Novi Sad J. Math.
Vol. 38, No. 3, 2008, 201-208

IMPACT OF HASH FUNCTION NON-UNIFORMITY ON DIGITAL SIGNATURE SECURITY[1]

Milan Tuba[2], Nadezda Stanarevic[3], Perica Strbac[4], Jasmina Novakovic[5]

Abstract. This paper examines the effect of the hash function irregularity on digital signature security. Digital signature is implemented as a hash function that maps large space of all possible messages to a smaller space of hash values. For the security of such a system it is important that it is difficult to find hash collision. Standard results in this area assume the best case, where hash function is regular. The irregularity of the hash function makes the security worse. We propose some irregularities and compute corresponding probabilities for finding hash collision.

AMS Mathematics Subject Classification (2000): 94A62, 60C05, 68U99

Key words and phrases: digital signature, birthday attack, irregular hash function, hash collision

1. Introduction

For any serious communication or data exchange it is essential to have cryptographic tools to preserve the safety of and integrity information. Changing or stealing data stored in electronic form is so widely spread that not having such protection would make that communication pointless.

The basic idea behind this paper, as a deviation from widely spread methods, is to examine the irregular hash functions that are not dependent on any particular algorithm. This method is universal, because the principles it is based on are applicable to various algorithms, independently of the mechanism they use.

Although the aforementioned hash functions are specific, and have not been considered in detail in previous research, the goal of this paper is to get results that can be applied in general when hash function algorithms are formulated.

2. Digital Signature

To prove the authenticity of legal, financial or other important documents in electronic form, we need to provide a mechanism analog to handwritten signature. Such method first and foremost has to be resistant to forgeries.

[1]This paper is partially supported by the organizers of the 12th Serbian Mathematical Congress

[2]Faculty of Mathematics, University of Belgrade, Serbia e-mail: tuba@matf.bg.ac.yu
[3]Megatrend University, Belgrade, e-mail: srna@stanarevic.com
[4]Megatrend University, Belgrade, e-mail: perica.s@sbb.co.yu
[5]Megatrend University, Belgrade, e-mail: jnovakovic@eunet.rs

A digital signature or digital signature scheme is a type of asymmetric cryptography used to simulate the security properties of a handwritten signature on the paper.

Digital signature schemes consist of at least three algorithms [1]:

1. a key generation algorithm,
2. a signature algorithm, and
3. a verification algorithm.

A digital signature mainly provides authentication of a basic message. In theory, it can also provide non-repudiation, meaning that the authenticity of signed messages can be publicly verified, not only by the intended recipient. Messages may be anything, from electronic mail to a contract, or even a message sent in a more complicated cryptographic protocol.

By encoding the basic message, sender does not ensure its integrity, even if the key has not been compromised. The technique that protects the data integrity is based on a one-way hash function h that maps a random length text into a fixed size array of bits, $\{0,1\}^m \rightarrow \{0,1\}^t$, where $m > t$. The hashing function has three important attributes [5, 4]:

1. Collision-resistance: An attacker should not be able to find a pair of messages $M \neq M'$ such that $h(M) = h(M')$ with less than about $2^{t/2}$ work.
2. Preimage-resistance: An attacker given a possible output value for the hash Y should not be able to find an input X so that $Y = h(X)$ with less than about 2^t work.
3. Second preimage-resistance: An attacker given one message M should not be able to find a second message, M' to satisfy $h(M) = h(M')$ with less than about 2^t work.

A collision attack on a t-bit hash function with less than $2^{t/2}$ work, or a preimage or second preimage attack with less than 2^t work, is formally a break of the hash function. Collision resistance is especially important for digital signature theft prevention. Otherwise, if a collision between two or more messages occurs, certain message's digital signature sent by some sender can be abused and added onto a randomly chosen message without that sender's consent or knowledge.

3. Birthday Attack

In the probability theory, the birthday problem pertains to the probability that in a set of randomly chosen people some pair of them will have the same birthday. With the assumption that $n \leq 365$, the probability of the event A_n, that no two people from a set of n randomly chosen people have common birthday is calculated according to the following formula:

$$
\begin{aligned}
p(A_n) &= 1 \cdot (1 - \frac{1}{365}) \cdot (1 - \frac{2}{365}) \dots \ (1 - \frac{n-1}{365}) \\
&= \frac{365 \cdot 364 \dots \ (365 - n + 1)}{365^n} \\
&= \frac{365!}{365^n \cdot (365 - n)!} \qquad (1)
\end{aligned}
$$

Event B_n, that some pair of them was born on the same day, represents a complementary event to event A_n. Contrary to the naive intuition, the required number n of people that will make the probability of some pair having the common birthday greater than 0.5 is not around 180, it is only 23. For 57 people, the probability of some pair having common birthday is more than 99%. This shows that attacker may not need to examine too many messages before he finds a collision.

In real world circumstances, the basic goal of the attacker is the forgery of digital signatures for messages that the real sender does not want to send. For almost identical messages that differ in only a few bits, for example a space replaced with a tab, there is a major difference in accompanying digital signatures. To succeed, the attacker produces two lists of possible messages M_1 and M_2. The first list consists of messages obtained from M_1 that the sender would be willing to sign, and that are seemingly the same, yet differ in a few bits. The second list consists of messages obtained from M_2 by changing a few bits and are all messages that the attacker wants to send. The essence of this method is to find appropriate pairs $M_1' \in M_1$ and $M_2' \in M_2$ so that:

$$h(M_1') = h(M_2') \qquad (2)$$

With the previously stated facts in mind, we come to the conclusion that attacker's failure is guaranteed only in the case of truly collision-resistant hash function h, while any other case is open to disastrous consequences for the security of a signature scheme.

Described results offer considering the method in which the attacker searches for collisions in randomly chosen hash function. The best known collision attack is the birthday attack. One-way hashing function h maps messages of random length into fixed size bit arrays, $\{0,1\}^m \rightarrow \{0,1\}^t$, where $m > t$, or in short $h : D \rightarrow R$. In the case of birthday attack, the attacker generates random messages $x_1, x_2, ..., x_q \in D$ and computes their hash values $y_i = h(x_i)$, for every $i = 1, \dots, q$. The attack is considered successful if for different values of i, j the following is true $h(x_i) = h(x_j)$, where q represents the number of attempts.

Let $P_h(q)$ be the probability of the birthday attack on hash function $h : D \rightarrow R$ succeeding in q attempts. To have the probability $P_h(q) \geq 0.5$ the number of necessary attempts is $\sqrt{2 \mid R \mid}$, where $\mid R \mid$ is the total number of possible hash values for the hash function in question [8]. To ensure the hash function's collision-resistance we must ensure that it maps messages to hash values consisting of t-bits where

$$2^{\frac{t+1}{2}} = \sqrt{2 \mid R \mid} \tag{3}$$

is sufficiently large that generating $2^{\frac{t+1}{2}}$ random messages and corresponding hash values is infeasible for the attacker.

The following estimate is often used:

If $h : \{0,1\}^m \to \{0,1\}^t$, $3 \leq t < m$, $n = 2^{\lceil \frac{t+1}{2} \rceil}$ and $M_1, \ldots , M_n \in \{0,1\}^m$ are chosen independently at random then $P[collision\ exists] > \frac{1}{2}$

We will do the more accurate calculation. Let us assume that the hash function h is regular. Thus for any fixed hash value $y \in \{0,1\}^t$ and random message M we have $P_r[h(M) = y] = \frac{1}{2^t}$. If we choose n random messages independently from $\{0,1\}^m$ then the probability that they all have distinct hash values is

$$P[no - collision] = \prod_{i=1}^{n-1}(1 - \frac{i}{2^t}) \tag{4}$$

We can now use the inequality

$$\begin{aligned} 1 - x &\leq e^{-x} \quad for \quad 0 \leq x \leq 1 \\ 1 + 2 + ... + n - 1 &= \frac{(n-1) \cdot n}{2} \\ P[no - collision] &\leq e^{-\frac{(n-1) \cdot n}{2^{t+1}}} \end{aligned} \tag{5}$$

To calculate the exact value of this probability we have to examine two cases:

1. when variable t is odd,
2. when variable t is even

The first case presumes that variable t is always an odd number, so the probability of no-collision is calculated according to the following formula:

$$\begin{aligned} P[no - collision] &\leq e^{-\frac{(n-1) \cdot n}{2^{t+1}}} \\ P[no - collision] &\leq e^{-\frac{(2^{t+1})}{2^{t+1}} + \frac{(2^{\frac{t+1}{2}})}{2^{t+1}}} \\ P[no - collision] &\leq e^{-1} \cdot e^{\frac{1}{2^{\frac{t+1}{2}}}} \end{aligned} \tag{6}$$

Since fraction $\frac{1}{2^{\frac{t+1}{2}}} \to 0$ when $t \to \infty$, the probability of no-collision approaches:

$$P[no - collision] \leq e^{-1} = 0.368 \tag{7}$$

The probability of the complement event, event that collision exists, is then:

$$P[collision - exists] > 0.632 \tag{8}$$

The second analysis direction is when variable t is an even number, or $n = 2^{\frac{t}{2}+1}$, so the probability of no-collision is calculated in the following manner:

$$P[no - collision] \quad \leq \quad e^{-\frac{2^{t+2}}{2^{t+1}} + \frac{2^{\frac{t+2}{2}}}{2^{t+1}}} \tag{9}$$

Since fraction $\frac{2^{\frac{t+2}{2}}}{2^{t+1}} \rightarrow 0$ when $t \rightarrow \infty$, the probability of no-collision approaches:

$$P[no - collision] \quad \leq \quad e^{-2} = 0.135 \tag{10}$$

Probability of the complement event, that collision exists, is:

$$P[collision - exists] > 0.865 \tag{11}$$

In both cases, since t is usually greater than 100, the inequalities become very close to the equality. The results confirm that there is a significant correlation between the security of the hash function and the number of generated messages, or, that the hash function's resistance to birthday attacks is ensured if the attacker is not capable of generating $2^{\frac{t}{2}}$ messages.

4. The Hash Function Irregularity

So far, studies of the birthday attack and the conditions necessary for collision to occur presume that the hash function h is regular, meaning, hash function is of uniform distribution. Although hash functions and their application in the field of digital signature have been widely known to the public in the past years, the literature in the field describes relatively small number of examples in which irregular hash functions are used and cover mostly theoretical, rather than actual, use cases.

Stinson [7] says that preimage resistance implies collision resistance under certain circumstances, such as, for example, when the hash function is "close to" uniform. Schneier [6] says that to prevent birthday attacks one should choose the output length t large enough that $2^{\frac{t}{2}}$ trials is infeasible. Buchmann discussion of the attack [3] concludes that the distribution on the corresponding hash values is a uniform distribution.

Aforementioned proofs and assumptions depend on the regularity of the hash functions and its uniform distribution, while no indications are made about irregular hash functions and the number of attempts that would be needed to establish collision in such case. Bellare [2] asks whether under such conditions the number of attempts to establish collision is considerably lower than $\sqrt{2 \mid R \mid}$?

Testing in practice shows that with the rise of hash function's irregularity there is a rise in success of the birthday attack. Intuitively, this statement is not a surprise. In extreme cases we can observe a function that maps all messages M_i into the same value m, [2]. In such case, it is easy to notice that the probability of birthday attack's success, or rather accomplishing collision, is: $P[collisionexists] = 1$. In the case of the hash function having uniform distribution, or when there is no possibility of the collision occurring, than $P[collisionexists] = 0$.

The first step in testing the digital signature's sensitivity to birthday attacks is to construct irregular hash functions by disturbing uniform distribution and in that way gaining irregularity.

For example, let hash function h map $h : \{0,1\}^m \rightarrow \{0,1\}^t$, where $N = 2^t$ is the total number of hash values. Let us assume that a fixed proportion (constant α) of messages are being mapped to a single hash value. Such mapping may be due to some hash algorithm property, for example every millionth message maps to the same hash value. From the set of all messages $\{0,1\}^m$ we observe n randomly chosen messages $M_1, \ldots, M_n$ that are mapped into different hash values $m_1, \ldots, m_n$. The probability that the random message M_i is mapped into the mentioned hash value is α, while the probability of the message M_i not mapping into that particular hash value is 1-α.

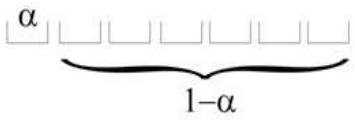

Figure 1: *Irregular hash function's mapping*

We can now differentiate two cases:

1. Case I - when all n messages map into n different hash values, where none of the hash values is the aforementioned hash value with the probability of α. The probability of this case is $P_I = (1-\alpha)^n$

2. Case II - when one of the n messages maps into the aforementioned hash value with the probability of α. The probability of this case can be written down as $P_{II} = n\alpha(1-\alpha)^{n-1}$

The total probability of event A, no-collision occurring, is a sum of probabilities P_I and P_{II}:

$$\begin{aligned} P[no-collision] &= (1-\alpha)^n + n\alpha(1-\alpha)^{n-1} \\ P[no-collision] &= (1-\alpha)^{n-1}((1-\alpha)+n\alpha) \\ P[no-collision] &= (1-\alpha)^{n-1}(1+n\alpha-\alpha) \\ P[no-collision] &= (1-\alpha)^{n-1}(1+\alpha(n-1)) \end{aligned} \tag{12}$$

Applying Bernoulli's inequality we get:

$$\begin{aligned} P[no-collision] &\leq (1-\alpha)^{n-1}(1+\alpha)^{n-1} \\ P[no-collision] &\leq (1-\alpha^2)^{n-1} \end{aligned} \tag{13}$$

When the constant $\alpha \to 1$, the probability of the event A, $(1-\alpha^2)^{n-1} \to 0$, which makes the complement event B, that collision does exist:

$$P[collision-exists] \to 1. \tag{14}$$

More importantly, for any constant α, the probability of $no-collision$ is very close to zero for any significant n, which is always the case.

This mathematical analysis shows that event B's probability, that collision exists, in the case of non-uniform hash function distribution increases as the hash function tends to map into a constant, meaning its irregularity increases.

With this example we can now introduce the idea of "irregularity amount" or hash function's balance. We can define balance as a real number between 0 and 1, where balance 1 indicates that the hash function is regular and balance 0 indicates that it is a constant function, meaning as irregular as can be. With analytical and experimental determination of the given hash function's balance we can establish how fast the attacker can succeed with the birthday attack. Examining the balance represents just one of the criteria we need to take into consideration when creating a hash function, but is not the only prerequisite to have the hash function be resistant to birthday attacks.

5. Conclusion

This paper provides quantitative information about the success-rate of the birthday attack on the irregular hash functions. The hash function's irregularity is accomplished by disturbing the uniform distribution of the observed functions. For our research we design irregular hash functions with different characteristics and show how "amount of irregularity" in the hash function h characterizes the behavior of the birthday attack on h, by showing the probability of finding a collision. The results of these examples determine the way we can model how collision resistance decreases as hash function's irregularity increases. Further research is directed toward establishing a general model of irregularity and quantitative relation between such irregularity and collision resistance.

Acknowledgment

This research was supported by the Ministry for Science and Technical Development of the Republic of Serbia, Project 144007.

References

[1] Abdalla, M., Reyzin, L., A New Forward-Secure Digital Signature Scheme. Advances in Cryptology - Asiacrypt 2000. Numer. Comp. 1976/-1, Springer, Berlin (2000), 116-129.

[2] Bellare, M., Kohno, T., Hash function balance and its impact on the birthday attack. Advances in Cryptology- Eurocrypt 2004. Numer. Comp. 3027, Springer Berlin, (2004), 401-418.

[3] Buchmann J., Introduction to cryptography. Springer, 335 pages, 2000.

[4] Halevi, S., Krawczyk, H., Strengthening Digital Signatures via Randomized Hashing. Advances in Cryptology - Crypto 2006. Numer. Comp. 4117, Springer Berlin, (2006), 41-59,

[5] Kelsey, J., Schneier, B., Second Preimages on n-Bit Hash Functions for Much Less than 2^n Work. Advances in Cryptology - Eurocrypt 2005. Numer. Comp. 3494, Springer Berlin, (2005), 474-490.

[6] Schneier B., Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code. in: C. John Wiley & Sons, Inc, 784 pages, 1996.

[7] Stinson, D., Some observations on the theory of cryptographic hash functions, Designs, Codes and Cryptography. Numer. Math. 38, Springer Netherlands, (2006), 259 - 277.

[8] Talnor, J., Welsh, D., Complexity and Cryptography an Introduction, Cambridge University Press, 292 pages, 2006.