Practical distinguisher for the compression function of Blue Midnight Wish

Jean-Philippe Aumasson

Nagravision SA, Cheseaux, Switzerland

Abstract. This note presents distinguishers for the compression functions of Blue Midnight Wish-256 and -512, with data complexity of 2^{19} pairs of images of uniformly random unknown inputs with a given difference.

Blue Midnight Wish (BMW) is a second round candidate in NIST's SHA-3 competition, which was "tweaked" after the first round to avoid weaknesses discovered by Thomsen [2]. So far no attack is known for the tweaked BMW (or for its compression function).

This note describes practical distinguishers for the compression functions of BMW-256 and BMW-512. We refer to [1] for a description of its algorithm and of the notations. Below we describe our strategy in detail for BMW-512: we first enumerate observations about the propagation of differences in its compression function, then we describe the actual distinguisher.

High-probability differentials for f_0. Given any difference Δ (with respect to XOR) in H_i and M_i, for some i in $\{0, \ldots, 15\}$, the output $Q_0, \ldots, Q_{15}$ has a difference only in $Q_{(i-1) \bmod 16}$. For random inputs, this difference is Δ with probability $2^{|\Delta \wedge \mathtt{7FF\ldots FF}|}$, where $|\cdot|$ denotes the Hamming weight.

For our distinguisher, we shall consider a difference Δ in H_1, M_1, H_5, M_5 (the choice of these indices and of Δ is explained below).

Pseudo-T-function behavior of the *expand*$_2$ function. Recall that a T-function $T : x \mapsto y$ acting on (say) a 64-bit word is such that the i-th bit of y does not depend on the $(i + j)$-th bits of x ($j = 1, \ldots, 65 - i$).

Observe that in *expand*$_2$, the $r_1, \ldots, r_7$ are rotations towards MSB's; hence as long as input differences are not too close to the MSB, they will only propagate towards MSB's. *expand*$_2$ also uses the functions s_4 and s_5 that are essentially right-shifts of one and two positions, which makes diffusion towards LSB's very slow.

We shall exploit that "pseudo T-function" behavior in order to minimize differences in the LSB's of the Q_i's, $i = 16, \ldots, 31$. But first, we explain how to avoid differences in the LSB's during the two calls to *expand*$_1$ (which precedes 14 calls to *expand*$_2$).

Good differences for the first *expand*$_1$ function. In *expand*$_1$, suppose we have a same difference Δ in (say) Q_{j-16} and Q_{j-12}; both these words enter the same function s_1, which is linear. Hence the two differences $s_1(\Delta)$ can cancel themselves in *expand*$_1$. In particular, Q_{16} can be free of difference. However, the bias exploited by our distinguisher does note come from a zero difference in Q_{16}, but rather from a difference Δ (as explained later).

Recall that from a difference Δ in H_1, M_1, H_5, M_5 we obtain after f_0 a difference Δ in Q_0 and Q_4 with high probability. We chose those indices to minimize the diffusion of differences (note that Q_0 is used once and Q_4 five times, which is optimal). We then searched for a difference Δ that minimizes the differences in the LSB's, and such that the sum of two differences $s_1(\Delta)$ gives Δ with high probability. Such a Δ exist, and the best we found is $\Delta = \mathtt{0000400}\ldots\mathtt{00}$. We can thus obtain a difference Δ in Q_{16} with high probability.

Good differences for the second *expand*$_1$ function. Note that in the second call to *expand*$_1$, only two of the state words used contain a difference (Q_4 and Q_{16}). Luckily, these words enter the same function s_0, and the two differences $s_0(\Delta)$ can cancel themselves.

For our distinguisher, however, we don't need Q_{17} to have zero differences, but only differences close to the MSB. After the second $expand_1$, we thus have differences only in the MSB's of the message and state words (note that the function *AddElement*, which processes message words, also has a slow diffusion towards LSB's). Since the subsequent calls to $expand_2$ are pseudo-T-functions, we can thus expect differences to propagate slowly towards LSB's.

Propagation of good differences and exploit. As an example, we give below an example of differential characteristic obtained with the above strategy:

Q_{16}	`0000400000000000`	Q_{24}	`26D7A46760968000`
Q_{17}	`0001C00000000000`	Q_{25}	`D2C4CFB637460000`
Q_{18}	`3F51D00000000000`	Q_{26}	`48D2B05C28EEC210`
Q_{19}	`11857C1800000000`	Q_{27}	`FA4544FE30A35110`
Q_{20}	`739DFB2600000000`	Q_{28}	`2A44095E9D7C9BAD`
Q_{21}	`B299152486000000`	Q_{29}	`B26C7ACE6D57F268`
Q_{22}	`BBD0F2CF26800000`	Q_{30}	`40421B4BD09BA528`
Q_{23}	`A94A246A0F380000`	Q_{31}	`17C315BA83521432`

Hence, with high probability the first four bits of Q_i, $i = 16, \dots, 27$, will have no difference. Now observe that in f_2 (i.e., the finalization of the compression function) the five LSB's of the new H_0 only depend on the five LSB's of $Q_0, Q_{16}, Q_{17}, \dots, Q_{24}$ and on the 6-th to 10-th bits of Q_{16}. As described above, given a difference in the 47-th bit of H_1, M_1, H_5, M_5, all those bits will be free of difference with high probability. One can thus distinguish the compression function of BMW-512 from a random function by querying for the images of random *unknown* pairs of values with that differences, and checking whether the differences in the four LSB's of the new H_0 are biased.

To estimate the number of samples required, we need to estimate the probability that Q_{16} has a difference Δ, and that in Q_{17} there is no difference in the least significant half of the word (that is, we need the bit difference(s) caused by $ROTL^{37}$ to vanish). First, Q_0 and Q_4 will both have difference Δ with probability 1/4. Given these differences, Q_{16} will have difference Δ with probability approximately $(1/2)^3 \times (1/2)^2 = 1/2^5$. Then, Q_{17} has a good difference with probability $(1/2)^2$. All the conditions are thus satisfied with probability approximately $1/2^9$, for (uniformly) random inputs. Empirically, the statistical deviation could be detected with probability close to one using 2^{19} pairs of inputs.

Application to BMW-256. The above strategy works as well for the compression function of BMW-256, with same complexity and for example input difference `00100000` (in same words as for BMW-512).

Conclusion. The compression functions of BMW-256 and BMW-512 do not behave ideally, as they admits strong differential biases. However, these seem difficult to exploit to build a distinguisher (or any other attack) for the hash function, because

1. the IV is fixed, hence an adversary cannot choose differences in the chaining values entering the compression function;
2. even if differences in the IV could be controlled, the additional "blank" invocation to the compression function would prevent an adversary from observing the output differences of the first compression function.

Therefore, our observations do not contradict the security claims of BMW.

Acknowledgments. I'd like to thank for their various help: Itai Dinur, Orr Dunkelman, Danilo Gligoroski, Jian Guo, Simon Knellwolf, Willi Meier, María Naya-Plasencia, and Petr Susil.

References

1. Danilo Gligoroski, Vlastimil Klima, Svein Johan Knapskog, Mohamed El-Hadedy, Jorn Amundsen, and Stig Frode Mjolsnes. Cryptographic hash function BLUE MIDNIGHT WISH. Submission to NIST (Round 2), 2009.
2. Søren S. Thomsen. Pseudo-cryptanalysis of the original Blue Midnight Wish. Cryptology ePrint Archive, Report 2009/478, 2009.