Relation between Successfulness of Birthday Attack on Digital Signature and Hash Function Irregularity

MILAN TUBA, NADEZDA STANAREVIC
Faculty of Computer Science
University Megatrend Belgrade
Bulevar umetnosti 29
SERBIA
tubamilan@ptt.rs, srna@stanarevic.com

Abstract: - In many network communications it is crucial to be able to authenticate both the contents and the origin of a message. Digital signatures based on public key schemas are used for such authentication. In order to provide message authentication the signature must depend on the contents of the message being signed. Since the public key-based signature schemes take too much time to compute, hash functions that map messages to short digests *h(M)* are used. Among other desirable properties of hash functions, an interesting one is that it should be collision-resistant, that is it should be difficult to find two messages with the same hash value. To find a collision the birthday attack is used, which shows that attacker may not need to examine too many messages before he finds a collision. Even worse, in estimates of attack successfulness it is always assumed that the hash function is regular, meaning that all points in the range have the same number of pre-images under *h*. If *h* is not regular, fewer trials are required to find a collision. In this paper we first compute tighter upper and lower bounds for the number of birthday attack trials when the hash function is regular. Then we examine different types of irregularity of the hash function and the quantitative changes in the required number of trials to find a collision which then compromises the digital signature system.

Key-Words: - Digital signature, Birthday attack, Irregular hash function, Hash collision

1 Introduction

Computer networks are used today for many applications (like banking, e-government etc.) where security is an absolute necessary. Changing or stealing data stored in electronic form is so widely spread that not having cryptographic tools to preserve the safety of information would make pointless any serious communication or data exchange. In general, the functions of security system are confidentiality, authentication, integrity and non-repudiation [1]. The last three functions are usually facilitated by digital signatures. In order to provide message authentication the signature must depend on the contents of the message being signed. The problem with the public key based signature schemes which are used for authentication is that for long messages the signature would take a long time to compute. To overcome that problem hash functions that map a message to a small digest *h(M)* are used. Such hash function has to be collision-resistant - it should be difficult to find two messages with the same hash value. Otherwise, the signature system is compromised.

Collision-resistance of a hash function is almost always estimated assuming that the hash function is regular, meaning that all points in the hash range have the same number of pre-images. This is not necessarily the case for all the hash functions that are used. It is interesting to estimate how the security becomes compromised when the hash function deviates from regularity. 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.

1.1 Digital Signature

To prove the authenticity of legal, financial or other important documents in electronic form, a mechanism analog to handwritten signature is needed [2]. Such method first and foremost has to be resistant to forgeries.

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 [3]:

- a key generation algorithm,
- a signature algorithm, and
- 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 key for 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. In order to provide message authentication the signature must depend on the contents of the message being signed.

The other problem with the public key-based signature schemes is that if the message is long then the signature will take a long time to compute. To overcome both of these problems hash functions that map a (possibly lengthy) message to a small digest *h(M)* are used.

1.2 Hash Function

A hash function *h* should map strings of bits of variable length to fix-length strings of bits, called the hash value of the message $\{0,1\}^m \rightarrow \{0,1\}^t$, where $m > t$ [4], [5]. Ideally it has the following properties [6], [7], [8]:

- The length of *h(M)* should be small so that messages can be signed efficiently.
- The function *h* should be a publicly known one-way function – it should be hard to find a message that hashes to a pre-specified value.
- It should destroy algebraic relationships between messages and signatures.
- It should be collision-resistant, that is it should be difficult to find two messages with the same hash value, or more precisely: an attacker should not be able to find a pair of messages M≠ M' such that *h(M)=h(M')* with less than about $2^{t/2}$ work.
- 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.
- 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.

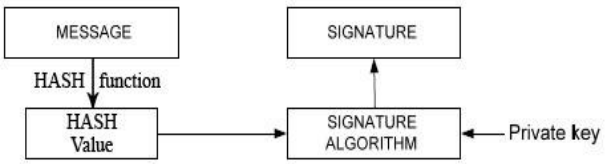


Fig. 1 Hash function

1.3 Collision Birthday Attack

In the probability theory, the birthday problem pertains to the probability that in a set of *n r*andomly chosen people some pair of them will have the same birthday. 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%.

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 is 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:

$$h(M'_1) = h(M'_2)$$

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 $i = 1,....q$. 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.

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.

The time required to find a collision is one of the most important measures in evaluating a hash algorithm. This time is also called the search cost of the algorithm [9].

1.4 The Balance Measure of Hash Functions

The most important properties of hash function, concerning a digital signature, is collision resistance - an attacker should not be able to find a pair of messages $M \neq M'$ such that $h(M)=h(M')$. The question is can we compute "irregularity amount" of hash function. The idea of "irregularity amount" or hash function's balance was introduced in [10]. Balance can be defined 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.

Let $h:D \rightarrow R$ be a function whose domain D and range $R=\{R_1,....,R_r\}$ have sizes d, $r \geq 2$, respectively. For $i \in [r]$ let $d_i = |h^{-1}(R_i)|$ denote the size of the pre-image of R_i under h. The balance of h, denoted $\mu(h)$, is defined as

$$\mu(h) = \log_r \left[\frac{d^2}{d_1^2 + \ldots + d_r^2} \right]$$

where $\log_r()$ denotes the logarithm in base r.

Note that

$$\frac{1}{r^{\mu(h)}} = \frac{d_1^2 + \ldots + d_r^2}{d^2}$$

is the probability that $h(a) = h(b)$ if a,b are drawn independently at random from the domain D.

It is easy to see that a regular function has balance one and a constant function has balance zero.

2 Bounds for Collision Probability

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 $\sqrt{2|R|}$, where $|R|$ is the total number of possible hash values for the hash function in question [3]. To ensure the hash function's collision-resistance we must ensure that it maps messages to hash values consisting of t-bits where

$$2^{(t+1)/2} = \sqrt{2|R|}$$

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

It is a known result that:

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

It is also known that

If $h:\{0,1\}^m \rightarrow \{0,1\}^t$ is regular, $3 \leq t < m$, $n = 2^{\frac{t-k}{2}}$, and $M_1,.....M_n \in \{0,1\}^m$ are chosen independently at random then

$$P[\text{ collision exists}] < \frac{1}{2^{k+1}}$$

Proof: Since h is regular we know that for each $y \in \{0,1\}^t$ we have $|h^{-1}(y)| = 2^{m-t}$. Let B_i be the event that the i-th message has a hash value that is the same as one of the earlier messages. Then

$$P[B_i] \leq \frac{i-1}{2^t}$$

$$P[\text{collision exists}] = P[B_1 \cup B_2 \cup \ldots\ldots \cup B_n]$$

$$P[\text{collision exists}] \leq \sum_{i=2}^{n} P[B_i]$$

$$P[\text{collision exists}] \leq \sum_{i=2}^{n} \frac{i-1}{2^t}$$

$$P[\text{collision exists}] \leq \frac{n(n-1)}{2^{t+1}}$$

$$P[\text{collision exists}] < \frac{n^2}{2^{t+1}} \leq \frac{2^{t-k}}{2^{t+1}} = \frac{1}{2^{k+1}}$$

We will calculate more precisely the lower and the upper bounds for the probability of complement event of no collision.

2.1 The Upper Bound

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

$Pr[h(M) = y] = \frac{1}{2^t}$.

For easier representation we introduce the following substitution $2^t = N$.

If we choose *n* random messages independently from $\{0, 1\}^m$ then the probability that they all have distinct hash values is

$$P[\text{no collision}] = \frac{N(N-1)(N-2)....(N-n+1)}{N^n} =$$

$$= (1-\frac{1}{N})...(1-\frac{(n-1)}{N}) = \prod_{i=1}^{n-1}(1-\frac{i}{2^t})$$

We can use approximations

$$e^{-X} \approx 1-x+\frac{x^2}{2!}-\frac{x^3}{3!}+\sigma(x^3)$$

$1 - x \le e^{-x}$ for $0 \le x \le 1$

to calculate

$$P[\text{no collision}] = \prod_{i=1}^{n-1}(1-\frac{i}{N}) \le \prod_{i=1}^{n-1} e^{-\frac{i}{N}}$$

$$P[\text{no collision}] \le e^{-\frac{1}{N}\sum_{i=1}^{n-1} i}$$

or

$$P[\text{no collision}] \le e^{-\frac{1}{N}\frac{(n-1)n}{2}}$$

$$P[\text{no collision}] \le e^{-\frac{(n-1)n}{2^{t+1}}}$$

where $(n-1)n = 2^{\frac{t^2}{4}-\frac{1}{4}}$ for *t* odd and $(n-1)n = 2^{\frac{t^2}{4}+\frac{t}{2}}$ for *t* even.

This result we use to calculate more precisely the estimate that was mentioned before that

$$P[\text{collision exists}] > \frac{1}{2}$$

To calculate the exact value of this probability we have to examine two cases depending on whether variable *t* is odd or 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:

$$P[\text{no collision}] \le e^{-\frac{(n-1)n}{2^{t+1}}} \quad (\text{and } n = 2^{\left\lceil \frac{t+1}{2} \right\rceil})$$

$$P[\text{no collision}] \le e^{-\frac{2^{t+1}}{2^{t+1}}+\frac{2^{\frac{t+1}{2}}}{2^{t+1}}}$$

$$P[\text{no collision}] \le e^{-1} * e^{\frac{1}{2^{\frac{t+1}{2}}}}$$

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

$$P[\text{no collision}] \le e^{-1} = 0.368$$

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

$$P[\text{collision exists}] > 0.632$$

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

$$P[\text{no collision}] \le e^{-\frac{2^{t+2}}{2^{t+1}}+\frac{2^{\frac{t+2}{2}}}{2^{t+1}}}$$

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

$$P[\text{no collision}] \le e^{-2} = 0.135$$

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

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

2.2 The Lower Bound

Now, for the lower bound for the probality of collision we use the approximation

$$e^{-X} \approx 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \sigma(x^3) \quad \text{za } 0 < x < 1$$

or

$$e^{-X} - \frac{x^2}{2} \le 1 - x$$

The probability of of the event that there is no collision can be estimated

$$P[\text{no collision}] = \prod_{i=1}^{n-1}(1 - \frac{i}{N})$$

$$P[\text{no collision}] \ge \prod_{i=1}^{n-1}(e^{-\frac{i}{N}} - \frac{i^2}{2N^2}) =$$

$$\left(e^{-\frac{1}{N}} - \frac{1}{2N^2}\right)\left(e^{-\frac{2}{N}} - \frac{4}{2N^2}\right)\ldots\left(e^{-\frac{n-1}{N}} - \frac{(n-1)^2}{2N^2}\right)$$

$P[\text{no collision}] \ge$

$$\prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \sum_{i=1}^{n-1}\left(\frac{i^2}{2N^2} * \prod_{\substack{j=1 \\ j\ne i}}^{n-1} e^{-\frac{j}{N}}\right) +$$

$$+ \sum_{\substack{i,j=1 \\ i<j}}^{n-1}\left(\frac{i^2}{2N^2} * \frac{j^2}{2N^2} * \prod_{\substack{k=1 \\ k\ne i,j}}^{n-1} e^{-\frac{k}{N}}\right) -$$

$$- \sum_{\substack{i,j,k=1 \\ i<j<k}}^{n-1}\left(\frac{i^2}{2N^2} * \frac{j^2}{2N^2} * \frac{k^2}{2N^2} * \prod_{\substack{l=1 \\ l\ne i,j,k}}^{n-1} e^{-\frac{l}{N}}\right)$$

$$+ \ldots\ldots +$$

$$\sum_{i_1<i_2<\ldots<i_x}\left(\frac{i_1^2}{2N^2} * \frac{i_2^2}{2N^2} * \ldots * \frac{i_x^2}{2N^2} * \prod_{k\ne i_1,i_2,\ldots i_x} e^{-\frac{k}{N}}\right) -$$

$$- \sum_{i_1<i_2<\ldots<j}\left(\frac{i_1^2}{2N^2} * \ldots * \frac{i_x^2}{2N^2} * \frac{j^2}{2N^2} * \prod_{k\ne i_1,i_2,\ldots,i_{x,j}} e^{-\frac{k}{N}}\right)$$

After some detailed calculations we can proove that the difference of the third and the fourth addend is positive

$$\sum_{\substack{i,j=1 \\ i<j}}^{n-1}\left(\frac{i^2 j^2}{4N^4} * \prod_{\substack{l=1 \\ l\ne i,j}}^{n-1} e^{-\frac{l}{N}}\right) - \sum_{\substack{i,j,k=1 \\ i<j<k}}^{n-1}\left(\frac{i^2 j^2 k^2}{4N^4 * 2N^2} * \prod_{\substack{l=1 \\ l\ne i,j,k}}^{n-1} e^{-\frac{l}{N}}\right) > 0$$

so when we remove both the inequality will continue to hold.

Similarly, the differences of the addends of the form of the last two addends in the previous inequality can also be shown to be positive and they can also be neglected and the inequality will hold:

$P[\text{no collision}] \ge$

$$\prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \sum_{i=1}^{n-1}\left(\frac{i^2}{2N^2} * \prod_{\substack{j=1 \\ j\ne i}}^{n-1} e^{-\frac{j}{N}}\right) \ge$$

$$e^{-\frac{1}{N}} e^{-\frac{2}{N}} \ldots e^{-\frac{n-1}{N}}\left[1 - \frac{1}{2N^2}\frac{1}{e^{-\frac{1}{N}}} - \frac{4}{2N^2}\frac{1}{e^{-\frac{2}{N}}} - \ldots - \frac{(n-1)^2}{2N^2}\frac{1}{e^{-\frac{n-1}{N}}}\right]$$

$$\ge \prod_{i=1}^{n-1} e^{-\frac{i}{N}}\left[1 - \sum_{i=1}^{n-1}\frac{i^2}{2N^2} * \frac{1}{e^{-\frac{i}{N}}}\right]$$

$$\ge \prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \frac{1}{2N^2} * \prod_{j=1}^{n-1} e^{-\frac{j}{N}} * \sum_{i=1}^{n-1}\frac{i^2}{e^{-\frac{i}{N}}}$$

Since $i \le n-1$ we have $-i^2 \ge -(n-1)^2$ and

$$\frac{i}{N} \le \frac{n-1}{N}, \qquad -e^{\frac{i}{N}} \ge -e^{\frac{n-1}{N}}$$

By applying this we get:

$P[\text{no collision}] \ge$

$$\prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \frac{1}{2N^2} * \prod_{j=1}^{n-1} e^{-\frac{j}{N}} * \sum_{i=1}^{n-1}\frac{(n-1)^2}{e^{-\frac{i}{N}}}$$

$$\ge \prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \frac{(n-1)^2}{2N^2} * e^{-\frac{(n-1)n}{2N}} * e^{\frac{n-1}{N}} * \sum_{i=1}^{n-1} 1$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{N}} - \frac{(n-1)^3}{2N^2} * e^{\frac{n-1}{N}(1-\frac{n}{2})}$$

Since $n = 2^{\frac{t+1}{2}}$ and $N = 2^t$ and also $2^{\frac{t+1}{2}} - 1 \leq 2^{\frac{t+1}{2}}$ and $-(2^{\frac{t+1}{2}} - 1) \geq -2^{\frac{t+1}{2}}$ the substitution gives:

P[no collision] ≥

$$\prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} - \frac{(2^{\frac{t+1}{2}} - 1)^3}{2^{2t+1}} * e^{\frac{2^{\frac{t+1}{2}} - 1}{2^t}(1-\frac{2^{\frac{t+1}{2}}}{2})}$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} - \frac{(2^{\frac{t+1}{2}})^3}{2^{2t+1}} e^{\frac{2^{\frac{t+1}{2}} - 1}{2^t}(1-\frac{2^{\frac{t+1}{2}}}{2})}$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} - 2^{\frac{1-t}{2}} * e^{2^{\frac{1-t}{2}} - 2^{\frac{1-t}{2}} * 2^{\frac{t-1}{2}}}$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} + \frac{-e^{-1+\frac{1}{2^{\frac{t-1}{2}}}}}{2^{\frac{t-1}{2}}}$$

It can be shown that

$$-e^{-1+\frac{1}{2^{\frac{t-1}{2}}}} > -1 + \frac{1}{2^{\frac{t-1}{2}}} - \frac{1}{2^{t-1}}$$

and substittution gives

P[no collision] ≥

$$\prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} + \frac{1}{2^{\frac{t-1}{2}}} * \left(-1 + \frac{1}{2^{\frac{t-1}{2}}} - \frac{1}{2^{t-1}} \right)$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} - \frac{1}{2^{\frac{t-1}{2}}} + \left(\frac{1}{2^{\frac{t-1}{2}}} \right)^2 - \left(\frac{1}{2^{\frac{t-1}{2}}} \right)^3$$

$$\geq \prod_{i=1}^{n-1} e^{-\frac{i}{2^t}} - \frac{1}{2^{\frac{t-1}{2}}} > e^{-\frac{(n-1)n}{2N}} - \frac{1}{2^{\frac{t-1}{2}}}$$

$$\geq e^{\frac{1}{2^{\frac{t+1}{2}}} - 1} - \frac{1}{2^{\frac{t-1}{2}}}$$

The last inequality is the lower bound for the probability that there is no collision. Inclusion of the previously calculated upper bound gives:

$$e^{\frac{1}{2^{\frac{t+1}{2}}} - 1} - \frac{1}{2^{\frac{t-1}{2}}} < P[\text{no collision}] < e^{\frac{1}{2^{\frac{t+1}{2}}} - 1}$$

or, for the complement event:

$$1 - e^{\frac{1}{2^{\frac{t+1}{2}}} - 1} < P[\text{collision exists}] < 1 - e^{\frac{1}{2^{\frac{t+1}{2}}} - 1} + \frac{1}{2^{\frac{t-1}{2}}}$$

3 Examples of 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 digital signature field have been widely known to the public in the past years, 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 [11] says that preimage resistance implies collision resistance under certain circumstances, such as, for example, when the hash function is "close to" uniform. Schneier [12] says that to prevent birthday attacks one should choose the output length *t* large enough that $2^{t/2}$ trials are infeasible. Buchmann's discussion of the attack says [13] that the distribution on the corresponding hash values is the uniform distribution.

Aforementioned proofs and assumptions depend on the regularity of the hash functions and its

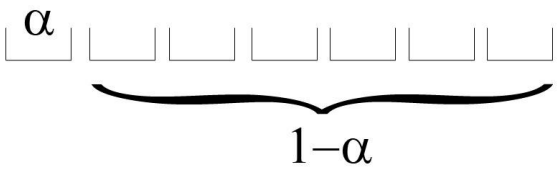
uniform distribution, while no indications are made about irregular hash functions and the number of attempts that would be needed to establish collision in such case. Bellare [10] asks whether under such conditions the number of attempts to establish collision is considerably lower than $\sqrt{2|R|}$.

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.

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.

Example 1. Irregular hash function with probability of mapping to one particular hash value equal to some constant.

Let hash function h maps 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 m^*. 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 $M_1,.....,M_n$ that are mapped into different hash values $m_1,....,m_n$. The probability that the random message M_i is mapped into the mentioned hash value m^* is α, while the probability of the message M_i not mapping into that particular hash value is $1-\alpha$.



We can now differentiate two cases:

- Case I – when all n messages map into n different hash values, where none of the hash values is the aforementioned hash value m^* with the probability of α. The probability of this case is $P_I=(1-\alpha)^n$
- Case II – when exactly one of the n messages maps into the aforementioned hash value m^* 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}: Here we use assumption that if all n messages miss the particular hash value m^*, there is no collision. Collision is possible in that case but since all other hash values have probability of being mapped to that approaches zero as t grows, probability of such collision is orders of magnitude smaller then hitting particular hash value m^*.

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

Applying Bernoulli's inequality we get:

$$P[\text{no-collision}] \le (1-\alpha)^{n-1}\,(1+\alpha)^{n-1}$$
$$P[\text{no-collision}] \le (1-\alpha^2)^{n-1}$$

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

$$P[\text{collision exists}] \rightarrow 1$$

More important, for any constant α no matter how small, 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.

Example 2. Irregular hash function where probabilities of mapping to different hash values are all different.

Let hash function h maps h: $\{0,1\}^m \rightarrow \{0,1\}^t$, where $N = 2^t$ is the total number of hash values. Let us assume that the probability of any message M being mapped to some hash value is different from such probability for some other hash value. The probability that the random message M is mapped into random hash value m_i is p_i, where $p_1+p_2+....+p_N = 1$ and $pi \ge 0$ for $1 \le i \le N$. From the set of all messages $\{0,1\}^m$ we observe n randomly chosen messages $M_1,.....,M_n$ that are mapped into different hash values $m_1,....,m_n$. We will show that probability of event ***A***, no collision occurring, is less in the case of non-uniform hash function distribution:

$$\sum_{i_1,\ldots i_n} p_{i_1}\,p_{i_2}\,\ldots.p_{i_n} \;\le\; \binom{N}{n}\,\frac{n!}{N^n}$$

where the sum is over all choices of distinct $i_1,i_2,..i_n$, satisfying $1 \le i_j \le N$, for $1 \le j \le n$.

The probability of event ***A***, no collision occurring, in the case of uniform hash function distribution has a form

$$P^U[\text{no-collision}] = \frac{N(N-1)(N-2)....(N-n+1)}{N^n}$$

$$= \frac{N(N-1)(N-2)....(N-n+1)}{N^n} \frac{(N-n)!}{(N-n)!}$$

$$= \frac{N!}{N^n (N-n)!} \frac{n!}{n!}$$

$$P^U[\text{no-collision}] = \binom{N}{n} \frac{n!}{N^n}$$

Let $L = \sum_{i_1,..i_n} p_{i_1} p_{i_2}.....p_{i_n}$ represents the sum of all probabilities. We consider L as a function of variables $p_1,.....,p_n$, where, $L = L(p_1,.....,p_n)$ and $i_j \in \{1.2……N\}$. Function $L(p_1...p_n)$ is a continuous function, polynomial, consists of N variables. This function is defined on the compact set

$D: p_1 \geq 0, p_2 \geq 0 ... p_N \geq 0, p_1+p_2+....+p_N = 1.$

For arbitrary $i, j \in \{1,2,.....,N\}$, where $i \neq j$ the following addends can be distinguished:

1. The addends in which appear both p_i, p_j.
2. The addends in which appear p_i or p_j
3. The remaining addends that do not contain p_i or p_j

Since $p_i * p_j \leq (\frac{p_i + p_j}{2})^2$ the following approximation can be done:

$$\sum_{i_1,..i_n} p_1 p_2 ... p_i p_{i+1}.....p_j p_{j+1}.....p_n \leq$$

$$\sum_{i_1,..i_n} p_1 p_2 ... \frac{p_i + p_j}{2} p_{i+1}..... \frac{p_i + p_j}{2} p_{j+1}.....p_n$$

In case when an addend contains one of the numbers p_i or p_j, the addends can be grouped in pairs so that the sum of such addends will not change if instead of p_i or p_j $\frac{p_i + p_j}{2}$ is substituted.

The function L reaches a maximum when all probabilities $p_1, p_2,....,p_N$ are mutually equal, $p_1=p_2=....= p_N= \frac{1}{N}$. As the number of addends is equal $\binom{N}{n} n!$, and the value of each addend equals $\frac{1}{N^n}$, follows that

$$\sum_{i_1,...i_n} p_{i_1} p_{i_2} .....p_{i_n} \leq \binom{N}{n} \frac{n!}{N^n}$$

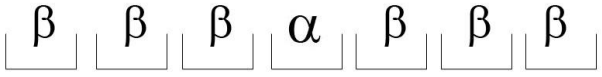
The equality is achieved in the case of uniform hash function distribution $p_1 = p_2 = ... = p_N = \frac{1}{N}$. From the above follows that probability of event ***A***, no collision occurring, is less in the case of non-uniform hash function distribution

$$P[\text{no-collision}] \leq P^U[\text{no-collision}]$$

The results of this mathematical analysis confirms that the probability in the case of non-uniform hash function distribution decreases as hash function's irregularity increases.

Example 3. Irregular hash function where probability of mapping to one particular hash value is greater than such probability for any other hash value.

Let hash function h maps h: $\{0,1\}^m \rightarrow \{0,1\}^t$, where $N = 2^t$ is the total number of hash values. Let us assume that the probability of a single random message M_i being mapped to one particular hash value is α which is greater than the probability β of being mapped to any other hash value. From the set of all messages $\{0,1\}^m$ we observe n randomly chosen messages $M_1,.....,M_n$ that are mapped into different hash values $m_1,....,m_n$.



We can now differentiate two cases:

- 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
  $P_I=(N-1)(N-2)(N-3)*.....*(N-n)\ \beta^n$
- 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
  $P_{II}=n\alpha\ (N-1)(N-2)(N-3)*.....*(N-n+1)\ \beta^{n-1}$

In the case when regular hash function with uniform distribution maps h:$\{0,1\}^m \rightarrow \{0,1\}^t$, the probabilities of all hash values are the same and equal $\frac{1}{N}$, where $N = 2^t$. In this example the hash function's uniformity is violated by increasing the probability of one hash value. We can assume that

$$\beta = \frac{1}{N} - x,$$

$$\alpha = \frac{1}{N} + (N-1)x$$

$$x = \frac{1}{N} - \beta$$

$$x < \frac{1}{N}$$

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

$$P[no\text{-}collision] = (N-1)(N-2)*.....*(N-n)\,\beta^n + n\alpha\,(N-1)(N-2)(N-3)*.....*(N-n+1)\,\beta^{n-1}$$

$$P[no\text{-}collision] = (N-1)(N-2)*.....*(N-n+1)\beta^{n-1}\,[(N-n)\beta + n\alpha]$$

$$P[no\text{-}collision] = (N-1)(N-2)*.....*(N-n+1)\beta^{n-1}\,[N\beta + n(\alpha-\beta)]$$

$$P[no\text{-}collision] = (N-1)(N-2)*.....*(N-n+1)\frac{(1-Nx)^{n-1}}{N^{n-1}}\,[N\beta + n(\alpha-\beta)]$$

$$P[no\text{-}collision] = \frac{(N-1)(N-2)*.....*(N-n+1)}{N^{n-1}}\,(1-Nx)^{n-1}[N\beta + n(\alpha-\beta)]$$

The probability of event ***A***, no collision occurring, in the case of uniform hash function distribution has a form:

$$P^U[no\text{-}collision] = \frac{(N-1)(N-2)....(N-n+1)}{N^{n-1}}$$

$$P[no\text{-}collision] = P^U[no\text{-}collision] * (1-Nx)^{n-1}[N\beta + n(\alpha-\beta)]$$

$$P[no\text{-}collision] = P^U[no\text{-}collision] * (1-Nx)^{n-1}[1-Nx+n(\frac{1}{N} + (N-1)x - \frac{1}{N} + x)]$$

$$P[no\text{-}collision] = P^U[no\text{-}collision] * (1-Nx)^{n-1}[1-Nx+nNx]$$

$$P[no\text{-}collision] = P^U[no\text{-}collision] * (1-Nx)^{n-1}[1+ (n-1)\,Nx]$$

By applying Bernoulli's inequality we get:

$$(1-Nx)^{n-1}[1+ (n-1)\,Nx] \le (1-Nx)^{n-1}(1+ Nx)^{n-1}$$
$$(1-Nx)^{n-1}[1+ (n-1)\,Nx] \le (1-N^2x^2)^{n-1}$$

Since $(1-N^2x^2)^{n-1} < 1$ it follows that

$$P[no\text{-}collision] < P^U[no\text{-}collision]$$

which makes the complement event ***B***, that collision does exist

$$P[collision\ exists] > P^U[collision\ exists]$$

This mathematical analysis shows that event ***B***'s probability, that collision exists, is greater in the case of non-uniform hash function distribution

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.

The ratio of probabilities of no collision for this third example and the uniform case is:

$$\frac{P[_{no-collision}]}{P^U_{[no-collision]}} = \left(1-Nx\right)^{n-1}\left(1-Nx+nNx\right)$$

Since variable x depends on N and n the substitution $nNx = a$ can be introduced:

$$\frac{P[_{no-collision}]}{P^U_{[no-collision]}} = \left(1-\frac{a}{n}\right)^{n-1}\left(1-\frac{a}{n}+a\right)$$

$$\frac{P[_{no-collision}]}{P^U_{[no-collision]}} = \left(1+\frac{1}{-\frac{n}{a}}\right)^{(-\frac{n}{a})\frac{n-1}{n}(-a)}\left(1-\frac{a}{n}+a\right)$$

When n $\to \infty$ ratio $\frac{n-1}{n} \to 1$, while ratio $\frac{a}{n} \to 0$. Since the natural logarithm can be defined as $e = \lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n$ the ratio of probabilities becomes:

$$\frac{P[_{no-collision}]}{P^U_{[no-collision]}} = \frac{1+a}{e^a}$$

The expression $\frac{P[_{no-collision}]}{P^U_{[no-collision]}}$ represents a real number from interval [0,1]. Maximum 1 is reached when hash function is regular and minimum 0 is

reached when hash function is constant meaning that all preimages map to a single hash value.

This is similar to balance function introduced by Bellare and Kohno [10] and can be used to quantitative estimate how sooner the birthday attack can succeed.

4 Conclusion

In this paper we provide 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 how we can model the collision resistance decrease as hash function's irregularity increases. This can be of interest for hash functions that are used in practice (like MD5, SHA-1 etc.) since they use Merkle-Damgard transform applied to the underlying compression function which does not preserve regularity. Further research is directed toward establishing a general model of irregularity and quantitative relation between such irregularity and collision resistance focused on the interesting area of hash functions that are close to regular.

References:

[1] Tsang-Yean, L. and Huey-Ming, L. Encryption and decryption algorithm of data transmission in network security, *WSEAS Transactions on Information Science and Applications.* Vol. 3, no. 12, 2006, pp. 2557-2562.

[2] Kovari, B., Albert, I., and Charaf, H. A general approach to off-line signature verification. *WSEAS Transactions on Computers* Vol.7, No. 10, 2008, pp. 1648-1657

[3] Abdalla, M., Reyzin, L., A New Forward-Secure Digital Signature Scheme, *Advances in Cryptology - Asiacrypt 2000*, Vol. 1976/2000, Springer Berlin, 2000. pp. 116-129

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

[5] Zivic, N. and Ruland, C. Probability of collisions in soft input decryption. *Proceedings of the American Conference on Applied Mathematics.* World Scientific and Engineering Academy and Society (WSEAS), 2008, pp. 362-366.

[6] S.M.M. Rahman, S.M. Masum, M.S.I. Khan, M.S. Alam, and M.I. Hasan, A New Message Digest Function for Message Authentication, *WSEAS Transactions on Computers*, Vol. 3, Issue 5, November 2004, pp. 1466 – 1469,.

[7] Halevi, S., Krawczyk, H., Strengthening Digital Signatures via Randomized Hashing, *Advances in Cryptology - Crypto 2006*, Vol. 4117/2006, Springer Berlin, 2006. pp. 41-59

[8] Kelsey, J., Schneier, B., Second Preimages on n-Bit Hash Functions for Much Less than 2^n Work, *Advances in Cryptology - Eurocrypt 2005*, Vol. 3494, Springer Berlin, 2005. pp. 474-490,

[9] Sun, N. and Nakamura, R. An alternative analysis of the open hashing algorithm. *In Proceedings of the 5th WSEAS international Conference on Applied Mathematics* World Scientific and Engineering Academy and Society (WSEAS), No. 12, 2004, pp. 1-6.

[10] Bellare, M., Kohno, T., Hash function balance and its impact on the birthday attack, *Advances in Cryptology-Eurocrypt 2004*, No. 3027/2004, Springer Berlin, 2004. pp. 401-418

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

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

[13] S Buchmann J., *Introduction to cryptography*, Springer, 335 pages, 2000.

Acknowledgment: This research is supported by Project 144007, Ministry of Science, Republic of Serbia.