X-Sieve: CMU Sieve 2.2
From: "Ron Steinfeld" <rons@ics.mq.edu.au>
To: <hash-function@nist.gov>
Subject: Hash Algorithm Requirements and Evaluation Criteria
Date: Fri, 27 Apr 2007 19:41:48 +1000
X-Mailer: Microsoft Outlook IMO, Build 9.0.6604 (9.0.2911.0)
Importance: Normal
X-Proofpoint-Virus-Version: vendor=fsecure engine=4.65.5502:2.3.11,1.2.37,4.0.164 definitions=2007-04-27_02:2007-04-26,2007-04-27,2007-04-27 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-0704270011
X-PP-SpamScore: 0
X-NIST-MailScanner: Found to be clean
X-NIST-MailScanner-From: rons@ics.mq.edu.au

Dear NIST,

I am writing in response to the call for comments on the NIST draft evaluation criteria for development of a new hash algorithm.

Before I state my main comments I'd like to explicitly state my main assumptions about NIST's goals for the new hash standard (to be called "the standard" from now on).

Assumptions

My assumptions are that NIST would like to specify a set of needed security properties for the standard which meet the following two requirements:

R1. The specified security properties are well-defined and achievable

This means that any needed security property for the hash function is well-known to the cryptographic community, is precisely specified, and is supported by theoretical evidence from the cryptographic theory literature that the property is really achievable (i.e. there exist functions published in the literature which were proven to achieve the property under a well known complexity theory assumption. Note that these functions may be impractical, but at least they show that the property is achievable in principle).

(Example: In the case of the AES competition, the security requirement on AES was that it is a Pseudo Random Permutation (PRP) Family, a well-known security property with precise definition in the crypto theory literature, and with constructions that achieve it under well-known complexity theory assumptions; in this case, the existence of a one-way function).

R2. The specified security properties guarantee the security of

the main hash function applications

This means that the security of the main applications of the hash standard were/can be demonstrated by a security reduction to follow from the specified security properties of the hash standard.

(Example: In the case of the AES competition, the PRP security requirement was known to imply security of popular block cipher "modes of operation" for encryption and authentication (e.g. CTR/CBC encryption modes and variants of CBC authentication modes are all proven secure in the crypto literature assuming the PRP security property on the underlying block cipher)).

I would like to clarify why I believe these requirements are important.

In R1, the importance of security requirements being "well-defined" is well-known in cryptography, where even apparently small subtle details can make a huge difference to security, since attacks usually exploit such subtle details. Moreover, precise definitions of security requirements are also essential to allow an objective evaluation and comparison of candidate functions. The importance of "achievable" security properties is that such properties are backed up by a sound theoretical foundation which supports their existence. Without this backing, the desired security may not be achievable and is more likely to collapse unexpectedly due to new developments. I think the importance of R2 is self-evident.

Comments

C1. The draft contains the following security requirement:

"The extent to which the algorithm output is indistinguishable from a random oracle."

This requirement is not well-defined. Although the definition of a random oracle is well known in the crypto community, it is NOT well known what it precisely means for the output of a concrete public algorithm to be indistinguishable from a random oracle.

To make this requirement well defined, one would have to specify how the input to the algorithm is chosen, and what information the adversary is given about this input. These choices can make a huge difference, for example:

- If a certain specified portion ("key") of the input to the algorithm is chosen at random from a certain well defined set, and the adversary is given access to an oracle that evaluates the ("keyed") algorithm at points of the adversary's choice (where the adversary chooses the portion of the input other than the "key"), then the requirement is the well-known and achievable

'Pseudorandom Function' (PRF) requirement.

- If the above "key" input is revealed to the adversary, then the requirement of the oracle output being indistinguishable from a random oracle is provably NOT achievable (as the adversary knows the all the inputs to the public algorithm and can evaluate it itself, and hence predict the oracle output, thus distinguishing it from the output of a random oracle, which is unpredictable except with negligible probability).

Note that many current standardised applications of hash functions (such as OAEP-RSA and PSS-RSA in the PKCS public key cryptography standard) model hash functions as random oracles in their security proofs. On the other hand, there is no known simple well-defined security requirement on a concrete public hash function algorithm that guarantees the security of all such applications (indeed, there are theoretical results on the impossibility of finding such a universal definition, see e.g. [1]). Thus in the case of existing random oracle applications such as OAEP-RSA, it may appear impossible to satisfy both requirements R1 and R2. This is indeed a difficult issue.

One option of dealing with this issue which NIST might like to consider (without significantly compromising requirements R1 and R2), is to explicitly identify the important random oracle applications that are to be supported by the standard (e.g. OAEP-RSA, PSS-RSA, and perhaps a few others), and state as a security requirement on the hash standard that each of those applications achieve their well-defined security goals (e.g. the well known IND-CCA2 indistinguishability under adaptive chosen ciphertext attack requirement for the OAEP-RSA public-key encryption scheme) when the hash standard is used to implement the random oracles in the applications (e.g. the oracles G and H in the OAEP scheme). At least in this way, the requirements are well-defined, guarantee the security of the main applications, and are believed to be achievable (the drawback is that there are no proofs of achievability under well known complexity theory assumptions, but this seems a relatively minor price to pay for the well-defined aspect, compared to a vague "indistinguishability from a random oracle" requirement).

C2. Currently it seems a main application of hash functions is the HMAC message authentication construction. Hence it seems that (by requirement R2) the standard should support this application. But none of the requirements of "first and second preimage resistance" and "collision resistance" are known to imply the security of HMAC. The latest security proof for HMAC assumes the compression function is a "Pseudo Random Function" (PRF) when keyed in a certain way (see [2]). Hence if HMAC applications are to be supported by the standard, this particular PRF requirement should be added to the needed security properties.

C3. With the current state of the art in practical provably secure cryptographic primitives, it is possible to provably achieve one of the needed hash function security properties (e.g. collision

resistance) based on a well studied hard problem, but seems difficult to achieve all security properties simultaneously (e.g. collision resistance and PRF). Note that many hash function applications only need a single security property, rather than all of them simultaneously (e.g. the collision-resistance security property alone suffices for message hashing for secure short-length signatures, for commitment schemes, and for database integrity checking). Hence, to allow provably secure solutions to be standardised for high security applications, NIST might also like to consider standardising hash functions with a single (but provable) security property (e.g. collision resistance only) for specific popular applications, rather than standardising only a single general purpose hash function possessing all security properties (which seems likely to exclude provably secure solutions).

C4. It is desirable that the security requirements for first and second preimage resistance and collision resistance, although well known, be made precise. In particular, the domain from which a random input is chosen in the preimage resistance attacks should be specified (note this is important since it is well known that collision resistance implies preimage resistance when the input domain is sufficiently large compared to the hash output space, but not when the input domain size is of the same order as the hash output space size, or smaller), and the desired security level should be specified (e.g. run-time/success probability ratio lower bound). For the collision-resistance requirement, NIST might like to consider specifying the well known achievable complexity-theory based requirement, which specifies that it is hard to find a collision for function selected from a family (by choosing a random key, e.g. "IV"), and then a collision found for the random key (if a single IV/key is to be published in the standard document for applications which do not wish to choose their own key, a single random key/IV could perhaps be generated using a publicly broadcast lottery process by NIST, and then published in the standard document). As discussed in detail in [3], without such a "family of functions", collision-resistance cannot be achieved by any function in the standard complexity theory model, but only in a "human ignorance" model, which seems less preferable.

Thank you for the opportunity to comment on the draft.

Best Regards,

Ron Steinfeld.

****

Ron Steinfeld

Centre for Advanced Computing - Algorithms and Cryptography
Department of Computing,
Macquarie University,
NSW 2109 Australia

References

[1] R. Canetti, O. Goldreich, S. Halevi, "The Random Oracle Methodology, Revisited", Journal of the ACM, Vol. 51, No. 4, July 2004, pp. 557--594.

[2] M. Bellare, "New Proofs for NMAC and HMAC: Security without Collision Resistance", In CRYPTO 2006, volume 4117 of LNCS, pages 602--619, Springer, 2006.

[3] P. Rogaway, "Formalizing Human Ignorance", In VIETCRYPT'06, volume 4341 of LNCS, pages 211-228, Springer, 2006.