Tiger (cryptography)

From Wikipedia, the free encyclopedia
(Redirected from Tiger (hash))

In cryptography, **Tiger**[1] is a cryptographic hash function designed by Ross Anderson and Eli Biham in 1995 for efficiency on 64-bit platforms. The size of a Tiger hash value is 192 bits. Truncated versions (known as Tiger/128 and Tiger/160) can be used for compatibility with protocols assuming a particular hash size. Unlike the SHA-2 family, no distinguishing initialization values are defined; they are simply prefixes of the full Tiger/192 hash value.

Tiger

General	
Designers	Ross Anderson and Eli Biham
First published	1996
Detail	
Digest sizes	192, 128, 160
Rounds	24

Tiger2[2] is a variant where the message is padded by first appending a byte with the hexadecimal value of 0x80 as in MD4, MD5 and SHA, rather than with the hexadecimal value of 0x01 as in the case of Tiger. The two variants are otherwise identical.

Contents [hide]

Algorithm [edit]

Tiger is designed using the nearly universal Merkle-Damgård paradigm. The one-way compression function operates on 64-bit words, maintaining 3 words of state and processing 8 words of data. There are 24 rounds, using a combination of operation mixing with XOR and addition/subtraction, rotates, and S-box lookups, and a fairly intricate key scheduling algorithm for deriving 24 round keys from the 8 input words.

Although fast in software, Tiger's large S-boxes (4 S-boxes, each with 256 64-bit entries totals 8 KiB) make implementations in hardware or small microcontrollers difficult.

Usage [edit]

Tiger is frequently used in Merkle hash tree form, where it is referred to as TTH (Tiger Tree Hash). TTH is used by many clients on the Direct Connect and Gnutella file sharing networks.

Tiger was considered for inclusion in the OpenPGP standard, but was abandoned in favor of RIPEMD-160.[3][4]

Byte Order [edit]

The specification of Tiger does not define the way the output of Tiger should be printed but only defines the result to be three ordered 64-bit integers. The "testtiger" program at the author's homepage was intended to allow easy testing of the test source code, rather than to define any particular print order. The protocols Direct Connect and ADC as well as the program tthsum use little-endian byte order, which is also preferred by one of the authors.[5]

Examples [edit]

In the example below, the 192-bit (24-byte) Tiger hashes are represented as 48 hexadecimal digits in little-endian byte order. The following demonstrates a 43-byte ASCII input and the corresponding Tiger hashes:

```
Tiger("The quick brown fox jumps over the lazy dog") =
```

```
6d12a41e72e644f017b6f0e2f7b44c6285f06dd5d2c5b075

Tiger2("The quick brown fox jumps over the lazy dog") =
976abff8062a2e9dcea3a1ace966ed9c19cb85558b4976d8
```

Even a small change in the message will (with overwhelming probability) result in a completely different hash, e.g. changing `d` to `c`:

```
Tiger("The quick brown fox jumps over the lazy cog") =
a8f04b0f7201a0d728101c9d26525b31764a3493fcd8458f

Tiger2("The quick brown fox jumps over the lazy cog") =
09c11330283a27efb51930aa7dc1ec624ff738a8d9bdd3df
```

The hash of the zero-length string is:

```
Tiger("") =
3293ac630c13f0245f92bbb1766e16167a4e58492dde73f3

Tiger2("") =
4441be75f6018773c206c22745374b924aa8313fef919f41
```

Cryptanalysis [edit]

Unlike MD5 or SHA-0/1, there are no known effective attacks on the full 24-round Tiger[6] except for pseudo-near collision.[7] While MD5 processes its state with 64 simple 32-bit operations per 512-bit block and SHA-1 with 80, Tiger updates its state with a total of 144 such operations per 512-bit block, additionally strengthened by large S-box look-ups.

John Kelsey and Stefan Lucks have found a collision-finding attack on 16-round Tiger with a time complexity equivalent to about 2^{44} compression function invocations and another attack that finds pseudo-near collisions in 20-round Tiger with work less than that of 2^{48} compression function invocations.[6] Florian Mendel et al. have improved upon these attacks by describing a collision attack spanning 19 rounds of Tiger, and a 22-round pseudo-near-collision attack. These attacks require a work effort equivalent to about 2^{62} and 2^{44} evaluations of the Tiger compression function, respectively.[8]

See also [edit]

- Comparison of cryptographic hash functions
- List of hash functions
- Serpent — A block cipher by the same authors

References [edit]

1. ^ Ross Anderson and Eli Biham, Tiger — A Fast New Hash Function, proceedings of Fast Software Encryption 3, Cambridge, 1996
2. ^ Project NESSIE, Tiger2 Test Vectors
3. ^ Callas, Jon (2004-08-18). "Re: re-consideration of TIGER". Archived from the original on 2014-07-06.
4. ^ Pornin, Thomas (2013-10-25). "How do you use the Tiger hash function with GPG?".
5. ^ Digest::Tiger Perl module
6. ^ *a* *b* John Kelsey and Stefan Lucks, Collisions and Near-Collisions for Reduced-Round Tiger, proceedings of Fast Software Encryption 13, Graz, 2006 (PDF)
7. ^ Mendel, Florian; Rijmen Vincent. "Cryptanalysis of the Tiger Hash Function". *ASIACRYPT 2007*. Springer Berlin / Heidelberg. pp. 536–550. doi:10.1007/978-3-540-76900-2_33.
8. ^ Florian Mendel, Bart Preneel, Vincent Rijmen, Hirotaka Yoshida, and Dai Watanabe, Update on Tiger, proceedings of Indocrypt 7, Kolkata, 2006

External links [edit]

- The Tiger home page

v · t · e **Hash functions & message authentication codes**

Security summary

Common functions	MD5 · SHA-1 · SHA-2 · SHA-3/Keccak
SHA-3 finalists	BLAKE · Grøstl · JH · Skein · Keccak (winner)
Other functions	FSB · ECOH · GOST · HAS-160 · HAVAL · LM hash · MDC-2 · MD2 · MD4 · MD6 · N-Hash · RadioGatún · RIPEMD · SipHash · Snefru · Streebog · SWIFFT · **Tiger** · VSH · WHIRLPOOL · crypt(3) (DES)
MAC algorithms	DAA · CBC-MAC · HMAC · OMAC/CMAC · PMAC · VMAC · UMAC · Poly1305-AES
Authenticated encryption modes	CCM · CWC · EAX · GCM · IAPM · OCB
Attacks	Collision attack · Preimage attack · Birthday attack · Brute force attack · Rainbow table · Distinguishing attack · Side-channel attack · Length extension attack
Design	Avalanche effect · Hash collision · Merkle–Damgård construction
Standardization	CRYPTREC · NESSIE · NIST hash function competition
Utilization	Salt · Key stretching · Message authentication

Cryptography

History of cryptography · Cryptanalysis · Cryptography portal · Outline of cryptography

Symmetric-key algorithm · Block cipher · Stream cipher · Public-key cryptography · Cryptographic hash function · Message authentication code · Random numbers · Steganography

This page was last modified on 25 June 2015, at 14:20.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.