```
X-Sieve: CMU Sieve 2.2
From: "Wallner, Debbie M" <dmwalln@orion.ncsc.mil>
To: "'hash-function@nist.gov'" <hash-function@nist.gov>
Cc: "'william.burr@nist.gov'" <william.burr@nist.gov>,
        "'Elaine Barker'"
         <elaine.barker@nist.gov>,
        "Wallner, Debbie M" <dmwalln@orion.ncsc.mil>
Subject: Hash Algorithm Requirements and Evaluation Criteria
Date: Fri, 13 Apr 2007 15:54:49 -0400
X-Mailer: Internet Mail Service (5.5.2653.19)
X-Proofpoint-Virus-Version: vendor=fsecure engine=4.65.5502:2.3.11,1.2.37,4.0.164 definitions=2007-04-
13_05:2007-04-13,2007-04-13,2007-04-13 signatures=0
X-PP-SpamDetails: rule=spampolicy2_notspam policy=spampolicy2 score=0 spamscore=0 ipscore=0
phishscore=0 adultscore=0 classifier=spam adjust=0 reason=mlx engine=3.1.0-0703060001
definitions=main-0704130040
X-PP-SpamScore: 0
X-NIST-MailScanner: Found to be clean
X-NIST-MailScanner-From: dmwalln@orion.ncsc.mil
```

Bill,

Please find attached comments for consideration.

Regards - Debby Wallner

<<hash_criteria_comments.doc>> See below

General Comment. Is a winner of the competition guaranteed? What if no suitable candidates are submitted?

A.3 The maximum message length should be at least 2^{64}, not 264.

B.1 First paragraph. This section seems to encourage the 40+ page "proofs of security" found in so many recent proposals. These sorts of proofs had little influence on the AES competition.

B.1 Second paragraph should be more specific. Is it enough to say that the number of rounds is a parameter, or must proposals explain how to increase the number of rounds? For instance, SHA-1 uses a different non-linear function for each 20 steps. Increasing the number of steps requires choosing a nonlinear function for the additional steps.

B.1 Second paragraph. Care must be taken when standardizing on a parameterized hash function. As was noted during the AES competition, parameterization permits the attacker to force the weakest version of the hash function. Wouldn't the real goal of strengthening an approved hash (say in 2025 SHA-3 is found to step too few times) be achieved by simply updating as was done with SHA to SHA-1?

B.2 This seems to suggest that submitters will have exactly two C programs, one for reference and one optimized version. How many variations should the optimized version account for? There are 4 hash output sizes, hardware version, software version (for 8, 32, and 64-bit processor), paralellizable version, etc. This was an issue with the AES competition – no one was entirely clear what weight these various factors held.

B.2 Additionally, there is some concern that optimized code will be used as a basis for differentiating the submissions. What if a great design just happens to have a bad programmer? One positive aspect of the AES competition was to have quasi-independent coding. Will that once again be provided (although it is not provided for in this call)?

B.5 Can the optimized software/hardware implementation be covered by IP?

C.1, first bullet. It would be nice to define second preimage resistance. There is a second pre-image attack on all SHA and MD hashes (Kelsey & Schneier, "Generic Second Preimages with less than 2^n work") and no one seems to care. Does NIST want hashes that resist this attack? This is significant for the hash competition because a hash that resists this attack will probably be slower than hashes that do not. It is reasonable to redefine second preimage resistance to exclude this attack.

C.1, second bullet. It is not certain that this is the best way to state this property. It is easy to distinguish the output of a hash from random – if one knows the message and the definition of the hash, just plug the message and either you get the hash output or not. A better quality to look for is whether the hash is appropriate for use as a pseudorandom number generator or for counter mode encryption. It would also be helpful to spell out the other intended uses of the hash (digital signatures, etc.)

C.2.1 and C.2.2. When NIST does their own tests, it would be helpful for competitors to know what platform, compiler, etcetera will be used. As with the AES competition, everyone will claim that their algorithm is the fastest or smallest in some way.