Some properties of an FSE 2005 Hash Proposal

Lars R. Knudsen
Department of Mathematics, Technical University of Denmark
knudsen@mat.dtu.dk

March 15, 2005

Abstract

We consider the hash function proposals by Mridul et al. presented at FSE 2005. For the proposed $2n$-bit compression functions it is proved that collision attacks require $\Omega(2^{2n/3})$ queries of the functions in question. In this note it is shown that with $\mathcal{O}(2^{n/3})$ queries one can distinguish the proposed compression functions from a randomly chosen $2n$-bit function with very good probability. Finally we note that our results do not seem to contradict any statements made the designers of the compression functions.

1 The 1/3 rate proposal from FSE 2005

[1] introduces several new constructions for hash function compression functions of varying hash rates, cf. later. We consider first the compression function of rate 1/3.

Let $f_i : \{0,1\}^{2n} \to \{0,1\}^n$ be independent random functions, for $i = 1, 2, 3$. Define the compression function $F : \{0,1\}^{3n} \to \{0,1\}^{2n}$

$$\begin{aligned} F(x,y,z) &= (F_1(x,y,z) \mid F_2(x,y,z)) \\ &= (f_1(x,y) \oplus f_2(y,z) \mid f_2(y,z) \oplus f_3(z,x)) \end{aligned}$$

This function has a rate of 1/3: it compresses one block of n bits with three evaluations of the f-functions.

First we note that $F_1(x,y,z) \oplus F_2(x,y,z) = (f_1(x,y) \oplus f_3(z,x))$ and thus this sum is independent of f_2. The idea of the distinguishing attack is to find two sets of values x_1, y_1, z_1 and x_2, y_2, z_2 such that

$$f_1(x_1, y_1) \oplus f_3(z_1, x_1) = f_1(x_2, y_2) \oplus f_3(z_2, x_2).$$

1. Fix x to an arbitrary n-bit value, i.e., $x = x_0$.
2. Generate s distinct (random) n-bit values, $y_1, \ldots, y_s$.
   (a) Evaluate $f_1(x_0, y_i)$, $i = 1, \ldots, s$, then store and sort the results.
   (b) Find t_1 pairs of values for which the exor of the $n/3$ least significant bits is α. Save the exor of these pairs in a table T_1.
3. Generate s distinct (random) n-bit values, $z_1, \ldots, z_s$.
   (a) Evaluate $f_3(z_i, x_0)$, $i = 1, \ldots, s$, then store and sort the results.
   (b) Find t_3 pairs of values for which the exor of the $n/3$ least significant bits is α. Save the exor of these pairs in a table T_2.
4. Sort the values in T_1 and T_2 and find u colliding pairs (a, b), where $a \in T_1$ and $b \in T_2$ and $a = b$ (i.e., pairs which are equal in all n bits).

Figure 1: Attack algorithm.

In this case one gets that

$$F(x_1, y_1, z_1) \oplus F(x_2, y_2, z_2) = \beta \mid \beta,$$

for some value of β.

Consider the algorithm of Figure 1 with $s = \sqrt{2} \cdot 2^{n/3}$. This algorithm is a minor modification of an algorithm by Wagner[2]. It follows that the expected value of both t_1 and of t_3 is

$$\binom{s}{2} / 2^{n/3} \simeq 2^{n/3}$$

and the expected value of u is $(2^{n/3})^2/2^{2n/3} = 1$. Thus we expect to find values i_1, i_2, j_1, j_2, such that

$$f_1(x_0, y_{i_1}) \oplus f_1(x_0, y_{i_2}) \oplus f_3(z_{j_1}, x_0) \oplus f_3(z_{j_2}, x_0) = 0.$$

Note that α (in Figure 1) can be of any value. Thus with $2 \cdot 2^{n/3}$ queries one expects to find a pair of inputs to F such that

$$F(x_1, y_1, z_1) \oplus F(x_2, y_2, z_2) = \beta \mid \beta.$$

The time complexity of the method is $\mathcal{O}(2^{n/3})$. If F was a truly random function one would need about $2^{n/2}$ queries to succeed finding such a pair.

Consider the algorithm of Figure 1 with $s = \sqrt{2} \cdot 2^{7n/12}$. Then it follows that $u = 1$ and one has pair of colliding inputs for F. However the total number of queries required to find the pair in this case is around 2^n, thereby larger than the bounds proved for this function[1].

2 The 2/3 rate proposal from FSE 2005

In [1] also two compression functions of rate 2/3 are proposed. The claimed level of security is the same as for the hash rate 1/3 proposal considered above. Our approach applies also to these two functions, here we describe only one of them.

Let $g_i : \{0,1\}^{3n} \rightarrow \{0,1\}^n$ be independent random functions, for $i = 1, 2, 3$. Define the compression function $G : \{0,1\}^{4n} \rightarrow \{0,1\}^{2n}$

$$\begin{aligned} G(x, y, z, w) &= (G_1(x, y, z, w) \mid G_2(x, y, z, w)) \\ &= (g_1(x, y, z) \oplus g_2(y, z, w) \mid g_2(y, z, w) \oplus g_3(x, z, w)) \end{aligned}$$

This function has a rate of 2/3, it compresses two blocks of n bits with three evaluations of the g-functions.

It follows that the above approach applies also to this compression function. Here one can fix the value of x and of z, then vary the values of y to attack g_1 and vary the values of w to attack g_3.

References

[1] M. Nandi, W. Lee, K. Sakurai, and S. Lee. Security Analysis of a 2/3-rate Double Length Compression Function in The Black-Box Model. Preproceedings of FSE 2005, Paris, France, February 2005.

[2] D. Wagner. A Generalized Birthday Problem. Proceedings Crypto'02, LNCS 2442, Springer-Verlag, pp. 288-303, 2002