Practical Attacks on the Maelstrom-0 Compression Function

Stefan Kölbl and Florian Mendel

Graz University of Technology

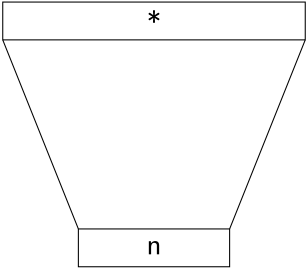
June 10th, 2011

Overview

- Cryptographic Hash Functions
- Maelstrom-0 Compression Function
- Differential Properties
- Attack on Maelstrom-0
- Results and Conclusion

Cryptographic Hash Functions

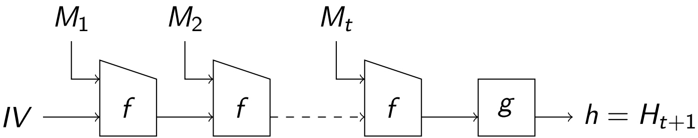
- Takes input of variable size and produces fixed size output



$$h\colon \{0,1\}^* \rightarrow \{0,1\}^n$$

Cryptographic Hash Functions

Iterative Construction



Security Properties

- Preimage Resistance: For a given output y find an input x' such that $y = h(x')$.
- Second Preimage Resistance: For given x and $y = h(x)$, find $x' \neq x$ such that $h(x') = y$.
- Collision Resistance: Find two distinct inputs x, x' such that $h(x) = h(x') = y$.

Security Properties

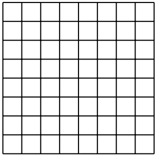
other non-random behaviour of interest

- semi-free-start collision: random chaining input, IV not fixed
- free-start collision: differences in the chaining input
- near-collision: difference in the output

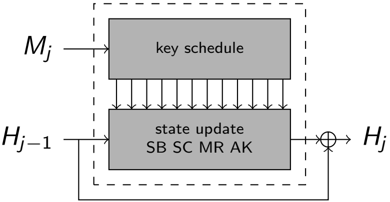
Maelstrom-0 Compression Function

Maelstrom-0 compression function

- tweaked version of Whirlpool which is standardized by ISO/IEC 10118-3:2003
- designed by Barreto, Filho and Rijmen
- designed to be faster and more robust
- byte-oriented using 8×8 states



Maelstrom-0 Compression Function



Maelstrom-0 compression function

- 10 rounds
- AES like round transformations are applied on the state
  - SubBytes: applies non-linear S-Box on every byte
  - ShiftColumn: rotates each column
  - MixRows: linear transformation for each row
  - AddKey: xors the round key to the state

Maelstrom-0 Key Schedule

Expands the 1024-bit key K by mapping it to two 8×8 states (K^{-2}, K^{-1}) and apply the following operations:

- $K^0 = K^{-2} \oplus K^{-1}$
- $K^1 = x^8 \cdot K^{-2} \oplus x^8 \cdot K^{-1} \oplus K^{-1}$
- adding of a round constant

For the actual round keys $SB \circ MR$ is applied to row 3 and 7

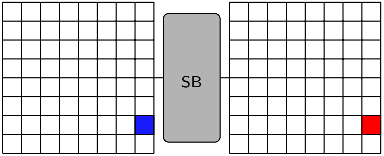
Differential Cryptanalysis

Basic idea of the attack

- observer how differences propagate through round transformations
- construct a differential path
- find a message following the path

Difference Propagation

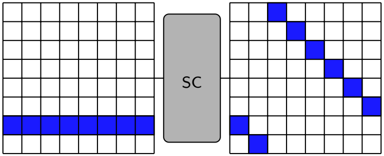
SubBytes:



- for a given input difference 101 possible output differences on average for the Whirlpool S-Box

Difference Propagation

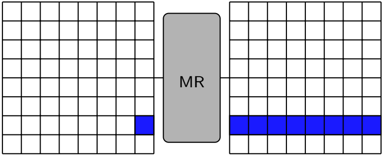
ShiftColumn:



- differences are rotated columnwise

Difference Propagation

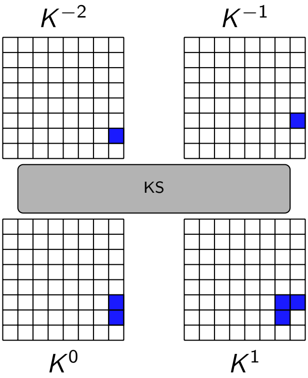
MixRows:



- one active byte will always propagate to 8 active bytes
- 8 active bytes can result in 1 to 8 active bytes
- probability for transition from a to b active bytes is in general $2^{(b-8)\cdot 8}$ for $a + b \geq 9$

Difference Propagation

KeySchedule:



- $K^0 = K^{-2} \oplus K^{-1}$
- $K^1 = x^8 \cdot K^{-2} \oplus x^8 \cdot K^{-1} \oplus K^{-1}$
- multiplication by x^8 equals bytewise rotation

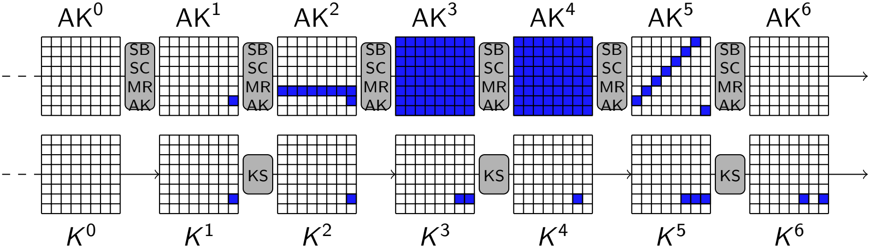
Attack on Maelstrom-0

The attack on the compression function can be split up into three parts

- construct the differential path
- determine the values of the differences
- construct a message following the path

Attack on Maelstrom-0

Differential path for 6 rounds



0 — 1 — 9 — 64 — 64 — 8 — 0

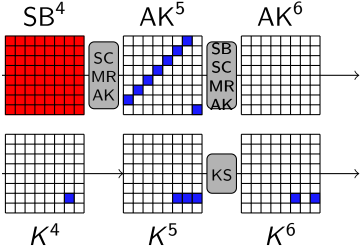
Attack on Maelstrom-0

Determine the differences

- same approach that has been used in the rebound attack on Whirlpool
- compute differences in forward and backward direction
- try to find a valid transition from AK^4 to SB^4

Attack on Maelstrom-0

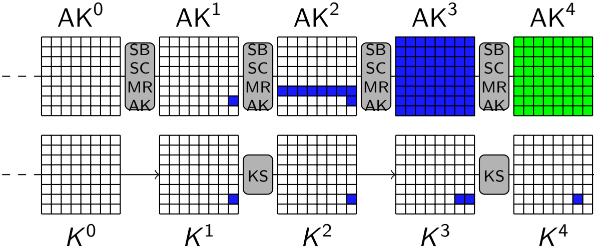
Backward direction



Values at SB^4 are fixed now

Attack on Maelstrom-0

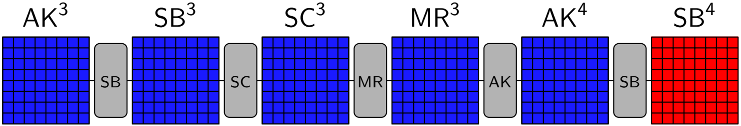
Forward direction



Values at AK^4 are fixed now

Attack on Maelstrom-0

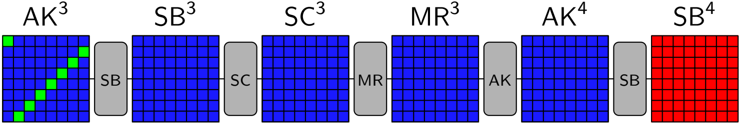
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

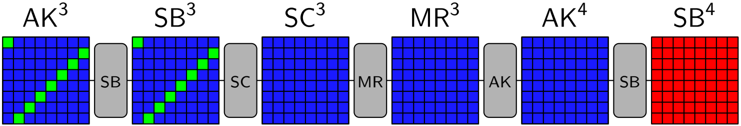
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

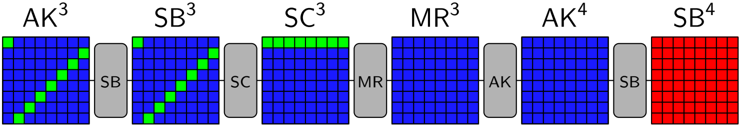
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

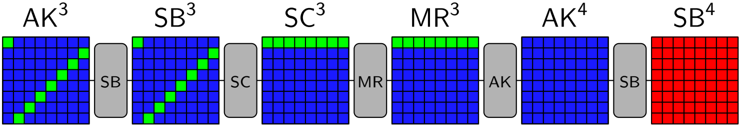
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

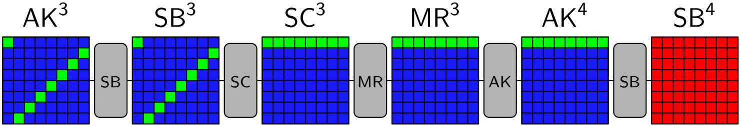
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

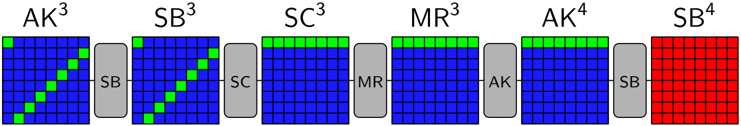
Finding the correct transition



- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Attack on Maelstrom-0

Finding the correct transition



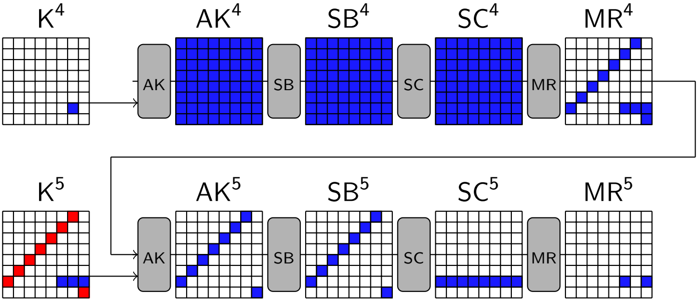
- Probability for one row is $2^{-10.72}$
- We can compute the rows individually

Complexity

$2^{13,72}$

Attack on Maelstrom-0

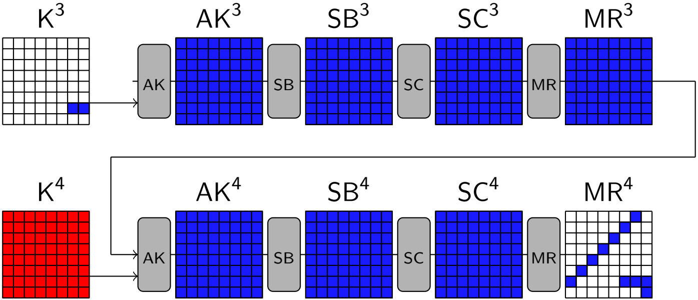
Constructing the message



Set values for SB^4 and use K^5 to correct the values for SB^5.

Attack on Maelstrom-0

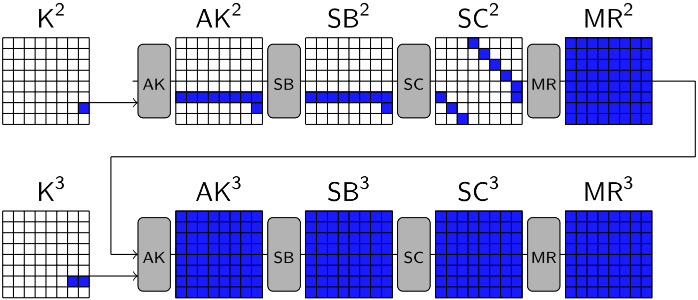
Constructing the message



Set values for SB^3 and use K^4 to correct the values for SB^4.

Attack on Maelstrom-0

Constructing the message

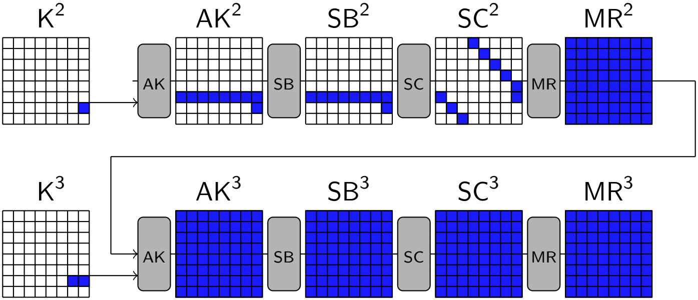


Apply inverse keyschedule to compute K^2 and K^3.

Use free bytes in K^5 to influence rows.

Attack on Maelstrom-0

Constructing the message



Apply inverse keyschedule to compute K^2 and K^3.
Use free bytes in K^5 to influence rows.

Complexity

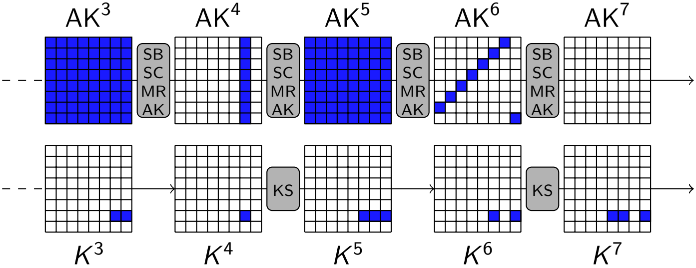
$\approx 2^{16} \cdot 2^8$

Colliding Message Pair

CV	0x62c411cf0e4eddeb	0x7e1f077cd784ae56	0xa48151b21e91d3fe	0x2308cd4ab8d482b9
	0x67891674f0e67d58	0x76e0faf9b68b019c	0x83d8d836e39e54f2	0x430c8558a09b3038
M_1	0x25fee7fa166f302b	0xc3038ed9793ad606	0x8e53d3da9b4133e0	0x66e6da065c9bf1f2
	0x311aff5ca1ac25cd	0x2f6e63a9840ed540	0x00c0d99f24ab7c20	0x1f2fd82fbcd2042a
	0x348c53c517b48735	0xe19c2ce81dfbdf80	0x973d460fee1d5d4b	0x635537c3de04888e
	0xb81392122cd28d8e	0xef3bfc5ab3446b7b	0xeff68042499a5dde	0x9f1bd8e9887fc473
M_2	0x25fee7fa166f302b	0xc3038ed9793ad606	0x8e53d3da9b4133e0	0x66e6da065c9bf1f2
	0x311aff5ca1ac25cd	0x2f6e63a9840ed540	0x00c0d99f24ab7c21	0x1f2fd82fbcd2042a
	0x348c53c517b48735	0xe19c2ce81dfbdf80	0x973d460fee1d5d4b	0x635537c3de04888e
	0xb81392122cd28d8e	0xef3bfc5ab3446b7b	0xeff68042499a5ddf	0x9f1bd8e9887fc473
H	0x6d85841532bdfc98	0xb6db1712edc5fe73	0xf5858ea793eab087	0xac8edab0e12082d8
	0x1532a861d53fbc93	0xbadd0a2bbb20871f	0x3245866ac24173df	0x3481634e4a1018a7

Attack on Maelstrom-0

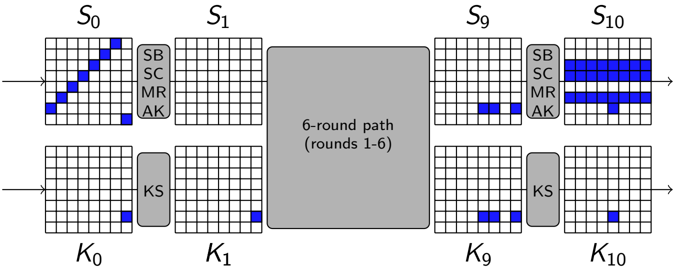
Collisions for 7 rounds



$$0 - 1 - 9 - 64 - 8 - 64 - 8 - 0$$

Attack on Maelstrom-0

extending the 6-round path



- appending 2-rounds to get near-collisions
- prepending 2-round to get free-start near-collisions

Summary

rounds	computational complexity	generic attack	type
6	2^{24}	2^{256}	semi-free-start collision
7	2^{128}	2^{256}	semi-free-start collision
8	2^{24}	2^{156}	semi-free-start near-collision
10	2^{24}	2^{124}	free-start near-collision

Conclusion

- The additional degrees of freedom in the key allows efficient attacks
  - practical collisions for 6 rounds
  - show non-random behaviour for full 10 rounds of the Maelstrom-0 compression function
- Future work
  - improvement of the attack on 7 rounds
  - attacks on the hash function

Thank you for your attention!