



HAL
open science

Symmetric lightweight primitives: (Design and) Cryptanalysis

María Naya-Plasencia

► **To cite this version:**

María Naya-Plasencia. Symmetric lightweight primitives: (Design and) Cryptanalysis. Lightweight Crypto Day 2018, Apr 2018, Tel Aviv, Israel. hal-01953947

HAL Id: hal-01953947

<https://inria.hal.science/hal-01953947>

Submitted on 13 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Symmetric lightweight primitives: (Design and) Cryptanalysis

María Naya-Plasencia
Inria, France

Tel Aviv, Lightweight Crypto Day 2018

Outline

- ▶ Symmetric lightweight primitives
- ▶ Most used cryptanalysis
 - *Impossible Differential Attacks*
 - *Meet-in-the-middle*
 - Dedicated attacks
- ▶ Conclusions and remarks

Symmetric Lightweight Primitives

Lightweight Primitives

- ▶ Lightweight primitives designed for **constrained environments**, like RFID tags, sensor networks.

- ▶ Real need \Rightarrow an **enormous amount of proposals** in the last years:

PRESENT, LED, KATAN/KTANTAN, KLEIN, PRINCE, PRINTcipher, LBLOCK, TWINE, XTEA, mCrypton, Iceberg, HIGHT, Piccolo, SIMON, SPECK, SEA, DESL...

- ▶ NIST competition to start around december 2018.

Cryptanalysis: Foundation of Confidence

Any attack better than the generic one is considered a “break” .

- ▶ Proofs on symmetric primitives need to make unrealistic assumptions.
- ▶ We need to perform an **empirical measure** of the security: cryptanalysis.

Lightweight Primitives

- ▶ Cryptanalysis of lightweight primitives: a fundamental task, responsibility of the community.
- ▶ Importance of cryptanalysis (especially on new proposals): the more a cipher is analyzed, the more confidence we can have in it...
- ▶ ...or know which algorithms are not secure to use.

Lightweight Primitives

- ▶ Lightweight: more 'risky' design, lower security margin, simpler components.
- ▶ Often innovative constructions: dedicated attacks
- ▶ Types of attacks: single-key/related-key, distinguisher/key-recovery, weak-keys, reduced versions.

On weakened versions

If no attack is found on a given cipher, what can we say about its robustness, security margin?

The security of a cipher is not a 1-bit information:

- Round-reduced attacks.
 - Analysis of components.
- ⇒ determine and adapt the security margin.

On high complexities

When considering large keys, sometimes attacks breaking the ciphers might have a very high complexity far from practical e.g.. 2^{120} for a key of 128 bits.

Still dangerous because:

- Weak properties not expected by the designers.
 - Experience shows us that **attacks only get better**.
 - Other existing ciphers without the "ugly" properties.
- ▶ When determining the **security margin**: find the highest number of rounds reached.

Main Objectives of this talk

- ▶ Perform a (non-exhaustive) survey of proposals and their security status.
- ▶ Provide the intuition of the “most useful attacks” against LW ciphers.
- ▶ Conclusions and remarks (link with hash functions).

Survey of Proposals ¹

- ▶ *Feistel Networks - best external analysis*
- DESLX - none
- ITUbee - self-similarity (8/20r)
- LBlock - **imposs. diff.** (24/32r)
- SEA - none
- SIMON and SPECK - **imposs. diff.**, diff, 0-correl.
- XTEA - **mitm** (23/64r)
- CLEFIA - **imposs. diff.** (13/18r)
- HIGHT - 0-correlation (27/32r)
- TWINE - **mitm,imposs. diff.**,0-corr (25/36r)

¹mainly from https://cryptolux.org/index.php/Lightweight_Block_Ciphers

Survey of Proposals

- ▶ *Substitution-Permutation Network*
 - KLEIN - **dedicated attack** (full round)
 - LED - EM generic attacks (8/12r, 128K)
 - Zorro - diff. (full round)
 - mCrypton - **mitm** (9/12r, 128K)
 - PRESENT - mult. dim. lin. (27/31r)
 - PRINTcipher - **invariant-wk** (full round)
 - PRIDE - diff (18/20r)
 - PRINCE - mult. diff (10/12r)
 - Fantomas/Robin - none/**invariant-wk** (full round)

Survey of Proposals

- ▶ *FSR-based*
 - KTANTAN/KATAN - **mitm** (153/254r)
 - Grain - correl./ cube attacks (some full)
 - Trivium - cube attacks (800/1152) -
 - Sprout - guess-and-determine (full round)
 - Quark -condit. diff (25%)
 - Fruit - divide and conquer (full)
 - Lizard - guess-and-det. (full)

Survey of Proposals

- ▶ *ARX*
 - Chaskey - diff-lin (7/8r)
 - Hight - 0-correl (27/32r)
 - LEA - diff. (14/24r)
 - RC5 - diff. (full round)
 - Salsa20 - diff (8/20r)
 - Sparx - **imposs. diff.** (15/24r)
 - Speck - diff. (17/32r)

More Proposals

For more details, primitives, classifications, see:

State of the Art in Lightweight Symmetric Cryptography,
by Alex Biryukov and Leo Perrin
<https://eprint.iacr.org/2017/511>

Most Successful Attacks

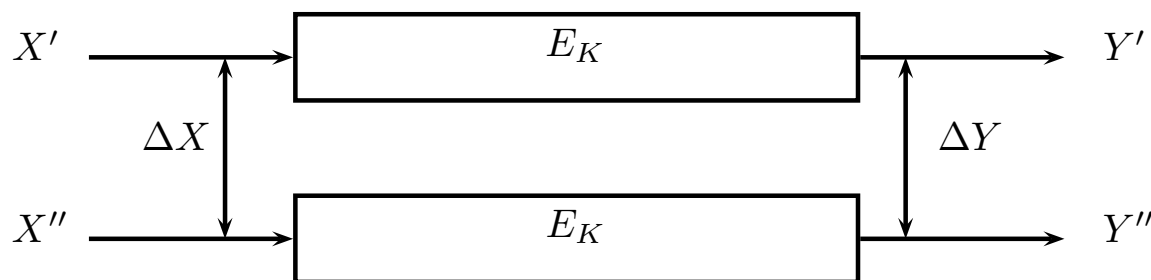
Families of attacks

- ▶ Impossible differentials (Feistel)
- ▶ Mitm / guess and determine (SPN, FSR)
- ▶ Dedicated: (differential/linear...)

Impossible Differential Attacks

Classical Differential Attacks [BS'90]

Given an **input difference** between two plaintexts, some **output differences** occur more often than others.

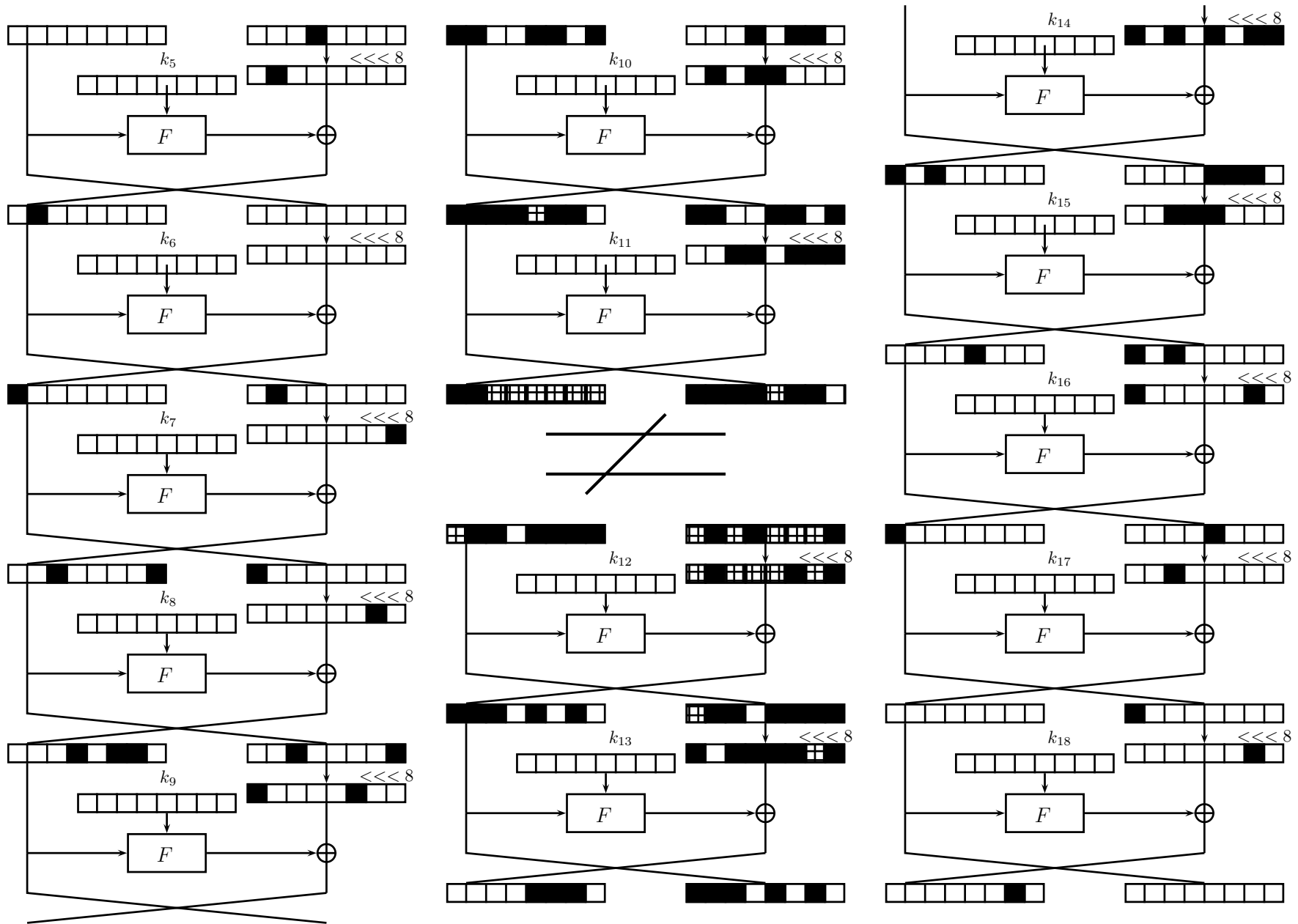


A differential is a pair (Δ_X, Δ_Y) .

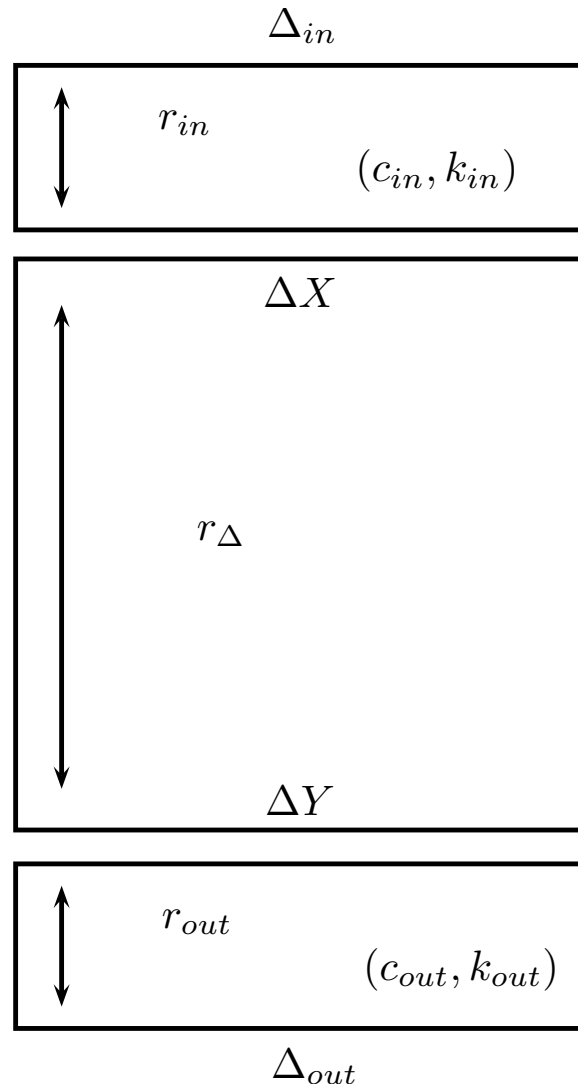
Impossible Differential Attacks [K,BBS'98]

- ▶ Impossible differential attacks use a differential with probability 0.
- ▶ We can find the impossible differential using the **Miss-in-the-middle [BBS'98]** technique.
- ▶ Extend it backward and forward \Rightarrow **Active Sboxes** transitions give information on the involved key bits.
- ▶ **Generic framework and improvements [BNPS14,BLNPS17]**

Impossible differential: 14 rounds



Impossible Differential Attack



Discarding Wrong Keys

- ▶ Given one pair of inputs with Δ_{in} that produces Δ_{out} ,
- ▶ all the (partial) keys that produce ΔX from Δ_{in} and ΔY from Δ_{out} differ from the correct one.
- ▶ If we consider N pairs verifying $(\Delta_{in}, \Delta_{out})$ the probability of NOT discarding a candidat key is

$$(1 - 2^{-c_{in} - c_{out}})^N$$

For the Attacks to Work

We need, for a state size s and a key size $|K|$:

$$C_{data} < 2^s$$

and

$$C_{data} + 2^{|k_{in} \cup k_{out}|} C_N + 2^{|K| - |k_{in} \cup k_{out}|} P 2^{|k_{in} \cup k_{out}|} < 2^{|K|}$$

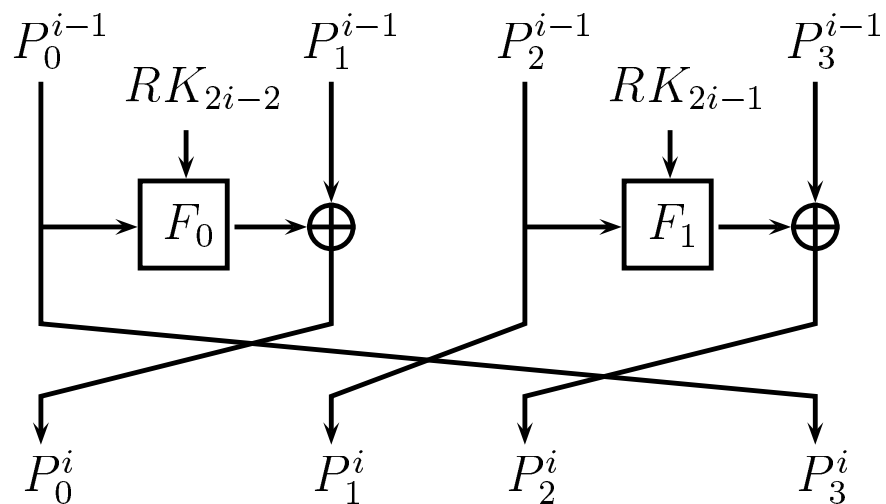
where C_{data} is the data needed for obtaining N pairs $(\Delta_{in}, \Delta_{out})$, C_N is the average cost of testing the pairs per candidate key (**early abort technique** [LKKD08]) and P is the probability of not discarding a candidate key.

Improvements from [BN-PS14, BLN-PS17]

- ▶ Multiple impossible differentials (related to [JN-PP13])
- ▶ Correctly choosing Δ_{in} and Δ_{out} (related to [MRST09])
- ▶ State-test technique (related to [MRST09])

Example: CLEFIA-128

- block size: $4 \times 32 = 128$ bits
- key size: 128 bits
- # of rounds: 18



Multiple Impossible Differentials

Formalize the idea of [Tsunoo et al. 08]:

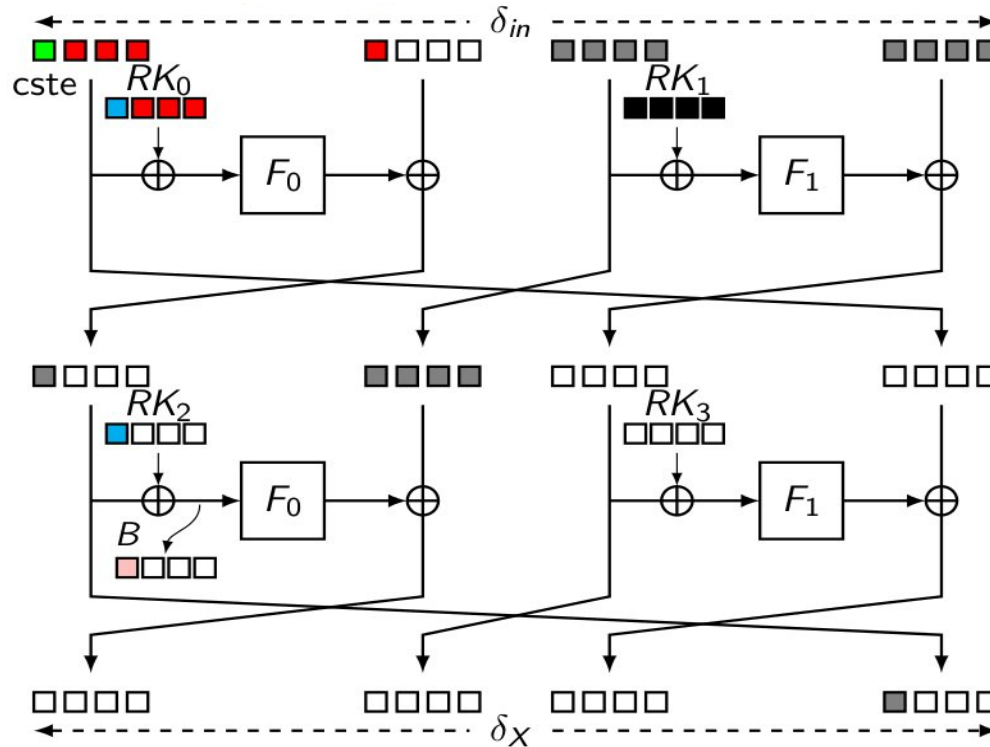
CLEFIA has two 9-round impossible differentials $((0, 0, 0, A) \not\rightarrow (0, 0, 0, B))$ and $((0, A, 0, 0) \not\rightarrow (0, B, 0, 0))$ when A and B verify:

<i>A</i>	<i>B</i>
$(0, 0, 0, \alpha)$	$(0, 0, \beta, 0)$ or $(0, \beta, 0, 0)$ or $(\beta, 0, 0, 0)$
$(0, 0, \alpha, 0)$	$(0, 0, 0, \beta)$ or $(0, \beta, 0, 0)$ or $(\beta, 0, 0, 0)$
$(0, \alpha, 0, 0)$	$(0, 0, 0, \beta)$ or $(0, 0, \beta, 0)$ or $(\beta, 0, 0, 0)$
$(\alpha, 0, 0, 0)$	$(0, 0, 0, \beta)$ or $(0, 0, \beta, 0)$ or $(0, \beta, 0, 0)$

24 in total: $C_{data} = 2^{113}$ becomes $C_{data} = 2^{113}/24$

State Test Technique

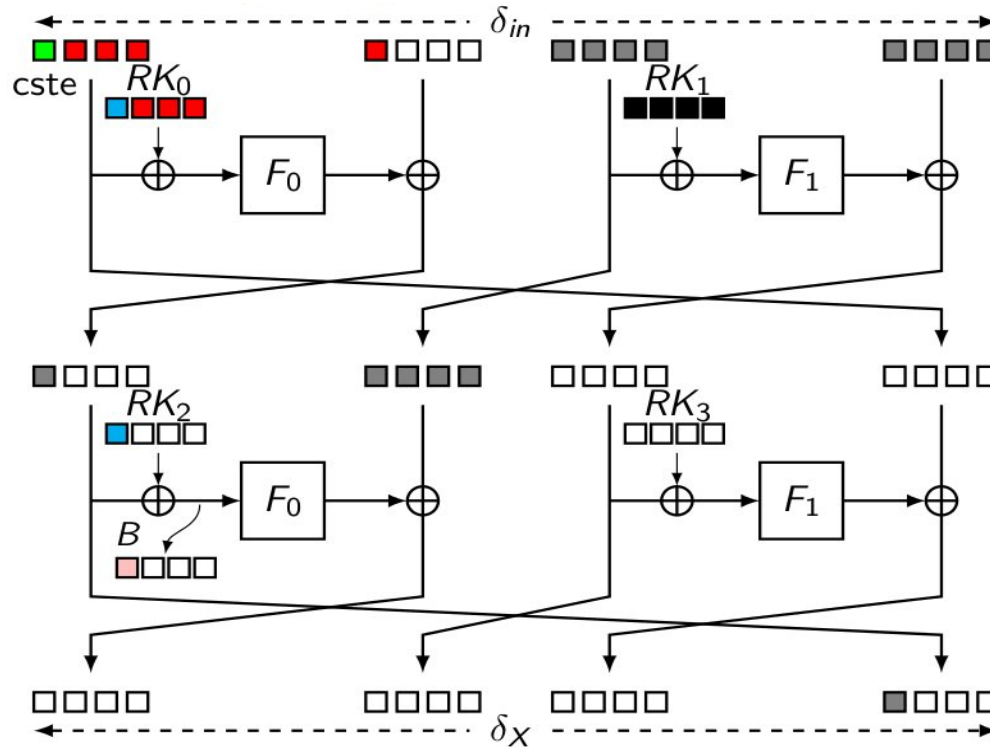
Reduce the number of key bits involved.



$$B = \blacksquare \oplus S_0(\blacksquare \oplus \color{green}\blacksquare) \oplus \color{red}\blacksquare$$

State Test Technique

Reduce the number of key bits involved.



$$B' = \blacksquare \oplus S_0(\blacksquare \oplus \blacksquare) \quad (\text{with } B = B' \oplus \blacksquare)$$

$$|k_{in} \cup k_{out}| = 122 \text{ bits} \quad \Rightarrow \quad |k_{in} \cup k_{out}| = 122 - 16 + \underbrace{8}_{B'} \text{ bits}$$

Applications of Improved Impossible Diff

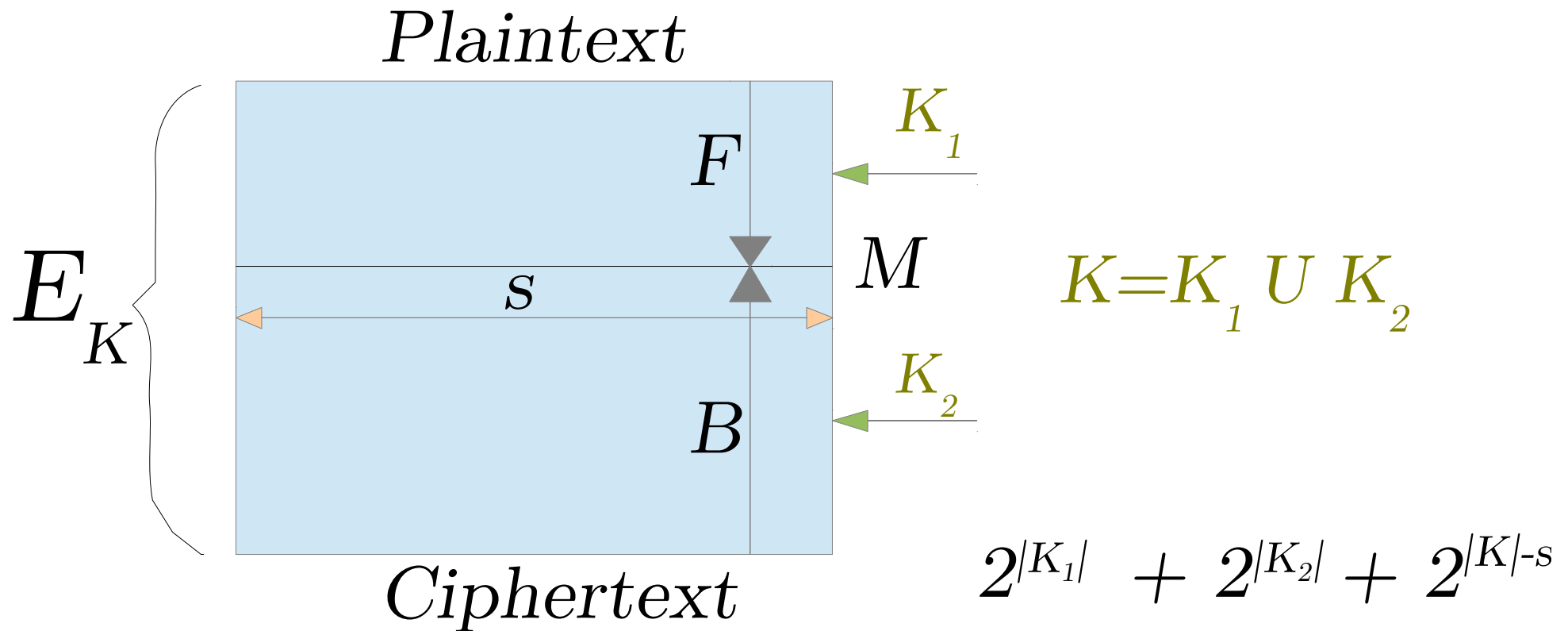
- ▶ CLEFIA: best attack on CLEFIA (13 rounds).
- ▶ Camellia: Improved best attacks for Camellia.
- ▶ AES: attacks comparable with best mitm ones (7 rounds).
- ▶ LBlock: best attack (on 24 rounds).

Meet-in-the-middle attacks

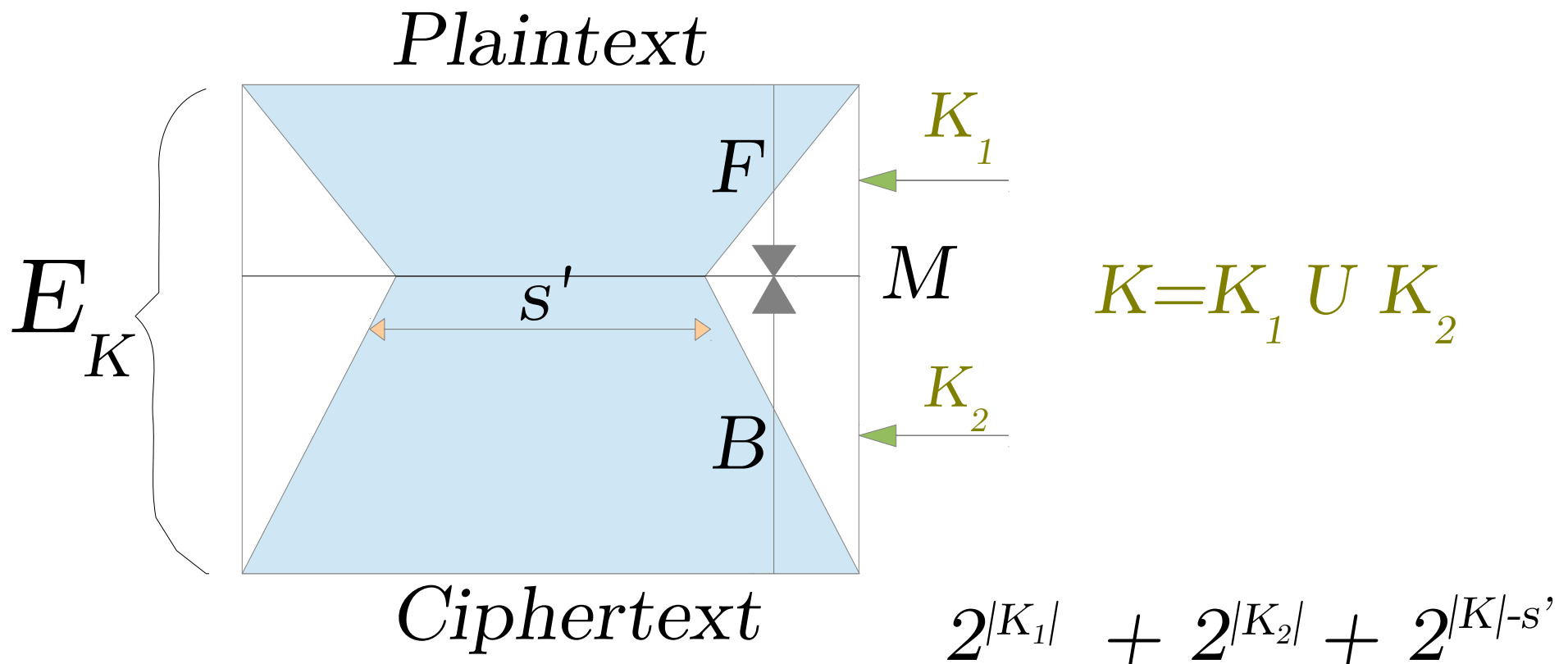
Meet-in-the-Middle Attacks

- ▶ Introduced by Diffie and Hellman in 1977.
- ▶ Largely applied tool.
- ▶ Few data needed.
- ▶ Many improvements: partial matching, bicliques, sieve-in-the-middle...

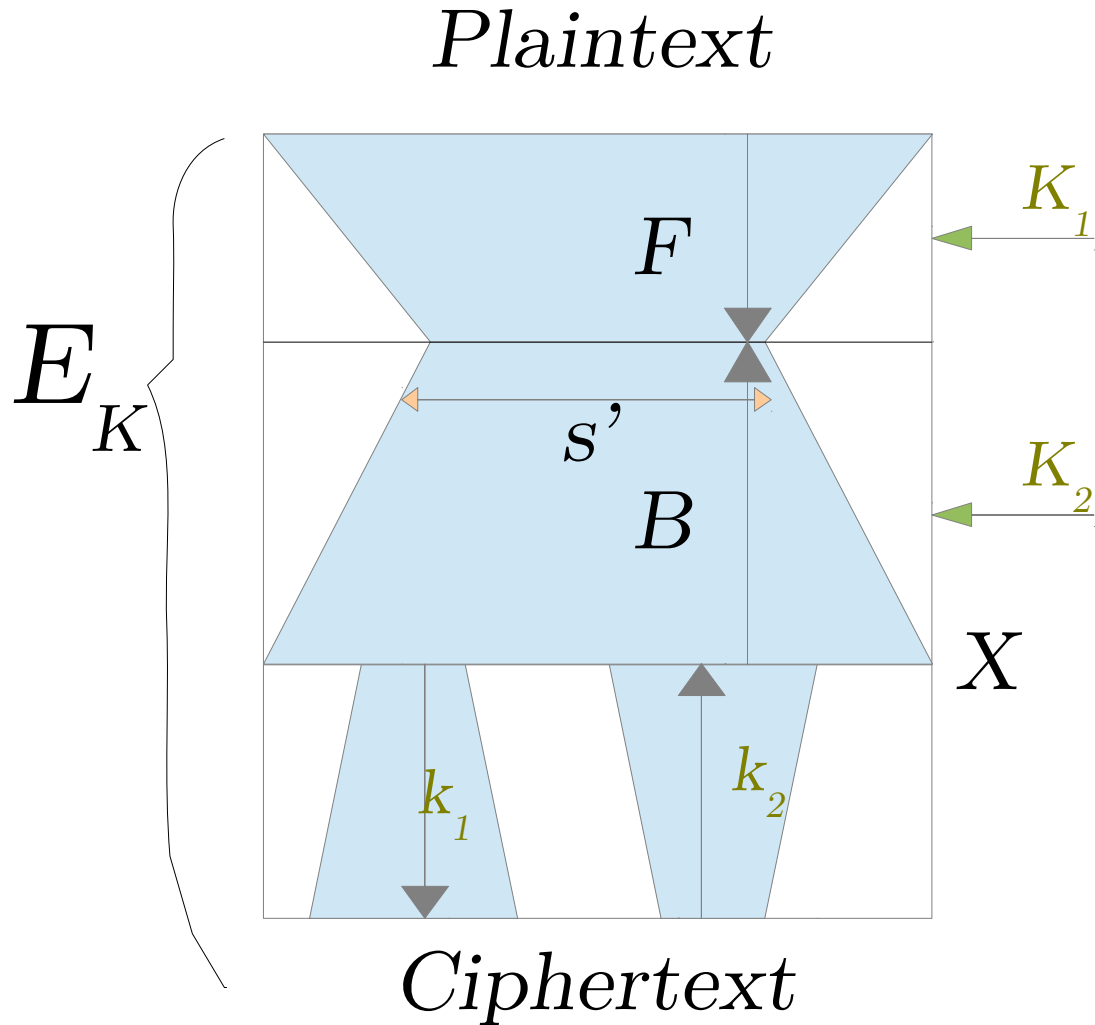
Meet-in-the-Middle Attacks [Diffie Hellman 77]



With Partial Matching [AS'08]



With Bicliques [KRS'11]



$$K = K_1 \cup K_2$$

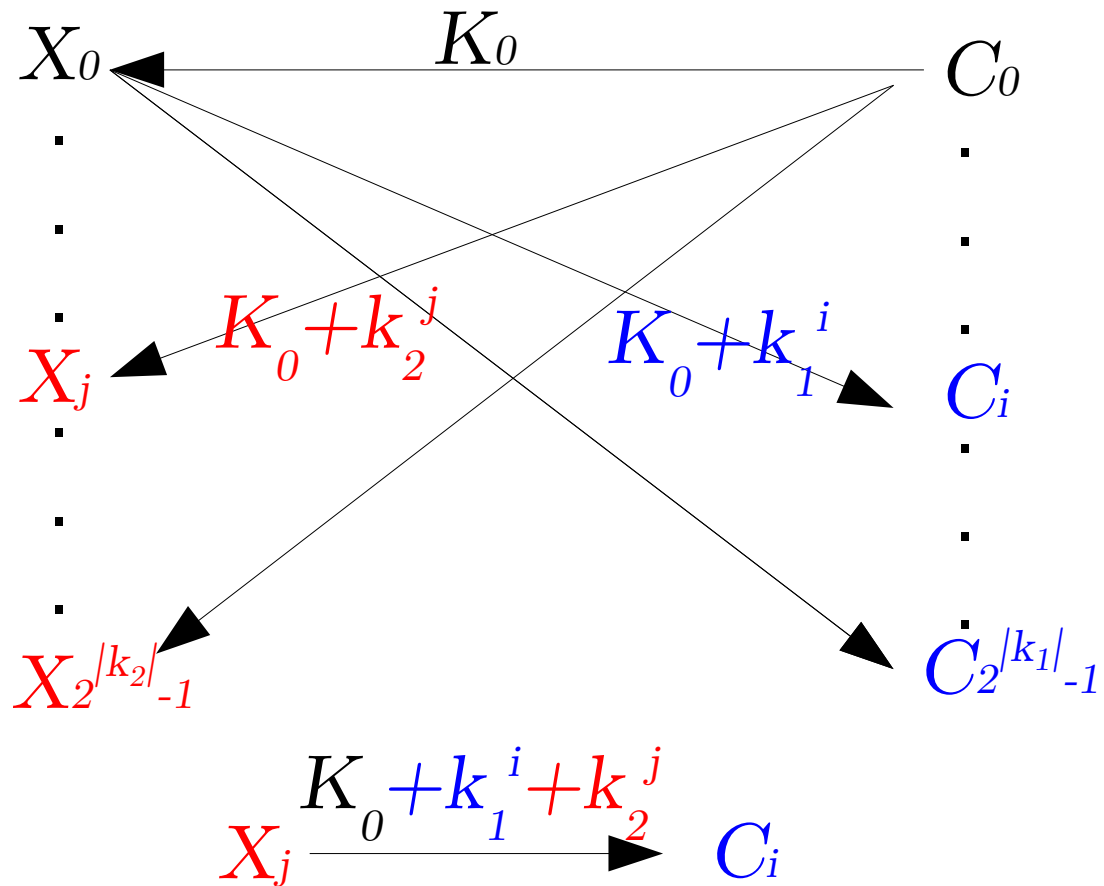
$$2^{|k_1|} + 2^{|k_2|} + 2^{|K_1|} + 2^{|K_2|} + 2^{|K| - s'}$$

Bicliques

- ▶ Improvement of MITM attacks, but also...
- ▶ It can always be applied to reduce the total number of computations (at least the precomputed part)
⇒ acceleration of exhaustive search [BKR'11]²
- ▶ Many other accelerated exhaustive search on LW block ciphers: PRESENT, LED, KLEIN, HIGHT, Piccolo, TWINE, LBlock ... (less than 2 bits of gain).
- ▶ Is everything broken? No.

²Most important application: best key-recovery on AES-128 in $2^{126.1}$ instead of the naive 2^{128} .

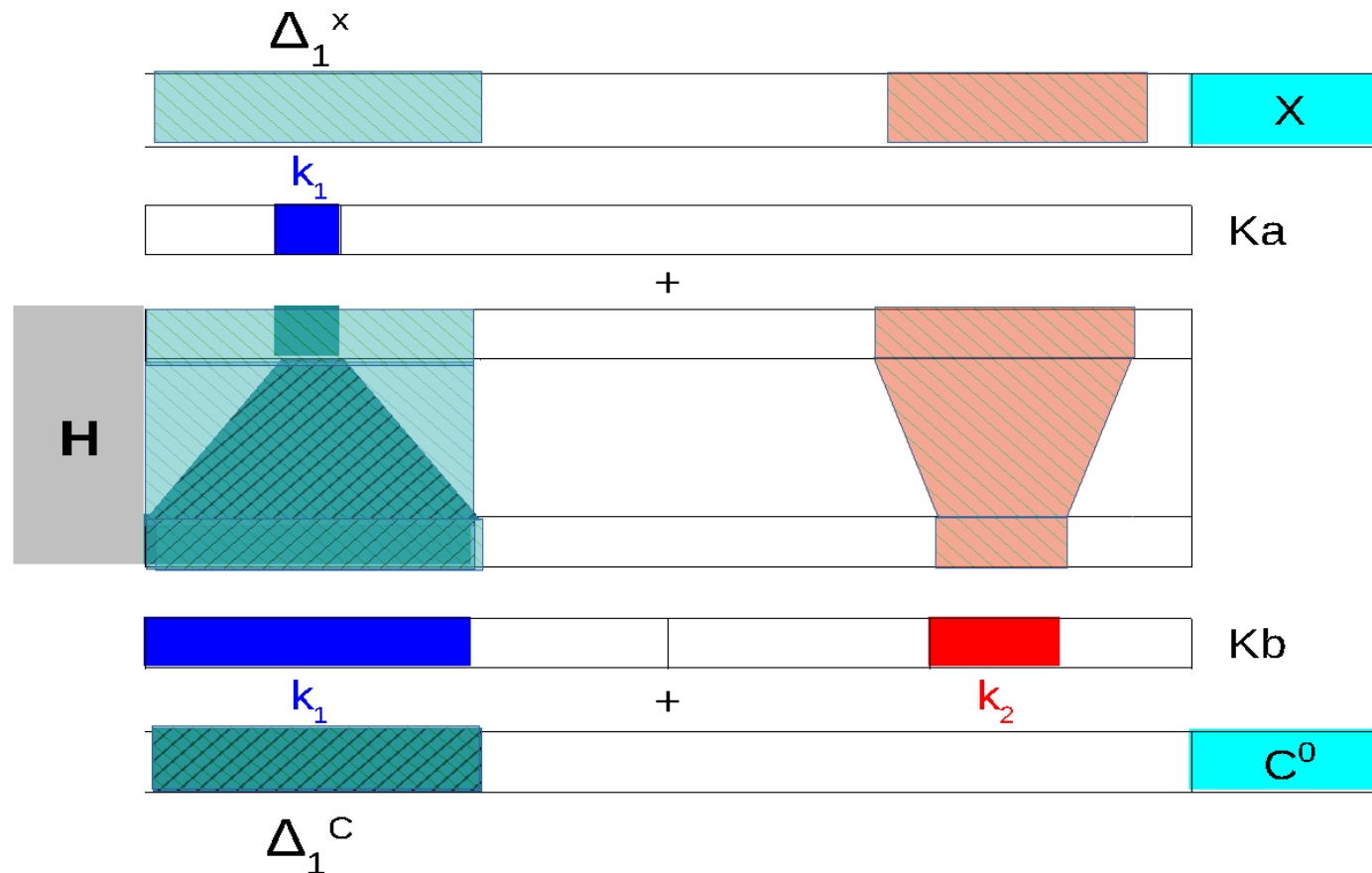
Bicliques



With
 $2^{|k_1|} + 2^{|k_2|}$
 computations,
 $2^{|k_1+k_2|}$
 Transitions.

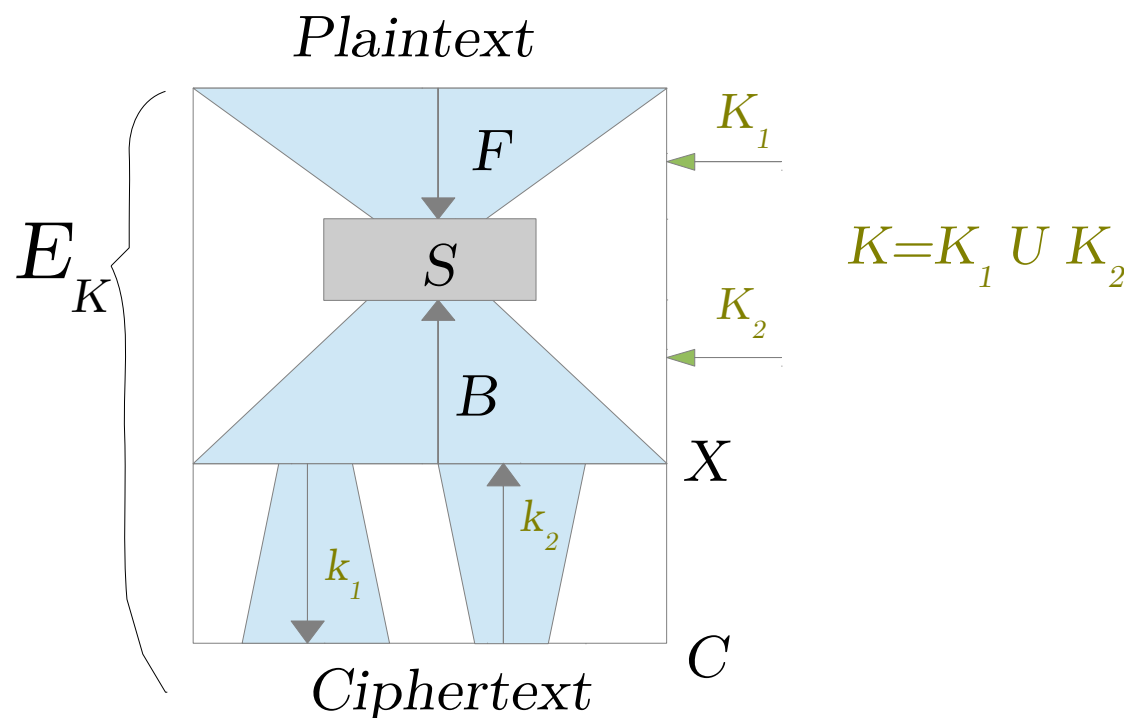
Improved Bicliques [CN-PV 13]

Can we build bicliques with only one pair of P-C?

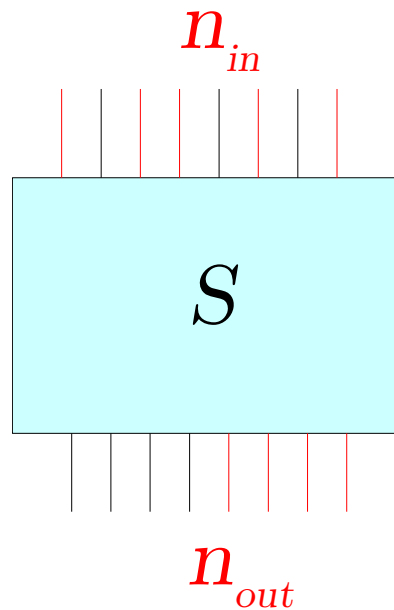


Sieve-in-the-Middle [CN-PV'13]

- ▶ Compute partial inputs and outputs of S
⇒ sieving with **transitions** instead of collisions.



When can we sieve?



- ▶ n_{in} known bits out of m : at most $2^{m-n_{in}}$ values for the n_{out} output bits.
- ▶ A transition exists with probability p .
- ▶ Sieve when $n_{in} + n_{out} > m \Rightarrow p < 1$

How do we sieve?

- ▶ We obtain a list L_A of partial inputs u and a list L_B of partial outputs $v \Rightarrow$ merge L_A and L_B with the condition (u, v) is a valid transition through S .
- ▶ Naive way costs $|L_A| \times |L_B| = 2^{|K_1|+|K_2|}$:
no gain with respect to exhaustive search.
- ▶ We need an efficient procedure.
Often S is a concatenation of S-boxes.

Merging the lists

Merging the lists with respect to R

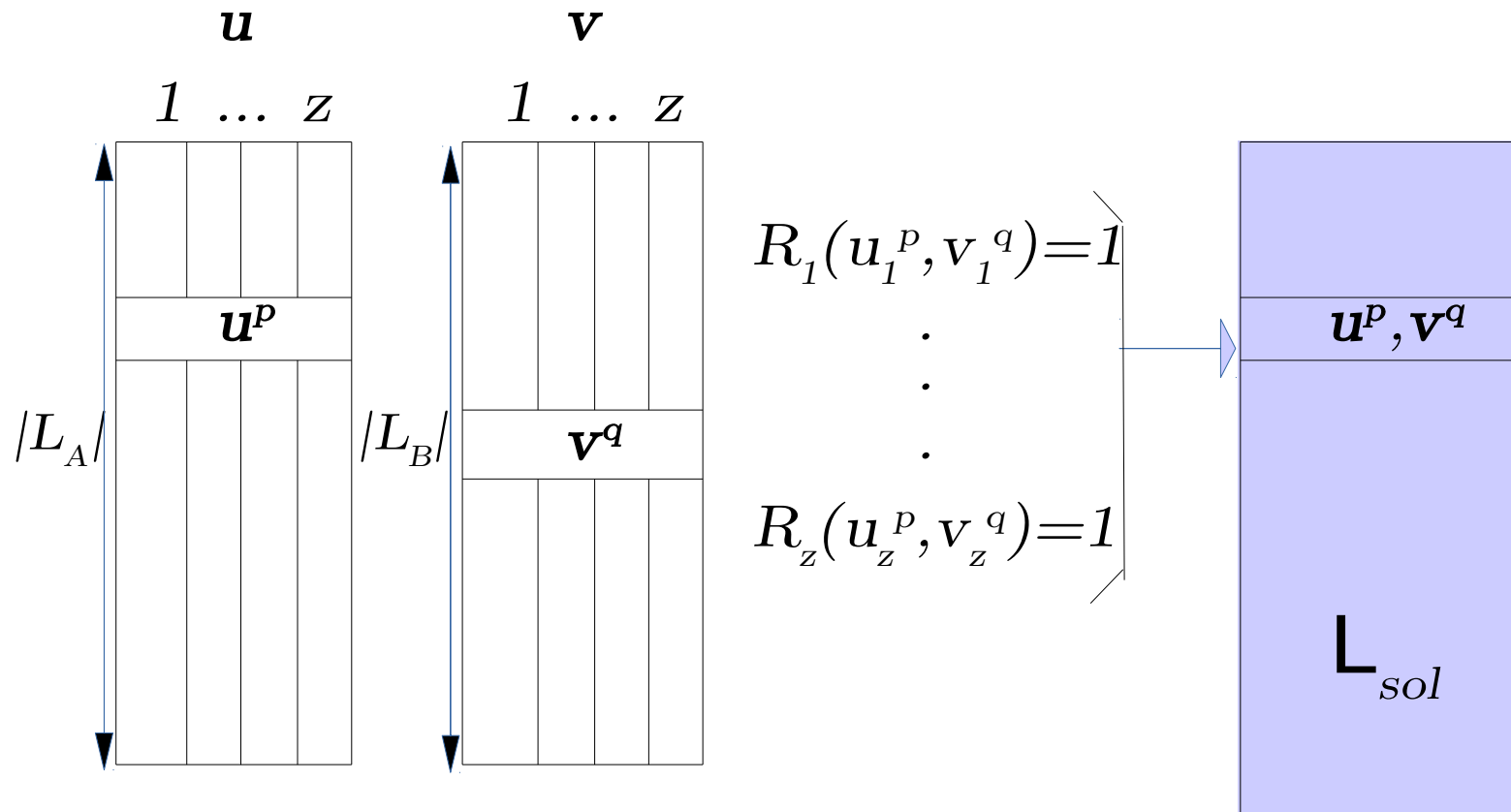
- ▶ R is group-wise, *i.e.* for z groups

$$R(u, v) = \prod_{i=1}^z R_i(u_i, v_i)$$

Find all $u \in L_A$ and $v \in L_B$ such that $R(u, v) = 1$.

- ▶ Subcase of the first problem in [N-P 11].
First studied for rebound attacks.

Group-wise relation



Merging Algorithms

- ▶ Problem also appears in divide-and-conquer attacks (and rebound attacks).
- ▶ Solutions from list merging algorithms [N-P-11] and dissection algorithms [DDKS 12]
- ▶ Many applications: ARMADILLO2 [ABN-PVZ 11], ECHO256 [JN-PS 11], JH42 [N-PTV 11], Grøstl [JN-PP 12], Klein [LN-P 14], AES-like [JN-PP 14], Sprout [LN-P 15], Ketje [FN-PR 18]...

Some Applications SITM

- ▶ Reduced-round: PRESENT, DES, PRINCE, AES-biclique [Canteaut N-P Vayssieres 13]
- ▶ Reduced-round LBlock [Altawy Youssef 14]
- ▶ Best reduced-round KATAN [Fuhr Minaud 14]
- ▶ Reduced-round Simon [Song et al 14]
- ▶ Low-data AES [Bogdanov et.al 15]
[Tao et al 15]
- ▶ MIBS80/PRESENT80 [Faghihi et al 16]

- ▶ Interesting for low data attacks...

Importance of Dedicated Cryptanalysis

Lightweight Dedicated Analysis

- ▶ Few cases broken by well known attacks (ex. Puffin or Puffin2 - multiple differentials)
- ▶ Happily, this is rare. Most of the times, new families or new ideas on known attacks exploiting the new properties are needed.
- ▶ Lightweight: more 'risky' design, lower security margin, simpler components.
- ▶ Often innovative constructions: dedicated attacks

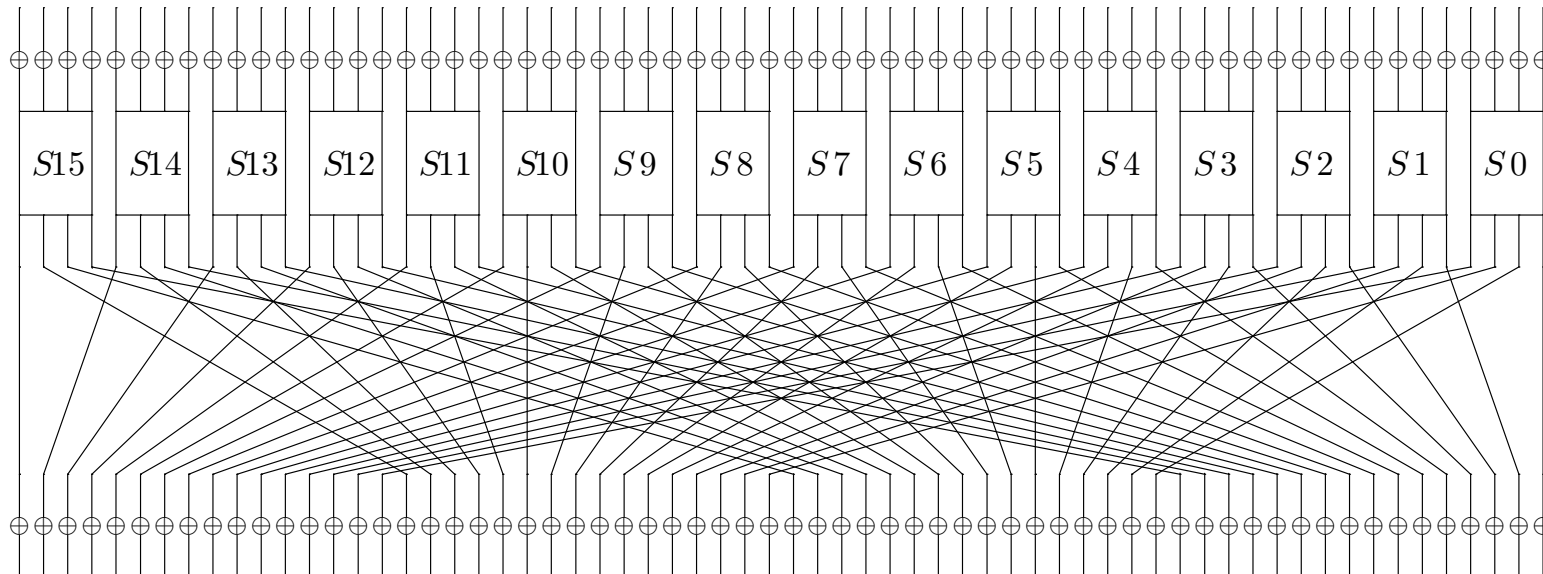
Ex: PRESENT and PRINTcipher

PRESENT [BKLPPRSV'07]

- ▶ One of the most popular ciphers, proposed in 2007, and now ISO/IEC standard.
- ▶ Very large number of analysis published (20+).
- ▶ Best attacks so far: multiple linear attacks (27r/31r).

PRESENT

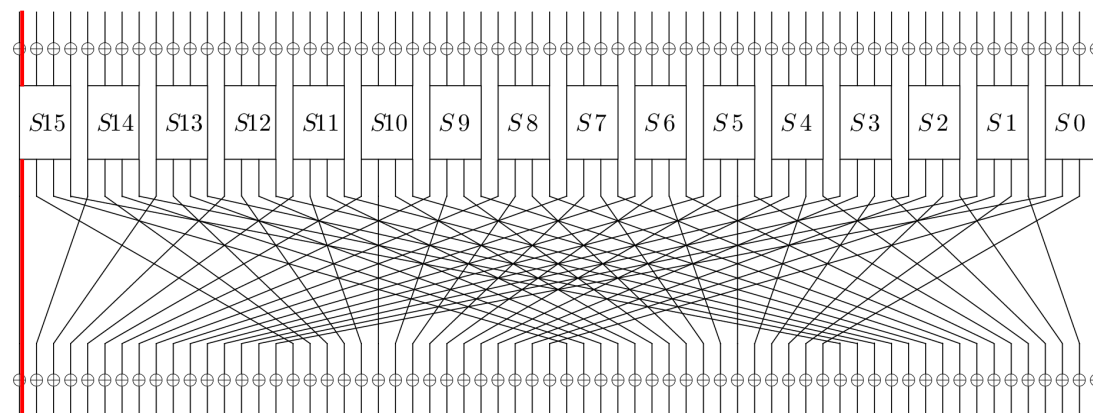
Block $n = 64$ bits, key 80 or 128 bits.



31 rounds + 1 key addition.

PRESENT

Linear cyptanalysis: because of the Sbox, a linear approximation 1 to 1 with bias 2^{-3} per round [O-09].

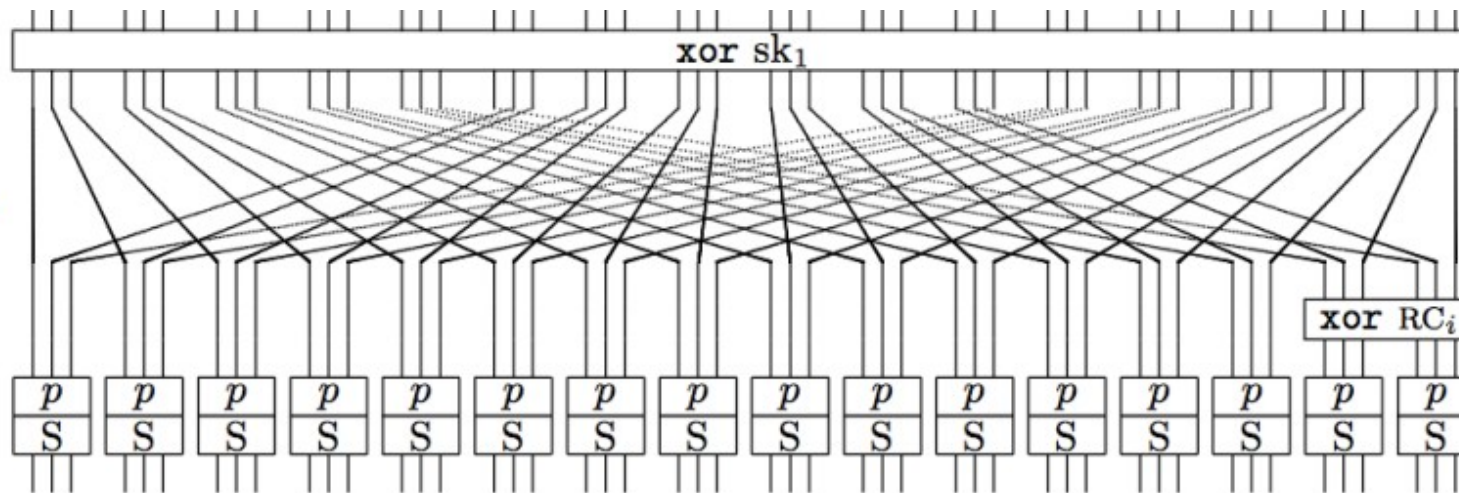


- ▶ Multiple linear attacks: consider several possible approxs simultaneously \Rightarrow up to 27 rounds out of 31 [BN-14].

PRINTcipher

- ▶ Many PRESENT-like ciphers proposed, like Puffin, PRINTcipher
- ▶ Usually, weaker than the original.
- ▶ PRINTcipher[KLPR'10]: first cryptanalysis: invariant subspace attack[LAAZ'11].

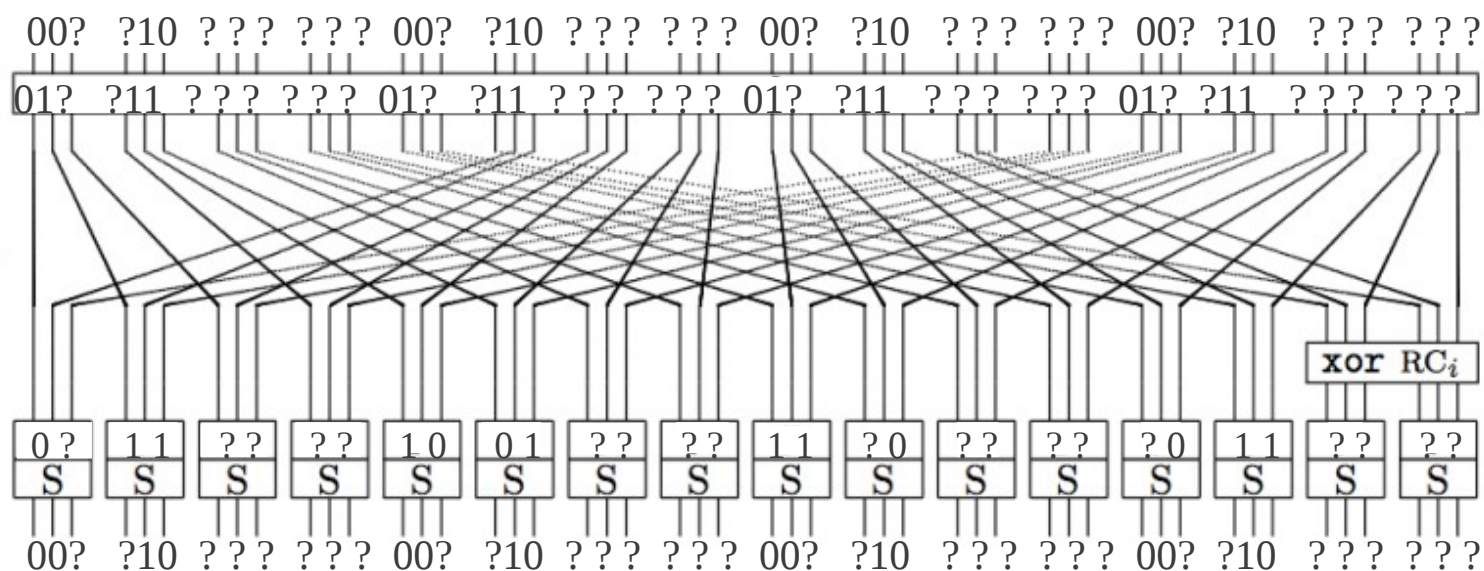
PRINTcipher



48 rounds.

The Invariant Subspace Attack [LAAZ'11]

With probability 1:



- ▶ Weak key attack, but a very bad property for 2^{51} keys...

The Invariant Subspace Attack

- ▶ More applications afterwards:
iScream, Robin, Zorro, Midori.
- ▶ Importance of generalizing/understanding
dedicated attacks:
new families/techniques might appear.

Final remarks

Zorro - Hash Functions links

- ▶ Lightweight block cipher proposed [GGN-PS13] for easy masking.
- ▶ A modified AES with only four sboxes per round (SPN with **partial non-linear layer**).
- ▶ **Bounds** on number of active Sboxes? Computed using **freedom degrees**.
- ▶ Many analyses published. Problem: MC property \Rightarrow devastating attack [BDDLT13, RASA13]

LED - Hash Functions links

- ▶ Lightweight block cipher proposed in [GPPR12].
- ▶ AES-like with simpler key-schedule and more rounds. Nice simple design.
- ▶ Analysis provided with respect to **known key distinguishers** (rebound-like). Seems like a lot of SHA-3 knowledge put into this design.

Hash functions links - Sum up

- ▶ Mitm, bicliques/initial structures:
used for both scenarios
- ▶ Early abort \leftarrow message modification techniques
- ▶ State-test tech. & choosing $\Delta_{in,out} \leftarrow$ Rebound attacks
- ▶ Mult. impos. diff. \leftarrow mult. limited birthday distinguishers
- ▶ Using freedom degrees for bounds?... be careful!!
- ▶ Merging lists from rebounds/sieve in the middle
 \rightarrow many applications
- ▶ *Other ex: AES distinguishers inspired on rebound attacks.*

Conclusion

To Sum Up

- ▶ Classical attacks, but also new **dedicated** ones exploiting the **originality** of the designs.
- ▶ Importance on generalizing: improvements, and dedicated might become well established techniques.
- ▶ Importance of **reduced-round analysis** to re-think security margin, or as first steps of further analysis.
- ▶ New **ideas inspired by SHA-3**: might help improving attacks further!
- ▶ Better identifying composite problems/ list merging situations might provide improved results.

To Sum Up³

A lot of ciphers to analyze/ a lot
of work to do!

³Thank you to Christina Boura and Leo Perrin for their help with the figures and the slides.