CubeHash attack analysis (2.B.5)

Daniel J. Bernstein *

Department of Computer Science
University of Illinois at Chicago
Chicago, IL 60607–7045
cubehash@box.cr.yp.to

This document is an analysis of CubeHash with respect to known attacks and their results.

Survey of attacks. The original CubeHash submission documents discussed three basic attack strategies against CubeHashr/b–h:

- Narrow-pipe attacks. The documents explained a standard generic collision attack and a standard generic preimage attack, each costing roughly 2^{512-4b} iterations, where each iteration involves $512r/b$ repetitions of the CubeHash round. The explanation is repeated below. A sufficiently small b stops these attacks.
- Differential attacks. A sufficiently large r stops these attacks.
- Output attacks, including slide attacks, length-extension attacks, etc. The CubeHash finalization stops these attacks.

The submission documents also discussed protection against insider attacks (i.e., protection against trap doors in the design): "CubeHash has a few constants that could be modified, but as far as I know there is no way that any design of this type could have a hidden vulnerability. See the CubeHash specification for discussion of the rotation distances, the hypercube structure, etc."

Of course, these attack strategies need to be compared to attacks that apply to *all* h-bit hash functions:

- Parallel collision search (1994 van Oorschot–Wiener), finding h-bit collisions in time roughly $2^{h/2}/A$ on circuits of total area A.
- Parallel quantum preimage search (1996 Grover), finding h-bit preimages in time roughly $2^{h/2}/A^{1/2}$ on quantum circuits of total area A.

Known quantum collision algorithms are, contrary to popular myth, *slower* than non-quantum collision search. I have a new paper discussing this issue in detail: see `http://cr.yp.to/papers.html#collisioncost`.

Narrow-pipe attacks. CubeHashr/b–h starts with an initial 128-byte state I, xors a b-byte message block m_0, applies an invertible transformation T to obtain $T(I \oplus m_0)$, xors a b-byte message block m_1, applies the transformation T to obtain $T(T(I \oplus m_0) \oplus m_1)$, etc. At the end it xors a particular constant c

* The author was supported by the National Science Foundation under grant ITR–0716498. Date of this document: 2009.09.14.

to the state, applies T ten more times, and outputs the first h bits of the final state.

Standard generic collision attack: The attacker searches for collisions in the last $128-b$ bytes of the intermediate state $T(T(I \oplus m_0) \oplus m_1)$ after two blocks m_0, m_1. If $T(T(I \oplus m_0) \oplus m_1)$ and $T(T(I \oplus m'_0) \oplus m'_1)$ share the last $128-b$ bytes then the attacker can immediately write down many CubeHash collisions, namely (m_0, m_1, m_2) and (m'_0, m'_1, m'_2) for any m_2, m'_2 satisfying

$$m_2 \oplus m'_2 = T(T(I \oplus m_0) \oplus m_1) \oplus T(T(I \oplus m'_0) \oplus m'_1),$$

and of course any extensions of those collisions.

(When I say "searches for collisions" I am assuming that the attacker uses state-of-the-art parallel low-memory collision search, as in 1994 van Oorschot–Wiener. Bloom and Kaminsky claim in "Single block attacks and statistical tests on CubeHash" that collision search "requires copious memory" and is not easily parallelizable; these claims are incorrect.)

More generally, the attacker searches for collisions in the last $128-b$ bytes of the intermediate state after n blocks, and then obtains $(n+1)$-block collisions in CubeHash. There are 2^{nb} possible n-block inputs, so $(128-b)$-byte collisions are likely to exist if $2nb > 1024 - 8b$, i.e., if $n > 512/b - 4$. Finding a collision in this way means evaluating T approximately $2^{521-4b-\lg b}$ times. The chance of success drops off quadratically with fewer T evaluations.

Standard generic preimage attack: The attacker expands the h-bit target arbitrarily into a 128-byte final state Z, works backwards to an end-of-message state $Y = c \oplus T^{-10}(Z)$, and searches for collisions between the last $128-b$ bytes of $T(T(I \oplus m_0) \oplus m_1)$ and $T^{-1}(T^{-1}(T^{-1}(Y) \oplus m_4) \oplus m_3)$, obtaining a CubeHash preimage

$$(m_0, m_1, T(T(I \oplus m_0) \oplus m_1) \oplus T^{-1}(T^{-1}(T^{-1}(Y) \oplus m_4) \oplus m_3), m_3, m_4).$$

More generally, the attacker searches for similar collisions involving n initial blocks and n final blocks. Finding a CubeHash preimage in this way means evaluating T approximately $2^{522-4b-\lg b}$ times. As above, the chance of success drops off quadratically with fewer T evaluations.

For example, if T is as fast as a single round of CubeHash, then a fantasy-universe attacker capable of 2^{511} bit operations would be able to evaluate T 2^{500} times, but still would have only about a 2^{-8} chance of breaking $b = 4$ with these attacks.

What is interesting about these attacks is that they do not disintegrate as r increases: they put a limit on the safe b's for any reasonable value of r.

Third-party analyses. After the original CubeHash submission there were several third-party analyses of differential attacks on reduced-round CubeHash:

- Aumasson, "Collision for CubeHash2/120-512".
- Dai, "Collisions for CubeHash1/45 and CubeHash2/89".
- Brier, Peyrin, "Cryptanalysis of CubeHash".

- Brier, Khazaei, Meier, Peyrin, "Attack for CubeHash-2/2 and collision for CubeHash-3/64".
- Brier, Khazaei, Meier, Peyrin, "Real Collisions for CubeHash-4/64".
- Brier, Khazaei, Meier, Peyrin, "Linearization framework for collision attacks: application to CubeHash and MD6".

The latest attacks are estimated to find second preimages

- in CubeHash2/2 using 2^{221} simple operations,
- in CubeHash3/4 using 2^{478} simple operations,
- in CubeHash4/3 using 2^{195} simple operations,
- in CubeHash5/64 using 2^{205} simple operations,
- in CubeHash6/4 using 2^{478} simple operations, and
- in CubeHash7/64 using 2^{447} simple operations;

and to find collisions

- in CubeHash2/2 using 2^{179} simple operations,
- in CubeHash3/12 using 2^{153} simple operations,
- in CubeHash4/3 using 2^{163} simple operations,
- in CubeHash5/64 using 2^{71} simple operations,
- in CubeHash6/16 using 2^{222} simple operations, and
- in CubeHash7/64 using 2^{203} simple operations.

Explicit collisions in CubeHash2/3, CubeHash3/64, and CubeHash4/48 have been computed by scaled-down versions of the same attacks.

There have also been several third-party analyses of other attacks:

- Aumasson, Meier, Naya-Plasencia, Peyrin, "Inside the hypercube": Variants of the standard generic preimage attack, trying to streamline the individual iterations.
- Khovratovich, Nikolic, Weinmann, "Preimage attack on CubeHash512-r/4 and CubeHash512-r/8": Republication of the same attack.
- Salaev, Rao, "Logical cryptanalysis of CubeHash using a SAT solver": Some automated attacks on CubeHash2/b, not as fast as previous attacks.
- Bloom, Janis, "Inference attacks on CubeHash": Attacks on CubeHash$r$/128, similar to previous attacks.
- Wang, Wilson, "Parallel collision search attack on hash function": Report of an implementation of the generic van Oorschot–Wiener attack.
- Bloom, Kaminsky, "Single block attacks and statistical tests on CubeHash": Slowdown of the standard generic preimage attack, approximately squaring the number of iterations required.

None of these attacks pose any risk to CubeHash8/1, CubeHash16/32, etc.