Understanding collision search

Andrey Dmukh, Grigory Marshalko

Consider a cryptographic hash function $f : X \to Y$, where X and Y – are finite sets. One of the main cryptographic properties of a hash function is a collision strength, i.e. the complexity of finding two different messages $x_1, x_2 \in X$ such that $f(x_1) = f(x_2)$. The complexity of such a search is assessed through a birthday problem, and is formulated in cryptographic literature in different ways: "'...a collision is expected in about $|Y|^{1/2}$ trials..."' [1] or "'in order to attain probability better than 1/2 of finding a collision in a hash function with n-bit output, it suffices to evaluate the function on approximately $1.2|Y|^{1/2}$ randomly chosen inputs"' [3] etc.

On the other hand in classic cryptanalysis of symmetric primitives, as long as in hash functions cryptanalysis based on specific properties of hash functions, the complexity of attacks is usually estimated as the expected number of steps of the cryptanalytic algorithm. In this respect, it would be reasonable to assess the complexity of collision search in terms of expected number of steps of collision finding algorithm, rather than in terms of number of steps required to achieve the desired probability. In order to do this we have to consider the probability $P_f(i)$ of finding collision on the i-th step of the algorithm. It's easy to see that $P_f(i) = C_f(i) - C_f(i-1)$, where $C_f(i)$ is the probability of finding collision in a sample of size i of equally distributed variables. The expected number of steps, that could be obtained by direct summation, is $\sqrt{\pi|Y|/2}$, and is equal to the expected number of steps of Pollard's ρ-method.

In [1] the notion of hash function balance as a measure of non-uniformity is introduced. It is noted that existing hash functions are unbalanced, and the complexity of a collision search for them is less than for balanced (regular) hash functions. Preneel et al. [2] argue that by showing the possibility of constructing for a regular hash function f the subset of X where f was unbalanced. They also note that by increasing the size of X one can make the probability of finding a collision for described types of functions close to each other.

A random mapping is a natural object for modeling cryptographic primitives. We performed experiments to assess the contiguity of the characteristics of the mapping defined by the reduced versions of several hash functions to the characteristics of the random mapping. The results show that corresponding characteristics (such as the number of components, points of specific type, height etc.) are extremely close to each other. As result we conclude that a random mapping is a good object for modeling hash functions and hash functions primitives, and that the theory of random allocations could be used to assess their characteristics. This leads immediately to the following conclusions:

- hash-codes are not equally distributed with high probability, which leads to the fact that, when X is fixed, the probability of constructing preimage for $y \in Y$ is less than 1. If $|X| = |Y|$ the expected number of y with no preimages is $(e^{-1})|X|$.
- when X is fixed the probability of constructing second preimage is also less than 1. If $|X| = |Y|$ this probability is about $1 - \frac{e^{-1}}{1-e^{-1}} \approx 0.418$.

We manage to assess the impact of the balance of a hash function, which could be modeled as a random mapping, on the expected number of steps of collision finding algorithm. We use the combinatorial approach described in [3] and show that in this case the expected number of steps is asymptotically $\sqrt{\pi|Y|/2}$ as in the uniform case. Our results show that

- the expected number of steps is better characteristics for assessing the complexity of collision finding algorithm, than the number of steps required to obtain collision with given probability;

- a random mapping is a good object for hash function modeling;

- a balance of a hash function, which is close to a random mapping, do not affect the average complexity of collision finding algorithm.

References

[1] M. Bellare, T. Kohno. Hash function balance and its impact on birthday attacks. In C. Cachin and J. Camenisch, editors. EUROCRYPT, 04, LNCS 3027, pp. 401-418, Springer.

[2] N. Mouha, G. Sekar, B. Preenel. Challenging the increased Resistance of regular hash function against birthday attacks. http://www.ecrypt.eu.org/hash2011/proceedings/hash2011_12.pdf.

[3] T.S. Nunnikhoven, A birthday problem solution for nonuniform birth sequences, The American Statistician, Vol. 46, No 4, 1992, pp. 270-274.

[4] I. Mironov, Hash functions: Theory, attacks and applications, http://research.microsoft.com/en-us/people/mironov/hash_survey.pdf