Applications of Surjection on Narrow-pipe Hash Functions

Xigen Yao
Wuxi Hengqi Electromechanical Device Co.Ltd.China
email : dihuo377@163.com

January 29, 2011

Abstract. Recently several reports of Cryptology ePrint Archive give analysis and conclusions,that for a normal narrow-pipe hash function the entropy and codomain will reduce greatly. And even generic collision attacks on narrow-pipe hash functions are faster than birthday paradox.However,the conclusions don't apply to all the narrow-pipe hash modes ,such as LAB Mode[5].In this paper,we use LAB Mode and give an effective way that we can design a composition of compression function F, such that: $F = g^* \circ F^*$.Where g^* is a simple surjective function, F^* is a ideal narrow-pipe compression function in LAB Mode , then,we can thwart the conclusions. .
keywords:narrow-pipe hash,LAB Mode ,surjection,entropy

1 Introduction:

The Merkle-Damgaard construction[1] is the most widely used to transform a secure compression function $C : \{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^n$ into a cryptographic hash function $h_c(\cdot)$. (where,n denotes the size of the chaining value,and m denotes the block size for the compression function .)

The iterating hash functions of M-D construction try to maintain the following three properties of the cryptographic secure compression functions :

1.**pre-image resistance:**

2.**second pre-image resistance:**

3.**collision resistance:**

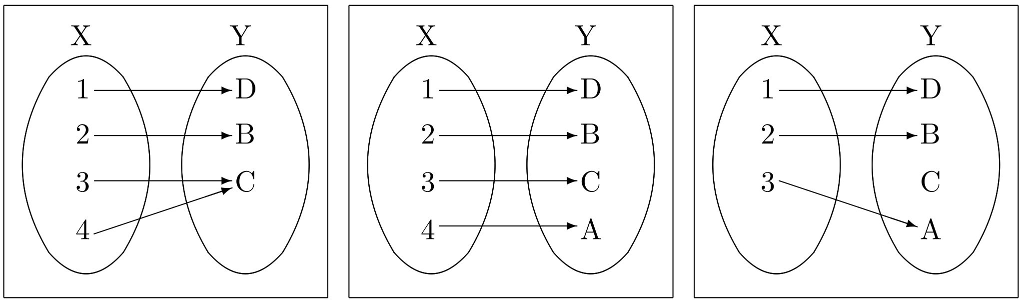
A number of attacks on hash functions have shown weaknesses of M-D construction,and recently hash designs are two types which called"wide-pipe "and "narrow-pipe" hash functions. A wide-pipe iterated hash can produce large-size internal chaining value,the size of internal chaining value is more larger than the final hash value . A narrow-pipe hash can use a big domain to produce internal chaining value ,the size of internal chaining value is equal to the hash value.

There are some reports giving analysis and conclusions: For any a narrow-pipe hash function the entropy and codomain will reduce greatly.The conclusions are mostly based on the process of last additional block in iterative hash functions.

In this text,we will firstly give a simple epimorphic round-function from MD5, then use LAB Mode and build a compression function F, such that: $F = g^* \circ F^*$,Where g^* is a epimorphic round-function , F^* is a normal ideal compression just as the narrow-pipe compression functions in LAB Mode, at last,we will try to avoid the conclusions on narrow-pipe hash functions.

2 Surjections

A function is said to be surjective or onto if its image is equal to its codomain. A function $f : X \to Y$is surjective if and only if for every y in the codomain Y there is at least one x in the domain X such that $f(x) = y$. A surjective function is called a surjection.If $f : X \to Y$ is surjective and B is a subset of Y, then $f(f^{-1}(B)) = B$. Thus, B can be recovered from its preimage $f^{-1}(B)$.



a. A surjective function. b.A bijective function. c.A non-surjective function.

3 The Probability of Empty Set

The conclusions on narrow-pipe hash functions are based on the basic mathematical facts bellow [2]:

Proposition 1. For finite narrow domain: Ideal random functions C map the domain of n-bit strings $X = \{0, 1\}^n$ to itself i.e. to the domain $Y = \{0, 1\}^n$,

Let $\mathcal{F}_C$ be the family of all functions $C : X \to Y$ and let for every $y \in Y$, $C^{-1}(y) \subseteq X$ be the set of preimages of y i.e. $C^{-1}(y) = \{x \in X | C(x) = y\}$. For a function $C \in \mathcal{F}_C$ chosen uniformly at random and for every $y \in Y$ the probability that the set $C^{-1}(y)$ is empty is approximately e^{-1} i.e.

$$P_r\{C^{-1}(y) = \varnothing\} \approx e^{-1}.$$

Proposition 2. For finite wide domain: Ideal random functions W map the domain of $(n+w)$-bit strings $X=\{0,1\}^{n+w}$ to the domain $Y=\{0,1\}^n$,

Let $\mathcal{F}_W$ be the family of all functions $W: X\rightarrow Y$ where $X=\{0,1\}^{n+w}$ and $Y=\{0,1\}^n$. Let for every $y\in Y$, $W^{-1}(y)\subseteq X$ be the set of preimages of y i.e.$W^{-1}(y)=\{x\in X|W(x)=y\}$.For a function $W\in\mathcal{F}_W$,chosen uniformly at random and for every$y\in Y$,the probability that the set $W^{-1}(y)$is empty is approximately e^{-2^w} i.e.

$$P_r\{W^{-1}(y)=\varnothing\}\approx e^{-2^w}.$$

Proposition 3.

Let $C_1: X\rightarrow Y$, $C_2: X\rightarrow Y$ are two particular functions, chosen uniformly at random (where $X=Y=\{0,1\}^n$). If we define a function $C: X\rightarrow Y$ as a composition:

$$C=C_1\circ C_2,$$

then for every $y\in Y$ the probability P_2 that the set $C^{-1}(y)$ is empty is $P_2=e^{-1+e^{-1}}$

Notice that:
1) **Proposition 2**and **Proposition 3** are based on **Proposition 1**,and somewhere the Domain X is according to the entropy namely" the effective and active domain ",e.g:If the last block is a fixed addition namely the entropy is 0,then the entropy of the last iteration only relies on the chaining value(CV),and the domain X equals to the domain of CV.
2) The Domain $X=\{0,1\}^n$ equals Codomain $Y=\{0,1\}^n$ of **Proposition 1**,and for a function $C\in\mathcal{F}_C$ chosen uniformly at random ,the probability that the set $C^{-1}(y)$ is empty is approximately e^{-1} ,it means for a function $C\in\mathcal{F}_C$, it's a general probability including the case of bijective function and non-bijective function.Since $X=Y=\{0,1\}^n$,C is a onto mapping if and only if C is a bijective function.
3) Then,for a general probability of **Proposition 2** $P_r\{W^{-1}(y)=\varnothing\}\approx e^{-2^w}$,it includes the probabilities of surjections and non-surjections.

4 MD5 Structure And It's Surjective Round

The Merkle-Damgard construction is the most common way to transform a compression function $C:\{0,1\}^n\times\{0,1\}^m\rightarrow\{0,1\}^n$ into a hash function $H_C(.)$,the Message Digest is n-bit value.

C denotes the compression function. M denotes the padded and Appended message , it is formatted as $16L$ words :$w_0,w_1,...,w_i,...,w_{16L-1}$ ie.,the message is made up of L m-bit blocks and each the block contains 16 words,for hash code H_C:

CV_i = Chaining variable ,$CV_0=IV$(given Initial Value),M_i = the i-th block

$CV_i = C(CV_{i-1}, M_i)$
$H_C(M) = CV_L$

MD5 Algorithm [3]:

The chaining variables are initialized as: $a_0 = 0x67452301; d_0 = 0x10325476; c_0 = 0x98badcfe; b_0 = 0xefcdab89$;

and for the ith iteration,the chaining variables a_0, b_0, c_0, d_0 are updated by CV_{i-1},i.e., $a_{i-1}, b_{i-1}, c_{i-1}, d_{i-1}$.Copy the i-th block M_i of 16 32-bit words into Buffer:

$m[16] \leftarrow w_{16i+j}$ $(0 \leq j \leq 15\ )$

The 1st Round :

Step1:$\Sigma_1 = a_0 + F(b_0, c_0, d_0) + m_0 + 0xd76aa478$, $a_1 = b_0 + \Sigma_1 <<< 7$;
Step2:$\Sigma_2 = d_0 + F(a_1, b_0, c_0) + m_1 + 0xe8c7b756$, $d_1 = a_1 + \Sigma_2 <<< 12$;
Step3:$\Sigma_3 = c_0 + F(d_1, a_1, b_0) + m_2 + 0x242070db$, $c_1 = d_1 + \Sigma_3 <<< 17$;
Step4:$\Sigma_4 = b_0 + F(c_1, d_1, a_1) + m_3 + 0xc1bdceee$, $b_1 = c_1 + \Sigma_4 <<< 22$;
.....
Step13:$\Sigma_{13} = a_3 + F(b_3, c_3, d_3) + m_{12} + 0x6b901122$, $a_4 = b_3 + \Sigma_{13} <<< 7$;
Step14:$\Sigma_{14} = d_3 + F(a_4, b_3, c_3) + m_{13} + 0xfd987193$, $d_4 = a_4 + \Sigma_{14} <<< 12$;
Step15:$\Sigma_{15} = c_3 + F(d_4, a_4, b_3) + m_{14} + 0xa679438e$, $c_4 = d_4 + \Sigma_{15} <<< 17$;
Step16:$\Sigma_{15} = b_3 + F(c_4, d_4, a_4) + m_{15} + 0x49b40821$, $b_4 = c_4 + \Sigma_{16} <<< 22$;

For the First Round of the ith iteration,no matter how the input values of chaining variables(updated a_0, b_0, c_0, d_0) are prescribed arbitrarily, the output of the chaining variables a_4, b_4, c_4, d_4 can achieve any values prescribed arbitrarily by selecting the input values of $m_{12}, m_{13}, m_{14}$and m_{15}. If we mark the first round as a function g,then g is a surjection,$g : X \to Y$,i.e,$g : (0,1)^n \times (0,1)^m \to (0,1)^n$.where X=$(0,1)^m$,$Y = (0,1)^n$,$m = n + w, n = 128, w = 384$.

Namely for any given input of chaining variable CV_{i-1},the codomain is recovered completely, the mapping from the first round is a surjection, this doesn't depend on the previous chaining variable CV_{i-1}.

For $g : X \to Y$, and let for every $y \in Y$, $g^{-1}(y) \subseteq X$ be the set of preimages of y i.e. $g_1^{-1}(y) = \{x \in X | g(x) = y\}$. For a function g chosen uniformly at random and for every $y \in Y$ the probability that the set $g^{-1}(y)$ is empty doesn't exist. i.e.

$$P_r\{g^{-1}(y) = \varnothing\} \approx e^{-2^w} \text{ doesn't exist.}$$

The rest 3 rounds of MD5 can be regarded as a function $C^* : (0,1)^n \times (0,1)^m \to (0,1)^n$,then the compression function C of MD5 is: $C = g \circ C^*$.

Since the codomain can be recovered completely in each iteration firstly, the case that the **continuous and cumulative reducing** of entropy does't exist.

There may be the probability that the set $C^{-1}(y)$ is empty is approximately e^{-2^w} in each iteration of the compression function C of MD5 ,i.e.

$$P_r\{C^{-1}(y) = \varnothing\} \approx e^{-2^w},$$

but this is discrete and independent,it's not cumulate,and the positions of empty set in each iteration are different .

It is necessary to make the difference between the input of Function C^* and of Function g,this can avoid the specificity of the mapping, we can see the sequences of the input$(m_0, m_1, ..., m_{15})$ in 4 rounds of MD5 are different.

5 Ideal Compression Function

Quote from Vlastimil Klima and Danilo Gligoroski [4]:
– hlen - the length of the chaining variable.
– mlen - the length of the message block.
– hashlen - the length of the hash function output.

If the compression function has the property, that for every value m the function $C(h, m) \equiv Ch(h)$ is an ideal random function of the variable h, we denote it as $IRF(h)$.

If the compression function has the property, that for every value h the function $C(h, m) \equiv Ch(m)$ is an ideal random function of the variable m, we denote it as $IRF(m)$.

The hash function is defined by a narrow-pipe compression function (NPCF), iff $hashlen = hlen = \frac{mlen}{2}$ and the compression function is $IRF(h)$ and $IRF(m)$.

The hash function is defined by a wide-pipe compression function (WPCF), iff $hashlen = \frac{hlen}{2} = \frac{mlen}{2}$and the compression function is $IRF(h)$ and $IRF(m)$.

Then,we can regard MD5 is a narrow-pipe approximate $IRF(m)$ of surjection but not a $IRF(h)$ of surjection.

If message input M chosen is fixed and invariant,there can't get surjection g so that the codomain can be recovered . There are continuous sets such $C^{-1}(y)$ is empty in each iteration, the entropy and the codomain will reduce greatly.We'd amend the function g again.

The conclusions on narrow-pipe hash functions are mostly based on the process of last additional block in iterative hash functions.One of the key questions is that processing the last block with additional bits in a normal iterative hash function,there's the entropy of CV_{L-1} only n bits,namely a n-bit domain X maps to a n-bit codomain Y,the probability of empty set is approximately e^{-1}.

There's already the way of adding CV_i into the next iteration such as :

$CV_{i+1} = CV_i + C(CV_i, M_i)$,the Davies-Meyer Mode ,which is also used in MD4, MD5, SHA-1, and SHA-2,but this is not sufficient enough.

For a narrow-pipe $IRF(m)$(where $hashlen = hlen = \frac{mlen}{2}$), firstly,we amend the input M_i as:$M_i = m^*(h_{i-1}, M_i)$*(This can take the mixing into each step function of computing and it is different from the Davies-Meyer Mode)*,and we can mark the new surjective function as g^*.

Then the codomain is recovered by g^* firstly,the compression function C is amended to be a approximate $IRF(m)$ and $IRF(h)$ of surjection F, such that:

$F = g^* \circ F^*$, i.e,the entropy and value of CV_{i-1} can be put into each message input M_i,and the codomain can be recovered in each iteration.

Firstly, the case that the continuous and cumulative reducing of entropy won't exist.

The second,by mixing CV_i into each step function,we can hold the entropy.

The third,LAB Mode increases a much big domain of input by providing additional block in each iteration, we can define the last block M_L^* and use the compression function F in LAB Mode to avoid the case of the other hashes in **Proposition 1**,in which the probability P_r of empty set is approximately e^{-1}.

6 One Mode of Narrow Pipe Hash

E.g.,for $hashlen = hlen = 512 = \frac{mlen}{2}$,$mlen = 1024$,amend a 512-bit r-round compression f of MD construction into ideal compression F of LAB Form C ,and to get a secure hash function construction .The compression F. :

$$CV_i = F(CV_{i-1}, \sum M_{i-1}, M_i) \ ;$$

where $\sum M_i$ is the sum (modulo addition) of the i blocks ,it can also be the xor operation of the i blocks.

And for each unit $\Sigma w_{i,j}$ of Block ,define :

$\sum M_i$ $(\Sigma m_{i,o}, \Sigma m_{i,1}, ..., \Sigma m_{i,15})$ $(1 \leq i \leq L$,$0 \leq j \leq 15)$,:

$\sum M_i = M_i + \sum M_{i-1}$ i.e.:

$\Sigma m_{i,j} = m_{i,j} + \Sigma m_{i-1,j}$

LAB Form C :

1.Append padding bits and append length just as M-D Structure:

The message is padded with single 1-bit followed by the necessary number of 0-bits,so that its length l congruent to 896 modulo1024 $[l \equiv 896(mod 1024)]$,append a block of 128 bits as an unsigned 128-bit integer(most significant byte first)and contains the length of the original message.M denotes the message after padding bits and appending length. message. M is split to be L blocks:$M_1 M_2 ... M_L$,i.e.,M is made up of $w_0, w_1, ..., w_{16L-1}$.

2.Define a additive block M_0,encode the size of hash value n into M_0, just like HAIFA [6].

Amend the last block as $M_L^* = M_{L-1} + M_L$(But this amending is not the necessary in LAB mode.).

3.Define an initial value IV,Set Array $A[16]$ and Array $B[16]$,for a r-round compression function F ,for i from 1 to L,do the following operations of each iteration,and get the hash value $h_F(M)$:

$CV_0 = IV$,$\sum M_0 = M_0$

$CV_i = F(h_{i-1}, \sum M_{i-1}, M_i)$

$CV_L = F(h_{L-1}, \sum M_{L-1}, M_L^*)$,The hash value is CV_L.

4.Truncate the final chaining value if needed.

The Details (e.g.):

Define :$\sum M_i = M_i + \sum M_{i-1}$ ($1 \le i \le L$,$0 \le j \le 15$) i.e.: $\Sigma m_{i,j} = m_{i,j} + \Sigma m_{i-1,j}$

Define :$\sum M_0 = M_0$ and $A[16] = \sum M_0$.

Split chaining variable CV_{i-1} into 8 words:$h_{i-1,0}, h_{i-1,1}, ..., h_{i-1,7}$.

For the i-th iteration,Update $A[16]$ and $B[16]$:

For $1 \le i \le L$,copy the i-th block M_i of 16 64-bit words and add the 8 chaining variables into Buffer:

1).$B[16] \leftarrow (w_{16i+j} + h_{i-1,jmod8})$ $(0 \le j \le 15\ )$

2).$A[16] \leftarrow (B[16] + A[16])$. ie., $A[16] \leftarrow \sum M_i$

On this wise,the entropy and the values of CV_{i-1} are completely mixed into $\sum M_i$ and M_i ,this is different from the Davies-Meyer Mode.

Amend all the nonlinearity step functions $s_{t,j}(m_j)$ $(1 \le t \le r, 0 \le j \le 15)$.

1.For the step functions $s_{t,j}(m_j)$ of the first round,amend them ,such that:

$S_{t,j} = s_{t,j}(\Sigma m_{i,j}) + m_{i,j}$ ($t = 1$,The first round is **Surjection** g^*.)

2.For the rest rounds ,

$S_{t,j} = s_{t,j}(m_{i,j}) + \Sigma m_{i,j}$ $(2 \le t \le r)$.

The input mode of the rest rounds is different from which of the first round.

In this case,for each iteration of LAB Mode C:$CV_i = F(h_{i-1}, \sum M_{i-1}, M_i)$,ideal random compression function F maps the domain $X = \{0,1\}^{n+w}$ of $(n+w)$-bit strings (block $\sum M_i$ and block M_i) to the domain $Y = \{0,1\}^n$,where $n = 512, m = 2 \times 1024 = 2048, w = m - n = 2048 - 512 = 1536$,
so,the probability of empty set is(According to **Proposition 2.**) :

$P_r\{F^{-1}(y) = \varnothing\} \approx e^{-2^w} = e^{-2^{1536}}$

The probability of empty set can be ignored in each iteration even Function g^* is not a surjection namely the codomain isn't recovered completely.

The computing of the last iteration is:

$CV_L = F(CV_{L-1}, \sum M_{L-1}, M_L^*)$,

where,$M_L^* = M_{L-1} + M_L$, $\sum M_{L-1} = M_{L-1} + \sum M_{L-2}$, this avoid the simpleness and specificity of the last block.Even the last block is fixed addition,the entropy of input ($CV_{L-1}, \sum M_{L-1}$and M_L^*)is much enough for the probability of empty set:$hlen = n = 512; n + w \ge 512 + 1024$;.

The probability of empty set in the last iteration is:

$P_r\{F^{-1}(y) = \varnothing\} \approx e^{-2^w} \le e^{-2^{1024}}$

7 About Wide-Pipe Hashes

By the way ,if a compression function doesn't recover the codomain in each iteration ,in some cases,the conclusions that the entropy and codomain will reduce greatly will apply .[4]

The only reason for wide-pipe hashes is that the entropy reduction after applying the compression function $C(Hi, Mi)$ to different message blocks starts from the value $hlen$ which is two times bigger than $hashlen$ i.e.,$hlen = 2hashlen$.

For a wide-pipe hash,since $hlen = mlen$,it maybe necessary to build a surjection g^* to recover the codomain in each iteration(the surjection g^* can be found if and only if the mapping is a bijective function,namely a one-to-one mapping.),for if the entropy and codomain reduced greatly,how about the distribution of the empty set and what about the matching between $hlen$ ($hlen = 2hashlen$)and $hashlen$?

However,it is meaningful for hash design that the observation of the reducing of entropy and codomain.

References

[1] Ralph C. Merkle, *One Way Hash Functions and DES*, Advances in Cryptology, proceedings of CRYPTO 1989, Lecture Notes in Computer Science 435, pp. 428 C 446, Springer-Verlag, 1990.

[2] Danilo Gligoroski, *Narrow-pipe SHA-3 candidates differ significantly from ideal random functions defined over big domains*, NIST hash-forum mailing list, 7 May 2010

[3] Jie Liang and Xuejia Lai , *Improved Collision Attack on Hash Function MD5*Cryptology ePrint Archive2005/425

[4] Vlastimil Klima and Danilo Gligoroski,*Generic collision attacks on narrow-pipe hash functions faster than birthday paradox, applicable to MDx, SHA-1, SHA-2, and SHA-3 narrow-pipe candidates*Cryptology ePrint Archive2010/430

[5] Xigen Yao,*LAB Form for Iterated Hash Functions*,Cryptology ePrint Archive2010/269

[6] Eli Biham, Orr Dunkelman, *A Framework for Iterative Hash Functions HAIFA*, NIST 2nd hash function workshop, Santa Barbara, August 2006.