Asymmetric Reply to SHA-3: Russian Hash Function Draft Standard

Sergey Grebnev, Andrey Dmukh, Denis Dygin, Dmitry Matyukhin,
Vladimir Rudskoy and Vasily Shishkin

In this talk we present new Russian hash function draft standard [1] and it's design rationale. We call new hash function `Streebog`. `Streebog` supposed to be the god of rash wind in ancient slavic mythology.

Cryptographic hash functions play an increasingly important role in secure communications. Originally created as a component part of digital signatures, now hash functions are widely used in various aspects of modern cryptography including authentication, key derivation and management, deterministic random bit generation.

Basic hash function properties include pre-image, second pre-image and collision resistance. Nevertheless some applications utilize hash function in a sense as an instantiation of a random mapping. This impose additional (certificational) properties hash function should satisfy.

For many years cryptographic community payed little attention to hash functions. Situation drastically changed after revolutionary A. Joux [2] and X. Wang et al. [3, 4, 5, 6] research results. These results show structural weaknesses in a common Merkle-Damgård construction and that the magic power of a differential cryptanalysis spreads not only on ciphers but on the hash functions as well. Cryptographic community response was a great development in hash functions theory and practice. As a part of this development we stress recent results of F. Mendel et al. [7, 8] with successful attack (fortunately, only theoretically) on GOST hash function and NIST concern about it's standards which resulted in SHA-3 competition announcement [9].

Anyway recent development in hash function studies waits for reply from standardization organizations. In contrast to NIST an asymmetric decision was made to propose Russian hash function draft standard without a competition.

We assume the following design principles as a basis. Proposing only one hash function as a draft standard we face the necessity of utilizing time-approved basic constructions and transformations. Moreover we have to leave large security margin. According to our design principles `Streebog` should have deeply conservative construction resisting common and recently developed attacks.

Actually `Streebog` is two separate hash functions: one with hash-code length 512 bits and another with hash-code length 256 bits. The only difference between these hash-functions is different IV's and truncation of the output in 256-bit variant.

As a basic construction we use slightly simplified HAIFA framework [10]. Each compression function which takes the message block depends on the number of bits hashed so far. Together with finalization part of the algorithm this prevents from second pre-image attacks like Kelsey and Schneier [11] and herding attack [12].

Finalization part of `Streebog` consists of two consecutive invocations of a compression function. Message blocks for them are: the length for the whole processed message (MD-strengthening) and the sum of all processed message blocks modulo 2^{512}. Proposed finalization part makes many attacks harder to apply. These attacks include multi-collision attacks, differential attacks, rebound attack etc. Likewise our finalization part and counter of the number of bits hashed so far prevents length-extension attack.

Compression function is built from a block cipher with Miyaguchi-Preneel mode, where block cipher is AES- and Whirlpool-like substitution-permutation network with block and key length equal to 512. There are 12 full and one (the last one) simplified rounds. Full round consists of xoring round key, substitution step – the S-box applies to each byte of the state, and linear transformation for the whole state. Simplified round is just xoring round key.

`Streebog` is designed to be fast on 64-bit CPUs. We have made the speed measurements on one core of 64-bit 2x Quad-Core Intel Xeon E5335. Performance evaluation showed that `Streebog` comparable with GOST hash function for short messages and starting with 400 byte messages exceeds it with the growing of the message length. Our plain C code for `Streebog` showed about 50 cpb for 64 Kb messages.

Bibliography

[1] ГОСТ Р 34.10-20_ (проект), Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи, 2012.

[2] Antoine Joux, Multicollisions in Iterated Hash Functions. Application to Cascaded Constructions, Proceedings of CRYPTO 2004, pp. 306-316.

[3] Xiaoyun Wang, Hongbo Yu, How to Break MD5 and Other Hash Functions, Proceedings of EUROCRYPT 2005, pp. 19-35.

[4] Xiaoyun Wang, Xuejia Lai, Dengguo Feng, Hui Chen, Xiuyuan Yu, Cryptanalysis of the Hash Functions MD4 and RIPEMD, Proceedings of EUROCRYPT 2005, pp. 1-18.

[5] Xiaoyun Wang, Hongbo Yu, Yiqun Lisa Yin, Efficient Collision Search Attacks on SHA-0, Proceedings of CRYPTO 2005, pp. 1-16.

[6] Xiaoyun Wang, Yiqun Lisa Yin, Hongbo Yu, Finding Collisions in the Full SHA-1, Proceedings of CRYPTO 2005, pp. 17-36.

[7] Florian Mendel, Norbert Pramstaller, Christian Rechberger, Marcin Kontak, Janusz Szmidt, Cryptanalysis of the GOST Hash Function, Proceedings of CRYPTO 2008, pp. 162-178.

[8] Florian Mendel, Norbert Pramstaller, Christian Rechberger, A (Second) Preimage Attack on the GOST Hash Function, Proceedings of FSE 2008, pp. 224-234.

[9] National Institute of Standards and Technology, Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHAЏ3) Family, Federal Register, November, 2007, Vol. 72, No. 212, pages 62212–62220.

[10] Eli Biham, Orr Dunkelman, A Framework for Iterative Hash Functions – HAIFA, Cryptology ePrint Archive, Report 2007/278, http://eprint.iacr.org/2007/278.pdf

[11] John Kelsey, Bruce Schneier, Second Preimages on n-Bit Hash Functions for Much Less than 2^n Work, Proceedings of EUROCRYPT 2005, pp. 474-490.

[12] John Kelsey, Tadayoshi Kohno, Herding Hash Functions and the Nostradamus Attack, Proceedings of EUROCRYPT 2006, pp. 183-200.