CubeHash appendix: complexity of generic attacks

Daniel J. Bernstein *

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

This appendix comments in more detail on the complexity of standard generic attacks against CubeHashr/b–h, i.e., attacks that immediately generalize from CubeHash's r-round transformation to *any* invertible transformation T having the same 128-byte state size.

CubeHash starts with an initial 128-byte state I, xors a b-byte message block m_0, applies the 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 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 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.

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 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 preimage in this way means evaluating

* The author was supported by the National Science Foundation under grant ITR–0716498. Date of this document: 2008.10.30.

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.

I'm not claiming that these standard generic attacks are the best possible generic attacks, and I'm certainly not claiming that they're the best possible attacks! On the contrary: I think that CubeHash1/b, for example, will need a smaller b than CubeHash2/b to be secure. But I expect most attacks on CubeHash$r$/b to disintegrate as r increases. What's interesting about generic attacks is that their cost grows only linearly with r.