Collision Attacks on the Reduced Dual-Stream Hash Function RIPEMD-128

Florian Mendel[1], Tomislav Nad[2], Martin Schläffer[2]

Katholieke Universiteit Leuven, ESAT/COSIC and IBBT, Belgium

Graz University of Technology, IAIK, Austria

FSE 2012

Outline

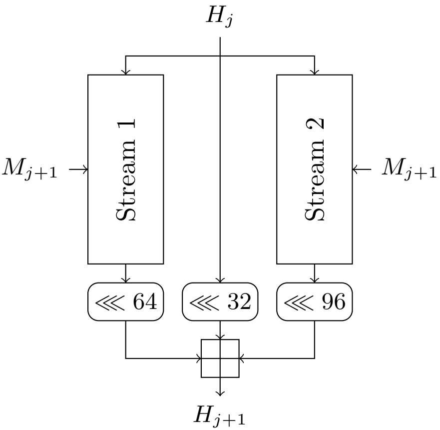
Outline

Motivation

- Cryptanalysis of ARX based designs is still important
- Very difficult without the right tools
- Even more for dual-stream hash functions
- Do the results on SHA-2 help to improve attacks on other designs?
- RIPEMD-128: shares some similarities with SHA-2

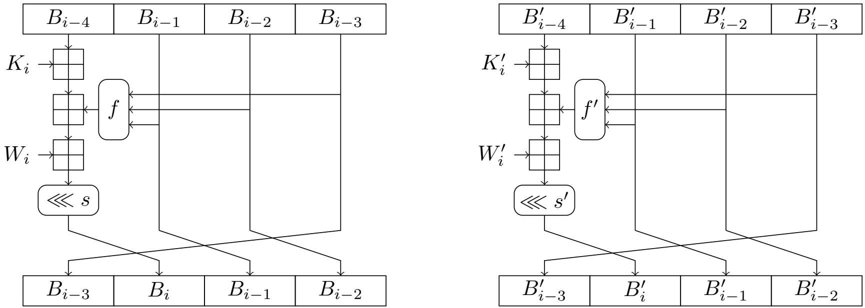
Outline

Description of RIPEMD-128



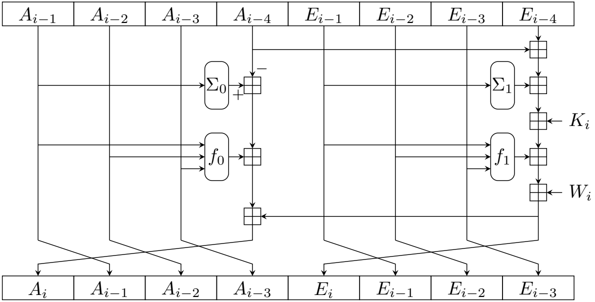
- ISO/IEC standard [DBP96]
- designed by Dobbertin, Bosselaers and Preneel
- iterated, Merkle-Damgård hash function
- dual stream compression function
- no output transformation
- 128-bit hash output

Step Update Transformation of RIPEMD-128



- one message word updates two state variables
- different message word permutations
- different rotation values and Boolean functions
- no interaction between streams (SHA-2: with interaction)
- 4 rounds of 16 steps

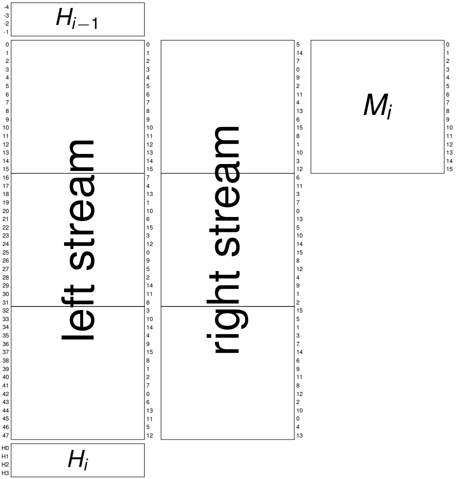
Step Update Transformation of RIPEMD-128



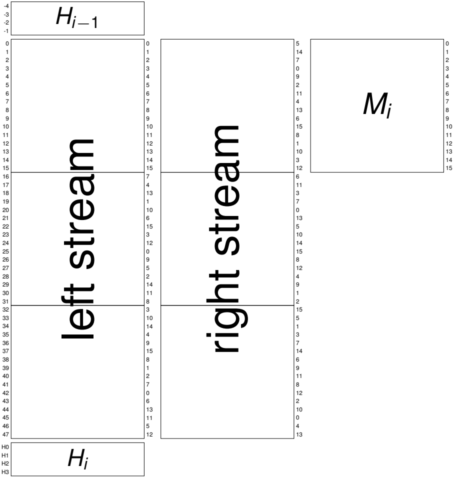
- one message word updates two state variables
- different message word permutations
- different rotation values and Boolean functions
- no interaction between streams (SHA-2: with interaction)
- 4 rounds of 16 steps

Outline

Overview of the Attack



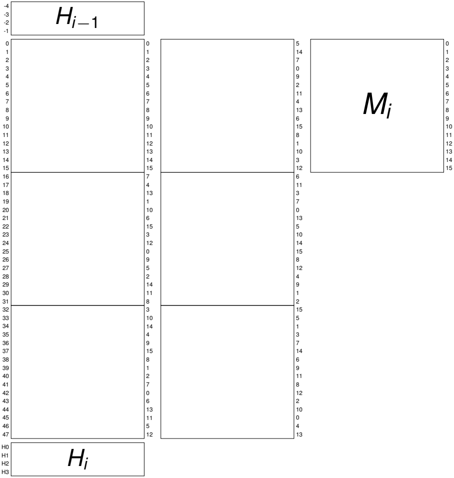
Overview of the Attack



1. choose a good starting point
   - few message word differences
   - high probability characteristic
2. search for a characteristics
   - very sparse in R2 and R3
   - sparse in one stream in R1
3. determine message pair
   - message modification in R1
   - exhaustive search for R2, R3

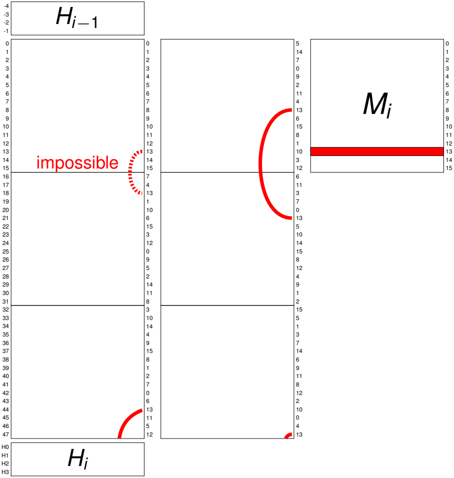
$\Rightarrow$ iterations between phases

Choosing a Starting Point



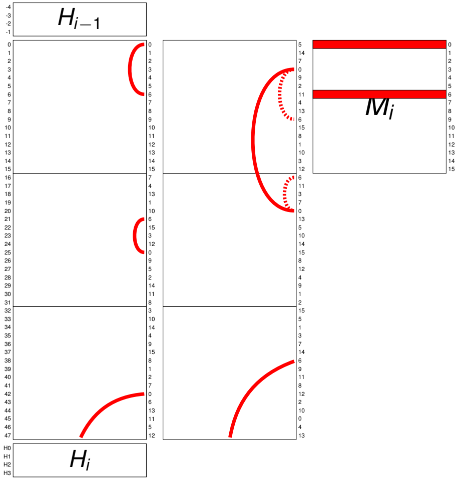
- which message words should contain differences?
  - as few words as possible
  - only words used late in R3
  - short local collisions in R2

Choosing a Starting Point



- which message words should contain differences?
  - as few words as possible
  - only words used late in R3
  - short local collisions in R2
- message word 13
  - single local collision (R1-R2)
  - impossible in left stream

Choosing a Starting Point



- which message words should contain differences?
  - as few words as possible
  - only words used late in R3
  - short local collisions in R2
- message word 13
  - single local collision (R1-R2)
  - impossible in left stream
- message word 0 and 6
  - left: two short local collisions
  - right: one long local collision
  - avoid overlapping of LCs
  - collision for 38 steps

Outline

Differences and Conditions

Generalized Conditions [DR06]

- take all 16 possible conditions on a pair of bits into account

(X_i, X_i^*)	(0,0)	(1,0)	(0,1)	(1,1)	(X_i, X_i^*)	(0,0)	(1,0)	(0,1)	(1,1)
?	✓	✓	✓	✓	3	✓	✓	-	-
-	✓	-	-	✓	5	✓	-	✓	-
x	-	✓	✓	-	7	✓	✓	✓	-
0	✓	-	-	-	A	-	✓	-	✓
u	-	✓	-	-	B	✓	✓	-	✓
n	-	-	✓	-	C	-	-	✓	✓
1	-	-	-	✓	D	✓	-	✓	✓
#	-	-	-	-	E	-	✓	✓	✓

2-bit Conditions [MNS11]

- linear relation between closely related bits: $X_i \oplus X_j = 0/1$
- 2-bit conditions on any generalized condition (-,x,?,...)
- used to determine critical bits (those with many relations)

Propagation of Differences and Conditions

- Stored conditions
  - all possible pairs on bits (generalized conditions)
  - all possible pairs on carries

Propagation of Differences and Conditions

- Stored conditions
  - all possible pairs on bits (generalized conditions)
  - all possible pairs on carries
- 2-bit conditions
  - all inputs and outputs of
  - Boolean functions
  - modular additions
  - even on carries (sign of carry)

Propagation of Differences and Conditions

- Stored conditions
  - all possible pairs on bits (generalized conditions)
  - all possible pairs on carries
- 2-bit conditions
  - all inputs and outputs of
  - Boolean functions
  - modular additions
  - even on carries (sign of carry)
- Efficiency
  - not all conditions in every iteration/phase
  - use table lookups when possible

Search Strategy

Search Algorithm [DR06, MNS11]

(1) Start with an unrestricted characteristic ('?' and '-')

(2) Successively impose new conditions on the characteristic
- path search: replace '?' by '-' and 'x' by 'n' or 'u'
- message search: replace '-' by '1' or '0'

(3) Propagate the conditions in a bitslice manner and check for consistency
- if a contradiction occurs then backtrack
- else proceed with step 2

(4) Repeat steps 2 and 3 until all bits of the characteristic are determined

Search Strategy

The difficulties are in the details...

- Which information to propagate (and when)?
  - path search: generalized conditions
  - message search: generalized conditions and 2-bit conditions
- Which bits (which area) to guess?
  - dedicated to hash function
  - bits with many 2-bit conditions (in message search)
  - lots of trial and error needed to find best strategy
- How to backtrack?
  - if a contradiction occurs on a bit, backtrack until bit can be set
  - keep and check a list of previous critical bits

Search Strategy

The difficulties are in the details...

- Which information to propagate (and when)?
  - path search: generalized conditions
  - message search: generalized conditions and 2-bit conditions
- Which bits (which area) to guess?
  - dedicated to hash function
  - bits with many 2-bit conditions (in message search)
  - lots of trial and error needed to find best strategy
- How to backtrack?
  - if a contradiction occurs on a bit, backtrack until bit can be set
  - keep and check a list of previous critical bits

$\Rightarrow$ Dedicated for every hash function (unfortunately)

Searching for a Differential Characteristic

i	∇B_i	$\nabla B'_i$	∇m_i
-4	--------------------------------		
-3	--------------------------------		
-2	--------------------------------		
-1	--------------------------------		
0	????????????????????????????????	--------------------------------	???????????????????????????????x
1	????????????????????????????????	--------------------------------	--------------------------------
2	????????????????????????????????	--------------------------------	--------------------------------
3	--------------------------------	????????????????????????????????	--------------------------------
4	--------------------------------	????????????????????????????????	--------------------------------
5	--------------------------------	????????????????????????????????	--------------------------------
6	--------------------------------	????????????????????????????????	????????????????????????????????
7	--------------------------------	????????????????????????????????	--------------------------------
8	--------------------------------	????????????????????????????????	--------------------------------
9	--------------------------------	????????????????????????????????	--------------------------------
10	--------------------------------	????????????????????????????????	--------------------------------
11	--------------------------------	????????????????????????????????	--------------------------------
12	--------------------------------	????????????????????????????????	--------------------------------
13	--------------------------------	????????????????????????????????	--------------------------------
14	--------------------------------	????????????????????????????????	--------------------------------
15	--------------------------------	????????????????????????????????	--------------------------------
16	--------------------------------	????????????????????????????????	
17	--------------------------------	--------------------------------	
18	--------------------------------	--------------------------------	
19	--------------------------------	--------------------------------	
20	--------------------------------	--------------------------------	
21	????????????????????????????????	--------------------------------	
22	--------------------------------	--------------------------------	
23	--------------------------------	--------------------------------	
24	--------------------------------	--------------------------------	
25	--------------------------------	--------------------------------	
26	--------------------------------	--------------------------------	
27	--------------------------------	--------------------------------	
28	--------------------------------	--------------------------------	
29	--------------------------------	--------------------------------	
30	--------------------------------	--------------------------------	
31	--------------------------------	--------------------------------	
32	--------------------------------	--------------------------------	
33	--------------------------------	--------------------------------	
34	--------------------------------	--------------------------------	
35	--------------------------------	--------------------------------	
36	--------------------------------	--------------------------------	
37	--------------------------------	--------------------------------	

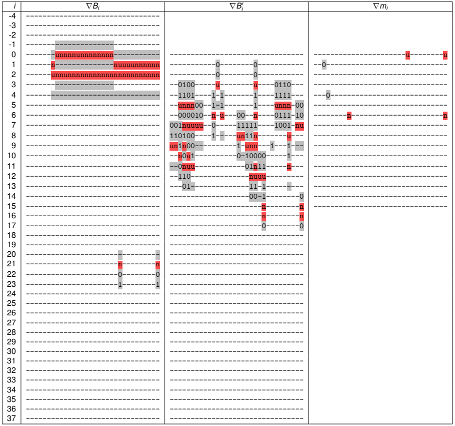
- Start characteristic
  - ? in words with difference
  - - in words without differences
  - x in LSB of word 0

Searching for a Differential Characteristic

i	∇B_i	$\nabla B'_i$	∇m_i
-4	--------------------------------		
-3	--------------------------------		
-2	--------------------------------		
-1	--------------------------------		
0	-------unnnnunnnnnnnn-----------	--------------------------------	----------------------u--------u
1	------n--------------nuuuunnnnnn	--------------------------------	--------------------------------
2	------unnunnnnnnnnnnnnnnnnnnnnnn	--------------------------------	--------------------------------
3	--------------------------------	-----------u--------u-----------	--------------------------------
4	--------------------------------	????????-----------?????????????	--------------------------------
5	--------------------------------	????????????????????????????????	--------------------------------
6	--------------------------------	????????????????????????????????	--------n----------------------n
7	--------------------------------	????????????????????????????????	--------------------------------
8	--------------------------------	????????????????????????????????	--------------------------------
9	--------------------------------	????????????????????????????????	--------------------------------
10	--------------------------------	????????????????????????????????	--------------------------------
11	--------------------------------	????????????????????????????????	--------------------------------
12	--------------------------------	????????????????????????????????	--------------------------------
13	--------------------------------	????????????????????????????????	--------------------------------
14	--------------------------------	????????????????????????????????	--------------------------------
15	--------------------------------	----------------------x--------x	--------------------------------
16	--------------------------------	----------------------n--------n	
17	--------------------------------	--------------------------------	
18	--------------------------------	--------------------------------	
19	--------------------------------	--------------------------------	
20	--------------------------------	--------------------------------	
21	----------------------n--------n	--------------------------------	
22	----------------------0--------0	--------------------------------	
23	----------------------1--------1	--------------------------------	
24	--------------------------------	--------------------------------	
25	--------------------------------	--------------------------------	
26	--------------------------------	--------------------------------	
27	--------------------------------	--------------------------------	
28	--------------------------------	--------------------------------	
29	--------------------------------	--------------------------------	
30	--------------------------------	--------------------------------	
31	--------------------------------	--------------------------------	
32	--------------------------------	--------------------------------	
33	--------------------------------	--------------------------------	
34	--------------------------------	--------------------------------	
35	--------------------------------	--------------------------------	
36	--------------------------------	--------------------------------	
37	--------------------------------	--------------------------------	

- Start characteristic
  - ? in words with difference
  - - in words without differences
  - x in LSB of word 0
- Separate search (phases)
  1. high probability in R2
  2. left stream in R1

Searching for a Differential Characteristic



- Start characteristic
  - ? in words with difference
  - - in words without differences
  - x in LSB of word 0
- Separate search (phases)
  1. high probability in R2
  2. left stream in R1
  3. find first block M_0
  4. right stream in R1

Outline

Finding a Colliding Message Pair

- Message modification
  - many dedicated techniques published
  - mostly hand-tuned (for MD5, RIPEMD, SHA-1, ...)

Finding a Colliding Message Pair

- Message modification
  - many dedicated techniques published
  - mostly hand-tuned (for MD5, RIPEMD, SHA-1, ...)
- Apply to RIPEMD-128?
  - difficult and time consuming
  - 1 message word updates 2 state words
  - different message permutations and rotations values

Finding a Colliding Message Pair

- Message modification
  - many dedicated techniques published
  - mostly hand-tuned (for MD5, RIPEMD, SHA-1, ...)
- Apply to RIPEMD-128?
  - difficult and time consuming
  - 1 message word updates 2 state words
  - different message permutations and rotations values
- Automatic message search
  - continue guessing '-' bits to '0' or '1'
  - guess on words (state, message) in order they appear

Finding a Colliding Message Pair

- Message modification
  - many dedicated techniques published
  - mostly hand-tuned (for MD5, RIPEMD, SHA-1, ...)
- Apply to RIPEMD-128?
  - difficult and time consuming
  - 1 message word updates 2 state words
  - different message permutations and rotations values
- Automatic message search
  - continue guessing '-' bits to '0' or '1'
  - guess on words (state, message) in order they appear
- Amortize costs
  - automatic message modification until word 13
  - brute-force with message words 14,15
  - complexity $2^?$

Outline

Results

previous results:

component	attack	steps	complexity	generic	reference
hash	preimage	33	$2^{124.5}$	2^{128}	[OSS10]
hash	preimage	interm. 35	2^{121}	2^{128}	[OSS10]
hash	preimage	interm. 36	$2^{126.5}$	2^{128}	[WSK$^+$11]

our results:

component	attack	steps	complexity	generic
hash	collision	38	example, 2^{14}	2^{64}
hash	near-collision	44	example, 2^{32}	$2^{47.8}$
hash	non-randomness	48	2^{70}	2^{76}
compression	collision	48	example, 2^{40}	2^{64}

Summary

- Strategy to analyze dual stream hash functions
- Automatic path search and automatic message modification
- Time consuming to find the right settings
- Once settings are found, collision can be found in minutes
- Still lots of work to be done for other (ARX based) hash functions
- Remember: it took 5 years to get from SHA-1 to SHA-2

References

- Hans Dobbertin, Antoon Bosselaers, and Bart Preneel.
  RIPEMD-160: A Strengthened Version of RIPEMD.
  In Dieter Gollmann, editor, *FSE*, volume 1039 of *LNCS*, pages 71–82. Springer, 1996.
- Christophe De Cannière and Christian Rechberger.
  Finding SHA-1 Characteristics: General Results and Applications.
  In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT*, volume 4284 of *LNCS*, pages 1–20. Springer, 2006.
- Florian Mendel, Tomislav Nad, and Martin Schläffer.
  Finding SHA-2 Characteristics: Searching through a Minefield of Contradictions.
  In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT*, LNCS. Springer, 2011.
  To appear.
- Chiaki Ohtahara, Yu Sasaki, and Takeshi Shimoyama.
  Preimage Attacks on Step-Reduced RIPEMD-128 and RIPEMD-160.
  In Xuejia Lai, Moti Yung, and Dongdai Lin, editors, *Inscrypt*, volume 6584 of *LNCS*, pages 169–186. Springer, 2010.
- Lei Wang, Yu Sasaki, Wataru Komatsubara, Kazuo Ohta, and Kazuo Sakiyama.
  (Second) Preimage Attacks on Step-Reduced RIPEMD/RIPEMD-128 with a New Local-Collision Approach.
  In Aggelos Kiayias, editor, *CT-RSA*, volume 6558 of *LNCS*, pages 197–212. Springer, 2011.