The Design Principle of Hash Function with Merkle-Damgård Construction

Duo Lei[1], Feng Guozhu[2], Li Chao[1], Feng Keqin[2], and Longjiang Qu[1]

[1] Department of Science, National University of Defense Technology, Changsha, China
Duoduolei@163.com
[2] Department of Math, Tsinghua University, Beijing, China

Abstract. The paper discusses the security of compression function and hash function with Merkle-Damgård construction and provides the complexity bound of finding a collision and primage of hash function based on the condition probability of compression function $y = F(x, k)$. we make a conclusion that in Merkle-Dammaård construction, the requirement of free start collision resistant and free start collision resistant on compression function is not necessary and it is enough if the compression function with properties of fix start collision resistant and fix start preimage resistant. However, the condition probability $P_{Y|X=x}(y)$ and $P_{Y|K=k}(y)$ of compression function $y = F(x, k)$ have much influence on the security of the hash function. The best design of compression function should have properties of that y is uniformly distributed for all x and k.

KeyWord: Hash Function, Block Cipher, Merkle-Damgård Construction

1 Introduction

Most of hash functions are iterated hash function and most of compression function are iterated by Merkle-Damgård structure with constant IV[3]. Since the MD5 and SHA1 are attacked by [8][14][16], more and more attentions have been paid on hash function, the discussion about hash function mainly include security of compression function, attacking methods on hash function and security of iterated structure.

Let the compression function $F : \{0,1\}^{\kappa} \times \{0,1\}^n \to \{0,1\}^n$, $x_h \in \{0,1\}^n$, $x_m \in \{0,1\}^{\kappa}$, $y \in \{0,1\}^n$, where $y = F(x_m, x_h)$, in hash iteration x_h is chaining value. The compression function of iterated hash function has four way to build[3]: based on block cipher, based on Modular Arithmetic, based on knapsack problem and dedicate hash function. No matter what way be used to design a compression function, the basic requirement on compression function is not invertible, or else we can build a collision on compression function,

since the one way permutation is difficult to build, the condition probability of all known compression function has properties of $\max_y P_{Y|X_h=x_h}(y) > \frac{1}{2^n}$ and $\max_y P_{Y|X_m=x_m}(y) > \frac{1}{2^\kappa}$. In this paper, we get conclusion of that if the compression function is collision resistant and preimage resistant for fix start x_h, then the hash function is secure, the requirement of free start collision resistant and free start preimage resistant are not required. But the condition probability $P_{Y|X_h=x_h}(y)$ and $P_{Y|X_m=x_m}(y)$ are the most important character which we have to consider in design of hash function and the best value are $\max_y P_{Y|X_h=x_h}(y) = \frac{1}{2^n}$ and $\max_y P_{Y|X_m=x_m}(y) = \frac{1}{2^\kappa}$.

The attacking methods on hash function are aimed at finding collision, $m \neq m'$ getting $H(m) = H(m')$, if we can find the collision then we can build forgery to replace the original message. If for any given h_{i-1}, h_i we can find preimage m_i satisfying $h_i = F(h_{i-1}, m_i)$ then we can build a collision in following way, selecting an m'_i randomly, compute $h'_i = F(h_{i-1}, m'_i)$, find m''_{i+1} and satisfy $h_i = F(h'_i, m''_{i+1})$, which implies finding collision of two message $m_i \| \dots \| m_1$ and $m''_{i+1} \| m'_i \| m_{i-1} \| \dots \| m_1$. Finding a second preimage also means finding a collision, so hash function should be immune to collision attack, preimage attack and second preimage attack. The original discussion about immune to attacks on hash function are defined as 'hard' to find the attacks, but the 'hard' is hard to evaluate the security of the hash function, for if n is very small then no 'hard' way to finding the collision no matter how nice the compression function be designed and when n is very large a failure design of hash also means hard to find the collision. The paper make a definition of that if the best way of finding the preimage and collision are exhaustive search, then it is immune against those attack. And also the complexity bounds are given based on condition probability of compression function $P_{Y|X_h=x_h}(y)$ and $P_{Y|X_m=x_m}(y)$. Our complexity is defined as the times needed for computing the compression function.

The most famous iterated structure is M-D structure, which is not immune to extend attack, fix point attack and multi-collision attack, moreover, some slight weakness in compression (like some special plaintexts can make collision) may result in failure of hash function, so some revised structures have been given, include wide-pipe hash and double-pipe hash. Commonly, the security of structure was discussed on condition of compression function be random oracle model, in this paper the security of those structures are given based on discussion about condition probability $P_{Z|X=x}(z)$ and $P_{Z|M=m}(z)$ of hash function H, where $H : \{0,1\}^{\kappa \cdot *} \times \{0,1\}^n \to \{0,1\}^n$, $x \in \{0,1\}^n$, $m \in \{0,1\}^{\kappa \cdot *}$, $z \in \{0,1\}^n$, and $z = H(m, x)$. We find if the compression function is designed with $\max_y P_{Y|X_h=x_h}(y) > \frac{1}{2^n}$, then $\max_z P_{Z|M=m}(z)$ may increased dramatically, but in random oracle model $\max_y P_{Y|X_h=x_h}(y) = \frac{1}{2^n}$, so reanalysis the structure of wide-pipe hash and double-pipe hash, and give some new hash structure which can vanish the increase of $\max_z P_{Z|M=m}(z)$. The padding is adding zero to end of message, so we assume the message length is multiple of block length.

2 Definition

A discrete random variable X is a mapping from the sample space Ω to an alphabed $\mathcal{X}$. X assigns a value $x \in \mathcal{X}$ to each elementary event in the Ω and the probability distribution of X is the function[5]

$$P_X : \mathcal{X} \to \Re : x \mapsto P_X(x) = P[X = x] = \sum_{\omega \in \Omega : X(\omega) = x} P[\omega].$$

If the conditioning event involves another random variable Y defined on the same sample space, the conditional probability distribution of X given that Y takes on a value y is:

$$P_{X|Y=y}(x) = \frac{P_{XY}(x, y)}{P_Y(y)}$$

whenever $P_Y(y)$ is positive . Two random variables X and Y are called independent if for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$:

$$P_{XY}(x, y) = P_X(x) \cdot P_Y(y).$$

Definition 1 (Perfect Secrecy[6]). *A cryptosystem has perfect secrecy if*

$$P_{X|Y=y}(x) = P_X(x)$$

for all $x \in \{0,1\}^n, y \in \{0,1\}^n$.

Definition 2 (Perfect Key Distribution). *A cryptosystem has perfect key distribution if*

$$P_{K|Y=y}(k) = P_K(k)$$

for all $x \in \{0,1\}^n, y \in \{0,1\}^n$.

In fact, $P_{XY}(xy) = P_{X|Y=y}P_Y(y) = P_{Y|X_h=x_h}(y)P_X(x)$, since $P_{X|Y=y}(x) = P_X(x)$, we get $P_{Y|X_h=x_h}(y) = P_Y(y)$.

Definition 3 (Random Oracles[12]). *A fixed-size random oracle is a function* $f : \{0,1\}^n \to \{0,1\}^n$*, chosen uniformly at random from the set of all such functions. For interesting sizes* a *and* b*, it is infeasible to implement such a function, or to store its truth table. Thus, we assume a public oracle which, given* $x \in \{0,1\}^n$*, computes* $y = f(x) \in \{0,1\}^n$.

Let the compression function $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$, $x_h \in \{0,1\}^n$, $x_m \in \{0,1\}^\kappa$, $y \in \{0,1\}^n$, where $y = F(x_m, x_h)$, in hash iteration, x_h is chaining value. Let $H : \{0,1\}^{\kappa \cdot *} \times \{0,1\}^n \to \{0,1\}^n$, $x \in \{0,1\}^n$, $m \in \{0,1\}^{\kappa \cdot *}$, $z \in \{0,1\}^n$, and $z = H(m, x)$.

Definition 4. *Let* $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$, $H : \{0,1\}^{\kappa \cdot *} \times \{0,1\}^n \to \{0,1\}^n$, $\Lambda \subset \{0,1\}^n$. *Let* $\Omega^F \triangleq \{(x_m, x_h, y)\}^F \triangleq \{(x_m, x_h, y) | x_h \in \{0,1\}^n, x_m \in \{0,1\}^\kappa, y \in \{0,1\}^n, y = F(x_m, x_h)\}$. *Let* $\Omega^H \triangleq \{(m, x, z)\}^H \triangleq \{(m, x, z) | x \in \{0,1\}^n, m \in \{0,1\}^{\kappa \cdot *}, z \in \{0,1\}^n, z = H(m, x)\}$. *The* σ*-algebra* $\mathcal{F}$ *is the subsets of* Ω, $\omega^F \in \Omega^F$.

The examples of restriction $\mathcal{E}$ on Ω are as followings:

- $\{(x_{h_0}, x_m, y)\}^F \triangleq \{(x_{h_0}, x_m, y) | (x_{h_0}, x_m, y) \in \Omega^F\}$;
- $\{(x_h, x_m, y) | x_h \in \Lambda\}^F \triangleq \{(x_h, x_m, y) | (x_h, x_m, y) \in \Omega^F, x_h \in \Lambda\}$
- $\{\{(x_h, x_m, y)\}^F\}_{x_h \in \Lambda} \triangleq \bigcup_{x_h \in \Lambda} \{\{(x_h, x_m, y)\}^F\}$

Definition 5 (Finding Preimage). *Finding Preimage of F or H is for given y_0 or z_0 finding $\omega^F \in \{(x_m, x_h, y_0)\}^F$ or $\omega^H \in \{(m, x, z_0)\}^H$.*

Definition 6 (Finding Collision). *Finding Collision of F or H is finding $\omega^F, \omega'^F \in A$ and $A \in \{\{(x_m, x_h, y_0)\}^F\}_{y_0 \in \{0,1\}^n}$ or finding $\omega^H, \omega'^H \in A$ and $A \in \{\{(m, x, z_0)\}^H\}_{z_0 \in \{0,1\}^n}$.*

Definition 7 (Free Start Preimage Resistant). *Preimage resistant of F is that if the best way to find $\omega^F \in \{(x_m, x_h, y_0)\}^F$ is exhaustive search. Preimage resistant of H is that if the best way to find $\omega^H \in \{(m, x, z_0)\}^H$ is exhaustive search.*

Definition 8 (Fix Start Preimage Resistant). *Let $\Lambda \subset \{0,1\}^n$, F is fix start preimage resistant, if the best way to find $\omega^F \in \{(x_{h_0}, x_m, y_0)\}^F$ is exhaustive search. H is fix start preimage resistant , if the best way to find $\omega^H \in \{(x_0, m, z_0)\}^H$ is exhaustive search.*

Definition 9 (Free Start Collision Resistant). *Collision resistant of F is that the best way to find $\omega^F, \omega'^F \in A$ and $A \in \{\{(x_m, x_h, y_0)\}^F\}_{y_0 \in \{0,1\}^n}$ is exhaustive search. Collision resistant of H is that the best way to find $\omega^H, \omega'^H \in A$ and $A \in \{\{(m, x, z_0)\}^H\}_{z_0 \in \{0,1\}^n}$ is exhaustive search.*

Definition 10 (Fix Start Collision Resistant). *Let $\Lambda \subset \{0,1\}^n$, Fix start collision resistant of F is that the best way to find $\omega^F, \omega'^F \in A$ and $A \in \{\{(x_m, x_h, y_0) | x_h \in \Lambda\}^F\}_{y_0 \in \{0,1\}^n}$ is exhaustive search. Fix start collision resistant of H is that the best way to find $\omega^H, \omega'^H \in A$ and $A \in \{\{(m, x, z_0) | x \in \Lambda\}^H\}_{z_0 \in \{0,1\}^n}$ is exhaustive search.*

In hash function attack, the probability of finding a primage or collision is different from tradition point of view of probability. If the compression function F is block cipher E with form of $E_k(x) = y$, then the probabilities of $P_{X|Y=y,K=k}(x)$, $P_{K|Y=y,X=x}(k)$ are both equal 0 or 1 (assume the cipher with perfect key distribution). However, for given y, k, the value x satisfying $y = E_k(x)$ can be found directly by computing $x = E_k^{-1}(y)$, but for given y, x the value k satisfying $y = E_k(x)$ can be found only by exhaustive search of k, that implies we should compute E for each guessing k. So we consider giving new definition about the complexity of finding collision or preimage based on the times computing F being made.

Definition 11. *Let $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$, $H : \{0,1\}^{\kappa \cdot *} \times \{0,1\}^n \to \{0,1\}^n$, $\Lambda \subset \{0,1\}^n$. $\mathcal{P}^F$ and $\mathcal{P}^H$ are defined as the minimum times required*

of computing F with probability of 1 finding a free start preimage of F and H, respectively. $\mathcal{P}_{\Lambda}^{F}$ and $\mathcal{P}_{\Lambda}^{H}$ are defined as the minimum times required of computing F with probability of 1 finding a fix start preimage of F or H, respectively. C^{F} or C^{H} is defined as the minimum times required of computing F with probability of 1 finding free start collision of F or H. C_{Λ}^{H} or C_{Λ}^{H} is defined as the minimum times required of computing F with probability of 1 finding fix start collision of F or H.

If F is block cipher $F(x_m, x_h) = F_{x_h}(x_m)$, from given y, x_h we can compute $x_m = F_{x_h}^{-1}(y)$ that means $\mathcal{P}^F = 1$. But we can't compute x_h directly from give y, x_m, the only way to find k is exhaustive search, we have $\mathcal{P}^F = P_{Y|X_h=x_{h_0}}(y_0)^{-1}$.

3 Hash Properties of Compression Function

Let compression function $y = F(x_m, x_h)$ with $q_{x_h} \stackrel{\triangle}{=} \max_y P_{Y|X_h=x_h}(y)2^{\kappa}$, $q_{x_m} \stackrel{\triangle}{=} \max_y P_{Y|X_m=x_m}(y)2^{n}$ and $q_y \stackrel{\triangle}{=} P_Y(y)2^n 2^{\kappa}$. The conclusions of this section are that the best design of $y = F(x_m, x_h)$ should satisfy $q_{x_h} = q_{x_m} = 1$. We make a assumption of $\frac{1}{0} = 0$.

3.1 Free Start Preimage Resistance

The conclusion of this subsection is Theorem1, the upper bound of free start preimage resistant of F is $\min_{x_m,x_h}\{\frac{2^{\kappa}}{q_{x_h}}, \frac{2^n}{q_{x_m}}\}$, which implies the best selection of free start collision resistant and free start preimage resistant have same requirement on F.

Theorem 1. *Let $y = F(x_m, x_h)$ is free start preimage resistant then:*

$$\mathcal{P}^F = \min_{x_m,x_h}\{\frac{2^n}{q_{x_h}}, \frac{2^{\kappa}}{q_{x_m}}\}. \tag{1}$$

Proof. $F(x_m, x_h)$ is preimage resistant, the only way to get preimage is exhaustive search. The exhaustive search has following ways:

- given y_0, x_h searching x_m with $y = F(x_m, x_h)$, the success probability is:

  $$p = P_{Y|X_h=x_h}(y_0)$$

  We get the minimum complexity is $\frac{2^{\kappa}}{q_{x_h}}$.
- For given y_0, x_m searching x_h, we get the minimum complexity is $\frac{2^n}{q_{x_m}}$.
- For given y_0, randomly searching x_h and x_m, the minimum complexity is $\frac{2^{\kappa}2^n}{q_y}$. □

3.2 Free Start Collision Resistance

Conclusion of this subsection is Theorem2, upper bound of free start collision resistant of F is smaller than $\max_{x_m,x_h,y}\{\sqrt{\frac{2^\kappa}{(q_{x_h}-1)}},\sqrt{\frac{2^n}{(q_{x_m}-1)}},\sqrt{\frac{2^{n+\kappa}}{(q_y-1)}}\}$, which implies the best design of F should satisfy y is uniformly distributed in $\{0,1\}^n$ for each $k\in\{0,1\}^\kappa$ and for each $x\in\{0,1\}^n$.

Theorem 2. *F is not invertible for x_h and x_m then*

$$\mathcal{C}^F=\max_{x_m,x_h,y}\{\sqrt{\frac{2^\kappa}{(q_{x_h}-1)}},\sqrt{\frac{2^n}{(q_{x_m}-1)}},\sqrt{\frac{2^{n+\kappa}}{(q_y-1)}}\} \tag{2}$$

Proof. The collision can be get only by exhaustive search.

- The fastest way to search for collision is the way based on birthday paradox. For random selected x_h searching $x_{m_1},x_{m_2},\dots x_{m_t}$ finding collision of $F(x_h,x_{m_i})=F(x_h,x_{m_j})$. The max probability of success is

$$p=1-\frac{2^\kappa(2^\kappa-2^\kappa P_{Y|X_h=x_h}(y_1))\dots(2^\kappa-\sum_i^{t-1}(2^\kappa P_{Y|X_h=x_h}(y_i))}{\binom{2^\kappa}{t}t!}$$

  Let denote $q_{x_h}\triangleq 2^\kappa\max_y P_{Y|X_h=x_h}(y)$ then

$$\begin{aligned}p&\le 1-\frac{(2^\kappa)(2^\kappa-q_{x_h})\dots(2^\kappa-q_{x_h}(t-1))}{(2^\kappa)(2^\kappa-1)\dots(2^\kappa-t+1)}\\&=1-\prod_{i=0}^{t-1}\frac{n-iq_{x_h}}{2^\kappa-i}=1-\prod_{i=0}^{t-1}(1-\frac{iq_{x_h}-i}{2^\kappa-i})=1-\prod_{i=0}^{t-1}(1-\frac{i}{2^\kappa-i}(q_{x_h}-1))\\&\approx 1-\prod_{i=0}^{t-1}\exp^{\frac{i}{2^\kappa-i}(q_{x_h}-1)}\approx 1-\prod_{i=0}^{t-1}\exp^{(\frac{i}{2^\kappa}+\frac{i^2}{2^{\kappa 2}})(q_{x_h}-1)}\end{aligned}$$

  Same as birthday paradox, when $t\ge\sqrt{2^\kappa/(q_{x_h}-1)}$, $q_{x_h}>1$ the success probability of collision is bigger than $1/2$. We get the complexity is $\min_{x_m}\sqrt{\frac{2^\kappa}{q_{x_h}-1}}$.
- similar as item 1, we get for selectedx_m the complexity is $\sqrt{\frac{2^n}{q_{x_m}-1}}$;
- similar as item 1, we get for searching x_m, x_h the complexity is $\sqrt{\frac{2^{n+\kappa}}{q_y-1}}$. □

3.3 Fix Start Preimage Resistance

The conclusions of this subsection are Theorem3.

Theorem 3. *Let $y = F(x_m, x_h)$, $\Lambda \subset \{0,1\}^n$ then:*

- *If F is invertible for (y, x_h) then*

$$\mathcal{P}_{\Lambda}^{F} = 1.$$

- *If F is invertible for (y, x_m) and fix start preimage resistant then*

$$\mathcal{P}_{\Lambda}^{F} \geq \frac{2^{\kappa}}{\sum\limits_{x_h \in \Lambda} q_{x_h}}$$

Proof. If F is invertible for (y, x_h), make notation of $x_m = F^{-1}(y, x_h)$.

- select $x_h \in \Lambda$, compute $x_{m_0} = F^{-1}(x_h, y)$, get x_{m_0}, So $\mathcal{P}_{\Lambda}^{F} = 1$.
- there are two ways to search the preimage:
  - select $x_h \in \Lambda$, search x_m satisfy $y_0 = F(x_m, x_h)$, the complexity is $\min\limits_{x \in \Lambda} \frac{2^{\kappa}}{q_{x_h}}$
  - for y_0, select x_m search x_h, for random selected x_m, the maximum probability of success is

$$p = \sum_{x_m} \sum_{x_h \in \Lambda} P_{X_h}(x_h) P_{X_m}(x_m) P_{Y|X_m = x_m, X_h = x_h}(y_0 = F(x_m, x_h))$$

  the minimum requirement of computation times are $\frac{2^{\kappa}}{\sum_{x \in \Lambda} q_{x_h}}$.

□

3.4 Fix Start Collision Resistance

The conclusion of this subsection are Theorem4 , which tell us the best design of F also should satisfy Y is uniformly distributed in $\{0,1\}^n$ for each $k \in \{0,1\}^{\kappa}$ and for each $x \in \{0,1\}^n$.

Theorem 4. *Let $y = F(x_m, x_h)$, $\Lambda \subset \{0,1\}^n$ then:*

- *If F is invertible for (y, x_h) then*

$$\mathcal{C}_{\Lambda}^{F} = \begin{cases} 2 & |\Gamma| > 1 \text{ or } q_{x_m} > 1 \\ 0 & else \end{cases} \tag{3}$$

- *If F is invertible for (y, x_m) and fix start preimage resistant then*

$$\mathcal{C}_{\Lambda}^{F} \geq \min_{x_h \in \Lambda} \{ \sqrt{\frac{2^{\kappa}}{(q_{x_h} - 1)}}, \frac{2^{\kappa}}{\sum\limits_{x_h \in \Lambda} q_{x_h} - 1}, \sqrt{\frac{2^{\kappa|\Lambda|}}{\sum\limits_{x_h \in \Lambda} q_{x_h} - 1}} \}. \tag{4}$$

Proof. If F is invertible for (y, x_h), make notation of $x_m = F^{-1}(y, x_h)$.

- select $x_h \in \Lambda$, and x_m compute $F(x_m, x_h)$, select $x'_h \in \Lambda$, get $x'_m = F^{-1}(x'_h, F(x_m, x_h))$, so $\mathcal{C}^F_\Lambda = 2$.
- The collision can be found in following ways:
  - Since F is fix start preimage resistant, for selected $x_h \in \Lambda$, the fastest way to get collision of x_m, x'_m is random select a $x_{m_1}, \ldots, x_{m_t}$ getting $y = F(x_h, x_{m_i}$, checking $F(x_h, x_{m_i}) = F(x_h, x_{m_j})$ equals or not, similar as proof of Theorem2, the minimum requirement of computation is $\sqrt{\frac{2^\kappa}{(q_{x_h}-1)}}$.
  - if $|\Lambda| > 1$, for given $x_h, x'_h \in \Lambda$ the fastest way to find x_m, x'_m is random select $x_{m_1}, \ldots, x_{m_t}$, compute $y_i = F(x_h, x_{m_i})$ and $y'_j = F(x_h, x_{m_j})$ then check y_i equals y'_j or not, since from Theorem2 we get the minimum requirement of computation is $\sqrt{\frac{2^{\kappa|\Lambda|}}{\sum\limits_{x_h \in \Lambda} q_{x_h}-1}}$.
  - for selected $x_h \in \Lambda$,x_m, get $F(x_m, x_h)$, then minimum computation required for finding $x'_h \in \Lambda$ with $F(x_m, x_h) = F(x'_h, x'_m)$ is $\sum\limits_{x \in \Lambda} \frac{q_{x_h}-1}{2^\kappa}$.

□

4 The Security of M-D Structure

In this section, we give the proves of that if the compression function is free start preimage resistant and collision resistant, then the hash function is free start preimage resistant and but not free start collision resistant, if the compression function is fix start collision resistant and preimage resistant then the hash function is fix start collision resistant and preimage resistant, and also the upper bounds of collision resistance and preimage resistance are given based on the condition probabilities $P_{Y|X_h=x_h}(y)$ and $P_{Y|X_m=x_m}(y)$. And also if the compression function is not immune to free start preimage resistant, then the compression function should be designed with minimum value of $\max_y P_{Y|X_h=x_h}(y)$ and $\max_y P_{Y|X_m=x_m}(y)$, which imply the best design require the Y is uniformly distributed in $\{0,1\}^n$ for each x_h and each x_m, if $n = \kappa$ then the best design of compression function is permutation for each x_h and each x_m.

Let $F : \{0,1\}^\kappa \times \{0,1\}^\kappa \to \{0,1\}^n$ is a compression function of hash function H, the H with M-D construction is defined as(Figure illustration is given in Fig1):

$$H : \{0,1\}^{\kappa \cdot *} \times \{0,1\}^n \to \{0,1\}^n$$

$$H(m, x_h) \stackrel{\triangle}{=} H(m_* \| \ldots \| m_1, x) = F(m_*, F(m_{*-1}, \ldots (F(m_1, x_h)) \ldots))$$

where $x_h \in \{0,1\}^n$, $y = F(x_m, x_h)$, $y \in \{0,1\}^n$, $m \in \{0,1\}^{\kappa \cdot *}$, $m = m_* \| \ldots \| m_1$, $z = F(m_*, \ldots F(m_1, x_h) \ldots)$.

Lemma 1. *Let $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$, $H : \{0,1\}^{\kappa \cdot t} \times \{0,1\}^n \to \{0,1\}^n$, $z = F(m_t, \ldots F(m_1, x) \ldots)$, and $m_1, \ldots, m_t$ are independent from each other then:*

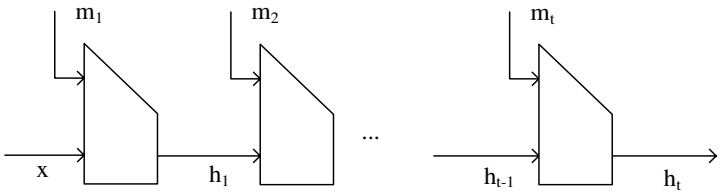

Fig. 1. The M-D Hash

- $P_{Z|M=m}(z) \leq \frac{q_{x_m}^t}{2^n}$
- $P_{Z|X_h=x_h}(z) \leq \frac{q_{x_h}}{2^\kappa}$.

Proof. It is clear $t = 1$ the inequality is correct, when $t = 2$:

$$
\begin{aligned}
&P_{Z|M=m}(z) = P_{Z|M=m_2\|m_1}(z) \\
&= \sum_{x_h} P_{X_h}(x_h) P_{Z|M=m_2\|m_1, X_h=x_h}(z = F(m_2, F(m_1, x_h))) \\
&= \sum_{x_h} \sum_{u} P_{X_h}(x_h) P_{Z|M=m_2\|m_1, X_h=x_h}(z = F(m_2, u), u = F(m_1, x_h)) \\
&= \sum_{u} P_{Z|M_2=m_2, U=u}(z = F(m_2, u)) \sum_{x_h} P_{X_h}(x_h) P_{U|M_1, X_h}(u = F(m_1, x_h)) \\
&= \sum_{u} P_{Z|M_2=m_2, U=u}(z = F(m_2, u)) P_{U|M_1=m_1}(u) \\
&\leq q_{x_m} \sum_{u} \frac{1}{2^n} P_{Z|M_2=m_2, U=u}(z = F(m_2, u)) \leq q_{x_m} P_{Z|M_2=m_2}(z)
\end{aligned}
$$

$$
\begin{aligned}
&P_{Z|X_h=x_h}(z) \\
&= \sum_{m_1, m_2} P_M(m_1) P_M(m_2) P_{Z|M=m_2\|m_1, X_h=x_h}(z = F(m_2, F(m_1, x_h))) \\
&= \sum_{m_1, m_2} \sum_{u} P_M(m_1) P_M(m_2) P_{Z|M=m_2\|m_1, X_h}(z = F(m_2, u), u = F(m_1, x_h)) \\
&= \sum_{m_2} \sum_{u} P_M(m_2) P_{Z|M_2, U}(z = F(m_2, u)) \sum_{m_1} P_M(m_1) P_{U|M_1, X_h}(u) \\
&= \sum_{m_2} \sum_{u} P_M(m_2) P_{Z|M_2, U=u}(z = F(m_2, u)) P_{U|X_h=x_h}(u) \\
&= \sum_{u} P_{Z|U=u}(z) P_{U|X_h=x_h}(u) \leq \frac{q_{x_h}}{2^\kappa} \sum_{u} P_{U|X_h=x_h}(u) = q_{x_h}/2^\kappa.
\end{aligned}
$$

Let assume when $t \leq l - 1$ the inequality is true, when $t = l$

$$P_{Z|M=m}(z)$$

$$
\begin{aligned}
&= \sum_{x_h} P_{X_h}(x_h) P_{Z|M'=m'\|m_1,X_h=x_h}(z = H(m', F(m_1, x_h))) \\
&= \sum_{u} P_{Z|M'=m',U=u}(z = H_{X_h}(m', u)) P_{U|M_1=m_1}(u) \\
&\leq q_{x_m} \sum_{u} \frac{1}{2^n} P_{Z|M'=m',U=u}(z = H_{X_h}(m', u)) \leq {q_{x_m}}^l 2^{-n}
\end{aligned}
$$

$$
\begin{aligned}
&P_{Z|X_h=x_h}(z) \\
&= \sum_{m',m_1} P_M(m') P_M(m_1) P_{Z|M=m'\|m_1,X_h=x_h}(z = H(m', (F(m_1, x_h)))) \\
&= \sum_{m',m_1,u} P_{M'}(m') P_M(m_1) P_{Z|M=m'\|m_1,X_h,U}(z = H(m', u), u = F(m_1, x_h)) \\
&= \sum_{m'} \sum_{u} P_{M'}(m') P_{Z|M'=m',U=u}(z = H(m', u)) P_{U|X_h=x_h}(u) \\
&= \sum_{u} P_{Z|U=u}(z) P_{U|X_h=x_h}(u) \leq \frac{q_{x_h}}{2^\kappa} \sum_{u} P_{U|X_h=x_h}(u) = \frac{q_{x_h}}{2^\kappa}.
\end{aligned}
$$

From induction principle we get the conclusions. □

Theorem 5. *If $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ is preimage resistant and collision resistant, $H : \{0,1\}^{\kappa \cdot t} \times \{0,1\}^n \to \{0,1\}^n, x \in \{0,1\}^n, m \in \{0,1\}^{\kappa \cdot t}, y \in \{0,1\}^n$, $z \in \{0,1\}^n$, $y = F(x_m, x_h)$ and $z = F(m_t, \ldots F(m_1, x) \ldots)$ then:*

- *if F is preimage resistant and collision resistant*

$$\mathcal{P}^H \geq \min_{x_m, x_h} \{\frac{2^\kappa}{q_{x_h}}, \frac{2^n}{q_{x_m}}\} \tag{5}$$

$$\mathcal{C}^H = 2 \tag{6}$$

- *If F is invertible for (y, x_h) then*

$$\mathcal{P}^H_\Lambda = \frac{|M|}{\kappa}$$

$$\mathcal{C}^H_\Lambda = \frac{|M| + |M'|}{\kappa}$$

- *If F is invertible for (y, x_m) and fix start preimage resistant then*

$$\mathcal{P}^H_\Lambda \geq \min\{\frac{2^\kappa}{\sum\limits_{x \in \Lambda} q_{x_h}}, \sqrt{\frac{2^\kappa}{q_{x_h}}}, \frac{2^n}{q_{x_m}^{\frac{|M|}{\kappa}}}\} \tag{7}$$

$$\mathcal{C}^H_\Lambda \geq \min_{x_h \in \Lambda, x_m} \{\frac{2^n}{q_{x_m}^{\frac{|M|}{\kappa}}}, \sqrt{\frac{2^\kappa}{(q_{x_h} - 1)}}, \frac{2^\kappa}{\sum\limits_{x \in \Lambda} q_{x_h} - 1}, \sqrt{\frac{2^{\kappa|\Lambda'|}}{\sum\limits_{x \in \Lambda'} q_{x_h} - 1}}\} \tag{8}$$

– *If F preimage resistant and collision resistant then*

$$\mathcal{P}_{\Lambda}^{H} \geq \min_{k\in\Gamma}\{\frac{2^n}{q_{x_m}}, \frac{2^n}{q_{x_h}^{\frac{|M|}{n}}}\} \tag{9}$$

$$\mathcal{C}_{\Lambda}^{H} \geq \min_{k\in\Gamma,x}\{\frac{2^n}{q_{x_h}^{\frac{|M|}{n}}}, \sqrt{\frac{2^n}{(q_{x_m}-1)}}, \sqrt{\frac{2^{n|\Gamma'|}}{\sum\limits_{k\in\Gamma'} q_{x_m}-1}}\} \tag{10}$$

Proof. If F is invertible, then denote $x_m = F^{-1}(y, x_h)$.

– If F is preimage resistant and collision resistant:
  - Let assume for given y find m, x satisfying $H(m_t\| \dots \|m_1, x) = y$ then we find $H(m_{t-1}\| \dots \|m_1, x), m_t$ satisfying $F(m_t, H(m_{t-1}\| \dots \|m_1, x)) = y$, from Theorem1 we get the conclusion.
  - Since $H(m_2\|m_1, x) = H(m_2, H(m_1, x))$, then we find collision.
– If $x_m = F^{-1}(y, x_h)$ then: The conclusions $\mathcal{P}_{\Lambda}^{H} = \frac{|M|}{\kappa}$, $\mathcal{C}_{\Lambda}^{H} = \frac{|M|+|M'|}{\kappa}$ can be get by the direct computation, since $x_m = F^{-1}(y, x_h)$.
– If F is fix start preimage resistant and fix start collision resistant:
  - there are two ways to find the preimage:
    * Case 1 : Using directly search way to find the preimage of z, directly searching $m \in \{0,1\}^{\kappa\cdot *}$ satisfying $z = H(m, x)$ where $x \in \Lambda$. F is fix start preimage resistant, which implies for given z, x the only way of finding m satisfying $z = H(m, x)$ is exhaustive search, more precisely, From Lemma1 and Theorem3 we get the requirement of minimum computation is $\min\{\frac{2^\kappa}{q_{x_h}}\frac{|M|}{\kappa}, \sum\limits_{x\in\Lambda}\frac{2^\kappa}{q_{x_h}}$.
    * Case 2 : Using meet in middle attack way to find the preimage, for given z, search $m' \in \{0,1\}^{\kappa\cdot t'}$, $m'' \in \{0,1\}^{\kappa\cdot t''}$, satisfying $z = H(m'', u)$ and $u = H(m', x)$ where $x \in \Lambda$:
      · Select m' randomly, searching m'', let $\Lambda' \triangleq \{H(m', x), x \in \Lambda\}$, the problem become case 1;
      · Select m'' randomly, get u from $z = H(m'', u)$, then searching m' satisfying $u = H(m', x)$, equals finding the preimage of u;
      · Guessing m' and m'', compute u and u' from $u = H(m', x)$ and $z = H(m'', u')$, let $t = |m''|$, the probability of $u = u'$ smaller than[**?**] $\sqrt{\frac{2^\kappa}{q_{x_h}}}$,
      · if the compression function is designed with property of that, $\exists\dot{z} \in \{0,1\}^n$, $\dot{m} \in \{0,1\}^{\kappa t}$ satisfy $P_{Z|M=\dot{m}}(\dot{z}) = q_{x_m}^t$ and $q_{x_m} > 1$, then the complexity of finding preimage of $\dot{z}$ is $\frac{2^n}{q_{x_m}^t}$, where we search m satisfy $\dot{z} = H(\dot{m}\|m, x)$.

    From Case 1 and Case 2, we get the conclution.
  - there are three ways to find the collision :

* Case 1: Directley finding collision of H: that means search $m' \in \{0,1\}^{\kappa \cdot t'}$, $m'' \in \{0,1\}^{\kappa \cdot t''}$ satisfying $H(m',x) = H(m'',x)$ with $x \in \Lambda$. F is preimage resistant implies for given z, x the only way of finding m satisfying $z = H(m,x)$ is exhaustive search. From Lemma1 and Theorem4 we get by directly search the minimum requirement of computation is $\min\limits_{x_h \in \Lambda}\{\sqrt{\frac{2^\kappa}{(q_{x_h}-1)}}, \frac{2^\kappa}{\sum\limits_{x \in \Lambda} q_{x_h}-1}, \sqrt{\frac{2^{\kappa|\Lambda|}}{\sum\limits_{x \in \Lambda} q_{x_h}-1}}\}$.
* Case 2: search $m \in \{0,1\}^{\kappa \cdot t}$, $m' \in \{0,1\}^{\kappa \cdot t'}$, $m'' \in \{0,1\}^{\kappa \cdot t''}$, satisfying $H(m,x) = H(m'',u)$ and $u = H(m',x)$ where $x \in \Lambda$:
  - if we randomly select m searching m', m'', the problem becomes finding a primage of $z = H(m,x)$;
  - If we randomly select m' get u from $u = H(m',x)$, then search m and m'' satisfying $H(m,x) = H(m'',u)$, let $\Lambda' \stackrel{\triangle}{=} \{H(m',x), x \in \Lambda\} \cup \Lambda$, the problem become case 1 where $x \in \Lambda'$;
  - If randomly select m'' search m, m' check $H(m'', H(m',x)) = H(m,x)$ being satisfied or not, which needs more computation than given m'' finding z and m' satisfying $z = H(m'', H(m',x))$.
* Case 3: search $m \in \{0,1\}^{\kappa \cdot t}$, $m' \in \{0,1\}^{\kappa \cdot t'}$, $\bar{m} \in \{0,1\}^{\kappa \cdot \bar{t}}$, $\bar{m}' \in \{0,1\}^{\kappa \cdot \bar{t}'}$ satisfy $H(m', H(m,x)) = H(\bar{m}', H(\bar{m},x)$ where $x \in \Lambda$, similar as case 2, case 3 needs more computation than case 2.

From Case 1, Case 2 and Case 3, we get the conclusion.

– if F is preimage resistant and collision resistant then the conclusion can be get directly from previous item. □

Theorem5 tell us on condition of the compression function F is free start preimage resistant and free start collision resistant, the best design of H and H_K have properties of $q_{x_m} = 1$ and $q_{x_h} = 1$.

5 Conclusion

The main conclusion of this paper is that if no way to design the compression $F(k,x)$ immune to free start preimage resistant, then the best design of compression function is a block cipher with perfect key distribution and perfect security where the hash function has M-D structure. So the design of block cipher and hash function can be one problem and the design of key schedule algorithm of block cipher become important than before.

References

1. B.Preneel: The State of Cryptographic Hash Functions. In Lectures on Data Security, Lecture Notes in Computer Science, Vol. 1561. Springer-Verlag, Berlin Heidelberg New York (1999) 158-182.
2. B. Preneel, R. Govaerts, and J. Vandewalle, " Hash functions based on block ciphers,", In Advances in Cryptology -CRYPTO'93, Lecture Notes in Computer Science,pages 368-378. Springer-Verlag, 1994.

3. B.Preneel, V. Rijmen, A.Bosselaers: Recent Developments in the Design of Conventional Cryptographic Algorithms. In State of the Art and Evolution of Computer Security and Industrial Cryptography. Lecture Notes in Computer Science, Vol 1528. Springer-Verlag, Berlin Heidelberg New York(1998) 106-131.
4. B. Van Rompay,Analysis and design of cryptographic hash functions, MAC algorithms and block cipher, K. U. Leuven, Juni 2004
5. C.Chchin. Entropy Measures and Uncoditional Security in Cryptography, PHD thesis.
6. C.E. Shannon. "Communication theory of secrecy systems,", Bell System Technical Journal, 28:656 – 715, 1949.
7. C. H. Meyer and S. M. Matyas. Cryptography: a New Dimension in Data Security. Wiley & Sons, 1982.
8. E.Biham and R.Chen. Near-Collisions of SHA-0,In Advances in Cryptology CRYPTO'2004, LNCS 3152,pp290-305,2004.
9. E.Biham and R.Chen. Near-Collisions of SHA-0 and SHA-1. In Selected Areas in Cryptography-SAC 2004.
10. M. O. Rabin. Digitalized Signatures. In R. A. Demillo, D. P. Dopkin, A. K. Jones, and R. J. Lipton, editors, Foundations of Secure Computation, pages 155-166, New York, 1978. Academic Press.
11. I.Damgård. A design principle for hash functions. In G. Brassard, editor, Advances in Cryptology-CRYPTO' 89, volume 435 of Lecture Notes in Computer Science. Springer-Verlag, 1990.
12. J.Black, P.Rogaway, and T.Shrimpton, "Black-box analysis of the block-cipher-based hashfunction constructions from PGV". In Advances in Cryptology - CRYPTO'02, volume 2442 of Lecture Notes in Computer Science. Springer-Verlag, 2002.pp.320-335.
13. J. Daemen and V. Rijmen: The Design of Rijndael: AES The Advanced Encryption Standard. Springer, 2002.
14. X. Wang, H. Yu, How to Break MD5 and Other Hash Functions, EUROCRYPT'2005, Springer-Verlag, LNCS 3494, pp19-35, 2005.
15. X. Lai and J. L. Massey: Hash functions based on block ciphers. In Advances in Cryptology Eurocrypt'92, Lecture Notes in Computer Science, Vol. 658. Springer-Verlag, Berlin Hei-delberg New York (1993) 55-70.
16. X. Wang, X. Lai, D.Feng and H.Yu., Cryptanalysis of the Hash Functions MD4 and RIPEMD, EUROCRYPT 2005, Springer-Verlag,LNCS 3494, pp1-18, 2005.