Wikipedia
The Free Encyclopedia

Comparison of cryptographic hash functions

The following tables compare general and technical information for a number of cryptographic hash functions. See the individual functions' articles for further information. This article is not all-inclusive or necessarily up-to-date. An overview of hash function security/cryptanalysis can be found at hash function security summary.

General information

Basic general information about the cryptographic hash functions: year, designer, references, etc.

Function	Year	Designer	Derived from	Reference
BLAKE	2008	Jean-Philippe Aumasson Luca Henzen Willi Meier Raphael C.-W. Phan	ChaCha20	Website (https://131002.net/blake/) Specification (https://web.archive.org/web/20201001184633/http://131002.net/blake/blake.pdf)
BLAKE2	2012	Jean-Philippe Aumasson Samuel Neves Zooko Wilcox-O'Hearn Christian Winnerlein	BLAKE	Website (https://blake2.net/) Specification (https://blake2.net/blake2.pdf) RFC 7693 (https://datatracker.ietf.org/doc/html/rfc7693)
BLAKE3	2020	Jack O'Connor Jean-Philippe Aumasson Samuel Neves Zooko Wilcox-O'Hearn	BLAKE2	Website (https://github.com/BLAKE3-team/BLAKE3) Specification (https://github.com/BLAKE3-team/BLAKE3-specs/blob/master/blake3.pdf)
GOST R 34.11-94	1994	FAPSI and VNIIstandart	GOST 28147-89	RFC 5831 (https://datatracker.ietf.org/doc/html/rfc5831)
HAVAL	1992	Yuliang Zheng Josef Pieprzyk Jennifer Seberry		Website (https://web.archive.org/web/20150111210116/http://labs.calyptix.com/haval.php) Specification (https://web.archive.org/web/20140411060613/http://labs.calyptix.com/files/haval-paper.pdf)
KangarooTwelve	2016	Guido Bertoni Joan Daemen Michaël Peeters Gilles Van Assche	Keccak	Website (https://keccak.team/kangarootwelve.html) Specification (https://keccak.team/files/KangarooTwelve.pdf)
MD2	1989	Ronald Rivest		RFC 1319 (https://datatracker.ietf.org/doc/html/rfc1319)
MD4	1990			RFC 1320 (https://datatracker.ietf.org/doc/html/rfc1320)
MD5	1992		MD4	RFC 1321 (https://datatracker.ietf.org/doc/html/rfc1321)
MD6	2008			Website (https://groups.csail.mit.edu/cis/md6/) Specification (https://groups.csail.mit.edu/cis/md6/docs/2009-04-15-md6-report.pdf)
RIPEMD	1992	The RIPE Consortium[1]	MD4	
RIPEMD-128 RIPEMD-256 RIPEMD-160 RIPEMD-320	1996	Hans Dobbertin Antoon Bosselaers Bart Preneel	RIPEMD	Website (http://homes.esat.kuleuven.be/~bosselae/ripemd160.html) Specification (https://homes.esat.kuleuven.be/~bosselae/ripemd160/pdf/AB-9601/AB-9601.pdf)
SHA-0	1993	NSA		SHA-0 (https://web.archive.org/web/20090130063617/http://w2.eff.org/Privacy/Digital_signature/?f=fips_sha_shs.info.txt)
SHA-1	1995		SHA-0	Specification (https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf)
SHA-256 SHA-384 SHA-512	2002			
SHA-224	2004			
SHA-3 (Keccak)	2008	Guido Bertoni Joan Daemen Michaël Peeters Gilles Van Assche	RadioGatún	Website (https://keccak.team/) Specification (https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf)
Streebog	2012	FSB, InfoTeCS JSC		RFC 6986 (https://datatracker.ietf.org/doc/html/rfc6986)
Tiger	1995	Ross Anderson Eli Biham		Website (https://www.cs.technion.ac.il/~biham/Reports/Tiger/) Specification (https://www.cs.technion.ac.il/~biham/Reports/Tiger/tiger/node3.html)
Whirlpool	2004	Vincent Rijmen Paulo Barreto		Website (https://web.archive.org/web/20171129084214/http://www.larc.usp.br/~pbarreto/WhirlpoolPage.html)

Parameters

Algorithm	Output size (bits)	Internal state size[note 1]	Block size	Length size	Word size	Rounds
BLAKE2b	512	512	1024	128[note 2]	64	12
BLAKE2s	256	256	512	64[note 3]	32	10
BLAKE3	Unlimited	256[note 4]	512	64	32	7
GOST	256	256	256	256	32	32
HAVAL	256/224/192/160/128	256	1024	64	32	3/4/5
MD2	128	384	128	–	32	18
MD4	128	128	512	64	32	3
MD5	128	128	512	64	32	64
PANAMA	256	8736	256	–	32	–
RadioGatún	Unlimited[note 5]	58 words	19 words[note 6]	–	1–64[note 7]	18[note 8]
RIPEMD	128	128	512	64	32	48
RIPEMD-128, -256	128/256	128/256	512	64	32	64
RIPEMD-160	160	160	512	64	32	80
RIPEMD-320	320	320	512	64	32	80
SHA-0	160	160	512	64	32	80
SHA-1	160	160	512	64	32	80
SHA-224, -256	224/256	256	512	64	32	64
SHA-384, -512, -512/224, -512/256	384/512/224/256	512	1024	128	64	80
SHA-3	224/256/384/512[note 9]	1600	1600 - 2*bits	–[note 10]	64	24
SHA3-224	224	1600	1152	–	64	24
SHA3-256	256	1600	1088	–	64	24
SHA3-384	384	1600	832	–	64	24
SHA3-512	512	1600	576	–	64	24
Tiger(2)-192/160/128	192/160/128	192	512	64	64	24
Whirlpool	512	512	512	256	8	10

Notes

1. The *internal state* here means the "internal hash sum" after each compression of a data block. Most hash algorithms also internally use some additional variables such as length of the data compressed so far since that is needed for the length padding in the end. See the Merkle–Damgård construction for details.
2. The size of BLAKE2b's message length counter is 128-bit, but it counts message length in bytes, not in bits like the other hash functions in the comparison. It can hence handle eight times longer messages than a 128-bit length size would suggest (one byte equaling eight bits). A length size of 131-bit is the comparable length size ($8 \times 2^{128} = 2^{131}$).
3. The size of BLAKE2s's message length counter is 64-bit, but it counts message length in bytes, not in bits like the other hash functions in the comparison. It can hence handle eight times longer messages than a 64-bit length size would suggest (one byte equaling eight bits). A length size of 67-bit is the comparable length size ($8 \times 2^{64} = 2^{67}$).
4. The full BLAKE3 incremental state includes a chaining value stack up to 1728 bytes in size. However, the compression function itself does not access this stack. A smaller stack can also be used if the maximum input length is restricted.
5. RadioGatún is an extendable-output function which means it has an output of unlimited size. The official test vectors are 256-bit hashes. RadioGatún claims to have the security level of a cryptographic sponge function 19 words in size, which means the 32-bit version has the security of a 304-bit hash when looking at preimage attacks, but the security of a 608-bit hash when looking at collision attacks. The 64-bit version, likewise, has the security of a 608-bit or 1216-bit hash. For the purposes of determining how vulnerable RadioGatún is to length extension attacks, only two words of its 58-word state are output between hash compression operations.
6. RadioGatún is not a Merkle–Damgård construction and, as such, does not have a block size. Its belt is 39 words in size; its mill, which is the closest thing RadioGatún has to a "block", is 19 words in size.
7. Only the 32-bit and 64-bit versions of RadioGatún have official test vectors

8. The 18 blank rounds are only applied once in RadioGatún, between the end of the input mapping stage and before the generation of output bits
9. Although the underlying algorithm Keccak has arbitrary hash lengths, the NIST specified 224, 256, 384 and 512 bits output as valid modes for SHA-3.
10. Implementation dependent; as per section 7, second paragraph from the bottom of page 22, of FIPS PUB 202.

Compression function

The following tables compare technical information for compression functions of cryptographic hash functions. The information comes from the specifications, please refer to them for more details.

<table>
<tr><th rowspan="2">Function</th><th colspan="7">Size (bits)[note 1]</th><th rowspan="2">Words × Passes = Rounds[note 2]</th><th rowspan="2">Operations[note 3]</th><th rowspan="2">Endian[note 4]</th></tr>
<tr><th>Word</th><th>Digest</th><th>Chaining values[note 5]</th><th>Computation values[note 6]</th><th>Block</th><th>Length [note 7]</th></tr>
<tr><td>GOST R 34.11-94</td><td>32</td><td colspan="3">×8 = 256</td><td>×8 = 256</td><td>32</td><td>4</td><td>A B L S</td><td>Little</td></tr>
<tr><td>HAVAL-3-128</td><td rowspan="15">32</td><td>×4 = 128</td><td colspan="2" rowspan="15">×8 = 256</td><td rowspan="15">×32 = 1,024</td><td rowspan="15">64</td><td rowspan="5">32 × 3 = 96</td><td rowspan="15">A B S</td><td rowspan="15">Little</td></tr>
<tr><td>HAVAL-3-160</td><td>×5 = 160</td></tr>
<tr><td>HAVAL-3-192</td><td>×6 = 192</td></tr>
<tr><td>HAVAL-3-224</td><td>×7 = 224</td></tr>
<tr><td>HAVAL-3-256</td><td>×8 = 256</td></tr>
<tr><td>HAVAL-4-128</td><td>×4 = 128</td><td rowspan="5">32 × 4 = 128</td></tr>
<tr><td>HAVAL-4-160</td><td>×5 = 160</td></tr>
<tr><td>HAVAL-4-192</td><td>×6 = 192</td></tr>
<tr><td>HAVAL-4-224</td><td>×7 = 224</td></tr>
<tr><td>HAVAL-4-256</td><td>×8 = 256</td></tr>
<tr><td>HAVAL-5-128</td><td>×4 = 128</td><td rowspan="5">32 × 5 = 160</td></tr>
<tr><td>HAVAL-5-160</td><td>×5 = 160</td></tr>
<tr><td>HAVAL-5-192</td><td>×6 = 192</td></tr>
<tr><td>HAVAL-5-224</td><td>×7 = 224</td></tr>
<tr><td>HAVAL-5-256</td><td>×8 = 256</td></tr>
<tr><td>MD2</td><td>8</td><td>×16 = 128</td><td>×32 = 256</td><td>×48 = 384</td><td>×16 = 128</td><td>None</td><td>48 × 18 = 864</td><td>B</td><td>N/A</td></tr>
<tr><td>MD4</td><td rowspan="2">32</td><td colspan="3" rowspan="2">×4 = 128</td><td rowspan="2">×16 = 512</td><td rowspan="2">64</td><td>16 × 3 = 48</td><td rowspan="2">A B S</td><td rowspan="2">Little</td></tr>
<tr><td>MD5</td><td>16 × 4 = 64</td></tr>
<tr><td>RIPEMD</td><td rowspan="5">32</td><td colspan="2" rowspan="2">×4 = 128</td><td rowspan="3">×8 = 256</td><td rowspan="5">×16 = 512</td><td rowspan="5">64</td><td>16 × 3 = 48</td><td rowspan="5">A B S</td><td rowspan="5">Little</td></tr>
<tr><td>RIPEMD-128</td><td rowspan="2">16 × 4 = 64</td></tr>
<tr><td>RIPEMD-256</td><td colspan="2">×8 = 256</td></tr>
<tr><td>RIPEMD-160</td><td colspan="2">×5 = 160</td><td rowspan="2">×10 = 320</td><td rowspan="2">16 × 5 = 80</td></tr>
<tr><td>RIPEMD-320</td><td colspan="2">×10 = 320</td></tr>
<tr><td>SHA-0</td><td rowspan="4">32</td><td colspan="3" rowspan="2">×5 = 160</td><td rowspan="4">×16 = 512</td><td rowspan="4">64</td><td rowspan="2">16 × 5 = 80</td><td rowspan="6">A B S</td><td rowspan="6">Big</td></tr>
<tr><td>SHA-1</td></tr>
<tr><td>SHA-256</td><td>×8 = 256</td><td colspan="2" rowspan="2">×8 = 256</td><td rowspan="2">16 × 4 = 64</td></tr>
<tr><td>SHA-224</td><td>×7 = 224</td></tr>
<tr><td>SHA-512</td><td rowspan="2">64</td><td>×8 = 512</td><td colspan="2" rowspan="2">×8 = 512</td><td rowspan="2">×16 = 1024</td><td rowspan="2">128</td><td rowspan="2">16 × 5 = 80</td></tr>
<tr><td>SHA-384</td><td>×6 = 384</td></tr>
<tr><td>Tiger-192</td><td rowspan="2">64</td><td>×3 = 192</td><td colspan="2" rowspan="2">×3 = 192</td><td rowspan="2">×8 = 512</td><td rowspan="2">64</td><td rowspan="2">8 × 3 = 24</td><td rowspan="2">A B L S</td><td rowspan="2">Not Specified</td></tr>
<tr><td>Tiger-160</td><td>×2.5=160</td></tr>
</table>

Tiger-128		×2 = 128							
Function	**Word**	**Digest**	**Chaining values**	**Computation values**	**Block**	**Length**	**Words × Passes = Rounds**	**Operations**	**Endian**
	Size (bits)								

Notes

1. The omitted multiplicands are word sizes.
2. Some authors interchange passes and rounds.
3. A: addition, subtraction; B: bitwise operation; L: lookup table; S: shift, rotation.
4. It refers to *byte* endianness only. If the operations consist of bitwise operations and lookup tables only, the endianness is irrelevant.
5. The size of message digest equals to the size of chaining values usually. In truncated versions of certain cryptographic hash functions such as SHA-384, the former is less than the latter.
6. The size of chaining values equals to the size of computation values usually. In certain cryptographic hash functions such as RIPEMD-160, the former is less than the latter because RIPEMD-160 use two sets of parallel computation values and then combine into a single set of chaining values.
7. The maximum input size = $2^{\text{length size}} - 1$ bits. For example, the maximum input size of SHA-1 = $2^{64} - 1$ bits.

See also

- List of hash functions
- Hash function security summary
- Word (computer architecture)

References

1. Dobbertin, Hans; Bosselaers, Antoon; Preneel, Bart (21–23 February 1996). *RIPEMD-160: A strengthened version of RIPEMD* (https://homes.esat.kuleuven.be/~bosselae/ripemd160/pdf/AB-9601/AB-9601.pdf) (PDF). Fast Software Encryption. Third International Workshop. Cambridge, UK. pp. 71–82. doi:10.1007/3-540-60865-6_44 (https://doi.org/10.1007%2F3-540-60865-6_44).

External links

- ECRYPT Benchmarking of Cryptographic Hashes (https://bench.cr.yp.to/results-hash.html) – measurements of hash function speed on various platforms
- The ECRYPT Hash Function Website (https://ehash.iaik.tugraz.at/wiki/The_eHash_Main_Page) – A wiki for cryptographic hash functions
- SHA-3 Project (https://csrc.nist.gov/projects/hash-functions/sha-3-project) – Information about SHA-3 competition