{ candidates } (\hat{X}, \hat{K})$$

- Add the 16 unguessed key bits:  $2^{88}$  keys
- Test with two more known pairs  $\Rightarrow$  3 known plaintexts

### Step 3: Merging the Two Sides

- Repeat Steps 1 and 2 on  $(X, C)$  over  $r_3 + r_4$  rounds:  $\mathcal{L}_4$ ,  $2^{120}$  entries
- Both lists contain  $\hat{X}$  (56 bits) and  $\hat{K}$  (112 bits)
- Intersect on 168 bits:

$$2^{120+120-168} = 2^{72} \text{ candidates } (\hat{X}, \hat{K})$$

- Add the 16 unguessed key bits:  $2^{88}$  keys
- Test with two more known pairs  $\Rightarrow$  3 known plaintexts

Re-running the attack on a second pair with only  $2^{72}$  candidates suggests that 2 pairs suffice.

Time:

$$2 \cdot \left( \underbrace{2^{80+4 \max\{r_1, r_2\}} + 2^{80+4 \max\{r_3, r_4\}}}_{\text{MitM per half}} + \underbrace{2^{52+4(r_1+r_2)} + 2^{52+4(r_3+r_4)}}_{\mathcal{L}_1, \mathcal{L}_2} \right) + \underbrace{2 \cdot 2^{120}}_{\mathcal{L}_3, \mathcal{L}_4}$$

**Time:**

$$2 \cdot \left( \underbrace{2^{80+4 \max\{r_1, r_2\}} + 2^{80+4 \max\{r_3, r_4\}}}_{\text{MitM per half}} + \underbrace{2^{52+4(r_1+r_2)} + 2^{52+4(r_3+r_4)}}_{\mathcal{L}_1, \mathcal{L}_2} \right) + \underbrace{2 \cdot 2^{120}}_{\mathcal{L}_3, \mathcal{L}_4}$$

Dominated by  $2^{120}$  as long as  $r_i < 10$  and  $r_1 + r_2, r_3 + r_4 < 18$

# Complexity

Time:

$$2 \cdot \left( \underbrace{2^{80+4 \max\{r_1, r_2\}} + 2^{80+4 \max\{r_3, r_4\}}}_{\text{MitM per half}} + \underbrace{2^{52+4(r_1+r_2)} + 2^{52+4(r_3+r_4)}}_{\mathcal{L}_1, \mathcal{L}_2} \right) + \underbrace{2 \cdot 2^{120}}_{\mathcal{L}_3, \mathcal{L}_4}$$

Dominated by  $2^{120}$  as long as  $r_i < 10$  and  $r_1 + r_2, r_3 + r_4 < 18$

$r_1 = r_2 = r_3 = r_4 = 8$  (32 rounds):

| Time      | Memory    | Data |
|-----------|-----------|------|
| $2^{121}$ | $2^{120}$ | 3 KP |

## Attacking More Rounds

- Below  $2^{128}$  as long as  $\max\{r_i\} < 12$  and  $r_1 + r_2, r_3 + r_4 < 19$

## Attacking More Rounds

- Below  $2^{128}$  as long as  $\max\{r_i\} < 12$  and  $r_1 + r_2, r_3 + r_4 < 19$
- Bottleneck becomes  $|\mathcal{L}_1| = 2^{52+4(r_1+r_2)}$

## Attacking More Rounds

- Below  $2^{128}$  as long as  $\max\{r_i\} < 12$  and  $r_1 + r_2, r_3 + r_4 < 19$
- Bottleneck becomes  $|\mathcal{L}_1| = 2^{52+4(r_1+r_2)}$

$r_1 = r_2 = r_3 = r_4 = 9$  (36 rounds):

| Time      | Memory    | Data |
|-----------|-----------|------|
| $2^{126}$ | $2^{124}$ | 3 KP |

## Attacking More Rounds

- Below  $2^{128}$  as long as  $\max\{r_i\} < 12$  and  $r_1 + r_2, r_3 + r_4 < 19$
- Bottleneck becomes  $|\mathcal{L}_1| = 2^{52+4(r_1+r_2)}$

$r_1 = r_2 = r_3 = r_4 = 9$  (36 rounds):

| Time      | Memory    | Data |
|-----------|-----------|------|
| $2^{126}$ | $2^{124}$ | 3 KP |

Adding 4 rounds to CRAFT is **not enough**

## Why Does Dissection Save Time Here?

|                      | MitM [LN26] | Dissection  |
|----------------------|-------------|-------------|
| Rounds per list      | $r$         | $r/2$       |
| Largest list         | $2^{4r+24}$ | $2^{52+2r}$ |
| After merging $L, R$ | $2^{64}$    | $2^{120}$   |
| $r = 32$             | $2^{153}$   | $2^{121}$   |

## Why Does Dissection Save Time Here?

|                      | MitM [LN26] | Dissection  |
|----------------------|-------------|-------------|
| Rounds per list      | $r$         | $r/2$       |
| Largest list         | $2^{4r+24}$ | $2^{52+2r}$ |
| After merging $L, R$ | $2^{64}$    | $2^{120}$   |
| $r = 32$             | $2^{153}$   | $2^{121}$   |

- The cipher splits in *two dimensions*:  $L/R$  and top/bottom

## Why Does Dissection Save Time Here?

|                      | MitM [LN26] | Dissection  |
|----------------------|-------------|-------------|
| Rounds per list      | $r$         | $r/2$       |
| Largest list         | $2^{4r+24}$ | $2^{52+2r}$ |
| After merging $L, R$ | $2^{64}$    | $2^{120}$   |
| $r = 32$             | $2^{153}$   | $2^{121}$   |

- The cipher splits in *two dimensions*:  $L/R$  and top/bottom
- Matching “from a distance” ( $S, S'$ ): enjoyed *twice*

## Why Does Dissection Save Time Here?

|                      | MitM [LN26] | Dissection  |
|----------------------|-------------|-------------|
| Rounds per list      | $r$         | $r/2$       |
| Largest list         | $2^{4r+24}$ | $2^{52+2r}$ |
| After merging $L, R$ | $2^{64}$    | $2^{120}$   |
| $r = 32$             | $2^{153}$   | $2^{121}$   |

- The cipher splits in *two dimensions*:  $L/R$  and top/bottom
- Matching “from a distance” ( $S, S'$ ): enjoyed *twice*
- Every merge removes the dependency on the number of rounds

# Attacks on Craft (Single Key)

| Rounds | Time         | Memory    | Data        | Attack  | Setting | Reference |
|--------|--------------|-----------|-------------|---------|---------|-----------|
| 21     | $2^{106.53}$ | $2^{100}$ | $2^{60.99}$ | ID      | CP      | [HSE23]   |
| 23     | $2^{111.46}$ | $2^{120}$ | $2^{60.99}$ | D       | CP      | [SYCHW24] |
|        | $2^{109}$    | $2^{36}$  | $2^{58}$    | TD-MitM | CC      | [MNN25]   |
| 24     | $2^{110}$    | $2^{34}$  | $2^{60}$    | TD-MitM | CC      | [MNN25]   |
| 25     | $2^{117.58}$ | $2^{48}$  | $2^{60}$    | TD-MitM | CC      | [MNN25]   |
|        | $2^{125}$    | $2^{100}$ | $2^1$       | MitM    | KP      | [LN26]    |
| 26     | $2^{118}$    | $2^{34}$  | $2^{64}$    | TD-MitM | CC      | [MNN25]   |

|       |                             |         |                             |
|-------|-----------------------------|---------|-----------------------------|
| D     | Differential                | MitM    | Meet-in-the-Middle          |
| ID    | Impossible Differential     | TD-MitM | Truncated-Differential MitM |
| CP/CC | Chosen Plaintext/Ciphertext | KP      | Known Plaintext             |

# Attacks on Craft (Single Key)

| Rounds | Time         | Memory    | Data        | Attack     | Setting | Reference |
|--------|--------------|-----------|-------------|------------|---------|-----------|
| 21     | $2^{106.53}$ | $2^{100}$ | $2^{60.99}$ | ID         | CP      | [HSE23]   |
| 23     | $2^{111.46}$ | $2^{120}$ | $2^{60.99}$ | D          | CP      | [SYCHW24] |
|        | $2^{109}$    | $2^{36}$  | $2^{58}$    | TD-MitM    | CC      | [MNN25]   |
| 24     | $2^{110}$    | $2^{34}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
| 25     | $2^{117.58}$ | $2^{48}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
|        | $2^{125}$    | $2^{100}$ | $2^1$       | MitM       | KP      | [LN26]    |
| 26     | $2^{118}$    | $2^{34}$  | $2^{64}$    | TD-MitM    | CC      | [MNN25]   |
| 32     | $2^{121}$    | $2^{120}$ | 3           | Dissection | KP      | this work |

D

Differential

ID

Impossible Differential

CP/CC

Chosen Plaintext/Ciphertext

MitM

Meet-in-the-Middle

TD-MitM

Truncated-Differential MitM

KP

Known Plaintext

# Attacks on Craft (Single Key)

| Rounds | Time         | Memory    | Data        | Attack     | Setting | Reference |
|--------|--------------|-----------|-------------|------------|---------|-----------|
| 21     | $2^{106.53}$ | $2^{100}$ | $2^{60.99}$ | ID         | CP      | [HSE23]   |
| 23     | $2^{111.46}$ | $2^{120}$ | $2^{60.99}$ | D          | CP      | [SYCHW24] |
|        | $2^{109}$    | $2^{36}$  | $2^{58}$    | TD-MitM    | CC      | [MNN25]   |
| 24     | $2^{110}$    | $2^{34}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
| 25     | $2^{117.58}$ | $2^{48}$  | $2^{60}$    | TD-MitM    | CC      | [MNN25]   |
|        | $2^{125}$    | $2^{100}$ | $2^1$       | MitM       | KP      | [LN26]    |
| 26     | $2^{118}$    | $2^{34}$  | $2^{64}$    | TD-MitM    | CC      | [MNN25]   |
| 32     | $2^{121}$    | $2^{120}$ | 3           | Dissection | KP      | this work |
| 36     | $2^{126}$    | $2^{124}$ | 3           | Dissection | KP      | this work |

D

Differential

MitM

Meet-in-the-Middle

ID

Impossible Differential

TD-MitM

Truncated-Differential MitM

CP/CC

Chosen Plaintext/Ciphertext

KP

Known Plaintext

- **First full-round attack** on CRAFT in the single-key model
  - Time  $2^{121}$ , memory  $2^{120}$ , 3 known plaintexts

- **First full-round attack** on CRAFT in the single-key model
  - Time  $2^{121}$ , memory  $2^{120}$ , 3 known plaintexts
- Extends to **36 rounds**:  $2^{126}$  time

- **First full-round attack** on CRAFT in the single-key model
  - Time  $2^{121}$ , memory  $2^{120}$ , 3 known plaintexts
- Extends to **36 rounds**:  $2^{126}$  time
- Decomposition + dissection: saves *time*, not only memory

- **First full-round attack** on CRAFT in the single-key model
  - Time  $2^{121}$ , memory  $2^{120}$ , 3 known plaintexts
- Extends to **36 rounds**:  $2^{126}$  time
- Decomposition + dissection: saves *time*, not only memory
- Root cause: key schedule aligned with the decomposition
  - A nibble permutation in the key schedule would prevent the attack

Questions?

- [BLMR19] C. Beierle, G. Leander, A. Moradi, S. Rasoolzadeh. *CRAFT: Lightweight Tweakable Block Cipher with Efficient Protection Against DFA Attacks*. ToSC 2019(1).
- [CE85] D. Chaum, J.-H. Evertse. *Cryptanalysis of DES with a Reduced Number of Rounds*. CRYPTO 1985.
- [DDKS19] I. Dinur, O. Dunkelman, N. Keller, A. Shamir. *Efficient Dissection of Bicomposite Problems with Cryptanalytic Applications*. J. Cryptology 32(4), 2019.
- [DH77] W. Diffie, M. Hellman. *Exhaustive Cryptanalysis of the NBS Data Encryption Standard*. Computer 10(6), 1977.
- [MH81] R. Merkle, M. Hellman. *On the Security of Multiple Encryption*. Comm. ACM 24(7), 1981.
- [DSP07] O. Dunkelman, G. Sekar, B. Preneel. *Improved Meet-in-the-Middle Attacks on Reduced-Round DES*. INDOCRYPT 2007.

- [HSE23] H. Hadipour, S. Sadeghi, M. Eichlseder. *Finding the Impossible: Automated Search for Full Impossible-Differential, Zero-Correlation, and Integral Attacks*. EUROCRYPT 2023.
- [LN26] E. Lambooi, P. Neumann. *A Known-Plaintext Attack with Minimal Data Complexity on 25-Round CRAFT*. ToSC 2026(1).
- [MNN25] D. M'foukh, M. Naya-Plasencia, P. Neumann. *The State-Test Technique on Differential Attacks: a 26-Round Attack on Craft and Other Applications*. ASIACRYPT 2025.
- [SYCHW24] L. Song, Q. Yang, Y. Chen, L. Hu, J. Weng. *Probabilistic Extensions: A One-Step Framework for Finding Rectangle Attacks and Beyond*. EUROCRYPT 2024.