Improved Linear Differential Attacks on CubeHash

Shahram Khazaei[1] Simon Knellwolf[2] Willi Meier[2]
Deian Stefan[3]

[1]EPFL, Switzerland

[2]FHNW, Switzerland

[3]The Cooper Union, USA

AFRICACRYPT 2010, Mai 03-06, Stellenbosch

Abstract

- Follow-up of

  [4] Brier, Khazaei, Meier, Peyrin: *Linearization Framework for Collision Attacks: Applications to CubeHash and MD6*, ASIACRYPT 2009.

- Better differences lead to improved collision attacks of CubeHash
- Main results

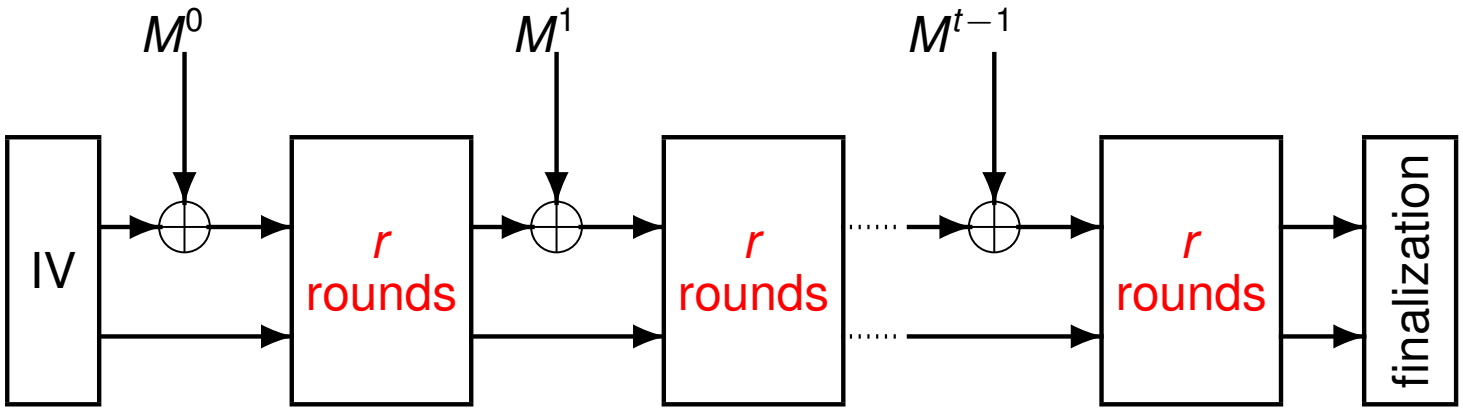
  5/96: practical collisions
  8/96: collision in 2^{80}

- No attack on official CubeHash-16/32

Outline

Description of CubeHash

- Hash function designed in 2008 by Dan Bernstein
- NIST SHA-3 second round candidate
- Internal state of 128 bytes
- Variants CubeHash-r/b (official version 16/32)



$M = M^0||M^1||\ldots||M^{t-1}$ padded message with *b*-byte blocks

One Round

Internal state = 32 words of 32 bits

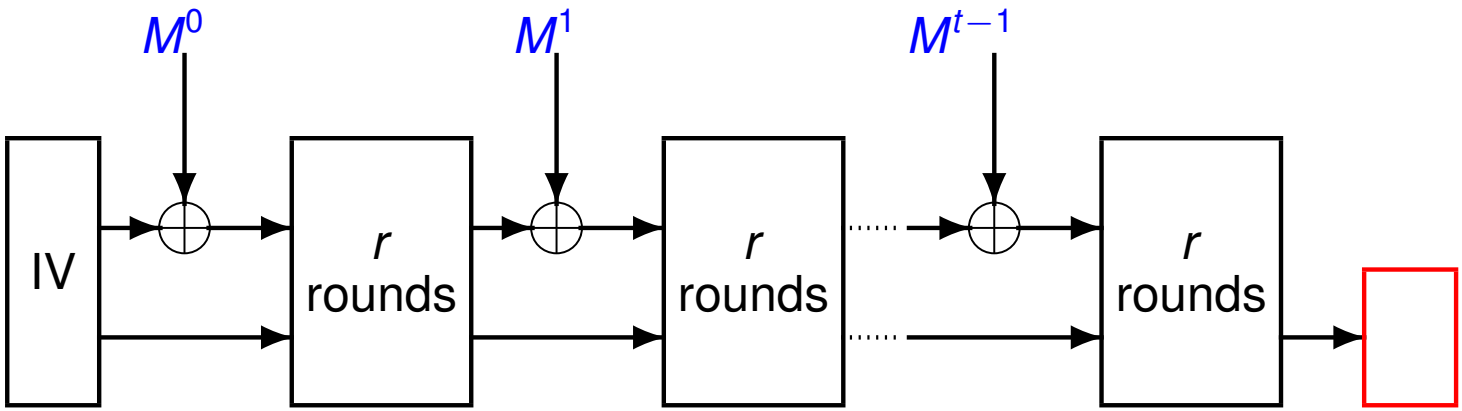
A round consists in:

- 32 additions
- 32 rotations
- 32 swaps
- 32 XORs

Linearization: replace additions by XORs

Attack using Linearization Framework

$Compress : \{0,1\}^{8bt} \rightarrow \{0,1\}^{1024-8b}$



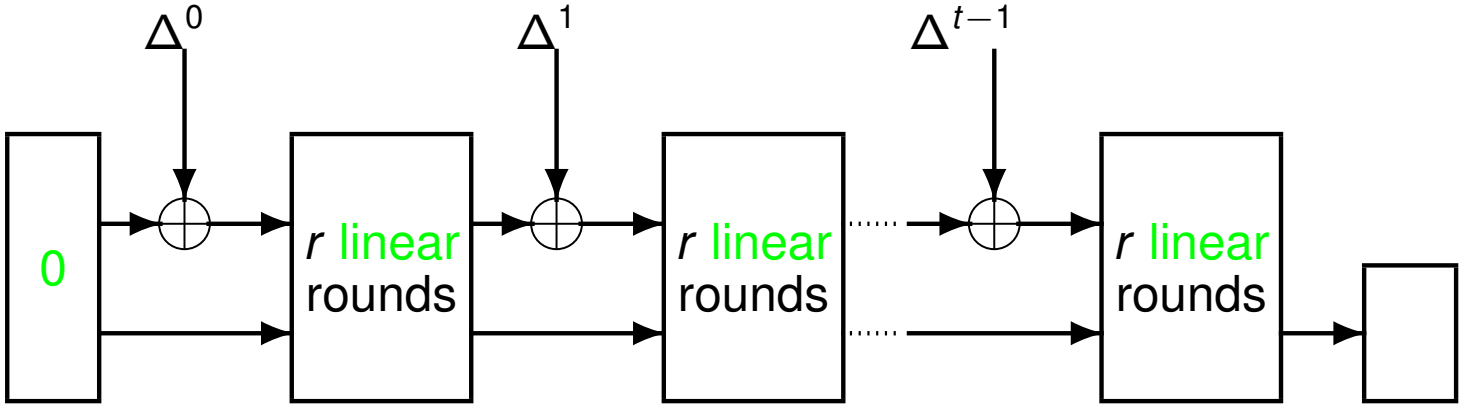
Collision attack: Find M and Δ such that

$$Compress(M) = Compress(M \oplus \Delta)$$

Every collision for *Compress* extends to a full collision.

Finding Δ

$Compress_{\text{lin}} : \{0,1\}^{8bt} \rightarrow \{0,1\}^{1024-8b}$



Δ in the kernel of $Compress_{\text{lin}}$

$\Rightarrow Compress_{\text{lin}}(M) = Compress_{\text{lin}}(M \oplus \Delta)$

$\alpha(\Delta)$, $\beta(\Delta)$: concatenation of left/right addends (XORSs) excluding MSBs

Number of conditions $y = \text{wt}(\alpha(\Delta) \vee \beta(\Delta))$

What is a good Δ?

Raw probability $p_\Delta = 2^{-y} \Rightarrow$ finding M takes 2^y queries

- can be reduced to c_Δ using concepts of [4]:
  - condition function
  - dependency table
  - backtracking algorithm

  $\approx$ automatic message modification techniques

  $c_\Delta =$ estimated complexity of the attack

Two aspects of a good Δ:

1. small total number of conditions
2. sparse conditions in later rounds

Finding Δ: Exhaustive Subset Search

Goal: Find Δ with small total number of conditions

Method in [4]:

1. Determine a kernel basis
2. Exhaustive search over linear combinations of at most 3 basis vectors

But: kernel basis is not unique!

y	6/32	6/64	6/96	7/96	8/96
[4]	400	351	142	251	266
NTL	700	700	165	652	329
c_Δ	2^{182}	2^{144}			

Finding Δ: Randomize the Search

Method adapted from:

[11] Pramstaller, Rechberger, Rijmen: *Exploiting Coding Theory for Collision Attacks on SHA-1*, IMA Int. Conf. 2005.

Kernel basis: $\Delta_0, \ldots, \Delta_{\tau-1}$

Build matrix $\mathcal{G} = \begin{pmatrix} \Delta_0 & || & \alpha(\Delta_0) & || & \beta(\Delta_0) \\ & || & \ldots & || & \\ \Delta_{\tau-1} & || & \alpha(\Delta_{\tau-1}) & || & \beta(\Delta_{\tau-1}) \end{pmatrix}$

- Choose random pivot $\mathcal{G}_{i,j}$
- Eliminate all one's in column j
- Keep row with lowest number of conditions

Finding Δ: Randomize the Search

Minimal number of conditions found with random search (RS)

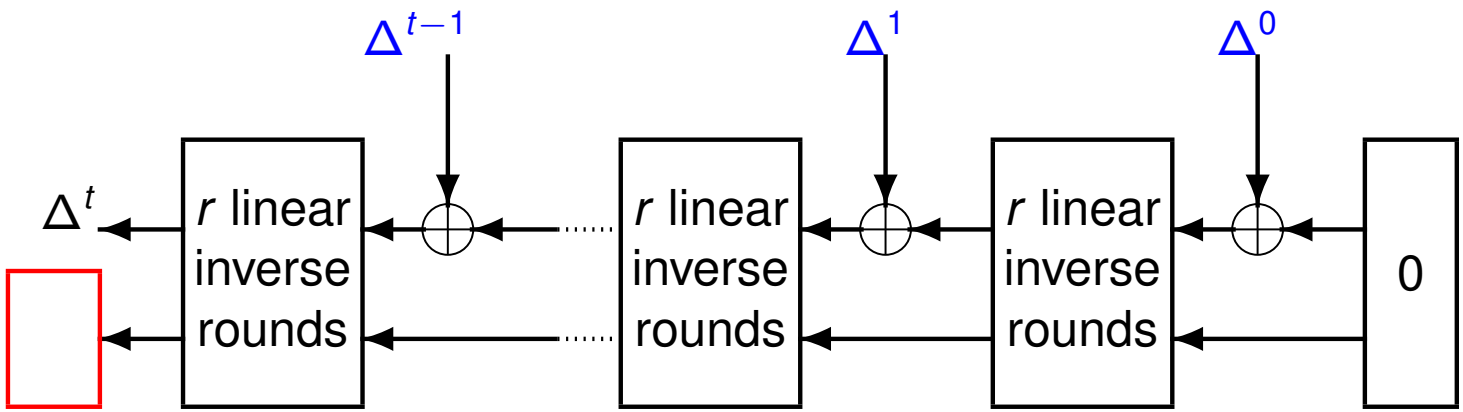
y	6/32	6/64	6/96	7/96	8/96
[4]	400	351	142	251	266
NTL	700	700	165	652	329
RS	394	309	90	251	151
c_Δ	2^{180}	2^{132}	2^{51}	2^{192}	2^{80}

Generic attack for $b = 96$ has complexity of about 2^{128}

Finding Δ: Backword Computation

Goal: Find Δ with sparse conditions in late rounds

$Compress^{b}_{lin} : \{0,1\}^{8bt} \rightarrow \{0,1\}^{1024-8b}$



$\Delta = \Delta^t||\Delta^{t-1}||\ldots||\Delta^1$ lies in the kernel of $Compress_{lin}$

CubeHash-5/96, t = 1

Two different distributions of conditions

y_i = number of conditions at round i

y	y_1	y_2	y_3	y_4	y_5	c_Δ
127	14	17	23	30	43	2^{69}
134	44	36	25	17	12	2^{32}

Collisions found after 2^{23} to $2^{32.25}$ function calls.

Collision for CubeHash-5/96

$M =$
F06BB068 487C5FE1 CCCABA70 0A989262 801EDC3A 69292196
8848F445 B8608777 C037795A 10D5D799 FD16C037 A52D0B51
63A74C97 FD858EEF 7809480F 43EB264C D6631863 2A8CCFE2
EA22B139 D99E4888 8CA844FB ECCE3295 150CA98E B16B0B92
3DB4D4EE 02958F57 8EFF307A 5BE9975B 4D0A669E E6025663
8DDB6421 BAD8F1E4 384FE128 4EBB7E2A 72E16587 1E44C51B
DA607FD9 1DDAD41F 4180297A 1607F902 2463D259 2B73F829
C79E766D 0F672ECC 084E841B FC700F05 3095E865 8EEB85D5

$\Delta =$
08000208 08000208 00000000 00000000 40000100 00000000
00400110 00000000 00000000 00000000 00000000 00000000
00000000 00000000 0800A000 00000000 08000888 08000208
00000000 00000000 40011000 00000000 00451040 00000000
80000000 00000000 80000000 00000000 00000000 00000000
00000000 00000000 00400000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000080
00000000 00000080 00000000 00040000 00000000 00000000

Summary of Improved Results

Backward Computation:

- Collisions for CubeHash-5/96 in practical time

Randomized Search

- Improved collision attacks

  6/32: 2^{180} 6/64: 2^{132} 6/96: 2^{51}

- First collision attack for 8 rounds

  8/96: 2^{80}

Far away from an attack on official CubeHash-16/32

Improved Linear Differential Attacks on CubeHash

Shahram Khazaei[1] Simon Knellwolf[2] Willi Meier[2]
Deian Stefan[3]

[1]EPFL, Switzerland

[2]FHNW, Switzerland

[3]The Cooper Union, USA

AFRICACRYPT 2010, Mai 03-06, Stellenbosch