Emerging SHA-3

brief comments on NIST SHA-3 hash function candidates

Stanisław Radziszowski
Rochester Institute of Technology
spr@cs.rit.edu

March 2011

NIST SHA-3 competition

shrinking pool of candidates

- 2008 – 64 submissions
- Round 1 – 51 acceptable candidates
- Round 2 – 14 candidates
  BLAKE, BMW, CubeHash, ECHO, Fugue, Grøstl, Hamsi, JH, Keccak, Luffa, Shabal, SHAvite-3, SIMD, Skein
- Round 3, December 2010 – 5 finalists
  BLAKE, Grøstl, JH, Keccak, Skein
- 2012 – the winner ax07bc6a

Security

General Attacks

- Trying to get any (pseudo)-collisions, or something close, for reduced (rounds) versions of candidates.
- Should we worry about preimage attacks on 256-bit hash requiring 2^{236} calls to a hash function?
- Related-key attacks on AES (Biryukov, Khovratovitch - 2009) may hurt some of the SHA-3 candidates.
- Computational statisticians, please do come forwards! How easily is $H(x)$ distinguishable from a random function?

Proofs of Security

- partial reductions
- resistance to known attacks
- still mainly hand-waiving

R·I·T
Department of Computer Science

Block cipher based designs

round 2/finalists

AES-based

- ECHO, Gilbert+ FR, wide-pipe
- Fugue, Jutla+ IBM, sponge-like
- Grøstl, Knudsen+ DK, wide-pipe
- SHAvite-3, Dunkelman+ IS, narrow-pipe

Other cipher-based

- BLAKE/ChaCha, Aumasson+ CH, narrow-pipe, ARX
- Hamsi $(+-)$, Küçük BE/TR, narrow-pipe, bitsliced
- JH/E, Wu SG, sponge-like, S-boxes
- Skein/Threefish, Schneier+ US, narrow-pipe, ARX

Which finalist is not on the list?

R·I·T Department of Computer Science

Keccak

The Keccak Team, BE

Team (from STMicroelectronics and NXP Semiconductors):
Guido Bertoni, Joan Daemen (of AES fame),
Michaël Peeters, Gilles Van Assche

- Elegant, convincing design, ideas from *Grindahl*
- Runs on a $5 \times 5 \times 2^l$ cube of bits,
  recommended 1600-bit state ($l = 6$)
- Nontrivial padding, message schedule
- Same rounds all over, except constants RC
- Sponge construction

This is my favorite hash!

Keccak

single round (out of 24), only one nonlinear step χ

$$\mathrm{R} = \iota \circ \chi \circ \pi \circ \rho \circ \theta, \text{ with}$$

$$\theta: \quad a[x][y][z] \leftarrow a[x][y][z] + \sum_{y'=0}^{4} a[x-1][y'][z] + \sum_{y'=0}^{4} a[x+1][y'][z-1],$$

$$\rho: \quad a[x][y][z] \leftarrow a[x][y][z-(t+1)(t+2)/2],$$

$$\text{with } t \text{ satisfying } 0 \le t < 24 \text{ and } \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}^t \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} \text{ in } \mathrm{GF}(5)^{2\times 2}$$

$$\text{or } t = -1 \text{ if } x = y = 0,$$

$$\pi: \quad a[x][y] \leftarrow a[x'][y'], \text{ with } \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix},$$

$$\chi: \quad a[x] \leftarrow a[x] + (a[x+1]+1)a[x+2],$$

$$\iota: \quad a \leftarrow a + \mathrm{RC}[i_r].$$

Keccak

cube attack

Dinur, Shamir - 2009+

discovering low-degree polynomial representation of keyed black box cryptographic boolean functions. Successful on the reduced version of the stream cipher Trivium, ... but not much more

Results by Joel Lathrop, MS-CS thesis

- cube attacks on unkeyed hash functions
- 4 rounds of 224- and 256-bit Keccak yield to cubes
- 7 rounds, perhaps, if we are lucky, in the future
- full 24-round Keccak is safe against cube attacks
- cited in the Keccak team reports, and in the February 2011 NIST SHA-3 Round-2 report.

R·I·T
Department of Computer Science

Skein

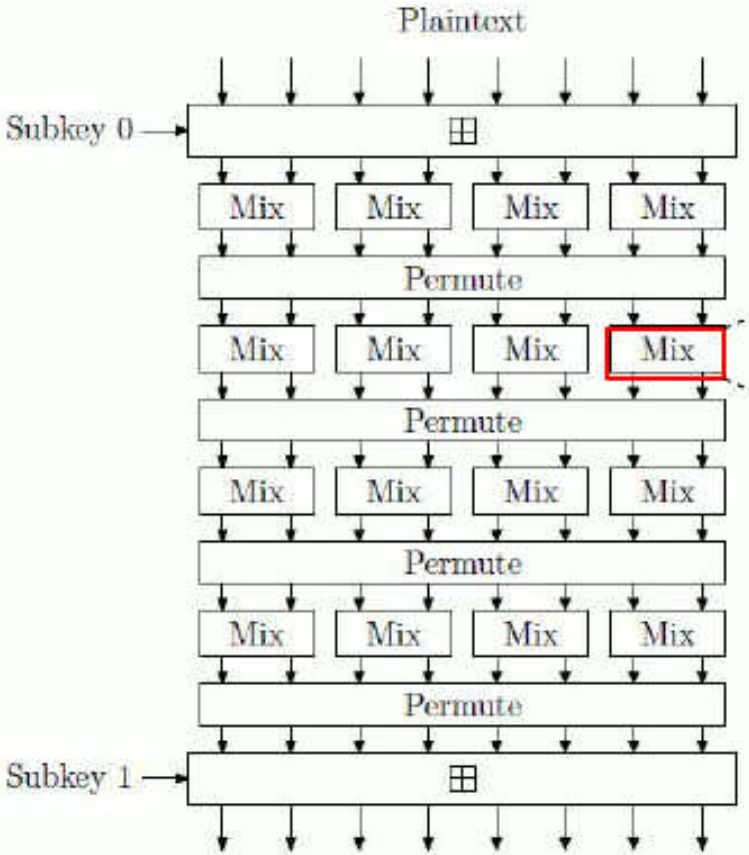
Schneier-Ferguson + team, US-UK

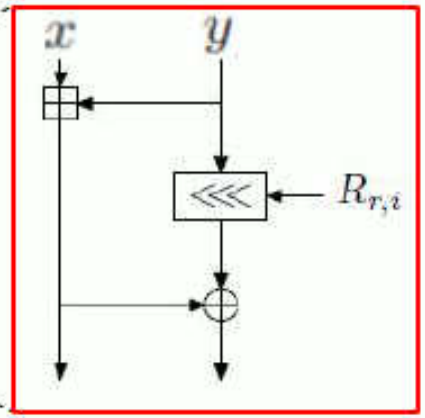
The main designers are well known authors of cryptography and security books

- Based on a tweakable block cipher Threefish (Twofish competed in 2000 to become AES)
- Nonlinearity from plain addition, no S-boxes
- Tree hashing strongly stressed
- Easy parallelizability

Skein

4 rounds (out of 72) of Threefish in Skein-512 on 64-bit blocks

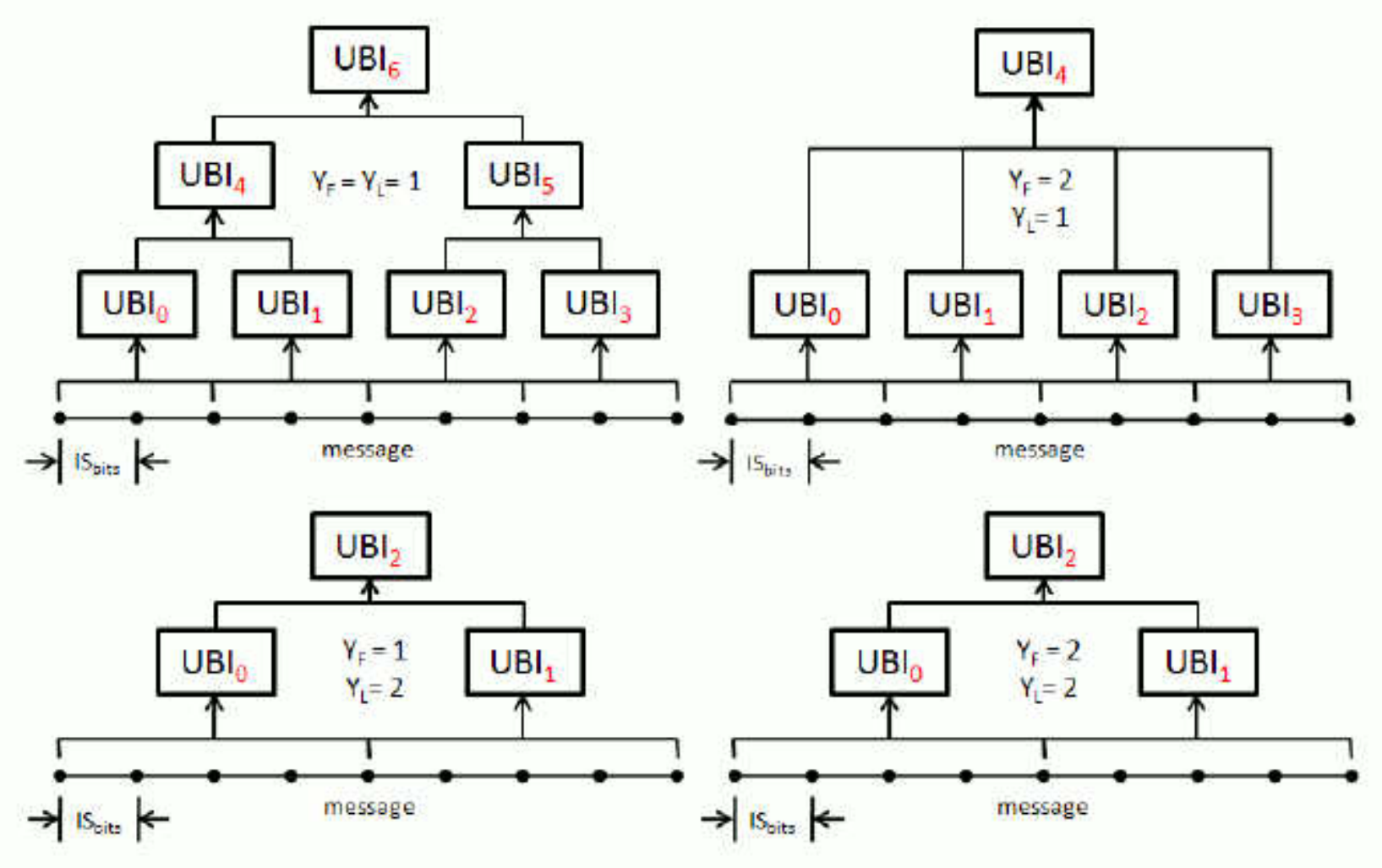




$$\mathrm{MIX}_{r,i}(x,y) = (x+y, (x+y) \oplus (y \lll R_{r,i}))$$

Skein

tree hashing modes on Unique Block Iterations



[Schorr 2010]

JH

Hongjun Wu, SG

The only solo-design among SHA-3 finalists

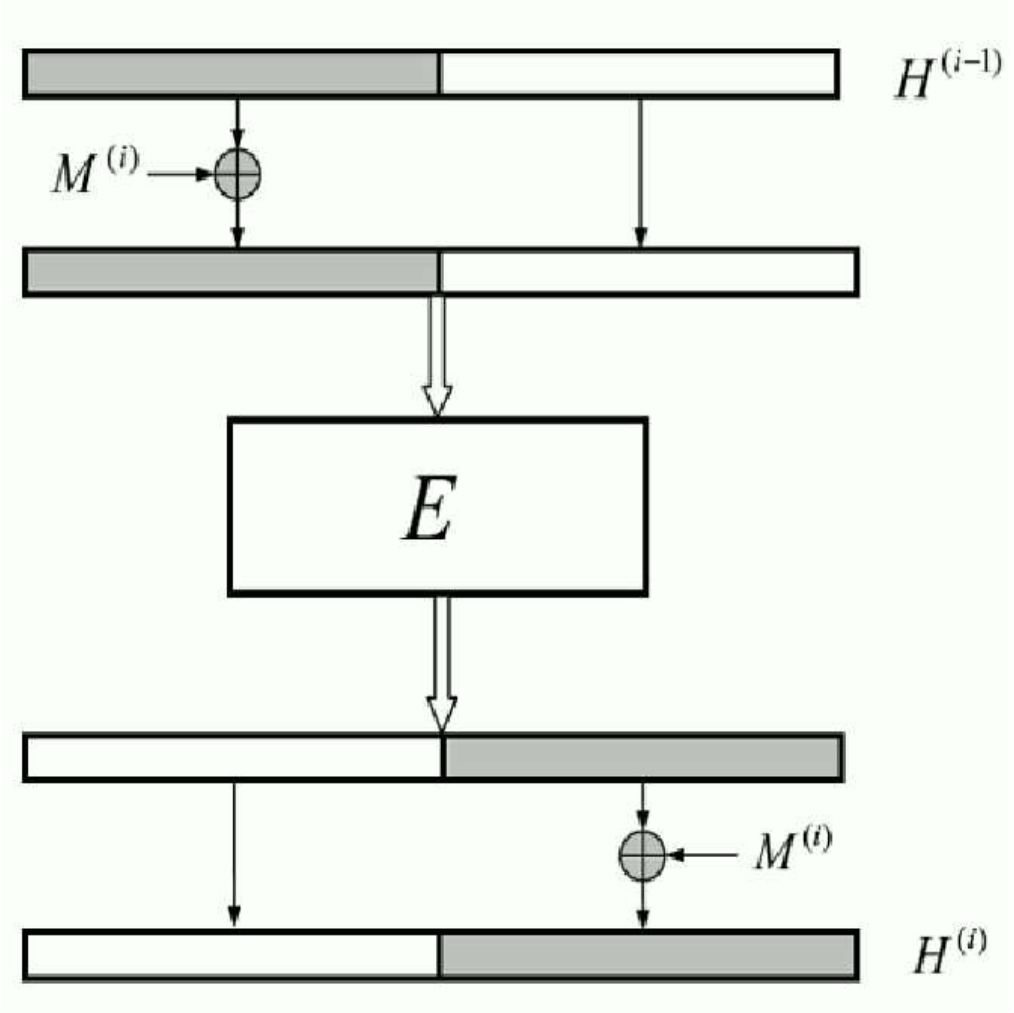
- 42-round compression within MD-template
- 1024-bit state, 512-bit input blocks
- bit slicing architecture
  many small 5/4-bit S-boxes, some say no S-boxes at all
- picks up some of the AES (Galois codes)
  and Serpent (small S-boxes) spirit
- 512-bit preimage attack using 2^{507} compressions,
  Bhattacharyya, Mandal, Nandi 2010

R·I·T
Department of Computer Science

JH

absorbing one block

Fixed keyless cipher E, 42-round SPN



Grøstl

Knudsen+team, DK

Strong AES-based candidate

- Wide-pipe Merkle-Damgård design
- Wide-trail design (not that clear how different from the latter)
- (512 or 1024)-bit blocks and chaining values
- (10 or 14) AES-type rounds on
  8 by (8 or 16) array of bytes
- AES S-box

Grøstl

Austrian Rochester-garbage-plate-like dish

- x
- Detailed slides by Joel Lathrop at
  `http://www.cs.rit.edu/ spr/gdn2010/groestl.pdf`
- x

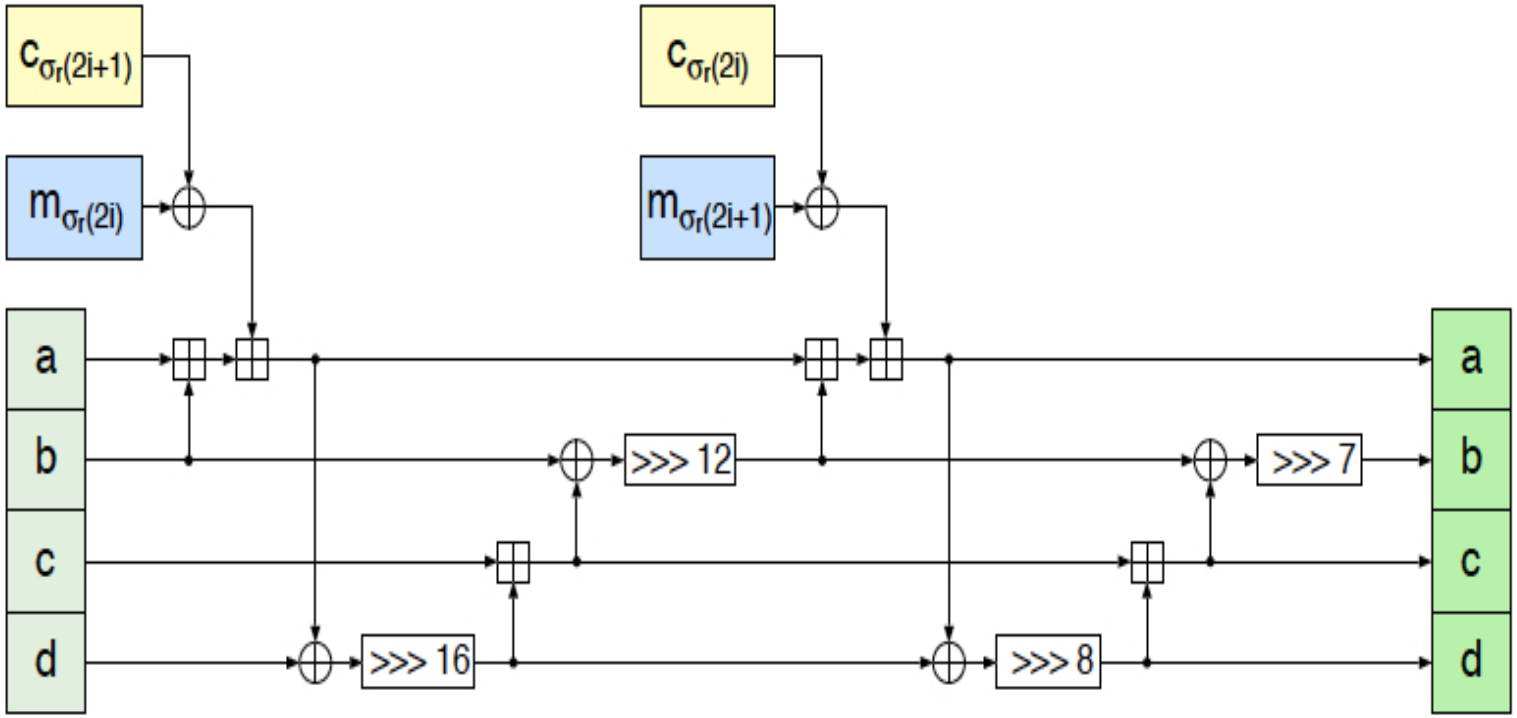
BLAKE

Aumasson+team, CH

- follows HAIFA - HAsh Iterative FrAmework
  Biham, Dunkelman - 2005, salt and the number of bits hashed so far injected into each compression
- state - 4×4 array of words (512 bits)
- compression based on ChaCha/Salsa20
  stream ciphers by Bernstein, 2008

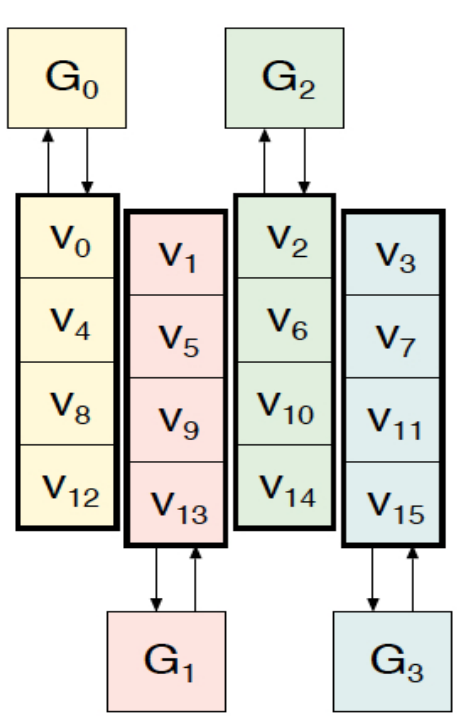
BLAKE

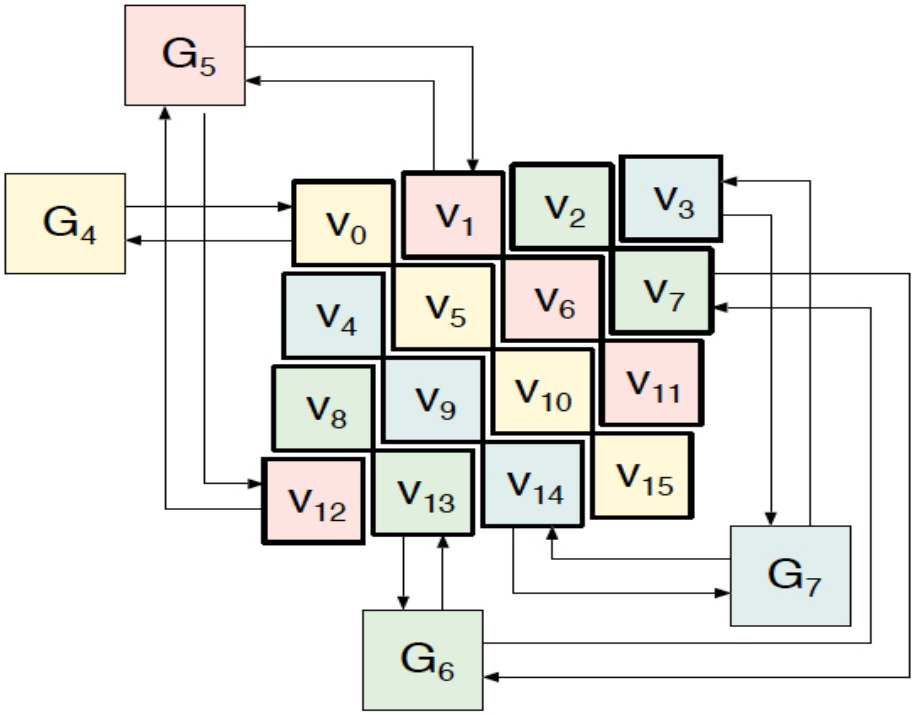
absorption



BLAKE

diffusion





Hash Summaries

The following 5 slides contain summaries of the final round hash function candidates, in alphabetical order, as described in the report NISTIR 7764, February 2011

Status Report on the First Round of the SHA-3 Cryptographic Hash Algorithm Competition

BLAKE

BLAKE follows a HAIFA iteration mode and operates on an inner state that can be represented as a four by four matrix of words. The inner state is initialized using an Initial Value (IV), a salt and a counter. The state is updated using the G function, which is based on the ChaCha stream cipher [35]. The G function updates columns and the disjoint diagonals of the state using modular addition, XOR, and rotate operations. The input message blocks and constants are selected using round-dependent fixed permutations. The nonlinearity of the design is achieved by the modular addition.

To generate 224-, 256-, 384- and 512-bit outputs, four instances are proposed: BLAKE-28, -32, -48, and -64. BLAKE-28 is similar to BLAKE-32, except for the IVs and the padding structure. A similar relationship holds for BLAKE-48 and BLAKE-64.

The tunable parameter of the design is the number of rounds, which is recommended to be ten for BLAKE-28 and -32, and fourteen for BLAKE-48 and -64, respectively. Four toy versions are defined by the submitters for analysis purposes: (i) BLOKE (with identity permutations), (ii) FLAKE (with no feed-forward), (iii) BLAZE (with zero constants), and (iv) BRAKE (with all the changes above).

BLAKE was selected as a finalist, due to its high security margin, good performance in software, and its simple and clear design.

.

Grøstl

Grøstl is a wide-pipe Merkle-Damgård hash algorithm with an output transformation. The compression function is a novel construction, using two fixed 2n-bit permutations together, in order to produce a 2n-bit compression function with the goal of achieving the collision and preimage resistance of an ideal n-bit-wide compression function. Intermediate chaining values in Grøstl-256 are 512 bits wide; for Grøstl-512, they are 1024 bits wide. The output transformation processes the final chaining state, and discards half the bits of the result, yielding an n-bit hash output.

The underlying fixed permutations are themselves based closely on the structure of AES, reusing the S-box, but expanding the size of the block to 512 bits (for Grøstl-256) or 1024 bits (for Grøstl-512) in a straightforward way.

Grøstl was selected as a finalist because of its well-understood design and solid performance, especially in hardware. While Grøstl's security margin is not ideal, NIST views it in light of the extensive amount of cryptanalysis that has been published, both on Grøstl itself and the larger AES structure on which Grøstl is based. Due to the large amount of existing cryptanalysis, NIST feels that future results are less likely to dramatically narrow Grøstl's security margin than that of the other candidates.

.

JH

The JH family of hash algorithms for different output sizes is based on a single compression function F 8 , which uses the fixed permutation E 8 . The different members of the JH family are distinguished by using different IVs. Hash values can be obtained just by a truncation of the final output. JH's domain extension is the chopped, wide-pipe Merkle-Damgård construction, where for output size s, the chaining value size n satisfies n 2s.

JH can be efficiently implemented in the bitslice mode. There are three components of the underlying permutation E 8 : four-bit S-boxes, an eight-bit L-permutation, and a P-permutation. For the S-boxes and L-permutation, JH can be directly implemented in the bitslice mode. However, for the P-permutation, 128 different shift values are used, so there is no efficient bitslice implementation for the P-permutation. Instead of implementing P directly, the submitter used seven different permutations P0P6, which can be efficiently implemented in the bitslice mode.

JH was selected as a finalist because of its solid security margin, good all-around performance, and innovative design.

.

Keccak

Keccak follows the sponge construction model [91]. The permutation can be considered as a substitution-permutation network with five-bit wide S-boxes, or as a combination of a linear mixing operation and a very simple nonlinear mixing operation. The construction of the permutation is the most innovative part of the Keccak design. The recommended security parameter for Keccak is twenty-four rounds, which was tweaked from eighteen rounds in the original submission. Additionally, the message block size was increased as the part of the first round tweak.

Keccak was selected as a finalist, mainly due to its high security margin, its high throughput and throughput-to-area ratio and the simplicity of its design.

.

Skein

Skein is an iterative hash algorithm built on a tweakable block cipher Threefish. Threefish is used to build the compression function of Skein using a modified Mateas-Meyer-Oseas construction, which is then iterated in a chaining mode similar to HAIFA. The designers refer to the whole construction as a Unique Block Iteration (UBI) mode. The UBI mode provides indifferentiability from a random oracle in the ideal cipher model [119]. Threefish is a 72-round substitution-permutation network using a 128-bit MIX function consisting of a 64-bit addition, rotate and XOR operations.

For the second round, Skein was tweaked by modifying the rotation constants, which the submitters felt had not been fully optimized for the first round.

Skein was selected as a finalist, mainly due to its high security margin and speed in software.

.

References

containing pointers to many other references

- NIST report NISTIR 7620, Status Report on the First Round of the SHA-3 Cryptographic Hash Algorithm Competition, September 2009
  `http://csrc.nist.gov/groups/ST/hash`
- The Second SHA-3 Candidate Conference, Santa Barbara, CA, Aug. 23-24, 2010
  `http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/Aug2010`
- ECRYPT SHA-3 Zoo
  `http://ehash.iaik.tugraz.at/wiki/The_SHA-3_Zoo`
- eBASH: ECRYPT Benchmarking of All Submitted Hashes
  `http://bench.cr.yp.to/ebash.html`
- NIST report NISTIR 7764, Status Report on the Second Round of the SHA-3 Cryptographic Hash Algorithm Competition, February 2011,
  `http://csrc.nist.gov/groups/ST/hash/sha-3`

.