Impossible Differentials Automation: Model Generation and New Techniques

Emanuele Bellini[2] Alessandro De Piccoli[1] David Gerault[2]
Paul Huynh[2] Simone Pelizzola[1] Andrea Visconti[1]

[1]Università degli Studi di Milano, Milano, Italia

[2]Technology Innovation Institute, Abu Dhabi, UAE

SAC, Toronto (CA) – August 14th, 2025

Aim of the paper

ID Cryptanalysis automation with Constraint Programming:

Aim of the paper

ID Cryptanalysis automation with Constraint Programming:

- Negative models, precise, cost 2^{2n}.

Aim of the paper

ID Cryptanalysis automation with Constraint Programming:

- Negative models, precise, cost 2^{2n}.
- Positive models, less precise (false positives, as we just saw, or false negatives, as we will see) but very efficient.

Aim of the paper

ID Cryptanalysis automation with Constraint Programming:

- Negative models, precise, cost 2^{2n}.
- Positive models, less precise (false positives, as we just saw, or false negatives, as we will see) but very efficient.

Our goal: improve state of the art of automation of ID Cryptanalysis in the second case.

Aim of the paper

ID Cryptanalysis automation with Constraint Programming:

- Negative models, precise, cost 2^{2n}.
- Positive models, less precise (false positives, as we just saw, or false negatives, as we will see) but very efficient.

Our goal: improve state of the art of automation of ID Cryptanalysis in the second case. What we do:

- Propose a model for finding Impossible Differential taking into account both direct and indirect transitions.
- Improve the key recovery automation by inserting in the model the hash table method for improved efficiency.

Impossible Differential Cryptanalysis

Idea: exploit differentials with probability 0

Differential Cryptanalysis	ID Cryptanalysis
$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert$ high	$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert = 0$

Impossible Differential Cryptanalysis

Idea: exploit differentials with probability 0

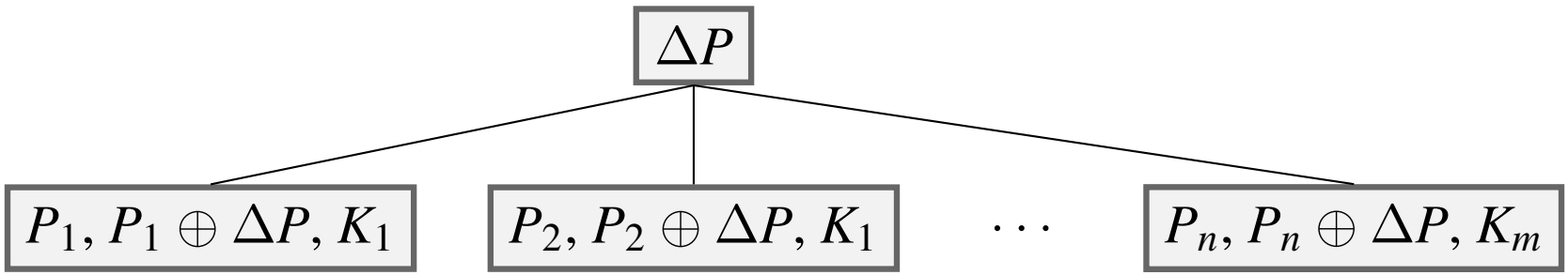
Differential Cryptanalysis	ID Cryptanalysis
$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert$ high	$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert = 0$

$$\boxed{\Delta P}$$

Impossible Differential Cryptanalysis

Idea: exploit differentials with probability 0

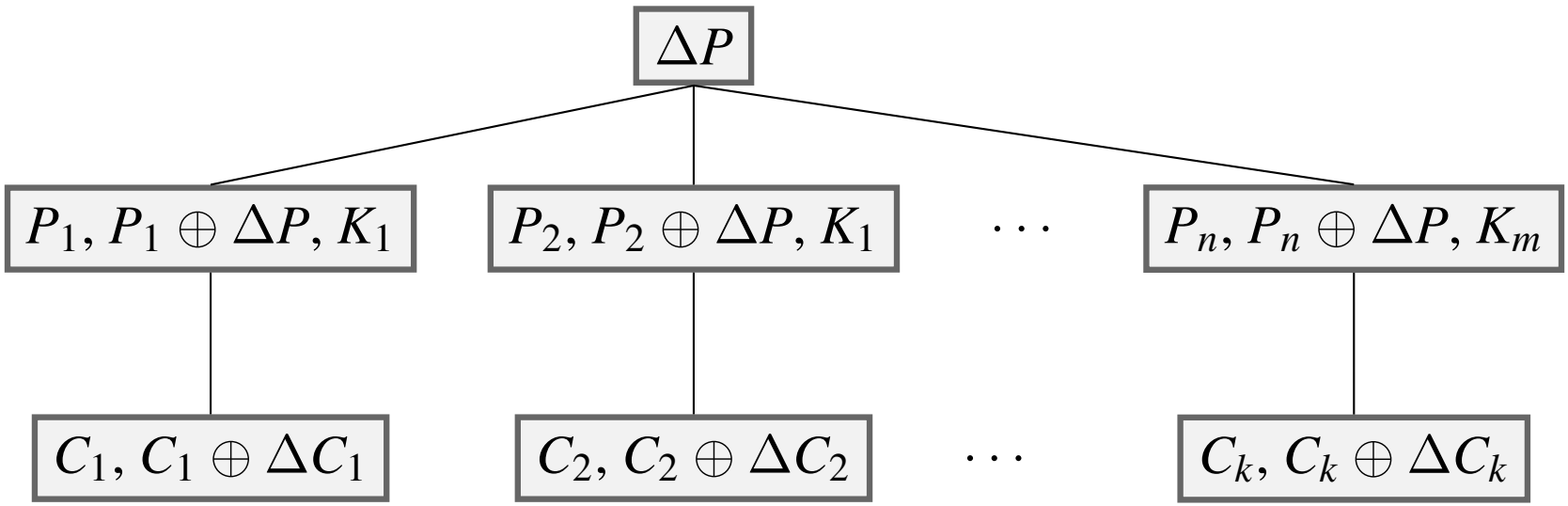
Differential Cryptanalysis	ID Cryptanalysis
$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert$ high	$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert = 0$



Impossible Differential Cryptanalysis

Idea: exploit differentials with probability 0

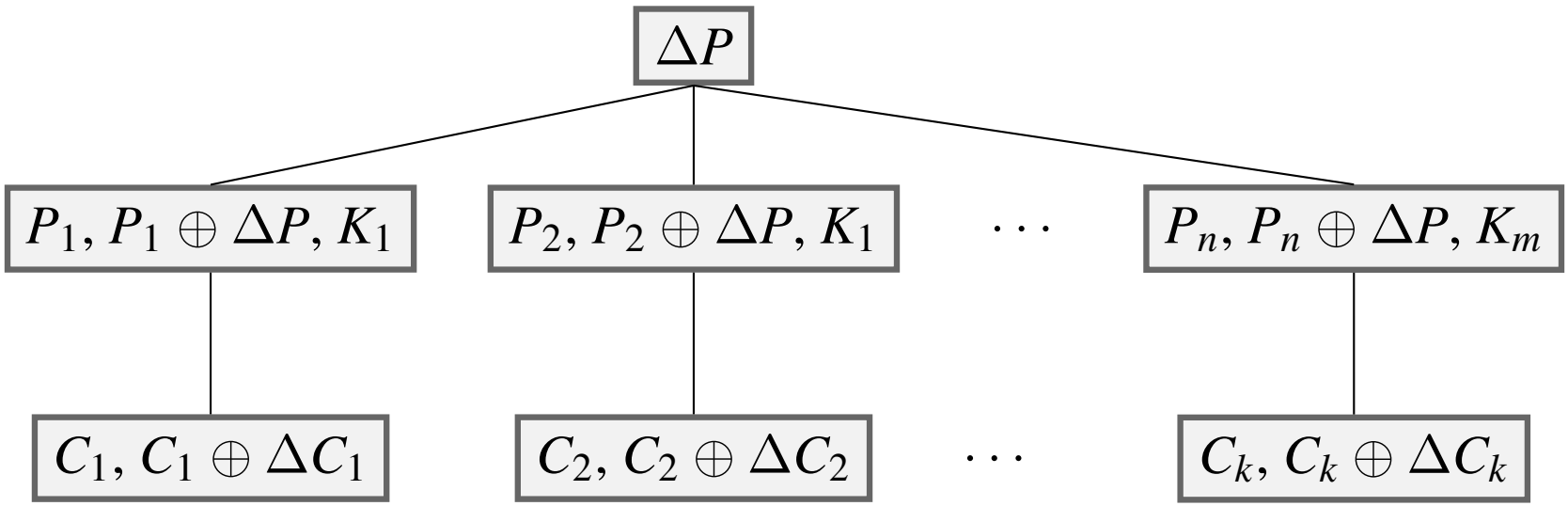
Differential Cryptanalysis	ID Cryptanalysis
$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert$ high	$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert = 0$



Impossible Differential Cryptanalysis

Idea: exploit differentials with probability 0

Differential Cryptanalysis	ID Cryptanalysis
$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert$ high	$\lvert\{P, K \; s.t. \; E(P,K) \oplus E(P \oplus \Delta P, K) = \Delta C\}\rvert = 0$



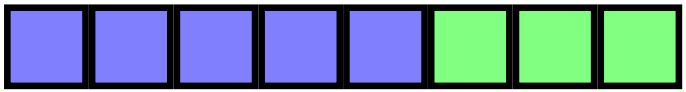
$$\Delta C \notin \{\Delta C_1, \ldots, \Delta C_k\} \implies \Delta P \nrightarrow \Delta C$$

Deterministic propagation

Any state bit difference can be either 0, 1 or unknown.

Deterministic propagation

Any state bit difference can be either 0, 1 or unknown.

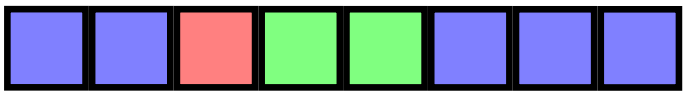
R 1 

R 2

R 3

R 3

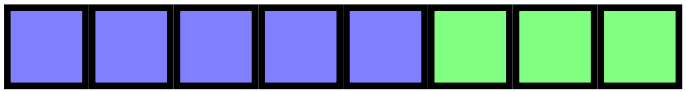
R 4

R 5 

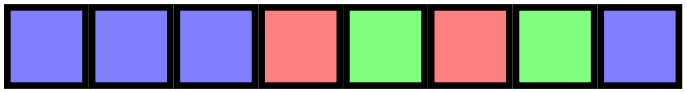
Deterministic propagation

Any state bit difference can be either 0, 1 or unknown.

R 1



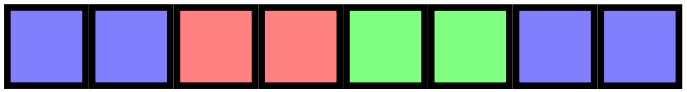
R 2



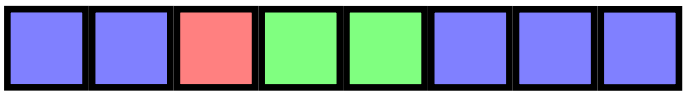
R 3

R 3

R 4

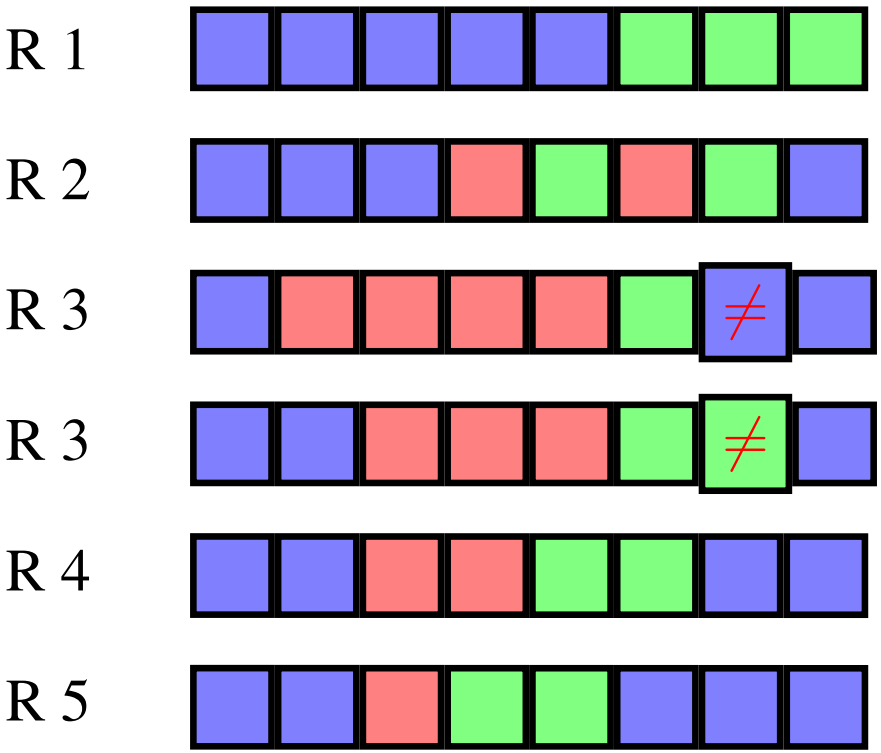


R 5



Deterministic propagation

Any state bit difference can be either 0, 1 or unknown.



Bitwise automatic modeling

Bit variables modeling:

- $0 \longrightarrow 0$
- $1 \longrightarrow 1$
- $\mathit{Unknown} \longrightarrow 2$

Bitwise automatic modeling

Bit variables modeling:

- $0 \longrightarrow 0$
- $1 \longrightarrow 1$
- $\mathit{Unknown} \longrightarrow 2$

Transition rules for bit operations, e.g.

$$AND(a,b) = \begin{cases} 0 & \text{if } \Delta_a = \Delta_b = 0 \\ 2 & \text{otherwise} \end{cases}$$

Wordwise automatic modeling

Word variables modeling: differential pattern δ + difference ζ (Sun et al., 2020)

Wordwise automatic modeling

Word variables modeling: differential pattern δ + difference ζ (Sun et al., 2020)

$$\delta X = \begin{cases} 0 & \text{if } \Delta X = 0 \\ 1 & \text{if } \Delta X \text{ is fixed, nonzero} \\ 2 & \text{if } \Delta X \text{ is nonzero} \\ 3 & \text{if } \Delta X \text{ is unknown} \end{cases}$$

Wordwise automatic modeling

Word variables modeling: differential pattern δ + difference ζ (Sun et al., 2020)

$$\delta X = \begin{cases} 0 & \text{if } \Delta X = 0 \\ 1 & \text{if } \Delta X \text{ is fixed, nonzero} \\ 2 & \text{if } \Delta X \text{ is nonzero} \\ 3 & \text{if } \Delta X \text{ is unknown} \end{cases}$$

$$\zeta X \in \begin{cases} \{0\} & \text{if } \delta X = 0 \\ \{1, \ldots, 2^{s-1}\} & \text{if } \delta X = 1 \\ \{-1\} & \text{if } \delta X = 2 \\ \{-2\} & \text{if } \delta X = 3 \end{cases}$$

The hybrid model

- Bitwise model: lose track of nonzero groups of unknown bits (*Unknown Nonzero* $\longrightarrow$????)

The hybrid model

- Bitwise model: lose track of nonzero groups of unknown bits (*Unknown Nonzero* $\longrightarrow$????)
- Wordwise model: lose granularity of difference knowledge (??0? $\longrightarrow$ *Fully Unknown*)

The hybrid model

- Bitwise model: lose track of nonzero groups of unknown bits (*Unknown Nonzero* $\longrightarrow$????)
- Wordwise model: lose granularity of difference knowledge (??0? $\longrightarrow$ *Fully Unknown*)

The hybrid model captures both properties in the bit representation:

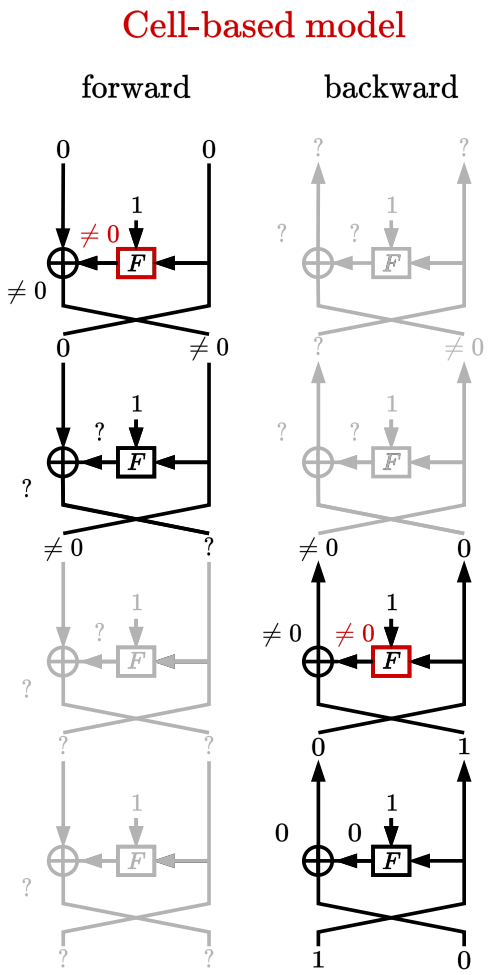
The hybrid model

- Bitwise model: lose track of nonzero groups of unknown bits (*Unknown Nonzero* $\longrightarrow$????)
- Wordwise model: lose granularity of difference knowledge (??0? $\longrightarrow$ *Fully Unknown*)

The hybrid model captures both properties in the bit representation:

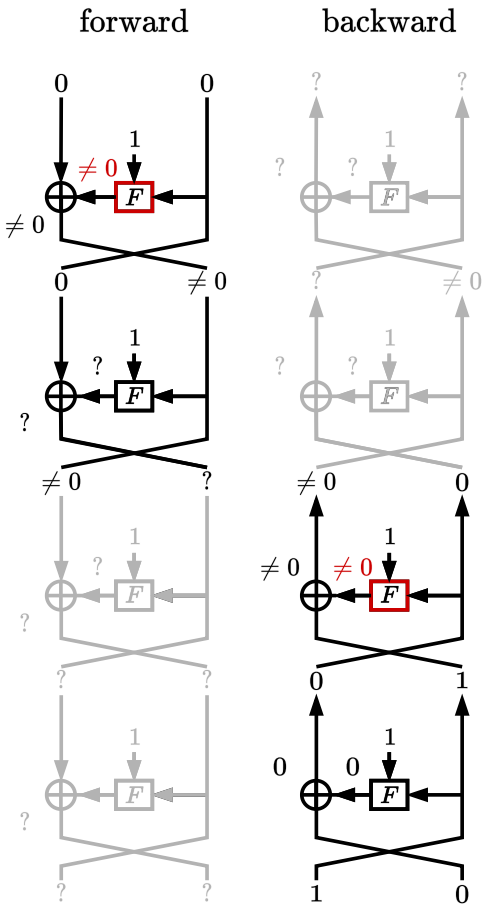
$$\delta X = \begin{cases} 0 & \text{if } \Delta X_{i,r} = 0 \\ 1 & \text{if } \Delta X_{i,r} = 1 \\ 2 & \text{if } \Delta X_{i,r} \text{ is unknown} \\ id_{s,r'} & \text{if } \Delta X_{i,r} \text{ is produced by S-box } s \text{ of round } r', \\ & \text{evaluated on a nonzero input difference} \end{cases}$$

An example

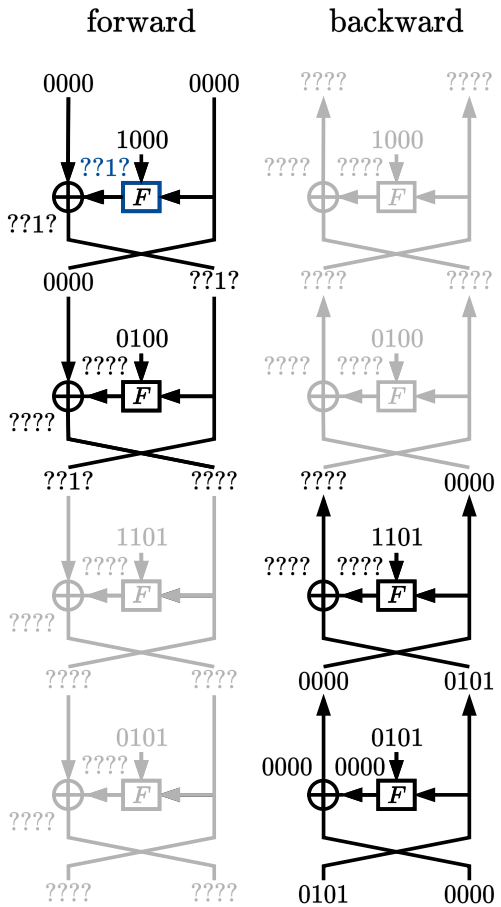


An example

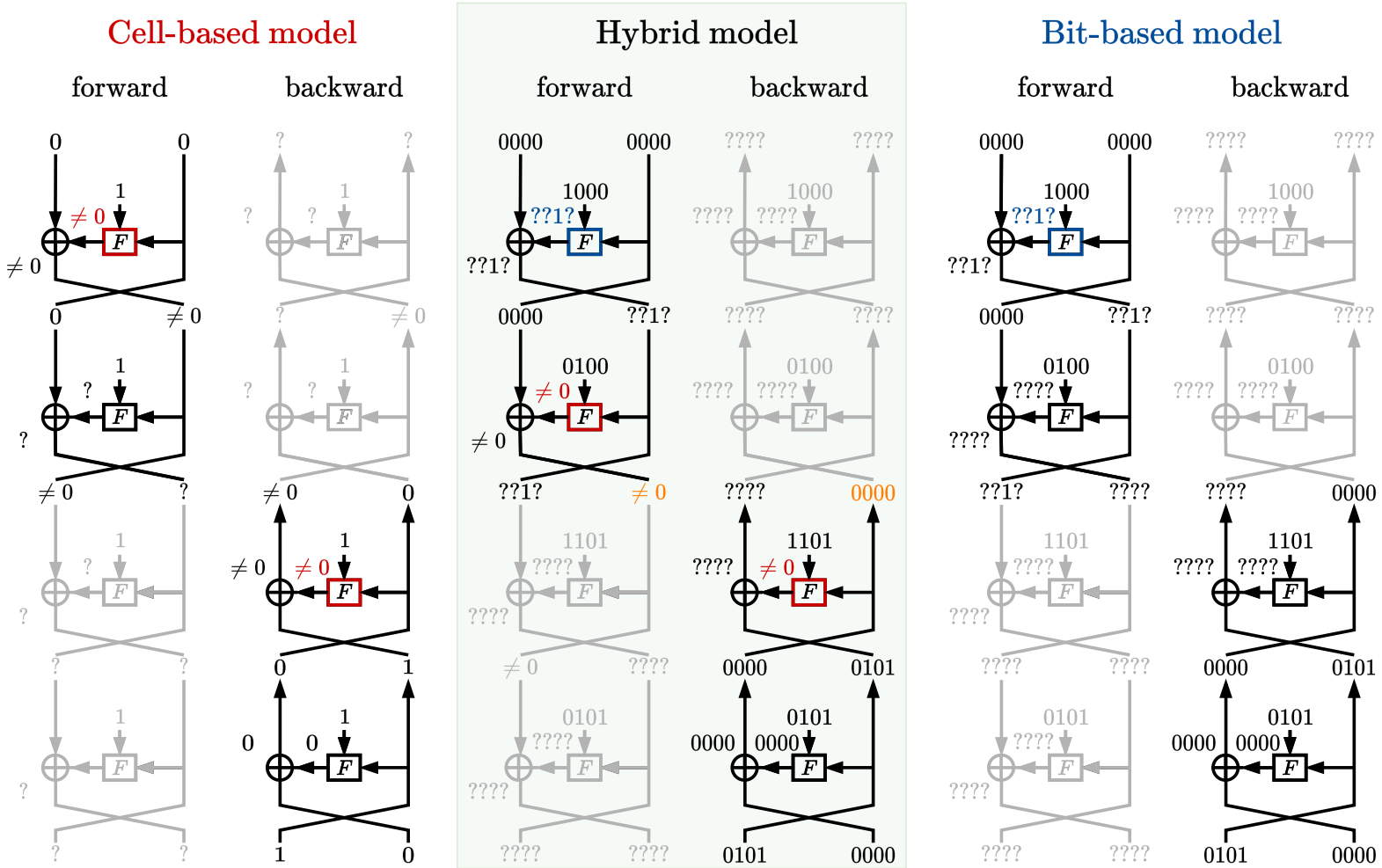
Cell-based model



Bit-based model



An example


Cell-based model
forward
backward
Hybrid model
forward
backward
Bit-based model
forward
backward

LBLOCK

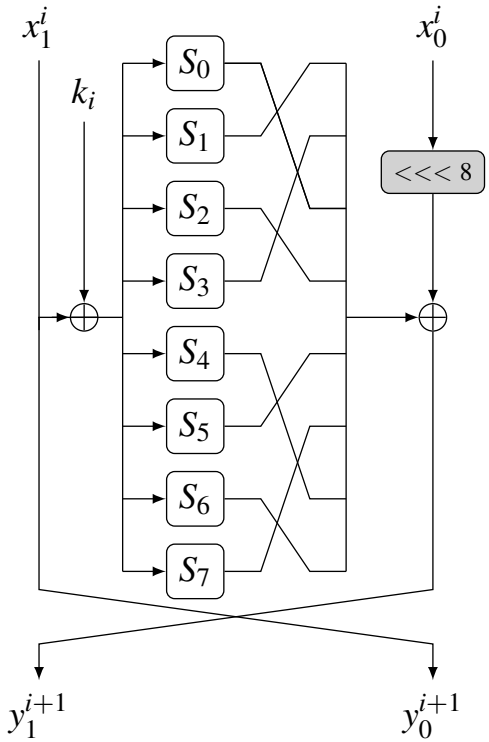
The hybrid model showed its best performances on LBLOCK.

LBLOCK

The hybrid model showed its best performances on LBLOCK.
LBLOCK is a lightweight block cipher with a Feistel structure, made of an SBox layer, a word permutation layer, a rotation and an internal XOR.

LBLOCK

The hybrid model showed its best performances on LBLOCK. LBLOCK is a lightweight block cipher with a Feistel structure, made of an SBox layer, a word permutation layer, a rotation and an internal XOR.



Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.

Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.
- The first 18-round improbable differential in a weak-key scenario was found with this model, which holds with probability $2^{-0.83}$ for an average of 2^{45} keys.

Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.
- The first 18-round improbable differential in a weak-key scenario was found with this model, which holds with probability $2^{-0.83}$ for an average of 2^{45} keys.

Weak-key scenario: allow some non deterministic transition in the key schedule .

Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.
- The first 18-round improbable differential in a weak-key scenario was found with this model, which holds with probability $2^{-0.83}$ for an average of 2^{45} keys.

Weak-key scenario: allow some non deterministic transition in the key schedule .

- Keep the trails which are impossible with a probability within a fixed threshold.

Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.
- The first 18-round improbable differential in a weak-key scenario was found with this model, which holds with probability $2^{-0.83}$ for an average of 2^{45} keys.

Weak-key scenario: allow some non deterministic transition in the key schedule .

- Keep the trails which are impossible with a probability within a fixed threshold.
- Group together the trails corresponding to a same impossible differential.

Hybrid model results

Results:

- The hybrid model retrieves new 16-rounds impossible differentials.
- The first 18-round improbable differential in a weak-key scenario was found with this model, which holds with probability $2^{-0.83}$ for an average of 2^{45} keys.

Weak-key scenario: allow some non deterministic transition in the key schedule .

- Keep the trails which are impossible with a probability within a fixed threshold.
- Group together the trails corresponding to a same impossible differential.

ID attack (Biham et al., 1998, Knudsen, 1998

Round 1

Round 2

Round 3

ID $\Delta P \nrightarrow \Delta C$

Round 3

Round 4

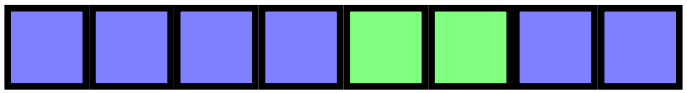
Round 5

ID attack (Biham et al., 1998, Knudsen, 1998

Round 1

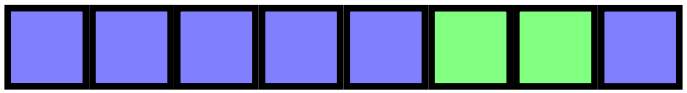
Round 2

Round 3



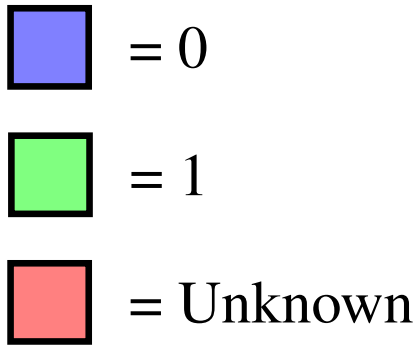
ID $\Delta P \nrightarrow \Delta C$

Round 3

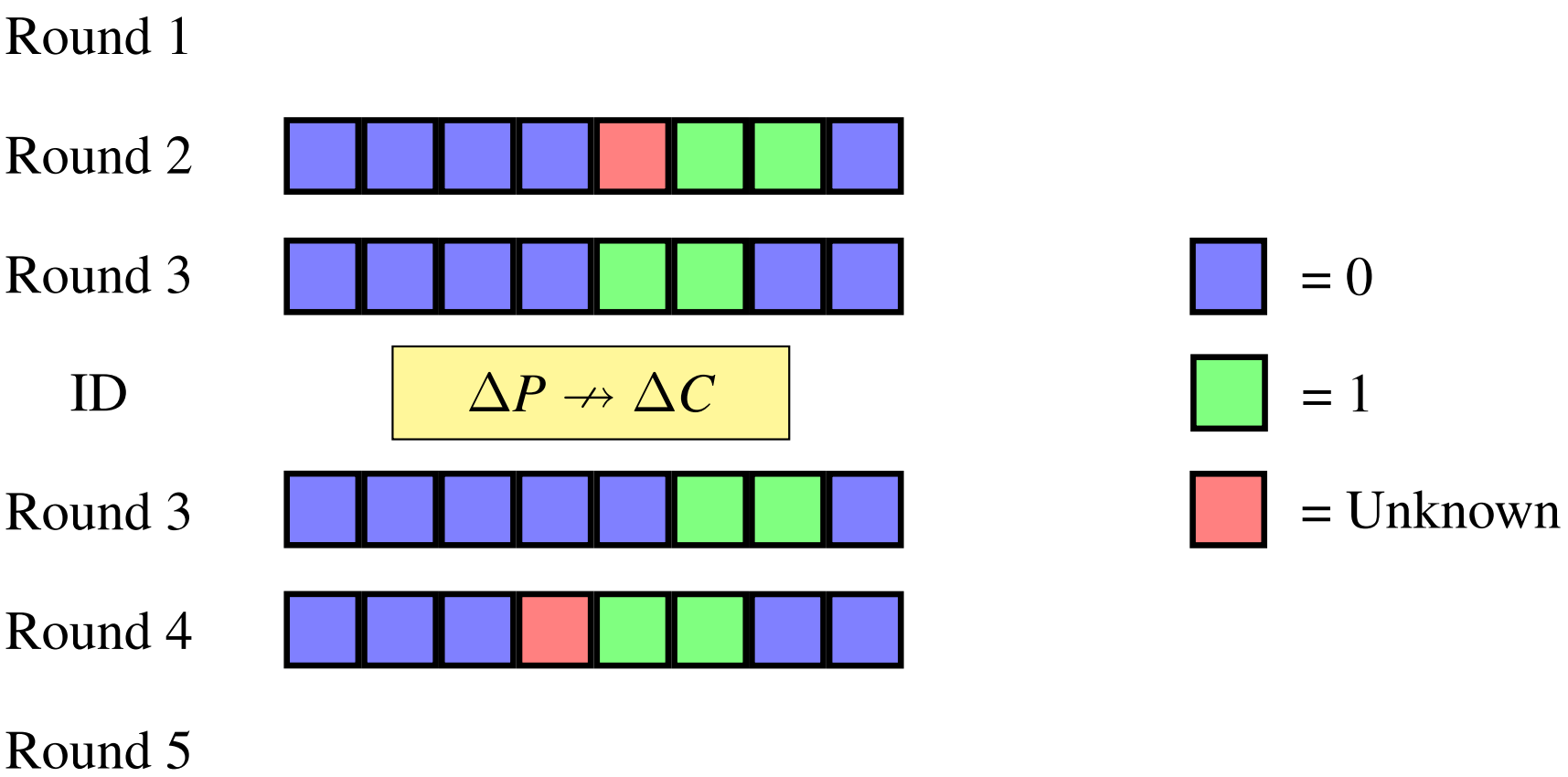


Round 4

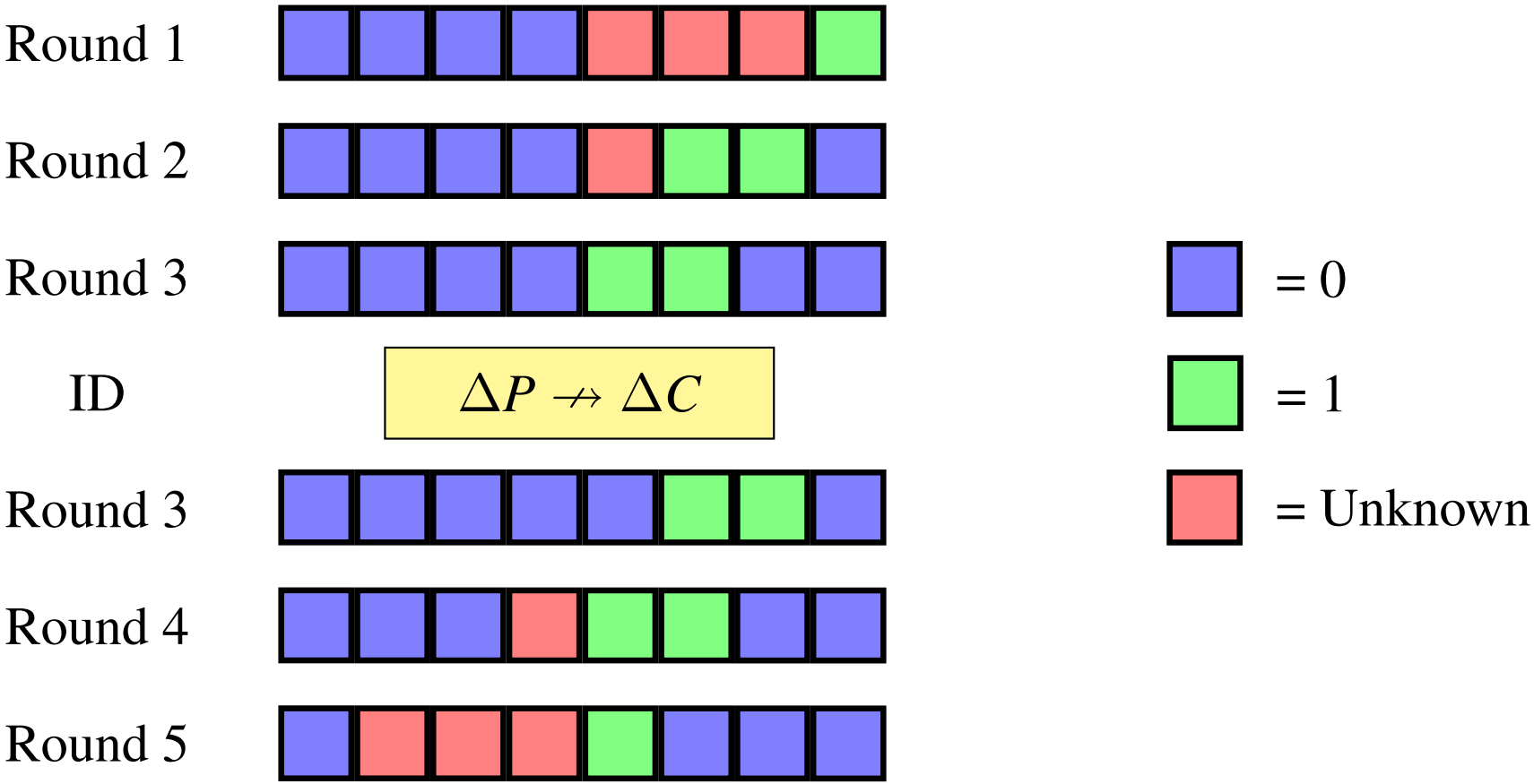
Round 5



ID attack (Biham et al., 1998, Knudsen, 1998



ID attack (Biham et al., 1998, Knudsen, 1998



ID attack

ID attack procedure:

1. Find an impossible differential
2. Compute its deterministic extensions
3. Compute plaintext/ciphertext pairs satisfying the plaintext/ciphertext differences

ID attack

ID attack procedure:

1. Find an impossible differential
2. Compute its deterministic extensions
3. Compute plaintext/ciphertext pairs satisfying the plaintext/ciphertext differences
4. Guess the key bits involved in computing the extensions

ID attack

ID attack procedure:

1. Find an impossible differential
2. Compute its deterministic extensions
3. Compute plaintext/ciphertext pairs satisfying the plaintext/ciphertext differences
4. Guess the key bits involved in computing the extensions
5. Filter out key guesses which can lead to the impossible differential

ID attack

ID attack procedure:

1. Find an impossible differential
2. Compute its deterministic extensions
3. Compute plaintext/ciphertext pairs satisfying the plaintext/ciphertext differences
4. Guess the key bits involved in computing the extensions
5. Filter out key guesses which can lead to the impossible differential
6. Exhaustively search the remaining keys

Hash tables for attack efficiency

Hash tables can be used to improve the complexity of the ID attack (Chen et al., 2011):

Hash tables for attack efficiency

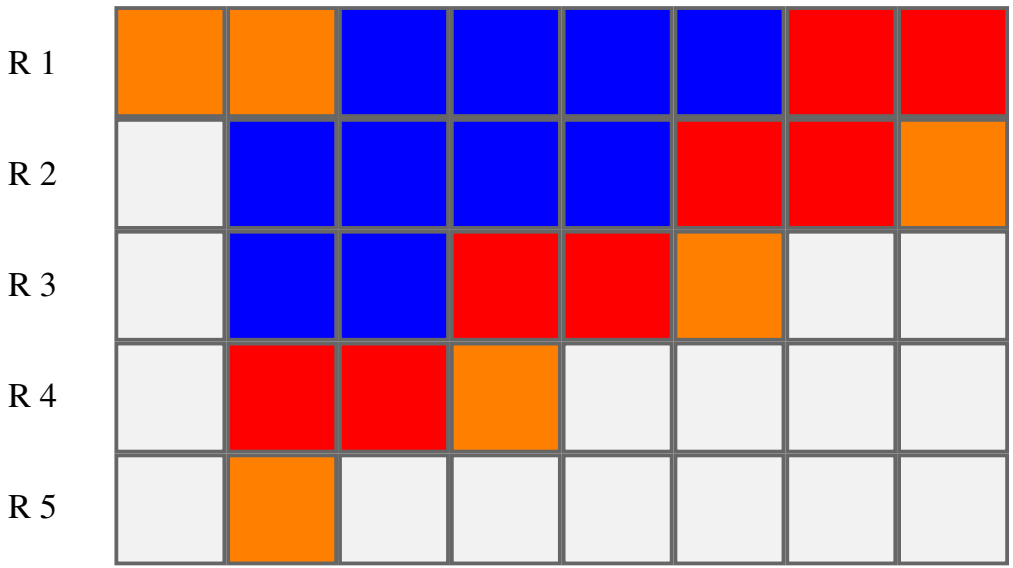
Hash tables can be used to improve the complexity of the ID attack (Chen et al., 2011):

1. exhaustively precompute portions of the impossible differential extensions
2. access the tables to get the correct key guesses when getting to the table key

Hash tables for attack efficiency

Hash tables can be used to improve the complexity of the ID attack (Chen et al., 2011):

1. exhaustively precompute portions of the impossible differential extensions
2. access the tables to get the correct key guesses when getting to the table key



HIGHT

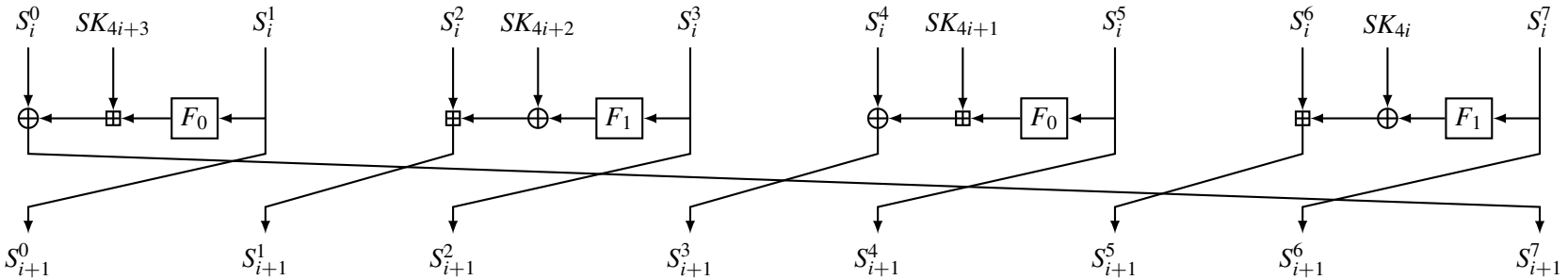
The hash tables attack technique was successfully used on HIGHT.

HIGHT

The hash tables attack technique was successfully used on HIGHT. HIGHT is a lightweight ARX block cipher with a substate structure similar to a Feistel.

HIGHT

The hash tables attack technique was successfully used on HIGHT. HIGHT is a lightweight ARX block cipher with a substate structure similar to a Feistel.



Improved results

Our automatic model for the attack with the hash tables support resulted in an improved 27 rounds attack to the HIGHT block cipher

Scenario	Rounds	Transformations	Time	Data	Memory
Single-key	18 (1-18)	Both	$2^{109.2}$ enc.	$2^{46.8}$ plaintexts	/
Single-key	25 (6-30)	Only final	$2^{126.75}$ enc.	2^{60} plaintexts	/
Single-key	26 (1-26)	Only final	$2^{119.53}$ enc.	2^{61} plaintexts	2^{109} B
Single-key	27 (4-30)	Both	$2^{126.6}$ enc.	2^{58} plaintexts	2^{120} B
Single-key	**27 (4-30)**	**Both**	$2^{124.5}$ **enc.**	2^{60} **plaintexts**	2^{116} **B**

Implementation

Implementation framework: CLAASP library.

Implementation challenges and improvements:

- Cipher inversion
- New hybrid model
- Hash tables inclusion
- Compliance with SAT, MILP and CP formalisms

Summing up

Our improvements:

- Automatic generation of a cipher's inverse in the CLAASP framework.

Summing up

Our improvements:

- Automatic generation of a cipher's inverse in the CLAASP framework.
- Automatic generation of bit-based, cell-based and hybrid models for searching impossible differentials.

Summing up

Our improvements:

- Automatic generation of a cipher's inverse in the CLAASP framework.
- Automatic generation of bit-based, cell-based and hybrid models for searching impossible differentials.
- CP modeling of the ID attack including the hash tables efficiency improvement.

Summing up

Our improvements:

- Automatic generation of a cipher's inverse in the CLAASP framework.
- Automatic generation of bit-based, cell-based and hybrid models for searching impossible differentials.
- CP modeling of the ID attack including the hash tables efficiency improvement.
- First 18-rounds improbable differential for LBLOCK.

Summing up

Our improvements:

- Automatic generation of a cipher's inverse in the CLAASP framework.
- Automatic generation of bit-based, cell-based and hybrid models for searching impossible differentials.
- CP modeling of the ID attack including the hash tables efficiency improvement.
- First 18-rounds improbable differential for LBLOCK.
- Most efficient attack on 27-rounds HIGHT.

Thank you!

Any questions?