An improved collision probability for CBC-MAC and PMAC

Avradip Mandal and Mridul Nandi
University of Waterloo, Canada

February 1, 2007

Abstract

In this paper we compute the coliision probability of CBC-MAC [3] for suitably chosen messages. We show that the probability is $\Omega(\ell q^2/N)$ where ℓ is the number of message block, N is the size of the domain and q is the total number of queries. For random oracle the probability is $O(q^2/N)$. This improved collision prbability will help us to have an efficient distinguishing attack and MAC-forgery attack. We also show collision probability for PMAC with collision probability $\Omega(q^2/N)$ (strictly more than birth day bound). We have used a purely combinatorial approach to obtain this bound. The similar analysis can be made for other MAC like XCBC [5], TMAC [9], OMAC [7] etc. We hope this apporach will help us to calculate related probabilties.

Keywords : MAC, CBC-MAC, PMAC, Distinguishing attack, random function.

1 Introduction

1.1 Message Authentication Codes (MAC) and its Security Notions

Definition of MAC

Message Authentication Code or MAC is a secret key version of digital signature. It is used as authentication of a message. A MAC is a family of functions $\{F_k\}_{k \in \mathcal{K}}$ where $F_k : \mathcal{M} \to T$, $\mathcal{M}$ is the message space, T is the set of all tag space and $k \in \mathcal{K}$ is a secret key chosen randomly from a key space. If $t = F_k(M)$ then t is called the tag of the message M.

In this paper we consider $T = D$, a group with addition $+$ and identity element 0 and $\mathcal{M} = D^\ell$ for fixed integer $\ell \geq 2$. In practice, $D = \{0,1\}^{128}$ and $\mathcal{M} = \{0,1\}^{\leq 2^{64}}$. In this case, we need some *padding rule* or preprocessing of a message (which can make message size multiple of 128 bits). We ignore details about padding in this paper. Here we are going to introduce and analyze some attacks on CBC-MAC (Cipher Block-Chaining) [3] and PMAC [6] (Parallelized Message Authentication Code) defined over a fixed length message block, namely D^ℓ.

Security Notions of MAC

There are two popular security notions. Those are distinguishing attack and forgery attack. There are some other variants.

1. **Distinguishing Attack :** Let Adversary $\mathcal{A}^O$ be an oracle algorithm where $O = F_k$ chosen randomly from $\mathcal{F} = \{F_k : k \in \mathcal{K}\}$ or $O = F$ chosen randomly from $Func := \{F : \mathcal{M} \to T\}$.

The adversary can make q queries adaptively and runs in time at most t. It returns either 1 or 0. The advantage for distinguishing attack is computed as follows :

$$\mathbf{Adv}_{pf}(\mathcal{A} : q, t) = |\mathbf{Pr}[\mathcal{A}^{\mathcal{F}} = 1] - \mathbf{Pr}[\mathcal{A}^{Func} = 1|$$

Intuitively, if the advantage is high then the attacker $\mathcal{A}$ can distinguish the random function class and MAC family of functions with high probability. If it is low, we sometimes say that the family $\mathcal{F}$ is a pseudo-random function family.

2. **MAC forgery :** The MAC forgery attackers make successive queries for F_k obtain responses. Let $(M_1, t_1), \cdots, (M_q, t_q)$ be all query-responses of the oracle. If attacker can return a pair (M, t) such that $(M, t) \neq (M_i, t_i)$ for all i and $t = F_k(M)$, then we say that the attacker forges. The probability for forging a message-tag pair is the advantage for MAC-forgery attack.

In general, if one can forge a message (say (M, t)) using this forgery attacker one can make a distinguishing attack (same as the forgery attacker except at the end it will submit the query M and checks whether the response is t or not). Thus, forgery attacker is much stronger. In this paper we will present forgery attacker and automatically it induces a distinguishing attack.

Examples of MAC

In this section we will briefly describe CBC-MAC and PMAC. Later we will analyze attacks on them.

1. **CBC-MAC :** Let f be a function on a group $(D, +)$ (i.e, from $(D, +)$ to $(D, +)$) where $|D| = N$. For a fixed $\ell \geq 1$, define the iterated functions recursively as follow :

$$f^{+}(x_1, \cdots, x_\ell) :=: f^{+}_{\ell}(x_1, \cdots, x_\ell) := f(f^{+}_{\ell-1}(x_1, \cdots, x_{\ell-1}) + x_\ell),$$

where $x_i \in D$, $f^{+}_{0}() :=: f^{+}_{0}(\lambda) := 0$ and λ is the empty string. We denote $f^{+}(x_1, \cdots, x_\ell)$ by $\mathbf{CBC}^f(x_1, \cdots, x_\ell)$.

2. **PMAC :** We define PMAC for fixed length. In this paper we will give an attack for message block length two. For general definition of PMAC (for any size message input) see [6]. Let $(x_1, \cdots, x_{\ell-1}, x_\ell) \in D^\ell$ where each $x_i \in D$ and $\ell > 1$. Compute $w = \sum_{i=1}^{\ell-1} f(x_i + c_i \cdot f(0))$ where $c_1, \cdots, c_{\ell-1}$ are known constants from D. The output of PMAC is $f(w + x_\ell)$. For $\ell = 2$, the value of PMAC at (x_1, x_2) and

$$\mathbf{PMAC}(x_1, x_2) = f(x_2 + f(x_1 + c_1 f(0))).$$

In this paper we consider $\ell = 2$ when we analyze the collision probability for suitable messages.

1.2 Known Results and Our Results

The only known attack on CBC-MAC is based on collision. Suppose we know a collision, $\mathbf{CBC}(X_1) = \mathbf{CBC}(X_2)$ then we know that $\mathbf{CBC}(X_1, x) = \mathbf{CBC}(X_2, x)$ for any $x \in D$. Thus we have the following forgery attack. Make query (X_1, x) and obtains the response t. Then $((X_2, x), t)$ is a valid message-tag pair. For distinguishing attack one can use this information to distinguish. Similar thing holds for PMAC or many other MAC like XCBC, TMAC, OMAC etc.

In this paper we will show how efficiently (in terms of the number of queries) one can obtain a collision for CBC-MAC. We will estimate the collision probability more closely for suitably chosen messages and we can use the similar calculation for PMAC and any other CBC-like construction also.

Our Attack Algorithm for CBC

Let $X_i = (x_i, 0, \cdots, 0) \in D^\ell$ be ℓ-tuples such that each $x_i \in D$'s are distinct, $1 \leq i \leq q$. Clearly, $\mathbf{CBC}^f(x_i, 0, \cdots, 0) = f^{(\ell)}(x_i)$. The function $f^{(i)}(x)$ is defined as follows for $i \geq 0$:

$$i \geq 1,\ f^{(i)}(x) = \overbrace{f \circ \cdots \circ f}^{i \text{ times}}(x) \text{ and } f^{(0)}(x) = x \text{ i.e., } f^{(i)}(x) = f(f^{(i-1)}(x)).$$

We want to find a lower bound of collision probability that is

$$\mathbf{Pr}_f[\mathbf{CBC}^f(X_i) = \mathbf{CBC}^f(X_j) \text{ for some } i \neq j]$$

$$= \mathbf{Pr}_f[f^{(\ell)}(x_i) = f^{(\ell)}(x_j) \text{ for some } i \neq j]$$

$$= \mathbf{Pr}_f[\bigcup_{1 \leq i < j \leq q} C_{i,j}]$$

$$\geq \sum_{i<j} \mathbf{Pr}_f[C_{i,j}] - 3 \sum_{i<j<k} \mathbf{Pr}_f[C_{i,j,k}] - \frac{1}{2} \sum_{\substack{i<j,k<m \\ \{i,j\} \cap \{k,m\} = \emptyset}} \mathbf{Pr}_f[C_{i,j} \cap C_{k,m}] \quad (1)$$

where $C_{i,j}$ denotes the event that $f^{(\ell)}(x_i) = f^{(\ell)}(x_j)$ and $C_{i,j,k}$ denotes the event that $f^{(\ell)}(x_i) = f^{(\ell)}(x_j) = f^{(\ell)}(x_k)$. The last inequality follows from Principle of Inclusion and Exclusion. Now for any event E,

$$\mathbf{Pr}_f[E] = \frac{|\{f : D \to D : E \text{ is true }\}|}{N^N}.$$

Note that N^N is the total number of $f : D \to D$. Thus, this is the probability that a uniform random function on D (or a function chosen uniformly from the set of all functions) satisfies the event E. To have an estimate in Equation 1, we are interested to compute the following bounds

- $|\mathcal{F}_{i,j,k} := \{f : f^{(\ell)}(x_i) = f^{(\ell)}(x_j) = f^{(\ell)}(x_k)\}| \leq 2\ell^2 N^{N-2} + 6\ell^6 N^{N-3}$ where x_i, x_j and x_k are distinct.
- $|\mathcal{F}_{i,j,k,m} := \{f : f^{(\ell)}(x_i) = f^{(\ell)}(x_j), f^{(\ell)}(x_k) = f^{(\ell)}(x_m)\}| \leq N^{N-2}\ell^2 + N^{N-3}(6\ell^3 + 2\ell^5) + 28\ell^8 N^{N-4}$ where x_i, x_j, x_k and x_m are distinct.
- $|\mathcal{F}_{i,j} := \{f : f^{(\ell)}(x_i) = f^{(\ell)}(x_j)\}| \geq N^{N-1} \exp(-\frac{4\ell^2}{N})$ where x_i and x_j are distinct and $1 \leq \ell \leq \frac{N}{4} + \frac{1}{2}$.

To prove above three bound we will use the notions of directed graphs. If $y = f(x)$ then (x, y) will be considered as an edge of a graph. The above event could be translated into a counting problem in graph theory. Mainly to compute the number of non-isomorphic graphs in a special class. We will describe these in more detail later. Now, we have the following main theorem of this paper (by using Equation 1 and three bounds obtained above).

Theorem : The collision probability, $\mathbf{Pr}_f[\mathbf{CBC}^f(X_i) = \mathbf{CBC}^f(X_j)$ for some $i \neq j]$, is at least

$$\Delta := \binom{q}{2}\frac{\ell}{n}exp(-\frac{4\ell^2}{N}) - 3\binom{q}{3}(\frac{2\ell^2}{N^2} + \frac{6\ell^6}{N^3}) - \frac{1}{2}\binom{q}{2}\binom{q-2}{2}(\frac{28\ell^8}{N^4} + \frac{6\ell^3 + 2\ell^5}{N^3} + \frac{\ell^2}{N^2})$$

For large N, the above expression is $\Omega(\frac{q^2\ell}{N}) - c(q, \ell, N)$, when $\frac{q^2\ell}{N} < \frac{8}{3}$ and $\ell = o(N^{\frac{1}{3}})$. Also $c(q, \ell, N)$ goes to zero when $q^2\ell = O(N), \ell = o(N^{\frac{1}{3}})$ and N is large.

Remark 1 *Bellare [1] proved that the CBC-MAC based on random permutation is secure and the advantage is bounded by $O(\ell q^2/N)$ provided $\ell = o(N^{1/3})$. Here we show that there is an attack for CBC based on random function with advantage $\Omega(\ell q^2/N)$. Our idea of the attack algorithm can not be easily extended to CBC based on random permutation. It seems that CBC based on random permutation is more secure than that is based on random function.*

Our Attack Algorithm for PMAC

We provide an analysis of collision probability for PMAC of two block message, that is $\ell = 2$. We choose messages $(x_1, 0), \cdots (x_q, 0) \in D^2$ and want to compute a lower bound for collision probability for these messages. Note that, for $\ell = 2$, PMAC acts very similar to CBC-MAC with $\ell = 2$. We have the following main results for PMAC.

Theorem : When $((x_1, 0), (x_2, 0), ..., (x_q, 0))$ are queried (x_i's are distinct and not equal to zero) to a PMAC oracle, the advantage with respect to a random oracle is at least $\Omega(\frac{q^2}{N}) - \frac{q}{N}$, when $\frac{q^2}{N} \leq c$ for some $d > 0$ such that $q \geq \frac{1}{d}, c \leq \frac{N}{3(1+d)}$, and $q \leq \frac{N}{4}$.

2 A Note on Graph Theory

Directed Graph

- Let $G = (D, E)$ be a directed graph where $E \subset D \times D$ and $|D| = N$. Denote the number of edges by $e(G)$.
- A *degree* of a vertex v (denoted as $\mathbf{deg}(v)$) is the sum of *out-degree* (or $\mathbf{deg_{out}}(v)$) and *in-degree* (or $\mathbf{deg_{in}}(v)$) of the vertex where

$$\mathbf{deg_{out}}(v) = |\{u : (v, u) \in E\}| \text{ and } \mathbf{deg_{in}}(v) = |\{u : (u, v) \in E\}|.$$

- Define $V(G) = \{v : \mathbf{deg}(v) > 0\}$. Denote the number of vertices with positive degree by $r(G)$ i.e., $|V(G)| = r(G)$. We denote $\Delta(G) = r(G) - e(G)$. It is an important parameter of a graph. The corresponding undirected graph of a directed graph is the graph considering all directed edges as undirected. So, it may contain a *self loop* and at most two *parallel edges.* For a connected undirected graph G, G is *tree* if and only if $\Delta(G) = 1$ and G contains exactly one cycle if and only if $\Delta(G) = 0$. We term them as *unicycle graphs.*
- In this paper we are interested in some special directed graphs. They are the following :
  - A *straight line path* of length k is a directed graph $G = (D, E)$ where $E = \{(x_0, x_1), (x_1, x_2), \cdots, (x_{k-1}, x_k)\}$ and $x_0, x_1, \cdots, x_k$ are distinct. Here x_0 is the *source node* and x_k is the *end points of the straight line path.* Here $\Delta(G) = 1$.

- A cycle is a directed graph $G = (D, E)$ where $E = \{(x_0, x_1), (x_1, x_2), \cdots, (x_{k-1}, x_k = x_0)\}$ and $x_0, x_1, \cdots, x_{k-1}$ are distinct. Here k can be 1. In this case, the cycle consists of a single self loop (x_0, x_0) (note that $x_1 = x_0$). Here $\Delta(G) = 0$.
- A s-unicycle is a directed graph $G = (D, E)$ where E is union of cycle C and $s_1(\leq s)$ distinct straight line paths $P_1, \cdots, P_{s_1}$ where the end points of P_i's are vertices of the cycle C. The paths P_i may not be disjoint. Each straight line path contributes at most one node in $V(G)$ with in-degree zero. Thus there are at most s nodes in $V(G)$ with in-degree zero. Here $\Delta(G) = 0$.

- Let $G_1 = (D, E_1)$ and $G_2 = (D, E_2)$ be two directed graphs. A function $\alpha : D \to D$ is an *isomorphism* from G_1 to G_2 if α is bijection and $(x, y) \in E_1$ if and only if $(\alpha(x), \alpha(y)) \in E_2$. In this case, we write $G_1 \cong G_2$. Moreover for $A \subset D$, if α is identity on A then α is called *A-isomorphism* and we write $G_1 \cong_A G_2$. G_1 and G_2 are said to be *A-isomorphic.* If there is no such A-isomorphism then they are *non A-isomorphic.* The notion of A-isomorphism is not standard and we need this notion in this paper.

Example 1 *In the Figure 1 all graphs are isomorphic but G_1 and G_2 are not A-isomorphic where $A = \{1, 2, 4, 5\}$. Clearly, G_1 and G_3 are A-isomorphic.*

Lemma 1 *The number of graphs isomorphic to a given graph $G = (D, E)$ is at most $N(N - 1) \cdots (N - r + 1)$ (which is less than N^r). If $A \subset V(G)$ with size s then the number of graphs A-isomorphic to G is at most $(N - s)(N - s + 1) \cdots (N - r + 1)$ (which is less than N^{r-s}).*

Proof. If $\alpha : G \to G' = (D, E')$ is an isomorphism for an isomorphic copy of G. The graph is completely determined by $V(G')$ and α (α determines uniquely the edge set E'). Now we can choose $V(G')$ in $\binom{N}{r}$ ways. For each choice there are at most $r!$ isomorphisms. Thus we have at most $\binom{N}{r} \times r!$ isomorphic copies of G. Similarly, we can prove for the second part. Note that if G' is an A-isomorphic copy of G and $A \subset V(G)$ then $A \subset V(G')$. Thus, we can choose $V(G')$ in $\binom{N}{r-s}$ ways and each choice there are exactly $(r - s)!$ isomorphisms. ■

Function Graph

- A directed graph $G = (D, E)$ is called a *function graph* if $(x, y_1), (x, y_2) \in E$ then $y_1 = y_2$. A partial function f on D can be uniquely characterized by a function graph and vice versa by the following rule :

$$f(x) = y \text{ if and only if } (x, y) \in E.$$

- Define domain of a function graph as $\mathbf{Dom}(G) = \{v : \mathbf{deg_{out}}(v) > 0\}$. Since it is a function graph $\mathbf{Dom}(G) = \{v : \mathbf{deg_{out}}(v) = 1\}$ and it is the domain of the corresponding partial graph. Clearly, $|\mathbf{Dom}(G)| = e(G)$ (map an edge $(v, w) \in E$ to $v \in \mathbf{Dom}(G)$).

Example 2 *The graphs in Figure 1 are function graphs that is there is no node with out-degree more than one. The partial function corresponding to G_1 is $f(1) = 3, f(3) = 7, f(7) = 8, f(8) = 9, f(9) = 9, f(2) = 3, f(4) = 6, f(6) = 7, f(5) = 7$. The domain of the graph G_1* $\mathbf{Dom}(G_1) = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

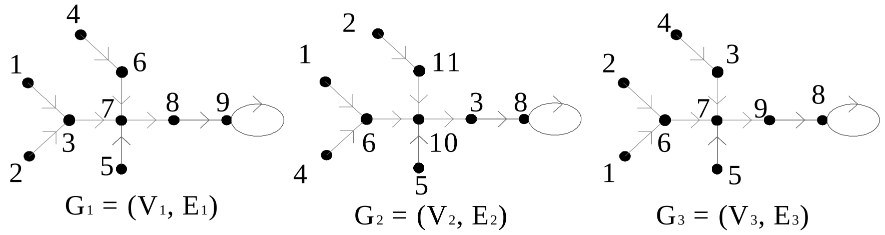

Figure 1: Tree or Unicycle Function Graph

- A function f is called *compatible* with a function graph $G = (D, E)$ if $f(x) = y$ whenever $(x, y) \in E$. Note that for a given function graph G there are N^{N-e} compatible functions $f : D \to D$ with G where $e = |e(G)| = |\mathbf{Dom}(G)|$

$$\left.\begin{array}{rcl} f(x) & = & y \quad \text{if } (x,y) \in E \text{ (in other words } x \in \mathbf{Dom}(G)) \\ & = & * \quad \text{otherwise} \end{array}\right\} \tag{2}$$

  Here " * " means that the function is defined arbitrarily. Now we know that, given a s-set $A \subset \mathbf{Dom}(G)$ there are at most N^{r-s} many A-isomorphic graphs where $r = |V(G)|$. Thus, $|\mathcal{F}_G = \{f : D \to D : f \text{ is compatible with some } G' \cong_A G\}| \leq N^{N-e} \times N^{r-s} = N^{N-s+\Delta}$ where $\Delta = \Delta(G)$.

- **Main trick of this paper :** Suppose we want to find an upper bound of $\mathbf{Pr}[E]$ where E is some event related to uniform random function and probability is computed with respect to the uniform random functions (recall that, each function $f \in \mathbf{Func}(D, D)$, set of all functions $f : D \to D\}$, has equal probability that is $\frac{1}{N^N}$). To do so, we count the number (or give an upper bound) of $f \in \mathbf{Func}(D, D)$ such that the event E is true. Let $\mathcal{F} = \{f : E \text{ is true }\}$. Then $\mathbf{Pr}[E] = \frac{|\mathcal{F}|}{N^N}$. We do the following main steps

  1. Associate each function $f \in \mathcal{F}$ to a function graph $G \in \mathcal{G}$ such that f is compatible with G where $\mathcal{G}$ is some classes of function graphs.
  2. Now for a suitable choice of s-set A, partition $\mathcal{G}$ by A-isomorphism that is $\mathcal{G} = \bigsqcup_{i=1}^{L} \mathcal{G}_i$ where all elements within $\mathcal{G}_i$ are A-isomorphic and graphs between two classes are non-A-isomorphic. Let $\Delta_i = \Delta(G)$ where $G \in \mathcal{G}_i$.
  3. Now we can give an upper bound as follows. We know that $|\mathcal{F}_G| \leq N^{N-s+\Delta_i}$ for $G \in \mathcal{G}_i$. We denote $\mathcal{F}_G$ by $\mathcal{F}_{\mathcal{G}_i}$. Note that, this is well defined. $\mathcal{F} \subseteq \bigsqcup_{i=1}^{L} \mathcal{F}_{\mathcal{G}_i}$ and hence $|\mathcal{F}| \leq \sum_{i=1}^{L} N^{N-s+\Delta_i}$. If we know that there are L_i many classes such that Δ value is $i \geq 0$ then
  $$|\mathcal{F}| \leq \sum_{i \geq 0} L_i \cdot N^{N-s+i}. \tag{3}$$
  4. So it would be enough if one can suitably associate a function to a function graph, choose a suitable set A and compute L_i for all possible values of i.

- Now we consider a special class of function graphs called (ℓ, A)*-iterated function graph* where $A \subset D$ is a s-set. A function graph G is called (ℓ, A)-iterated graph (or function graph) if there exists a function f such that the domain of the graph $\mathbf{Dom}(G) = \{y : f^{(i)}(x) = y, x \in$

$A,\ 0 \le i \le \ell - 1\}$. We denote $G_{\ell,A}[f]$ by the (ℓ, A)-iterated function graph for the function f (this is unambiguous since the graph is completely determined the tuple (ℓ, A, f)). For $A = \{x\}$, we sometimes write $G_{\ell,x}[f]$. Thus $G_{\ell,A}[f]$ is union (may not be disjoint) of $G_{\ell,x}[f]$. Now $G_{\ell,x}[f]$ can be one of the following :

1. $f^{(0)}(x) = x, f^{(1)}(x) = f(x), \cdots, f^{(\ell)}(x)$ are distinct. In this case the graph is a straight line path.
2. $f^{(0)}(x) = x = w_0, f^{(1)}(x) = f(x) = w_1, \cdots, w_\ell = f^{(\ell)}(x)$ are not distinct that is $w_0, \cdots, w_i$ are distinct and $w_{i+1} = w_j,\ j \le i < \ell$. This is a 1-unicycle (or sometimes called ρ straight line path as the structure looks like ρ).

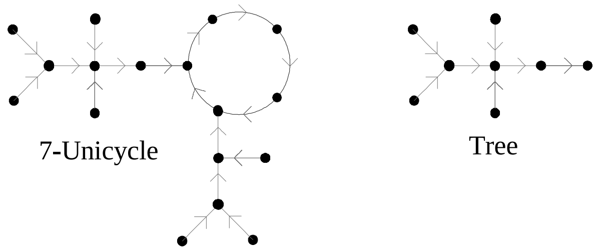

Figure 2: Tree or Unicycle Function Graph

Theorem 2 $G_{\ell,A}[f]$ *is union of unicycles or straight line paths. More particularly, it is disjoint union of tree or unicycle paths where the number of nodes in* $V(G_{\ell,A}[f])$ *with zero out-degree is at most* $s = |A|$.

Proof. We skip the proof as it is straightforward and needs some more notations. One can see this by looking different examples (for example Figure 2).

3 A Lower Bound of Probability for Collision Events

Now we prove the bounds for $|\mathcal{F}_{i,j}|$, $|\mathcal{F}_{i,j,k}|$ and $|\mathcal{F}_{i,j} \cap \mathcal{F}_{k,m}|$.

Upper Bound of $|\mathcal{F}_{i,j,k}|$

First consider $\mathcal{F}_{i,j,k}$ and define $A = \{x_i, x_j, x_k\}$ and associate each $f \in \mathcal{F}_{i,j,k}$ to the function graph $G_{\ell,A}[f] \in \mathcal{G}$ where $\mathcal{G} = \{G_{\ell,A}[f] : f \in \mathcal{F}_{i,j,k}\}$. Now note that $G \in \mathcal{G}$ is either a tree (in which case $\Delta = 1$) or 3-unicycle (in this case $\Delta = 0$). Now we have to count L_i, the number of non-A-isomorphic graphs for $\Delta = i,\ i = 0, 1$.

1. $\Delta = 1$: Let x be the collision node, that is $f^{(\ell)}(x_1) = f^{(\ell)}(x_2) = f^{(\ell)}(x_3) = x$. Each tree is determined by the point of intersection of x_2 and x_1 paths (ℓ choices) and point of intersection of x_3 and x_1, x_2 path (2ℓ choices). Thus, $L_1 \le 2\ell^2$.

2. $\Delta = 0$: Each 3-unicycle graph is determined by the length of the cycle, distance distance from x_i's to the circle, location of the point of intersection of x_2 path and union of x_1 path and cycle and location of point of intersection of x_3 and union of x_1, x_2 path and cycle. Each choice is at most ℓ except the location of point of intersection (for x_3) has 2ℓ choices. Thus $L_0 \leq 6\ell^6$.

Thus, $|\mathcal{F}_{i,j,k}| \leq 2\ell^2 N^{N-2} + 6\ell^6 N^{N-3}$.

Upper Bound of $|\mathcal{F}_{i,j,k,m}|$

We define $A = \{x_i, x_j, x_k, x_m\}$ and associate each $f \in \mathcal{F}_{i,j,k,m}$ to the function graph $G_{\ell,A}[f] \in \mathcal{G}$ where $\mathcal{G} = \{G_{\ell,A}[f] : f \in \mathcal{F}_{i,j,k}\}$. Now we have the following cases :

note that $G \in \mathcal{G}$ is either a tree (in which case $\Delta = 1$) or 3-unicycle (in this case $\Delta = 0$). Now we have to count L_i, the number of non-A-isomorphic graphs for $\Delta = i$, $i = 0, 1$. We can make similar analysis here as follows. We skip all detail.

1. $\Delta = 2$: Disjoint union of two trees consisting two paths each. Let $f^{(\ell)}(x_1) = f^{(\ell)}(x_2) = x$ and $f^{(\ell)}(x_3) = f^{(\ell)}(x_4) = y$. Each tree is determined by the point of intersection of x_2 and x_1 paths (ℓ choices) and point of intersection of x_3 and x_1, x_2 path (2ℓ choices). Thus, $L_1 \leq 4\ell^4$. $L_2 \leq \ell^2$.
2. $\Delta = 1$: G is either a tree or union of 2-unicycle and a tree consisting two paths. $L_1 \leq 6\ell^3 + 2\ell^5$.
3. $\Delta = 0$ Either G is 4-unicycle graph or union of two disjoint 2-unicycle graphs. Similar analysis. $L_0 \leq 28\ell^8$.

Thus, $|\mathcal{F}_{i,j,k,m}| \leq N^{N-2}\ell^2 + N^{N-3}(6\ell^3 + 2\ell^5) + 28\ell^8 N^{N-4}$.

Lower Bound of $|\mathcal{F}_{i,j}|$

Let $\mathcal{F}_{i,j} = \{f : f^{(\ell)}(x_i) = f^{(\ell)}(x_j)\}$ for distinct $x_i, x_j \in D$. Let $\mathcal{F}^k_{i,j} = \{f \in \mathcal{F}_{i,j} : f^{(k)}(x_i) = f^{(k)}(x_j)$ and $f^{(k_1)}(x_i), f^{(k_2)}(x_j)$ are distinct $1 \leq k_1, k_2 \leq k-1\}$ where $1 \leq k \leq \ell$. That is it is the set where all intermediate outputs are distinct before k round and at kth round we have a collision. Clearly they are disjoint sets and $\mathcal{F}_{i,j} \supseteq \bigsqcup_{k=1}^{\ell} \mathcal{F}^k_{i,j}$. So, $|\mathcal{F}_{i,j}| \geq \sum_{k=1}^{\ell} |\mathcal{F}^k_{i,j}|$. The following lemma is a straightforward counting so we skip the proof.

Lemma 3 *For $2 \leq k \leq \ell$, $|\mathcal{F}^k_{i,j}| = (N-2)(N-3)...(N-2k+1)N^{N-2k+1}$ and $|\mathcal{F}^1_{i,j}| = N^{N-1}$.*

Lemma 4 *For $1 \leq \ell \leq \frac{N}{4} + \frac{1}{2}, \mathbf{Pr}_f[C_{i,j}] \geq \frac{\ell}{N}\exp(-\frac{4\ell^2}{N})$*

Proof. From above Lemma we have $|\mathcal{F}| \geq N^{N-1}(1 + \sum_{i=2}^{\ell}(1 - \frac{2}{N})(1 - \frac{3}{N})...(1 - \frac{2k-1}{N}))$.

$$\mathbf{Pr}_f[C_{i,j}] \geq (1 + \sum_{k=2}^{\ell}(1 - \frac{2}{N})(1 - \frac{3}{N})...(1 - \frac{2k-1}{N}))\frac{1}{N}$$

$$\geq \frac{\ell}{N}\prod_{k=1}^{2\ell-1}(1 - \frac{k}{N})$$

$$\geq \frac{\ell}{N} \exp(-\frac{4\ell^2}{N})$$

In the last step we have used the inequality $1 - x \geq \exp(-2x)$. Which is true for $0 \leq x \leq 0.5$ and hence we need that $\ell \leq N/4 + 1/2$. ■

Theorem 5 *The collision probability is at least*

$$\delta = \binom{q}{2} \frac{\ell}{n} exp(-\frac{4\ell^2}{N}) - 3\binom{q}{3}(\frac{2\ell^2}{N^2} + \frac{6\ell^6}{N^3}) - \frac{1}{2}\binom{q}{2}\binom{q-2}{2}(\frac{28\ell^8}{N^4} + \frac{6\ell^3 + 2\ell^5}{N^3} + \frac{\ell^2}{N^2})$$

For large N, the above expression is $\Omega(\frac{q^2\ell}{N}) - c(q, \ell, N)$, when $\frac{q^2\ell}{N} < \frac{8}{3}$ and $\ell = o(N^{\frac{1}{3}})$. Also $c(q, \ell, N)$ goes to zero when $q^2\ell = O(N), \ell = o(N^{\frac{1}{3}})$ and $\ell \leq N/4 + 1/4$ is large.

Proof.

$$\delta \geq \frac{q^2\ell}{3N} \exp(\frac{-4\ell^2}{N}) - \frac{q^3}{2}(\frac{2\ell^2}{N^2} + \frac{6\ell^6}{N^3}) - \frac{q^4}{8}(\frac{28\ell^8}{N^4} + \frac{6\ell^3 + 2\ell^5}{N^3} + \frac{\ell^2}{N^2}).$$

Now putting $\alpha = \frac{q^2\ell}{N}$ and $c(q, \ell, N) = \frac{1}{2}(\frac{q^2\ell}{N})^{1.5}(\frac{2\ell^{0.5}}{N^{0.5}} + \frac{6\ell^{4.5}}{N^{1.5}}) + \frac{1}{8}(\frac{q^2\ell}{N})^2(\frac{28\ell^6}{N^2} + \frac{6\ell + 2\ell^3}{N})$ we get,

$$\delta \geq \alpha(\frac{1}{3} - \frac{\alpha}{8}) - c(q, \ell, N),$$

when $\ell = o(N^{\frac{1}{3}})$ and N is large. Thus,

$$\delta \geq d\alpha - c(q, \ell, N),$$

when $\alpha \leq \frac{8}{3} - d$ and hence

$$\delta \geq \Omega(\frac{q^2\ell}{N}) - c(q, \ell, N),$$

as long as $\frac{q^2\ell}{N} < \frac{8}{3}$. Also $c(q, \ell, N)$ goes to zero when $q^2\ell = O(N), \ell = o(N^{\frac{1}{3}})$ and N is large. ■

Exact computation of collision probability

We can use this to have a distinguishing attack and MAC-forgery. Here we will see how this bound is practically meaningful. We will compute the collision probability numerically for a suitable choices ℓ and q. This calculation is important as sometimes the constant can make a real difference.

Example 3 *MAC forgery for* 64 *bit.*

Taking $q = c_1N^{\frac{1}{3}}$ and $\ell = c_2N^{\frac{1}{3}}$, $\alpha = c_1^2c_2$ we get,
$\delta \approx \frac{\alpha}{2} - \frac{\alpha^2}{8} - (3\alpha^{\frac{3}{2}}c_2^{\frac{9}{2}} + \frac{7}{2}\alpha^2c_2^6 + \frac{1}{4}\alpha^2c_2^3)$
So for small c_2, $\delta \approx \frac{\alpha}{2} - \frac{\alpha^2}{8}$ To maximize δ we take $\alpha = 2$, and we get $\delta \approx 0.5$.
Hence taking $q = \sqrt{20} \cdot 2^{\frac{64}{3}}$, $\ell = 0.1 \times 2^{\frac{64}{3}}$, we get $\delta = 0.499$. ■

Example 4 *MAC forgery for* 128 *bit.*

Taking $q = \sqrt{20} \cdot 2^{\frac{128}{3}}$, $\ell = 0.1 \cdot 2^{\frac{128}{3}}$, we get $\delta = 0.499$. ■

4 Collision Probability for PMAC

Let $((x_1, 0), (x_2, 0), ..., (x_q, 0))$ be the q queries to the PMAC oracle. Let $y_i = x_i + cf(0), w_i = f(y_i), z_i = f(w_i)$, for $1 \leq i \leq q$.

Property I: $(0, y_1, y_2, ..., y_q)$ is a non collision $q + 1$-tuple and $(w_1, w_2, ..., w_q)$ is a collision q-tuples, $w_i \neq 0$. In other words, $(0, w_i)$and $(0, x_i + cf(0))$ are non collision 2-tuple for $1 \leq i \leq q$, $(w_1, w_2, ..., w_q)$ is a collision q-tuple. As x_i's are distinct.We call a $(q+1)$-tuple $(f(0), w_1, w_2, ..., w_q)$ as *kind-I* $(q + 1)$-tuple if it satisfy the above property.

Property II: Similarly, we call a $(2q + 1)$-tuple $(f(0), w_1, w_2, ..., w_q, z_1, z_2, ..., z_q)$ as *kind-II* $(2q + 1)$-tuple if $(0, y_1, y_2, ..., y_q, w_1, w_2, ..., w_q)$ is a non collision $2q + 1$-tuple and $(z_1, z_2, ..., z_q)$ is a collision q-tuple.

Now corresponding to each kind-I $(q + 1)$-tuple there are N^{N-q-1} collision functions. We call such functions as *kind-I collision function.* And corresponding to each kind-II $(2q + 1)$-tuple there are N^{N-2q-1} collision functions. We call such functions as *kind-II collision function.*

- There are $(N - q)((N^q - (N - 1)(N - 2)...(N - q))$ kind-I $(q + 1)$-tuples.
- There are $(N - q)(N - q - 1)(N - q - 2)...(N - 2q)(N^q - N(N - 1)...(N - q + 1))$ kind-II $(2q + 1)$-tuples.

Hence we get the following result, which we mention as Lemma 6.

Lemma 6 *There are at least* $(N - q)((N^q - (N - 1)(N - 2)...(N - q))N^{N-q-1} + (N - q)(N - q - 1)(N - q - 2)...(N - 2q)(N^q - N(N - 1)...(N - q + 1))N^{N-2q-1}$ *many collision functions.*

Theorem 7 *When* $((x_1, 0), (x_2, 0), ..., (x_q, 0))$ *are queried to a PMAC oracle The advantage with respect to a random oracle is at least* $\Omega(\frac{q^2}{N}) - \frac{q}{N}$*, when* $\frac{q^2}{N} \leq c$ *for some* $d > 0$ *such that* $q \geq \frac{1}{d}, c \leq \frac{N}{3(1+d)}$ *and* $q \leq \frac{N}{4}$.

Proof. From Lemma 6 we get collision probability is at least δ. Where,
$\delta = (1 - \frac{q}{N})(1 - (1 - \frac{1}{N})...(1 - \frac{q}{N})) + (1 - \frac{q}{N})...(1 - \frac{2q}{N})(1 - (1 - \frac{1}{N})...(1 - \frac{q-1}{N}))$.

$$\delta \geq (1-(1-\frac{1}{N})...(1-\frac{q-1}{N}))-\frac{q}{N}(1-(1-\frac{1}{N})...(1-\frac{q}{N}))+(1-\frac{q}{N})...(1-\frac{2q}{N})(1-(1-\frac{1}{N})...(1-\frac{q-1}{N}))$$

Hence, $\mathbf{Adv} \geq (1 - \frac{q}{N})...(1 - \frac{2q}{N})(1 - (1 - \frac{1}{N})...(1 - \frac{q-1}{N})) - \frac{q}{N}(1 - (1 - \frac{1}{N})...(1 - \frac{q}{N}))$ So When $\frac{1}{d} \leq q \leq \frac{N}{4}$, we get $\mathbf{Adv} \geq \exp(\frac{-3(1+d)q^2}{N})(1 - \exp(\frac{-(1-d)q^2}{2N}) - \frac{q}{N})$
When $q^2 \leq \frac{N}{3(1+d)}$, the above expression is

$$\geq (\frac{1-d}{2})(\frac{q^2}{N})(1 - 3(1 + d)(\frac{q^2}{N}))(1 - (\frac{1-d}{4})(\frac{q^2}{N}) - \frac{q}{N})$$

Now if there exist constant c such that $\frac{q^2}{N} \leq c \leq \frac{1}{3(1+d)}$ then $\mathbf{Adv} \geq \Omega(\frac{q^2}{N}) - \frac{q}{N}$ ■

Example 5 *PMAC forgery for 128 bit.*

Writing $\alpha = \frac{q^2}{N}$, we get $\mathbf{Adv} \approx \frac{\alpha}{2}(1 - \frac{3\alpha}{2})(1 - \frac{\alpha}{4})$ This expression attains maximum value 0.083 at $\alpha = 0.3183$. Hence if we make $\sqrt(0.3183) \cdot 2^{64}$ queries we get at least 0.083 advantage. ■

5 Conclusion

In this paper we study collision probability for CBC-MAC and PMAC for suitably chosen messages. The calculation is made purely combinatorially and the idea of counting can be adopted to other similar calculation. We have found that CBC-MAC based on random function has collision probability $\Omega(q^2\ell/N)$ for ℓ-block messages. We can have similar collision probability for XCBC , TMAC, OMAC etc. (in fact, all constrcution similar to CBC). For PMAC, it is not easy to have similar bound, but we can do same thing for $\ell = 2$ and we have better collision probability than that of random oracle.

All these calculation are for MAC based on random function. Thus, it would be interesting to see a similar probability analysis for MAC based on random permutation.

References

[1] M. Bellare, K. Pietrzak and P. Rogaway. Improved Security Analysis for CBC MACs. Advances in Cryptology - CRYPTO 2005. Lecture Notes in Computer Science, Volume **3621**, pp 527-545.

[2] M. Bellare, A. Boldyreva, L. Knudsen and C. Namprempre. On-Line Ciphers and the Hash-CBC constructions. Advances in Cryptology - CRYPTO 2000. Lecture Notes in Computer Science, Volume **2139**, pp 292-309.

[3] M. Bellare, J. Killan and P. Rogaway. The security of the cipher block chanining Message Authentication Code. Advances in Cryptology - CRYPTO 1994. Lecture Notes in Computer Science, Volume **839**, pp 341-358.

[4] Daniel J. Bernstein. A short proof of the unpredictability of cipher block chaining (2005). URL: http://cr.yp.to/papers.html#easycbc. ID 24120a1f8b92722b5e1 5fbb6a86521a0.

[5] J. Black and P. Rogaway. CBC MACs for arbitrary length messages. Advances in Cryptology - CRYPTO 2000. Lecture Notes in Computer Science, Volume **1880**, pp 197-215.

[6] J. Black and P. Rogaway. A Block-Cipher Mode of Operations for Parallelizable Message Authentication. Advances in Cryptology - Eurocrypt 2002. Lecture Notes in Computer Science, Volume **2332**, pp 384-397.

[7] T. Iwata and K. Kurosawa. OMAC : One-Key CBC MAC. Fast Software Encryption, 10th International Workshop, FSE 2003. Lecture Notes in Computer Science, Volume **2887**, pp 129-153.

[8] C. S. Jutla. PRF Domain Extension using DAG. Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006. Lecture Notes in Computer Science, Volume **3876** pp 561-580.

[9] K. Kurosawa and T. Iwata. TMAC : Two-Key CBC MAC. Topics in Cryptology - CT-RSA 2003: The Cryptographers' Track at the RSA Conference 2003. Lecture Notes in Computer Science, Volume **2612**, pp 33-49.