The First Two Rounds of MD4 are Not One-Way

Hans Dobbertin

German Information Security Agency
e-mail: `dobbertin@skom.rhein.de`

March 21, 1997

In [1] it was shown that there are very effective attacks leading to collisions for the hash function MD4 designed by R. Rivest [3]. A summary of the status of hash functions of the MD4-family with respect to collision-resistence can be found in [2] and [4]. However, attacking the one-wayness of a hash function is a much more demanding challenge, and in case of success it has much more devastating consequences. No result along this line is known for MD4 and its successors. Therefore it is worth to explore how the recently developed new analytic methods for finding collisions can be applied to construct preimages or second preimages. As a first step, we state here the following partial result:[1]

Denote by $\mathsf{MD4}^{[12]}$ the reduced version of MD4, where the third round of its underlying three-round compression function is cancelled, but everything else of its specification is kept (e.g. initial value, padding rule).

$$\boxed{\mathsf{MD4}^{[12]} \textit{ is not one-way.}}$$

It takes less than one hour to find preimages on a PC. Second preimages take a few minutes or even less than a millisecond if the initial value is free.

Example. Assume we want to find a preimage of

$$V = \texttt{0x00000000 0x00000000 0x00000000 0x00000000}.$$

We have constructed the following message $M = M_0||M_1||...||M_{28}$, which is hashed to this value V by $\mathsf{MD4}^{[12]}$:

$M_0 =$ 0xB5B6AC17	$M_8 =$ 0xFF9405C3	$M_{16} =$ 0x814F4825	$M_{24} =$ 0x847064AD
$M_1 =$ 0xB5B6AC17	$M_9 =$ 0xFF9405C3	$M_{17} =$ 0x814F4825	$M_{25} =$ 0x05DDD0F5
$M_2 =$ 0xB5B6AC17	$M_{10} =$ 0xFF9405C3	$M_{18} =$ 0x814F4825	$M_{26} =$ 0xD462FA71
$M_3 =$ 0x85574A58	$M_{11} =$ 0xC26EA1D5	$M_{19} =$ 0x814F4825	$M_{27} =$ 0x56A79DEC
$M_4 =$ 0x4353212D	$M_{12} =$ 0x015CB5D0	$M_{20} =$ 0x9919C508	$M_{28} =$ 0x00000080
$M_5 =$ 0x4353212D	$M_{13} =$ 0x81BBD193	$M_{21} =$ 0x9919C508	
$M_6 =$ 0x4353212D	$M_{14} =$ 0x1DEF9763	$M_{22} =$ 0x9919C508	
$M_7 =$ 0x3E30333E	$M_{15} =$ 0xADE9028B	$M_{23} =$ 0x2FD7B0F9	

[1] The attack will be explained in the forth-coming paper *"Cryptanalysis of MD4"* (extended version of [1]), to appear in the Journal of Cryptology.

We anticipate that a similar attack works for the last two rounds of MD4.

Technical Details for Checking the Example. According to the padding rule, before processing, a message has to be extended by a bit string

$$P = \texttt{100...0 (bin)} \,||\, \ell,$$

where ℓ is the 64-bit representation of the bit-length of the (unextended) message. In P, between `1` on the left and ℓ on the right side, the minimal number of zeros is placed such that the bit-length of the extended message becomes a multiple of $512 = 16 \times 32$ (to allow an iterative application of the compression function, which takes 16 words as input). The above M has bit-length 29×32. This means that

$$\ell = \texttt{00000000000003A0 (hex)},$$

and P is a 96-bit string with the little-endian representation $P = P_0 \,||\, P_1 \,||\, P_2$:

$$P_0 = \texttt{0x00000080},$$
$$P_1 = \texttt{0x000003A0},$$
$$P_2 = \texttt{0x00000000}.$$

Denote by $\mathsf{MD4}^{[12]}\text{-}\mathsf{Compress}$ the compression function of $\mathsf{MD4}^{[12]}$, i.e. the first two rounds of the MD4 compression function. In order to compute the $\mathsf{MD4}^{[12]}$ hash value of M, first $\mathsf{MD4}^{[12]}\text{-}\mathsf{Compress}$ is applied to $M_0, ..., M_{15}$ with the following fixed initial value, which is a part of the specification of MD4:

$$IV = \texttt{0x67452310 0xEFCDAB89 0x98BADCFE 0x10325476}.$$

This gives the output

$$C = \mathsf{MD4}^{[12]}\text{-}\mathsf{Compress}(IV; M_0, ..., M_{15})$$
$$= \texttt{0xA86FDECC 0x25BF84C9 0xDB95C842 0xD0B260B9}.$$

C is then the initial value for the second application of $\mathsf{MD4}^{[12]}\text{-}\mathsf{Compress}$ with $M_{16}, ..., M_{28}, P_0, P_1, P_2$ as input. The output is the hash value of M:

$$\mathsf{MD4}^{[12]}(M) = \mathsf{MD4}^{[12]}\text{-}\mathsf{Compress}(C; M_{16}, ..., M_{28}, P_0, P_1, P_2)$$
$$= \texttt{0x00000000 0x00000000 0x00000000 0x00000000}.$$

References

1. H. Dobbertin, *Cryptanalysis of MD4*, Fast Software Encryption (Third Workshop on Cryptographic Algorithms, Cambridge 1996), Lecture Notes in Computer Science, Springer-Verlag 1996, pp. 55-72.
2. H. Dobbertin, *The status of MD5 after a recent attack*, CryptoBytes, The technical newsletter of RSA Laboratories, vol. 2/2, Sommer 1996, pp. 1-6.
3. R. Rivest, *The MD4 message-digest algorithm*, Request for Comments (RFC) 1320, Internet Activities Board, Internet Privacy Task Force, April 1992.
4. M.J.B. Robshaw, *On recent results for MD2, MD4 and MD5*, Bulletin 4, RSA Laboratories, November 1996 (see http://www.rsa.com/PUBS/).