Cryptographic Hash Functions: Theory and Practice

Bart Preneel

Katholieke Universiteit Leuven and IBBT
Dept. Electrical Engineering-ESAT/COSIC,
Kasteelpark Arenberg 10 Bus 2446, B-3001 Leuven, Belgium
bart.preneel@esat.kuleuven.be

Abstract. Cryptographic hash functions are an essential building block for security applications. Until 2005, the amount of theoretical research and cryptanalysis invested in this topic was rather limited. From the hundred designs published before 2005, about 80% was cryptanalyzed; this includes widely used hash functions such as MD4 and MD5. Moreover, serious shortcomings have been identified in the theoretical foundations of existing designs. In response to this hash function crisis, a large number of papers has been published with theoretical results and novel designs. In November 2007, NIST announced the start of the SHA-3 competition, with as goal to select a new hash function family by 2012. About half of the 64 submissions were broken within months. This talk will present an outline of the state of the art of hash functions half-way the competition and attempts to identify open research issues.

Cryptographic hash functions map input strings of arbitrary length to short fixed length output strings. They were introduced in cryptology in the 1976 seminal paper of Diffie and Hellman on public-key cryptography [4]. Hash functions can be used in a broad range of applications: to compute a short unique identifier of a string (e.g. for a digital signature), as one-way function to hide a string (e.g. for password protection), to commit to a string in a protocol, for key derivation and for entropy extraction.

Until the late 1980s, there were few hash function designs and most proposals were broken very quickly after their introduction. The first theoretical result is the construction of a collision-resistance hash function based on a collision-resistant compression function, proven independently by Damgård [3] and Merkle [10] in 1989. Around the same time, the first cryptographic algorithms were proposed that are intended to be fast in software; the hash functions MD4 [14] and MD5 [15] fall in this category. Both were picked up quickly by application developers as they were ten times faster than DES; in addition they were not patent-encumbered and they posed less export problems than an encryption algorithm. As a consequence, hash functions were also used to construct MAC algorithms (e.g., HMAC as analyzed by Bellare et al. [2,1]) and even block ciphers and stream ciphers.

During the 1990s, a growing number of hash functions were proposed [13], but unfortunately very few of these designs have withstood cryptanalysis. Notable

M. Soriano, S. Qing, and J. López (Eds.): ICICS 2010, LNCS 6476, pp. 1–3, 2010.

results were obtained by Dobbertin, who found collisions for MD4 in 1995 [5]. Very few theoretical results were available in the area. At the same time however, MD5 and SHA-1, the latter introduced in 1995 by NIST (National Institute for Standards and Technology, US) [7], were deployed in an ever growing number of applications, resulting in the name "Swiss army knifes" of cryptography.

Wang et al. made substantial progress in the differential cryptanalysis of hash functions of the MD4 type: in 2004 they found collisions for MD4 by hand and for MD5 in a few minutes [17]. They managed to reduce the cost of collisions for SHA-1 by three orders of magnitude [16]. Suddenly hash functions moved to the center stage in cryptology: many new theoretical results were obtained, new designs were proposed and the cryptanalytic techniques of Wang et al. were further developed. Today RIPEMD-160 [6] seems to be one of the few older 160-bit hash functions for which no shortcut attacks are known. In 2002, NIST introduced the SHA-2 family of hash functions [8] with as goal to match the security levels provided by 3-DES and AES (output results of 224 to 512 bits). Even if attempts to cryptanalyzed SHA-2 have failed so far, there is a concern that the attacks of Wang et al. would also apply to these functions, which have design principles that are quite similar to those of SHA-1.

In November 2007, NIST announced that it would organize an open competition to select the SHA-3 algorithm [11]. In October 2008, 64 candidates were submitted; 51 of these were admitted to the first round and in July 2009, 14 were selected for the second round. In December 2010, NIST will announce 4 to 6 finalists; the final winner will be announced in the second Quarter of 2012.

This talk presents an overview of the state of hash functions. We discuss the main theoretical results, describe some of the most important attacks, including the rebound attack [9]. Next we give an update on the status of the SHA-3 competition and explain why SHA-3 will be a hash function that is very different from SHA-2. One can expect that the SHA-3 competition will result in a robust hash function with a good performance, that will co-exist with SHA-2. One can also expect that NIST will standardize a tree mode for hash functions to obtain improved performance on multi-core processors (see [3,12] and several SHA-3 submissions). For the long term, we face the challenging problem to design an efficient hash function for which the security can be reduced to a mathematical problem that is elegant and for which we have a convincing security reduction.

References

1. Bellare, M.: New proofs for NMAC and HMAC: security without collisionresistance. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 602–619. Springer, Heidelberg (2006)
2. Bellare, M., Canetti, R., Krawczyk, H.: Keying hash functions for message authentication. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 1–15. Springer, Heidelberg (1996)
3. Damgård, I.B.: A design principle for hash functions. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 416–427. Springer, Heidelberg (1990)
4. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Trans. on Information Theory IT-22(6), 644–654 (1976)

5. Dobbertin, H.: Cryptanalysis of MD4. Journal of Cryptology 11(4), 253–271 (1998); see also, In: Gollmann, D. (ed.) FSE 1996. LNCS, vol. 1039, pp. 53–69. Springer, Heidelberg (1996)
6. Dobbertin, H., Bosselaers, A., Preneel, B.: RIPEMD-160: a strengthened version of RIPEMD. In: Gollmann, D. (ed.) FSE 1996. LNCS, vol. 1039, pp. 71–82. Springer, Heidelberg (1996)
7. FIPS 180-1, Secure Hash Standard, Federal Information Processing Standard (FIPS), Publication 180-1, National Institute of Standards and Technology, US Department of Commerce, Washington D.C. (April 17, 1995)
8. FIPS 180-2, Secure Hash Standard, Federal Information Processing Standard (FIPS), Publication 180-2, National Institute of Standards and Technology, US Department of Commerce, Washington D.C. (August 26, 2002) (Change notice 1 published on December 1, 2003)
9. Lamberger, M., Mendel, F., Rechberger, C., Rijmen, V., Schläffer, M.: Rebound distinguishers: results on the full Whirlpool compression function. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 126–143. Springer, Heidelberg (2009)
10. Merkle, R.: One way hash functions and DES. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 428–446. Springer, Heidelberg (1990)
11. NIST SHA-3 Competition, `http://csrc.nist.gov/groups/ST/hash/`
12. Pal, P., Sarkar, P.: PARSHA-256 – A new parallelizable hash function and a multithreaded implementation. In: Johansson, T. (ed.) FSE 2003. LNCS, vol. 2887, pp. 347–361. Springer, Heidelberg (2003)
13. Preneel, B.: Analysis and design of cryptographic hash functions. Doctoral Dissertation, Katholieke Universiteit Leuven (1993)
14. Rivest, R.L.: The MD4 message digest algorithm. In: Menezes, A., Vanstone, S.A. (eds.) CRYPTO 1990. LNCS, vol. 537, pp. 303–311. Springer, Heidelberg (1991)
15. Rivest, R.L.: The MD5 message-digest algorithm. Request for Comments (RFC) 1321, Internet Activities Board, Internet Privacy Task Force (April 1992)
16. Wang, X., Yin, Y.L., Yu, H.: Finding collisions in the full SHA-1. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 17–36. Springer, Heidelberg (2005)
17. Wang, X., Yu, H.: How to breakMD5 and other hash functions. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 19–35. Springer, Heidelberg (2005)