X-Sieve: CMU Sieve 2.2
DKIM-Signature: a=rsa-sha1; c=relaxed/relaxed;
d=gmail.com; s=beta;
h=domainkey-signature:received:received:message-id:date:from:to:subject:mime-version:content-type;
b=hpQBvkv1eVIC20ITx1GF9QzMkMgj45SpDwp7VWz0W/LJW29eYzpEgcm7TdB40KSAeRITOmQF5LppqiFT7kxwNxf6+Htrs+f7fILPPfFTlxd1YKBtWyKTqUgItzIKSRRZBBrZJTPK1wGzyLKu+e6zCcGKjyY3YkapX8pfGFbd6YI=
DomainKey-Signature: a=rsa-sha1; c=nofws;
d=gmail.com; s=beta;
h=received:message-id:date:from:to:subject:mime-version:content-type;
b=ZtPlFdzF3BkdtbUVdSwNPSE9pyZrEW1hgpYD8nNzItks99BtCqv54peixXGrjfKxpWY5aena0sKBR8MsEDjW2TJ2nJqHJqfiG5uTel2mylE7QBhUsTTMKdlxq9n2st6kPWm28FcVjY38aKEJbNYS4WpOdCuhRgeb7siyRVibt98=
Date: Sun, 8 Apr 2007 03:43:18 +0800
From: "Hongjun Wu" <wuhongjun@gmail.com>
To: hash-function@nist.gov
Subject: Hash Algorithm Requirements and Evaluation Criteria
X-Proofpoint-Virus-Version: vendor=fsecure engine=4.65.5502:2.3.11,1.2.37,4.0.164 definitions=2007-04-07_02:2007-04-04,2007-04-07,2007-04-07 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-0704070041
X-PP-SpamScore: 0
X-NIST-MailScanner: Found to be clean
X-NIST-MailScanner-From: wuhongjun@gmail.com

Dear Sir/Madam,

Below are some comments and questions on the hash function requirement and evaluation.

1. comments on software: B.2 together with C.2.1 and C.2.2

B.2 An ANSI C source language reference implementation and an optimized implementation. The optimized code will be used to compare software performance and memory requirements to the implementations of other submitted algorithms.
C.2.1 Computational efficiency:
C.2.2 Memory requirements:

There may be some confusion on the use of the term "an optimized implementation". The code optimized for software throughput may not be the same as the code optimized for memory and code size. Thus do we need two optimized implmentations, one optimized for the computational efficiency evaluation, and another one optimized for the memory requirement evaluation?

The problem may become more complicated: an optimized implementation on which platform? The optimized code on the 32-bit platform may be different from the optimized code on the 64-bit platform.

2. comments on the security requirement

Is it possible to provide some numerical data to clearly specify the security requriements so as to eliminate the ambiguity? For example, for the hash function with 256-bit output, the collision resistance is with complexity 2^{128}; (first- and second-) preimage resistance is with complexity 2^{256}.

One more question: For a hash function with 256-bit output, it is supposed to use this hash function in the environment where no one can carry out 2^{128} computations. So what would be the meaningful requirement on the preimage resistance of such hash function ?

Best Regards,
hongjun