Quantum attacks against Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein

Daniel J. Bernstein *

Department of Computer Science (MC 152)
The University of Illinois at Chicago
Chicago, IL 60607–7053
djb@cr.yp.to

Abstract. This paper presents attacks that clearly violate the explicit security claims of 11 of the 14 second-round submissions to the SHA-3 competition: Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein. The attacks are structured-first-preimage attacks, the most devastating type of hash-function attack. The attacks use a quantum computer, but not a particularly large quantum computer. The attacks are not instantaneous, but they are much faster than the minimum attack cost claimed in the submission documents.

1 Introduction

NIST's call for SHA-3 submissions required each submission to contain, among other things, security claims:

> 2.B.4 A statement of the expected strength (i.e., work factor) of the algorithm shall be included, along with any supporting rationale, for each of the security requirements specified in sections 4.A.ii and 4.A.iii, and for each message digest size specified in section 3.

The security requirements include collision resistance, preimage resistance, etc. The specified output sizes are "224, 256, 384, and 512 bits." Most, although not all, of the second-round SHA-3 submissions obey this requirement and contain explicit claims of preimage resistance for 224-bit output, 256-bit output, 384-bit output, and 512-bit output.

This document disproves the claims of preimage resistance for Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein. Specifically, this document presents attacks finding preimages in each of these hash functions using time much, much, much smaller than 2^{224}: specifically, using only about 2^{112} simple operations. The exponent gap is so large that it cannot be explained by the difference between "224 bits" and "approximately 224 bits."

The attacks are more powerful than first-preimage attacks: they are *structured*-first-preimage attacks, finding first preimages with practically any format desired by the attacker. This type of attack is a complete break, trivially allowing the attacker to carry out every other standard attack notion in the literature: for example, being able to find structured first preimages implies being able to find second preimages.

* Permanent ID of this document: 0152ab005327cb177476138d8ca74674. Date of this document: 2010.11.12.

The attacks rely on a quantum computer, but not a large quantum computer; the required size is discussed later. The attacks do not raise any issues of parallelism, large-scale error correction, etc. Building a quantum computer that carries out 2^{112} simple operations is of course a highly nontrivial engineering challenge, but one can reasonably argue that this challenge will be met within the lifetime of SHA-3, whereas building any sort of computer that carries out 2^{224} simple operations is obviously not possible for the foreseeable future.

To summarize: Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein each claim 2^{224} preimage resistance but are in fact breakable with only 2^{112} simple operations. Similar comments apply to other output sizes, but those attacks are slower than 2^{112} simple operations and therefore less threatening than the attacks discussed in this document.

2 Security claims made by Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein

The Blue Midnight Wish submission [13, Section 3.8, "Statements about security, support for applications, HMACs and randomized hashing"] contains the following columns in Table 3.8, "Cryptographic strength of the Blue Midnight Wish":

Algorithm abbreviation	···	Work factor for finding a preimage	···
BMW224	···	$\approx 2^{224}$	···
BMW256	···	$\approx 2^{256}$	···
BMW384	···	$\approx 2^{384}$	···
BMW512	···	$\approx 2^{512}$	···

This is, in particular, a claim that finding preimages in 224-bit BMW has a "work factor" of $\approx 2^{224}$. There is no warning regarding the impact of quantum computers: BMW claims security against all methods of "finding a preimage," not just pre-quantum methods.

The ECHO submission [3, Part III, "Security claims and analysis"] contains the following claims:

> When ECHO is used to generate a hash output of n bits, where n takes the values 224, 256, 384, and 512 bits, we claim that ... the work effort to compromise preimage resistance is 2^n operations ...

This is, in particular, a claim that finding preimages in 224-bit ECHO takes 2^{224} "operations." There is no warning regarding the impact of quantum computers: ECHO claims "preimage resistance," not merely pre-quantum preimage resistance.

The Fugue submission [15, Section 1.2, "Statement of expected strength"] contains the following claim:

> We expect that the best attacks against Fugue are the generic ones. That is, the best collision attack against Fugue-X will have work factor of $2^{X/2}$, and the best pre-image and second-pre-image attacks will have work factor of 2^X.

This is, in particular, a claim that finding preimages in 224-bit Fugue has "work factor" 2^{224}. There is no warning regarding the impact of quantum computers: Fugue claims security against all "pre-image attacks," not merely pre-quantum pre-image attacks.

The Grøstl submission [12, Section 6.5, "Security claims and summary of known attacks"] contains the following claim:

> With the number of rounds proposed in Section 3.4.6, we claim the following security levels for the Grøstl-n hash function. In the second preimage attack, the first preimage is assumed to be of length at most 2^k blocks.

Attack type	Claimed complexity	Best known attack
⋮		
Preimage	2^n	2^n
⋮		

This is, in particular, a claim that finding preimages in 224-bit Grøstl has "complexity" 2^{224}. There is no warning regarding the impact of quantum computers: Grøstl claims security against any "Preimage" attack, not merely pre-quantum preimage attacks.

The Hamsi submission [16, Section 2.1] contains the following table:

Table 2.1: Hamsi variants and security claims

Variant	···	Collision resistance	Preimage resistance	···
⋮				
Hamsi-224	···	112	224	···
⋮				

The table entries are separately defined as being in "bits," and it is clear to the reader that b bits of security mean that attacks use 2^b operations. This is, in particular, a claim of 2^{224} preimage resistance for 224-bit Hamsi. There is no warning regarding the impact of quantum computers: Hamsi claims "preimage resistance," not merely pre-quantum preimage resistance.

The JH submission [19, Section 9, "Security analysis of JH"] contains the following table:

> The security of JH hash algorithms are stated below ...

	collision	second-preimage	preimage
JH-224	2^{112}	2^{224}	2^{224}
JH-256	2^{128}	2^{256}	2^{256}
JH-384	2^{192}	2^{384}	2^{384}
JH-512	2^{256}	$2^{512-\log_2 l}$	2^{512}

This is, in particular, a claim of 2^{224} security against preimages for 224-bit JH. There is no warning regarding the impact of quantum computers: JH claims security against any "preimage" attack, not merely pre-quantum preimage attacks.

The Keccak submission [6, Section 6.1.2] contains the following claim:

> For the first four Keccak candidates with fixed digest length, the output length n satisfies $n = c/2$. This means that using Keccak as a hash

> function provides collision resistance of $2^{n/2}$, (second) preimage resistance of 2^n and resistance to length-extension.

This is, in particular, a claim of 2^{224} preimage resistance for 224-bit Keccak. There is no warning regarding the impact of quantum computers: Keccak claims "preimage resistance," not merely pre-quantum preimage resistance.

The Shabal submission [8, Part 2.B.4, "A statement of the expected strength"] contains the following claim in Section 10.2, "Preimage resistance":

> **Security Claim 2.** For any $\ell_h \in \{192, 224, 256, 384, 512\}$, any preimage attack against Shabal with ℓ_h-bit message digests requires at least 2^{ℓ_h} calls to the message round function.

This is, in particular, a claim that at least 2^{224} "calls to the message round function" are required for a preimage attack against 224-bit Shabal. There is no warning regarding the impact of quantum computers: Shabal claims security against "any preimage attack," not merely pre-quantum preimage attacks.

The SHAvite-3 submission [7, Section 3.4.4, "Security against second preimage attacks"] contains the following claim (which must be read together with the separate statement that SHAvite-3 "is a HAIFA hash function"):

> HAIFA offers full security against second preimage attacks, i.e., finding a second preimage or a chosen target preimage of an m-bit digest requires 2^m compression functions calls.

This is, in particular, a claim that at least 2^{224} "compression functions calls" are required for a preimage attack against 224-bit SHAvite-3. There is no warning regarding the impact of quantum computers: SHAvite-3 claims security against any method of "finding a second preimage," not merely pre-quantum methods.

The SIMD submission [17, Chapter 3, "Expected strength"] contains the following claim:

> In particular this means that we believe that a collision attack on SIMD-n has a complexity of $2^{n/2}$, and a preimage or second preimage attack has a complexity of 2^n.

This is, in particular, a claim of "complexity" 2^{224} for any preimage attack against 224-bit SIMD. There is no warning regarding the impact of quantum computers: SIMD claims security against any "preimage attack," not merely pre-quantum attacks.

The Skein submission [11, Section 6.1, "Basic security claims for Skein"] contains the following claim:

> Below, we write n for the state size, and m for the minimum of state and output size. We claim the following levels of security against standard attacks:
> - First preimage resistance up to 2^m.
> - Second preimage resistance up to 2^m.
> - Collision resistance up to $2^{m/2}$.
> - ...

This is, in particular, a claim of 2^{224} preimage resistance for 224-bit Skein (whether Skein-256-224 or Skein-512-224). There is no warning regarding the impact of quantum computers: Skein claims "preimage resistance," not merely pre-quantum preimage resistance.

3 Attacks violating the security claims for Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, SIMD, and Skein

This section presents an attack that finds structured first preimages in 224-bit SIMD using only about 2^{112} simple operations on a rather small quantum computer. The main tool in the attack is Grover's algorithm [14]; this section borrows heavily from the discussion of Grover's algorithm in my paper [5, Section 3]. This section concludes by discussing the other 10 hash functions.

The attack is parameterized by an attacker-selected structure: a function φ that expands a 224-bit string $b_0b_1 \ldots b_{223}$ into (e.g.) a 1024-bit string $c_0b_0c_1b_1 \ldots c_{223}b_{223}c_{224}$, where $c_0, c_1, \ldots, c_{224}$ are constant strings that in total contain 800 bits. The attack is given a 224-bit hash h as output and finds a SIMD preimage of h having the form $\varphi(b)$: i.e., a 224-bit string b such that $\text{SIMD}(\varphi(b)) = h$. The same attack idea can also handle more general functions φ, although very complicated functions φ make the attack more expensive.

Define $f(b)$ as 0 if $\text{SIMD}(\varphi(b)) = h$, and 1 otherwise. The first step in constructing the attack algorithm is to write down a fully unrolled circuit for this function f. The size of this circuit is the number of bit operations used by SIMD on a 1024-bit input, minus a few operations that can be eliminated given the constants in φ, plus a few operations to compare the result to h. To simplify subsequent steps I will assume that the circuit uses only NAND gates; other gates such as XOR can be expressed as small combinations of NAND gates, although this is not as efficient as building them directly from transistors.

The next step is to write down a "reversible" circuit for f: a non-erasing circuit built from Toffoli gates $(x, y, z) \mapsto (x, y, z + xy)$ rather than NANDs. This is a standard circuit transformation, costing small constant factors in the number of input bits and in the size of the circuit. At this point the circuit size is still within an order of magnitude of the number of bit operations used by SIMD on a 1024-bit input.

The next step is to convert the circuit into a quantum circuit: replace each bit by a qubit, and replace each Toffoli gate by a quantum Toffoli gate. This requires reliable qubits, but not a huge number of them: the total number of qubits is bounded by the total number of qubit operations, which is still within an order of magnitude of the number of bit operations used by SIMD on a 1024-bit input. I do not claim that this size is optimal: for example, many designs of quantum computers do not require full unrolling of the original circuit.

The last step is to combine this quantum circuit with a quantum rotation and a Hadamard transformation, as explained by Grover in [14]. Iterating the resulting quantum computer approximately $2^{224/2} = 2^{112}$ times has a good chance of finding a root of f, i.e., a structured preimage of SIMD. This disproves the claim of 2^{224} complexity for preimage attacks against 224-bit SIMD.

The attacks on Blue Midnight Wish, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Shabal, SHAvite-3, and Skein are analogous to the above attack on SIMD. Each attack uses approximately 2^{112} iterations, disproving the 2^{224} preimage-resistance claims for each of these functions. The exact complexity of each iteration, and the required number of qubits, varies somewhat from function to function, but is on a far smaller scale than the 2^{112} complexity gap.

4 What about BLAKE, CubeHash, and Luffa?

This section discusses the other three second-round SHA-3 submissions, namely BLAKE, CubeHash, and Luffa.

The BLAKE submission document [2, Section 1.3, "Expected strength"] says

> For all BLAKE hash functions, there should be no attack significantly more efficient than standard bruteforce methods for
> - finding collisions, with same or distinct salt
> - finding (second) preimages, with arbitrary salt

but does not say what these "standard bruteforce methods" are or how efficient they are.

Given the lack of quantification here, I certainly cannot say that I have disproven the BLAKE security claims. On the other hand, I don't think that it was NIST's intent to allow such a vague security statement. Users who are aware only of 2^{224} attacks against 224-bit hash functions are likely to interpret the BLAKE documentation as claiming 2^{224} preimage security, and will be unpleasantly surprised when quantum computers find preimages in BLAKE using only 2^{112} simple operations.

Similarly, I have been unable to find a statement of expected strength anywhere in the Luffa submission documents [9] and [10], so I cannot say that I have disproven the Luffa security claims. However, I cannot see how the Luffa submission can be viewed as compliant with NIST requirement 2.B.4. It seems clear that NIST's intent is for the SHA-3 standard to explicitly state the security levels provided by (at least) the 224-bit, 256-bit, 384-bit, and 512-bit SHA-3 options. Most SHA-3 submissions contain such statements, and those statements have now been disproven. BLAKE and Luffa evade this only by failing to quantify their security statements.

CubeHash is the only second-round SHA-3 submission whose 2.B.4 security statements are (1) explicitly quantified for each of the four required sizes and (2) resistant to quantum computers. For example, [4] states

> **224-bit preimage resistance.** CubeHash–224 is expected to provide preimage resistance of approximately 224 bits, but quantum computers are expected to reduce preimage resistance to approximately 112 bits.

A recent paper by one of the SIMD authors claims that an attack similar to the attack in Section 3 "violates the expected security of CubeHash." This claim ignores the explicit security statements in the CubeHash submission documents. There have been no attacks violating any of the CubeHash security claims.

References

[1] — (no editor), *Proceedings of the twenty-eighth annual ACM symposium on the theory of computing, held in Philadelphia, PA, May 22–24, 1996*, Association for Computing Machinery, 1996. ISBN 0-89791-785-5. MR 97g:68005. See [14].

[2] Jean-Philippe Aumasson, Luca Henzen, Willi Meier, Raphael C.-W. Phan, *SHA-3 proposal BLAKE* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/BLAKE_Round2.zip, file `blake.pdf`. Citations in this document: §4.

[3] Ryad Benadjila, Olivier Billet, Henri Gilbert, Gilles Macario-Rat, Thomas Peyrin, Matt Robshaw, Yannick Seurin, *SHA-3 proposal: ECHO* (2008). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/ECHO_Round2.zip, file `echo_description.pdf`. Citations in this document: §2.

[4] Daniel J. Bernstein, *CubeHash expected strength (2.B.4)* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/CubeHash_Round2.zip, file `strength.pdf`. Citations in this document: §4.

[5] Daniel J. Bernstein, *Grover vs. McEliece*, in PQCrypto 2010 [18] (2010), 73–80. URL: http://cr.yp.to/papers.html#grovercode. Citations in this document: §3.

[6] Guido Bertoni, Joan Daemen, Michaël Peeters, Gilles Van Assche, *Keccak sponge function family main document* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Keccak_Round2.zip, file `Keccak-main-2.0.pdf`. Citations in this document: §2.

[7] Eli Biham, Orr Dunkelman, *The SHAvite-3 hash function* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/SHAvite-3_Round2.zip, file `Shavite.pdf`. Citations in this document: §2.

[8] Anne Canteaut, Benoît Chevallier-Mames, Aline Gouget, Pascal Paillier, Thomas Pornin (editors), *Shabal, a submission to NIST's cryptographic hash algorithm competition* (2008). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Shabal_Round2.zip, file `description.pdf`. Citations in this document: §2.

[9] Christophe De Cannière, Hisayoshi Sato, Dai Watanabe, *Hash function Luffa: specification ver. 2.0* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Luffa_Round2.zip, file `Luffa_Specification.pdf`. Citations in this document: §4.

[10] Christophe De Cannière, Hisayoshi Sato, Dai Watanabe, *Hash function Luffa: supporting document* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Luffa_Round2.zip, file `Luffa_SupportingDocument.pdf`. Citations in this document: §4.

[11] Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, Jesse Walker, *The Skein hash function family* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Skein_Round2.zip, file `skein1.2.pdf`. Citations in this document: §2.

[12] Praveen Gauravaram, Lars R. Knudsen, Krystian Matusiewicz, Florian Mendel, Christian Rechberger, Martin Schläffer, Søren S. Thomsen, *Grøstl—a SHA-3 candidate* (2008). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Grostl_Round2.zip, file `Groestl.pdf`. Citations in this document: §2.

[13] Danilo Gligoroski, Vlastimil Klima, Svein Johan Knapskog, Mohamed El-Hadedy, Jørn Amundsen, Stig Frode Mjølsnes, *Cryptographic hash function Blue Midnight Wish* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Blue_Midnight_Wish_Round2.zip, file `BlueMidnightWishDocumentation.pdf`. Citations in this document: §2.

[14] Lov K. Grover, *A fast quantum mechanical algorithm for database search*, in [1] (1996), 212–219. MR 1427516. Citations in this document: §3, §3.

[15] Shai Halevi, William E. Hall, Charanjit S. Jutla, *The hash function "Fugue"* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Fugue_Round2.zip, file `fugue.pdf`. Citations in this document: §2.

[16] Özgül Küçük, *The hash function Hamsi* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Hamsi_Round2.zip, file `Hamsi_Spec_2ndRound.pdf`. Citations in this document: §2.

[17] Gaëtan Leurent, Charles Bouillaguet, Pierre-Alain Fouque, *SIMD is a message digest* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/SIMD_Round2.zip, file `SIMD.pdf`. Citations in this document: §2.

[18] Nicolas Sendrier (editor), *Post-quantum cryptography, third international workshop, PQCrypto, Darmstadt, Germany, May 25–28, 2010*, Lecture Notes in Computer Science, 6061, Springer, 2010. See [5].

[19] Hongjun Wu, *The hash function JH* (2009). URL: http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/JH_Round2.zip, file `jh20090915.pdf`. Citations in this document: §2.