Cryptanalysis of RadioGatún

Thomas Fuhr [1] Thomas Peyrin [2]

[1]Direction Centrale de la Sécurité des Systèmes d'Information

[2]Ingenico

FSE 2009 - February 22-25 - Leuven

Outline

1. Description of RadioGatún
2. Symmetric differential cryptanalysis
3. Path search algorithm
4. Collision search algorithm

Hash functions - Definition and security

Definition

A *hash function* is a function $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^n$

Security against...

- Collision attacks: find $M \neq M'$ s.t. $\mathcal{H}(M) = \mathcal{H}(M')$
- 2^{nd}-preimage attacks: given M, find $M' \neq M$ s.t. $\mathcal{H}(M) = \mathcal{H}(M')$
- Preimage attacks: given h, find M s.t. $\mathcal{H}(M) = h$

Hash functions - Definition and security

Definition

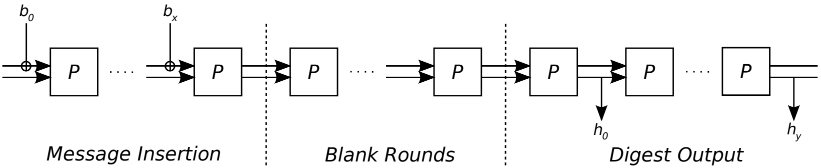
A *hash function* is a function $\mathcal{H} : \{0,1\}^* \rightarrow \{0,1\}^n$

Security against...

- Collision attacks: find $M \neq M'$ s.t. $\mathcal{H}(M) = \mathcal{H}(M')$
- 2^{nd}-preimage attacks: given M, find $M' \neq M$ s.t. $\mathcal{H}(M) = \mathcal{H}(M')$
- Preimage attacks: given h, find M s.t. $\mathcal{H}(M) = h$

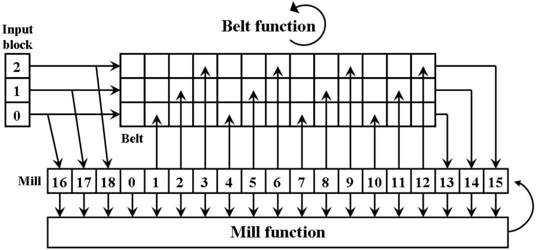
Overview of RadioGatún

- A family of *stream-oriented* hash functions
- Designed by Bertoni *et al.* (2006)
- Based on a *round permutation* of a large internal state
- Parameters: w (size of variables), n (digest length)
  - Notation: RadioGatún[w]
  - Usually 32 or 64
  - *Word*: w-bit variable
- Three stages

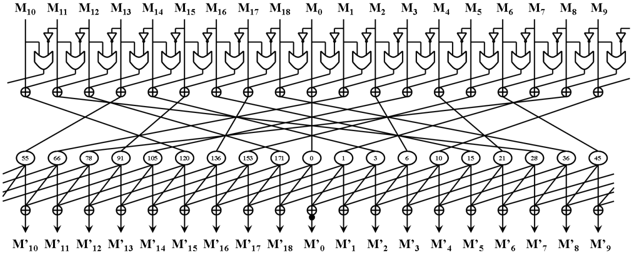


The *belt-and-mill* structure

- State (58 words) = Belt (3×13 words) + Mill (19 words)
- Message block: 3 words
- Mill to belt and belt to mill x-ors
- Rotation of the belt
- Nonlinear update of the mill



The *mill* function



- 5 steps, the first one is nonlinear
- Permutation, rotation, diffusion and disymmetry

Security claims and previous results on RadioGatún

- Maximum digest size: $19w$
  - Collisions: birthday bound in $2^{9.5w}$
- Best generic collision search: $2^{27.5w}$
- Bouillaguet and Fouque: $2^{24.5}$ hash computations for RadioGatún[1] (SAC2008)
- Khovratovich (2008): semi-free-start collisions in 2^{18w}

Our attack

- Collision on the internal state before the blank rounds
- A symmetric differential path
- Independent from w
- Collision search complexity: 2^{11w} computations of the state update function
- A 148-block collision for RadioGatún[2]

Differential cryptanalysis

- Choose equal-length message pairs $\{M, M'\}$ with a specific *difference*
    - Our paper: x-or difference
- Find a *differential path*
    - Probabilistic propagation through elementary operations
    - For each pair of equivalent variables: a set of admissible differences
    - Succession of admissible differences = differential path
    - No difference on the digests

RadioGatún and differential cryptanalysis

- RadioGatún properties:
  - Blank rounds → No freedom degrees to control difference propagation
  - Large internal state → No easy automated search for differential path
  - Shorter digests → Security margin on the internal state

Symmetric differential cryptanalysis

- A tool introduced by Bertoni *et al.*
- Restriction to a *linear subspace* of the differential path space
- Improving a probabilistic search for a differential path
- For each word: no difference, or differences on all bits

X	X'	$X \oplus X'$	Δ_X
01100011	01100011	00000000	0^w
10100110	01011001	11111111	1^w
01011010	11001100	10010110	$\perp$

Symmetric differential propagation for RadioGatún

- Deterministic differential propagation through linear functions
- Nonlinear part of the mill: $c = a \vee \bar{b}$

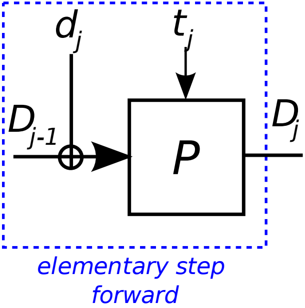
Δ_a	Δ_b	$\Delta_{a \vee \bar{b}}$	Probability	Condition
0^w	0^w	0^w	1	
0^w	1^w	0^w	2^{-w}	$a = 1^w$
0^w	1^w	1^w	2^{-w}	$a = 0^w$
1^w	0^w	0^w	2^{-w}	$b = 0^w$
1^w	0^w	1^w	2^{-w}	$b = 1^w$
1^w	1^w	0^w	2^{-w}	$a \oplus b = 0^w$
1^w	1^w	1^w	2^{-w}	$a \oplus b = 1^w$

Differential path search

- Meet-in-the-middle technique to find a path
- Elimination of too complex paths
  - Computation of a list of differential transitions for the mill function
  - Use of the *entropy* to evaluate the path complexity

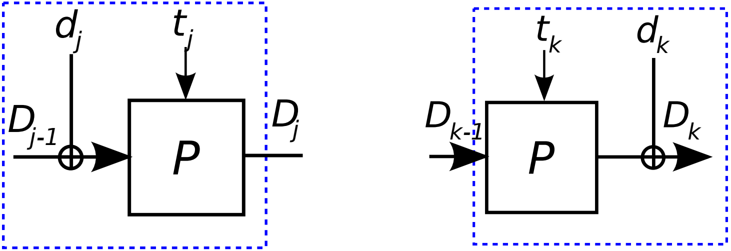
Differential path search

- Computation of 2^{27} forward paths
  - Width-first search
- Depth-first search for a matching backward path
  - Collision on a 55-bit variable
  - Cost : $2^{55-27} = 2^{28}$



Differential path search

- Computation of 2^{27} forward paths
  - Width-first search
- Depth-first search for a matching backward path
  - Collision on a 55-bit variable
  - Cost : $2^{55-27} = 2^{28}$

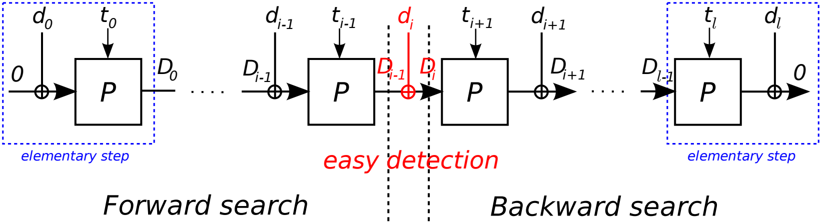


elementary step forward

elementary step backward

Differential path search

- Computation of 2^{27} forward paths
  - Width-first search
- Depth-first search for a matching backward path
  - Collision on a 55-bit variable
  - Cost : $2^{55-27} = 2^{28}$

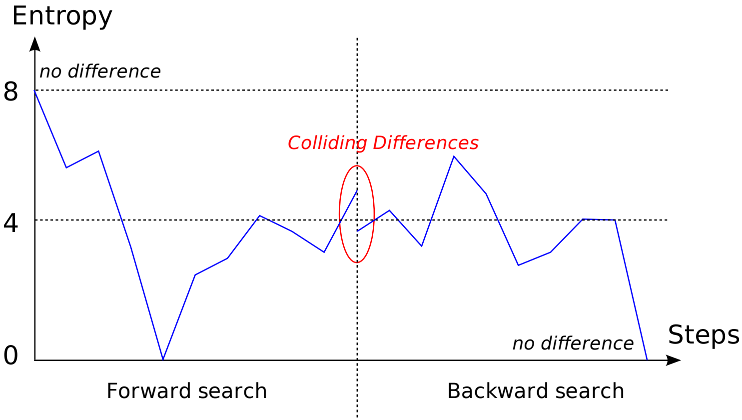


Entropy

- Evaluation of the path complexity
- Defined recursively from the last step of a differential path
- $H_k = max(H_{k+1} + c_k - 3, 0), H_\ell = 0$
- c_k conditions on the mill words before round permutation k
  - Logarithmic value of the expected number of prefixes of length k to get a collision
  - Computing forward: the expected number of available prefixes of length k (logarithmic value)
- No path with a maximum entropy below 8

Entropy bounds

- Backward search: maximum entropy of 8
- Forward search: entropy 8 at the starting point



Summary of the collision search algorithm

- Block per block computation of colliding messages
- Backtracking when no suitable block can be found
- Round k complexity:
  - $B_k \times P_k$
  - P_k: Number of prefixes of length k
  - B_k: Cost of the message blocks search

Message insertion and conditions

- Influence of message insertion k:
  - After message insertion, round k
  - After message insertion, round $k+1$
  - After message insertion, round $k+2$

Variable	M_0	$M_0 \oplus M_1$	M_1	$M_1 \oplus M_2$	M_2	$M_2 \oplus M_3$	M_3	$M_3 \oplus M_4$
Round	$k+2$	$k+1$	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$
Variable	M_4	$M_4 \oplus M_5$	M_5	$M_5 \oplus M_6$	M_6	$M_6 \oplus M_7$	M_7	$M_7 \oplus M_8$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	$k+2$
Variable	M_8	$M_8 \oplus M_9$	M_9	$M_9 \oplus M_{10}$	M_{10}	$M_{10} \oplus M_{11}$	M_{11}	$M_{11} \oplus M_{12}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+2$	$k+2$	$k+1$
Variable	M_{12}	$M_{12} \oplus M_{13}$	M_{13}	$M_{13} \oplus M_{14}$	M_{14}	$M_{14} \oplus M_{15}$	M_{15}	$M_{15} \oplus M_{16}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	k
Variable	M_{16}	$M_{16} \oplus M_{17}$	M_{17}	$M_{17} \oplus M_{18}$	M_{18}	$M_{18} \oplus M_0$		
Round	k	k	k	k	k	k		

- Conditions on these variables: not affected after message insertion k

Message insertion and conditions

- Influence of message insertion k:
  - After message insertion, round k
  - After message insertion, round $k+1$
  - After message insertion, round $k+2$

Variable	M_0	$M_0 \oplus M_1$	M_1	$M_1 \oplus M_2$	M_2	$M_2 \oplus M_3$	M_3	$M_3 \oplus M_4$
Round	$k+2$	$k+1$	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$
Variable	M_4	$M_4 \oplus M_5$	M_5	$M_5 \oplus M_6$	M_6	$M_6 \oplus M_7$	M_7	$M_7 \oplus M_8$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	$k+2$
Variable	M_8	$M_8 \oplus M_9$	M_9	$M_9 \oplus M_{10}$	M_{10}	$M_{10} \oplus M_{11}$	M_{11}	$M_{11} \oplus M_{12}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+2$	$k+2$	$k+1$
Variable	M_{12}	$M_{12} \oplus M_{13}$	M_{13}	$M_{13} \oplus M_{14}$	M_{14}	$M_{14} \oplus M_{15}$	M_{15}	$M_{15} \oplus M_{16}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	k
Variable	M_{16}	$M_{16} \oplus M_{17}$	M_{17}	$M_{17} \oplus M_{18}$	M_{18}	$M_{18} \oplus M_0$		
Round	k	k	k	k	k	k		

- Conditions on these variables: not affected after message insertion k

Message insertion and conditions

- Influence of message insertion k:
  - After message insertion, round k
  - After message insertion, round $k+1$
  - After message insertion, round $k+2$

Variable	M_0	$M_0 \oplus M_1$	M_1	$M_1 \oplus M_2$	M_2	$M_2 \oplus M_3$	M_3	$M_3 \oplus M_4$
Round	$k+2$	$k+1$	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$
Variable	M_4	$M_4 \oplus M_5$	M_5	$M_5 \oplus M_6$	M_6	$M_6 \oplus M_7$	M_7	$M_7 \oplus M_8$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	$k+2$
Variable	M_8	$M_8 \oplus M_9$	M_9	$M_9 \oplus M_{10}$	M_{10}	$M_{10} \oplus M_{11}$	M_{11}	$M_{11} \oplus M_{12}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+2$	$k+2$	$k+1$
Variable	M_{12}	$M_{12} \oplus M_{13}$	M_{13}	$M_{13} \oplus M_{14}$	M_{14}	$M_{14} \oplus M_{15}$	M_{15}	$M_{15} \oplus M_{16}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	k
Variable	M_{16}	$M_{16} \oplus M_{17}$	M_{17}	$M_{17} \oplus M_{18}$	M_{18}	$M_{18} \oplus M_0$		
Round	k	k	k	k	k	k		

- Conditions on these variables: not affected after message insertion k

Message insertion and conditions

- Influence of message insertion k:
  - After message insertion, round k
  - After message insertion, round $k+1$
  - After message insertion, round $k+2$

Variable	M_0	$M_0 \oplus M_1$	M_1	$M_1 \oplus M_2$	M_2	$M_2 \oplus M_3$	M_3	$M_3 \oplus M_4$
Round	$k+2$	$k+1$	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$
Variable	M_4	$M_4 \oplus M_5$	M_5	$M_5 \oplus M_6$	M_6	$M_6 \oplus M_7$	M_7	$M_7 \oplus M_8$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	$k+2$
Variable	M_8	$M_8 \oplus M_9$	M_9	$M_9 \oplus M_{10}$	M_{10}	$M_{10} \oplus M_{11}$	M_{11}	$M_{11} \oplus M_{12}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+2$	$k+2$	$k+1$
Variable	M_{12}	$M_{12} \oplus M_{13}$	M_{13}	$M_{13} \oplus M_{14}$	M_{14}	$M_{14} \oplus M_{15}$	M_{15}	$M_{15} \oplus M_{16}$
Round	$k+1$	$k+1$	$k+1$	$k+1$	$k+2$	$k+1$	$k+2$	k
Variable	M_{16}	$M_{16} \oplus M_{17}$	M_{17}	$M_{17} \oplus M_{18}$	M_{18}	$M_{18} \oplus M_0$		
Round	k	k	k	k	k	k		

- Conditions on these variables: not affected after message insertion k

Reduction of B_k

- A polynomial system in $3w$ variables
- Interesting conditions:
  - At round k: linear dependence on the message
  - At round $k+1$: bitwise dependence on the message for some conditions
- w 3-variable independent subsystems
- Overall complexity of the collision search algorithm:
  - Sum of the round complexities
  - Approximated by the *crowded round* complexity

Our Results

- A 143-block path
- The crowded round complexity: 2^{11w}
- A RadioGatún[2] collision that confirms the complexity analysis
- More details: `http://eprint.iacr.org/2008/515`

Towards breaking the designers' security claims ?

- Increasing the size of the path space
  - Use $(01)^{w/2}$ and $(10)^{w/2}$ differences
  - May lead to paths with a better complexity
  - Problem: the state space has $2^{2\times 55}$ elements
- Tradeoff: path search vs minimal complexity

Thank you for your attention