X-Sieve: CMU Sieve 2.2
From: Tolga Acar <Tolga.Acar@microsoft.com>
To: "hash-function@nist.gov" <hash-function@nist.gov>
CC: Brian LaMacchia <bal@exchange.microsoft.com>
Date: Fri, 27 Apr 2007 15:44:15 -0700
Subject: Hash Algorithm Requirements and Evaluation Criteria
Thread-Topic: Hash Algorithm Requirements and Evaluation Criteria
Thread-Index: AceJHZWLVSgrbhlBTzC5YoBPvV9a6w==
Accept-Language: en-US
X-MS-Has-Attach:
X-MS-TNEF-Correlator:
acceptlanguage: en-US
X-Proofpoint-Virus-Version: vendor=fsecure engine=4.65.5502:2.3.11,1.2.37,4.0.164
definitions=2007-04-27_06:2007-04-27,2007-04-27,2007-04-27 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-0704270152
X-PP-SpamScore: 0
X-NIST-MailScanner: Found to be clean
X-NIST-MailScanner-From: tolga.acar@microsoft.com

Attn: Hash Algorithm Requirements and Evaluation Criteria

We are pleased to submit the following comments on draft minimum acceptability requirements, submission requirements, and evaluation criteria for the candidate hash algorithms on behalf of Microsoft Corporation, Redmond, WA.

1. Input message length
   a. The message length should be measured in bytes instead of bits.
   b. Following our comment above, we believe that all candidate algorithms should be required to accept messages with lengths >= $2^{64}$ bytes, with slight preference given to functions that can accept longer messages.
2. Performance, Power, and Memory Usage
   a. Performance of a hash function candidate should be evaluated on 8, 32, and 64-bit platforms, and 64-bit platform benchmark figures should be given a higher priority than others. We expect most servers (where the majority of performance requirements come from) to move to 64-bit platforms.
   b. Run-time memory footprint (both code and data) should be part of the evaluation criteria, with preference given to smaller footprint implementations.
   c. Gate count and performance of possible hardware implementations, such as Trusted Platform Modules (TPM), should be part of the evaluation criteria, with preference given to fewer number of gates and faster implementations.
   d. In addition to desktop and server platforms, run-time power consumption requirements on mobile platforms should also be taken into account in the evaluation criteria. Preference should be given to designs with possible low-power implementations on mobile platforms.
3. Testing
   a. In addition to traditional known answer tests, a design should allow very long message testing without having to feed the entire message. For instance, in existing hash functions described in FIPS 180-2, this might be accomplished by setting the internal state and message length, and finalizing the hash function to produce an output.
   b. We suggest that known answer tests should be provided with input message length crossing well-known boundaries, such as $2^8$, $2^{16}$, $2^{32}$, $2^{48}$, etc, where it would trigger an update in more than one internal message length variables. For example,

we would like to see known-answer tests with messages of length $2^8$-1 bytes, $2^8$ bytes, and $2^8$+1 bytes long where the internal length is kept in bytes.

c. If there is randomness inherent in the design, it must be possible to fix the random data or let the caller provide the random data to produce a pre-computed answer.

Best regards,

- Tolga Acar, Brian LaMacchia