3kf9: Enhancing 3GPP-MAC beyond the Birthday Bound

Liting Zhang[2], Wenling Wu[1], Han Sui[2], and Peng Wang[2]

[1] Institute of Software, Chinese Academy of Sciences
State Key Laboratory of Information Security
[2] Institute of Information Engineering, Chinese Academy of Sciences
{zhangliting,wwl,suihan}@is.iscas.ac.cn, wp@is.ac.cn

Abstract. Among various cryptographic schemes, CBC-based MACs belong to the few ones most widely used in practice. Such MACs iterate a blockcipher E_K in the so called Cipher-Block-Chaining way, i.e. $C_i = E_K(M_i \oplus C_{i-1})$, offering high efficiency in practical applications. In the paper, we propose a new deterministic variant of CBC-based MACs that is provably secure beyond the birthday bound. The new MAC 3kf9 is obtained by combining $f9$ (3GPP-MAC) and EMAC sharing the same internal structure, and so it is almost as efficient as the original CBC MAC. 3kf9 offers $O(\frac{l^3q^3}{2^{2n}} + \frac{lq}{2^n})$ PRF-security when its underlying n-bit blockcipher is pseudorandom with three independent keys. This makes it more secure than traditional CBC-based MACs, especially when they are applied with lightweight blockciphers. Therefore, 3kf9 is expected to be a possible candidate MAC in resource-restricted environments.

Keywords: MAC, Birthday Bound, CBC, Mode of Operation.

1 Introduction

1.1 Background

Birthday Bound. In cryptography, birthday attack is a generic attack that exploits no specific properties within cryptographic schemes, but just takes the advantage of birthday paradox in probability theory. This paradox says, approximately $2^{n/2}$ independently random n-bit points will collide with a probability close-to-1, where $2^{n/2}$ is called the birthday bound [28,20]. The birthday attack itself is not fatal to the practical security of cryptographic schemes, because people can choose long-enough security parameters to defend, e.g. by restricting the output length of hash functions to be no shorter than 224 bits [3], or by preventing attackers from getting sufficient number of input-output pairs, to make this attack infeasible in recent years.

However, being constrained by some particular software/hardware environments, there still exist many actual applications using short security parameters. For example, the 64-bit blockcipher KASUMI is currently a standard algorithm in mobile communication systems [7]. With the rapid developments of Internet

X. Wang and K. Sako (Eds.): ASIACRYPT 2012, LNCS 7658, pp. 296–312, 2012.

of Things, several lightweight primitives have been proposed in recent years, e.g. PRESENT and PHOTON [11,14]. These algorithms take small-size internal states and output values, usually are much easier to be realized in software and require smaller area in hardware, offering better performance than normal-size ones. Unfortunately, their small sizes imply vulnerability when they are used with traditional modes of operation, most of which are only secure within the birthday bound [19,2]. To ensure practical security in such cases, those modes have to be combined with stateful or random values, or to limit the lengths of their input messages, or to update secret keys frequently, resulting in inconveniences and security risks if misused.

MAC. Message Authentication Code is a widely-used cryptographic scheme for data integrity protection and data origin authentication. Practical applications usually require them to be not only secure (outputting unpredictable tags for new messages) but also efficient. A common way to design a MAC algorithm is to iterate a blockcipher $E : \mathcal{K}_E \times \{0,1\}^n \rightarrow \{0,1\}^n$ in the Cipher-Block-Chaining (CBC) manner. That is, in each step, a new chaining value C_i is obtained by encrypting the XOR result of the current message block M_i and the previous chaining value C_{i-1}, i.e. $C_i = E_K(M_i \oplus C_{i-1})$. The CBC structure is so common in the design of many cryptographic schemes that it has been considerably studied for many years [8,27,9,16,24].

Up to now, many excellent CBC-based MACs have been proposed, e.g. EMAC, XCBC, OMAC, CMAC and GCBC [27,9,16,4,24]. Besides, PMAC takes a fully parallelizable construction and can offer extremely high speed in parallel environments [10]. All of the above MAC algorithms are deterministic (needing no stateful or random values), and provably secure when their underlying blockcipher is assumed to be a pseudorandom permutation (PRP). However, their security bounds all fall within the birthday bound, and can not be further improved because there exist birthday attacks on them, i.e. the birthday bound is tight for them [19,2].

There are also a few CBC-based MACs with provable security beyond the birthday bound. For example, RMAC replaces the second key in EMAC by XORing its first key and a random value [18,2], and MAC-R1 and MAC-R2 inject n-bit randomness into the internal states of CBC-based MACs [23]. Obviously, their high security relies on not only the PRP security of blockciphers but also the randomness of the injected values.

In fact, all the deterministic blockcipher-based MACs fall within the birthday bound until Yasuda shows algorithm 6 in the ISO standard is an exception, conditioned on some restrictions on messages [1,30]. In the same paper, Yasuda also introduces SUM-ECBC to reduce the key size in algorithm 6, by XORing the results from two CBC-based MACs, providing half of the efficiency that normal CBC-based MACs offer in serial implementations (rate 2 [1]). On the other hand, Dodis and Steinberger build a MAC from unpredictable blockciphers, with

[1] For each message of l blocks long, it has to call the underlying blockcipher roughly $2l$ times.

security beyond the birthday bound, but pay by very high efficiency cost [12]. Very recently, Yasuda proposes PMAC_Plus that improves PMAC beyond the birthday bound [31]. By pre-calculating sufficiently large number (as many as the number of message blocks) of masks, this MAC would provide high efficiency due to the fully parallelizable structure in PMAC and rate-1 design.

3GPP-MAC. To promote the global system for mobile communications, the 3rd Generation Partnership Project (3GPP) proposes f9 as its first MAC algorithm, which is based on blockcipher KASUMI and produces 32-bit tags [6]. $f9$ inherits the structure of original CBC MAC, but in the end encrypts the sum of all chaining values, other than the last chaining value, to obtain the tag. The analysis for $f9$ tends to be tough due to this particular feature [17]. Knudsen and Mitchell are the first to give birthday attacks on $f9$, which need $2^{(n+1)/2}$ known (Message, MAC) pairs and $2^{n/2+1}$ chosen (Message, MAC) pairs to make a forgery against $f9$ without truncations [20]. Then, Iwata and Kohno proved that when KASUMI is secure against a special kind of related-key attacks (RK-PRP), a generalized version of $f9$ (named with $f9'$) is PRF-secure within the birthday bound [15]. This implies the previous birthday attack is the best one without knowledge of internal information.

Despite the fact that the birthday attacks on MACs need on-line invocations, making it much more harder than those on hash functions (needing only off-line computations), people still take several countermeasures for large enough security margin. For example, in the practical applications of $f9$, it has been demanded that each message should be prepended with a fresh value, the length of messages should be no longer than 20000 bits, the secret key should be changed after each invocation, and the outputs should be truncated [5,6].

1.2 Our Work

In this paper, we attempt to design a rate-1 CBC-based MAC with provable security beyond the birthday bound. A direct application of such a scheme is to enforce the security level of current CBC-based MACs, especially in the situations where small-size (lightweight) blockciphers are used, e.g. 3GPP and smart cards. Another application is to make it serve as a highly-secure pseudorandom number generator for various protocols, which therefore would improve the security level of the latter.

To do this, stateful or random values (e.g. counter, fresh) can help, but we would not consider them for practical convenience. Another possible way is to enlarge the size of internal states but still output normal-size tags. As for CBC-based MACs, however, their internal states have the same size as their underlying blockcipher, so one may want to use a large-size blockcipher in CBC-based MACs and truncate their outputs. Unfortunately, the efficiency of such a solution will not be satisfying, because a large-size blockcipher usually runs no faster than a small-size one, not to mention many other costs, e.g. memory and area requirements.

Our starting point is $f9$, in favor of its double-blocksize internal states providing a possible chance to resist the birthday attacks. Inspired by the design of SUM-ECBC and `PMAC_Plus`, we append one more blockcipher invocation to the end of the $f9$ structure, as illustrated in Fig. 1. The resulting MAC is named with 3kf9, for it enhances $f9$ and needs three independent keys. From another point of view, it is also an extension of EMAC [27], ignoring E_{K_3} and the last XOR operation.

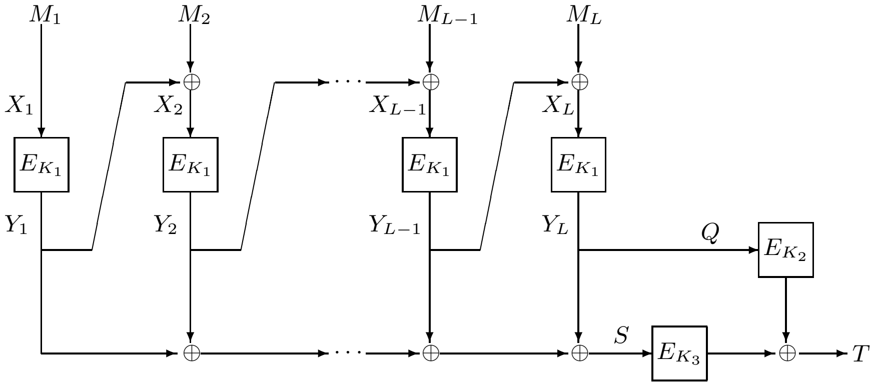

Fig. 1. Illustration of 3kf9

When authenticating messages, 3kf9 can start to work without stateful values or message length information (on-line), requires no pre-computation and only two block-size memory for internal states, besides those for its underlying blockcipher. Specially, it needs no multiplications, comparing with `PMAC_Plus`. Therefore, 3kf9 will provide high efficiency in serial implementations.

A more detailed comparison with related MACs is given in Table 1.

Table 1. Comparison among 3kf9 and its related deterministic MACs

	key size	rate	structure	multi.	upper bounds	bBB. [a]	Ref.
Alg. 6 in ISO std.[b] SUM-ECBC	$6k$ $4k$	2	CBC	none	$O(\frac{l^4q^3}{2^{2n}})$ or restricted $O(\frac{l^3q^3}{2^{2n}})$	conditional	[1] [30]
`PMAC_Plus` 3kf9	$3k$	1	parallel CBC	$4l-1$ none	$O(\frac{l^3q^3}{2^{2n}}+\frac{lq}{2^n})$	yes	[31] This Work
$f9$ EMAC	k [c] $2k$	1	CBC	none	$O(\frac{l^2q^2}{2^n})$	no	[15] [27]

[a] bBB stands for "beyond the Birthday Bound".

[b] It has been removed from the latest version ISO/IEC 9797-1:2011.

[c] Its second key is obtained by $K_2 = K_1 \oplus \text{KM}$, where KM is a non-zero k-bit value.

1.3 Organization

The rest of this paper is organized as follows. Section 2 introduces necessary symbols and 3kf9 specification. Section 3 gives our provable security analysis for 3kf9, including security definitions, the main result and its proof. The proof will be completed in Section 4. In Section 5, we give some suggestions for practical usages of 3kf9. Finally, we conclude this work in Section 6.

2 Symbols and Specification

$\{0,1\}^n$ is the set of all n-bit strings and $\{0,1\}^*$ is the set of all strings. For strings $a, b \in \{0,1\}^*$, $a||b$ is a concatenation of a and b, and $|a|$ is its length in bits. If a, b have equal lengths then $a \oplus b$ is their bitwise XOR. Denote $\mathrm{Perm}(n)$ and $\mathrm{Rand}(n, n)$ as the sets of all permutations and functions over $\{0,1\}^n$ respectively. $\mathrm{Rand}(*, n)$ stands for the set of all functions whose range belongs to $\{0,1\}^n$. If A is a set, then $\#A$ denotes the size of set A, and $x \overset{\$}{\leftarrow} A$ means that x is chosen from set A uniformly at random.

A message M can be alternatively seen as a bit string $M \in \{0,1\}^*$. Then, by $M \leftarrow M||10^{n-1-|M| \bmod n}$ we mean we append a single bit "1" to the end of M, followed by as many as $n - 1 - |M| \bmod n$ bit "0"s such that the length of the padded string is a multiple of n. For any such string M ($|M| = nL$), $M_1 M_2 \cdots M_L \leftarrow$ **Partition**(M) means we break M into L successive n-bit blocks such that $M_1||M_2||\cdots||M_L = M$.

```
MAC Algorithm 3kf9[$E$]
Input: $K_1, K_2, K_3 \overset{\$}{\leftarrow} \mathcal{K}$, $M \in \{0,1\}^*$
Output: $T \in \{0,1\}^n$
$M \leftarrow M||10^{n-1-|M| \bmod n}$
$M_1 M_2 \cdots M_L \leftarrow$ Partition$(M)$
$S \leftarrow 0^n$
$Y_0 \leftarrow 0^n$
for $l \leftarrow 1$ to $L$ do
    $X_l \leftarrow Y_{l-1} \oplus M_l$
    $Y_l \leftarrow E_{K_1}(X_l)$
    $S \leftarrow S \oplus Y_l$
end for
$T \leftarrow E_{K_2}(Y_L) \oplus E_{K_3}(S)$
return $T$
```

Fig. 2. Specification of 3kf9

For any message $M \in \{0,1\}^*$, 3kf9 takes a blockcipher $E : \mathcal{K}_E \times \{0,1\}^n \rightarrow \{0,1\}^n$ as its underlying primitive, calling it iteratively as specified in Fig. 2 to deal with M, and finally outputs $T \in \{0,1\}^n$ as a tag. If necessary, T can be truncated to be of some particular length less than n.

3kf9 needs three keys K_1, K_2 and K_3, each of which should be independently selected from $\mathcal{K} = \mathcal{K}_E$ uniformly at random. We use $3\mathrm{kf9}[E_{K_1}, E_{K_2}, E_{K_3}]$ to stand for this MAC algorithm and we also write it as $3\mathrm{kf9}[E]$ for short.

3 Security Proof

3.1 Security Definitions

W need to introduce PRP/PRF definitions here, which are frequently used in the analysis of modes of operation for blockciphers [8,27,9,16,24].

These two definitions focus on the randomness of a keyed function f_K, which is selected from a function family $f : \mathcal{K}_f \times \{0,1\}^* \to \{0,1\}^n$ by selecting a random key K. To measure its randomness, f_K is compared with a random function $R \overset{\$}{\leftarrow} \mathrm{Rand}(*, n)$ (or a random permutation $P \overset{\$}{\leftarrow} \mathrm{Perm}(n)$ if f consists of only permutations).

The comparison is done as, informally, allowing adversaries (without knowing K) to query an oracle, which is either f_K or R with equal probability. The oracle will answer with the corresponding outputs. After some number of queries, the adversaries are asked to tell what the oracle is. The precise definition is given by

$$\begin{cases} \mathbf{Adv}_f^{\mathrm{prf}}(\mathcal{A}) \overset{\mathrm{def}}{=} |\Pr[K \overset{\$}{\leftarrow} \mathcal{K}_f : \mathcal{A}^{f_K(\cdot)} = 1] - \Pr[R \overset{\$}{\leftarrow} \mathrm{Rand}(*, n) : \mathcal{A}^{R(\cdot)} = 1]|, \\ \mathbf{Adv}_f^{\mathrm{prf}}(t, q, \mu) \overset{\mathrm{def}}{=} \max_{\mathcal{A}} \{\mathbf{Adv}_f^{\mathrm{prf}}(\mathcal{A})\}, \end{cases}$$
$$\begin{cases} \mathbf{Adv}_f^{\mathrm{prp}}(\mathcal{A}) \overset{\mathrm{def}}{=} |\Pr[K \overset{\$}{\leftarrow} \mathcal{K}_f : \mathcal{A}^{f_K(\cdot)} = 1] - \Pr[P \overset{\$}{\leftarrow} \mathrm{Perm}(n) : \mathcal{A}^{P(\cdot)} = 1]|, \\ \mathbf{Adv}_f^{\mathrm{prp}}(t, q, \mu) \overset{\mathrm{def}}{=} \max_{\mathcal{A}} \{\mathbf{Adv}_f^{\mathrm{prp}}(\mathcal{A})\}, \end{cases}$$

and the maximum is over all adversaries taking time at most t, making oracle queries at most q, whose total length is at most μ bits. If $\mathbf{Adv}_f^{\mathrm{prf}}(t, q, \mu)$ (or $\mathbf{Adv}_f^{\mathrm{prp}}(t, q, \mu)$) is sufficiently small, we say function family f is a pseudorandom function (PRF) (or a pseudorandom permutation (PRP)).

It has been proved that a PRF is a secure MAC [8].

3.2 Main Results

Let $3\mathrm{kf9}[P_1, P_2, P_3]$ stand for $3\mathrm{kf9}[E_{K_1}, E_{K_2}, E_{K_3}]$ when blockcipher E with three independent keys are replaced by three independently random permutations P_1, P_2 and P_3, and we further write it as $3\mathrm{kf9}[P]$ for simplicity. Then, the following theorem says that $3\mathrm{kf9}[P]$ is a PRF with an upper bound beyond the birthday bound.

Theorem 1 (Main Theorem). *For any computationally unbounded adversary $\mathcal{A}$, after querying the oracle q times, with each query no longer than $l_{\max}$ blocks, its advantage to distinguish* 3kf9[P] *from a random function $R \xleftarrow{\$} \mathrm{Rand}(*, n)$ is upper bounded by*

$$|\Pr[\mathcal{A}^{3\mathrm{kf9}[P]} = 1] - \Pr[\mathcal{A}^{R} = 1]| \leq \frac{ql_{\max}+q}{2^{n-2}} + \frac{2q^3l_{\max}^3+q^3l_{\max}^2+2q^3l_{\max}+2q^3}{2^{2n-1}}.$$

We conclude this theorem by the "coefficient H technique" initially proposed by Patarin [25,26]. This method is a useful tool for proving pseudorandom properties of blockcipher structures and modes of operation, and it has been frequently used before [25,13,16,24].

To simplify our proof, we also adopt the framework used in the proofs for SUM-ECBC and `PMAC_Plus` [30,31], which separates the inputs to P_2 and P_3 into four cases. Taking advantage of some known results for CBC structure, $f9$ and sum of PRPs [9,15,22], the first three cases can be easily upper bounded. For the last case, we prove it by Lemma 1 in the next section.

Proof. Since $\mathcal{A}$ is computationally unbounded, w.l.o.g. we assume $\mathcal{A}$ is a deterministic algorithm, otherwise we can maximize $\mathcal{A}$ by running it over all possible cases and choose the most powerful one. Based on this, the i-th query $M^i \notin \{M^1, M^2, \cdots, M^{i-1}\}$ $\mathcal{A}$ would make is fully determined by the previous $i-1$ input-output pairs (M^1, T^1), (M^2, T^2), $\cdots$, (M^{i-1}, T^{i-1}). Then, if we fix a q-tuple $\overrightarrow{T} = (T^1, T^2, \cdots, T^q)$, we know

- all $\mathcal{A}$'s queries are uniquely determined,
- the number of queries q is uniquely determined, and
- the output of $\mathcal{A}$ (0 or 1) is uniquely determined.

Denote $\mathrm{Tset}_1 = \{(T^1, T^2, \cdots, T^q)\}$ is the set that contains all q-tuple $\overrightarrow{T} = (T^1, T^2, \cdots, T^q)$ such that $\mathcal{A}$ outputs 1, and $N = \#\mathrm{Tset}_1$. Then we have

Evaluation for random function R.

$$\Pr[\mathcal{A}^{R} = 1] = \textstyle\sum_{\overrightarrow{T} \in \mathrm{Tset}_1} \Pr[R(M^i) = T^i, i = 1, 2, \cdots, q] = \frac{N}{2^{qn}}.$$

Evaluation for 3kf9[P].

$$\begin{aligned}
&\Pr[\mathcal{A}^{3\mathrm{kf9}[P]} = 1] \\
&= \sum_{\overrightarrow{T} \in \mathrm{Tset}_1} \Pr[3\mathrm{kf9}[P](M^i) = T^i, i = 1, 2, \cdots, q] \\
&\geq \sum_{\overrightarrow{T} \in \mathrm{Tset}_1} (\Pr[3\mathrm{kf9}[P] \text{ outputs } q \text{ random values}] \times (\frac{1}{2^n})^q) \\
&= \frac{N}{2^{qn}} \times \Pr[3\mathrm{kf9}[P] \text{ outputs } q \text{ random values}]. \qquad (1)
\end{aligned}$$

Denote $\underline{\mathrm{CBC}}[P_1]$ as the internal structure of 3kf9$[P]$, i.e. $(Q, S) \leftarrow \underline{\mathrm{CBC}}[P_1](M)$, and 3kf9$[P](M) = P_2(Q) \oplus P_3(S) = T$, as in Fig. 1. In the following analysis, we do step by step for each $i = 1, 2, \cdots, q$. Suppose in the previous $i-1$ queries, the $i-1$ outputs $T^1, T^2, \cdots, T^{i-1}$ are independently random values. Let $\mathrm{Domain}[P_2] = \{Q^1, Q^2, \cdots, Q^{i-1}\}$ and $\mathrm{Domain}[P_3] = \{S^1, S^2, \cdots, S^{i-1}\}$. Then, for the i-th query M^i, its corresponding $(Q^i, S^i) \leftarrow \underline{\mathrm{CBC}}[P_1](M^i)$ will definitely fall into one of the following four cases,

Case A: $Q^i \in \mathrm{Domain}[P_2]$ and $S^i \notin \mathrm{Domain}[P_3]$,
Case B: $Q^i \notin \mathrm{Domain}[P_2]$ and $S^i \in \mathrm{Domain}[P_3]$,
Case C: $Q^i \notin \mathrm{Domain}[P_2]$ and $S^i \notin \mathrm{Domain}[P_3]$,
Case D: $Q^i \in \mathrm{Domain}[P_2]$ and $S^i \in \mathrm{Domain}[P_3]$.

For Case A, Black and Rogaway have shown that the probability for any two messages to collide in CBC structure (with an independent ending blockcipher invocation, e.g. EMAC, ECBC) is upper bounded by the birthday bound, i.e. $\Pr[Q^j = Q^i] \leq \frac{4(l_{\max}+1)^2}{2^n}$ (See Lemma 3 in [9]). In such a case, we still have randomness for $T^i = P_2(Q^i) \oplus P_3(S^i)$ because $S^i \notin \mathrm{Domain}[P_3]$ and we can do lazy sampling $P_3(S^i)$. Since at this moment $\#\mathrm{Domain}[P_3] \leq i-1$, the advantage to distinguish $P_3(S^i)$ from a random value $r \xleftarrow{\$} \{0,1\}^n$ is no more than $\frac{i-1}{2^n}$. Then, the advantage to distinguish T^i from r is upper bounded by $\binom{i-1}{1}\frac{4(l_{\max}+1)^2}{2^n} \times \frac{i-1}{2^n}$.

For Case B, Iwata and Kohno have pointed out that the probability for any two messages to collide in $f9$ (with an independent ending block cipher invocation) is also upper bounded by the birthday bound, i.e. $\Pr[S^j = S^i] \leq \frac{(2l_{\max}+2)^2+2^2}{2^{n+1}} = \frac{2l_{\max}^2+4l_{\max}+4}{2^n}$ (See Lemma B.1 in [15], and note that we apply $\sigma \leq 2l_{\max} + 2$ and $q = 2$ here). Then, by lazy sampling for $P_2(Q^i)$, we know the advantage to distinguish T^i from r is upper bounded by $\binom{i-1}{1}\frac{2l_{\max}^2+4l_{\max}+4}{2^n} \times \frac{i-1}{2^n}$.

For Case C, Lucks has proved that the advantage to distinguish $T^i = P_2(Q^i) \oplus P_3(S^i)$ from r is upper bounded by $\frac{(i-1)^2}{(2^n-(i-1))^2} \leq \frac{4(i-1)^2}{2^{2n}}$ (See the proof for Theorem 5 in [22]).

As for Case D, we will show by Lemma 1 in the next section that $\Pr[\exists i \in [1, q]:$ Case D occurs$] \leq \frac{ql_{\max}+q}{2^{n-2}} + \frac{q^3 l_{\max}^3}{2^{2n-2}}$.

Denote $[T^i \nsim r]$ as the event that T^i is not an independently random value. Then, based on the none occurrence of Case D, we get

$$\begin{aligned}
&\Pr[T^i \nsim r] \\
&= \Pr[\text{Case A}]\Pr[T^i \nsim r|\text{Case A}] + \Pr[\text{Case B}]\Pr[T^i \nsim r|\text{Case B}] + \\
&\quad \Pr[\text{Case C}]\Pr[T^i \nsim r|\text{Case C}] \\
&\leq \binom{i-1}{1}\frac{4(l_{\max}+1)^2}{2^n} \times \frac{i-1}{2^n} + \binom{i-1}{1}\frac{2l_{\max}^2+4l_{\max}+4}{2^n} \times \frac{i-1}{2^n} + 1 \times \frac{4(i-1)^2}{2^{2n}} \\
&= \frac{(i-1)^2(3l_{\max}^2+5l_{\max}+6)}{2^{2n-1}}.
\end{aligned}$$

This allows us to have

$$\begin{aligned}
&\Pr[\text{3kf9}[P] \text{ doesn't output } q \text{ random values}] \\
&\le \Pr[\text{Case D}] + \sum_{i=1}^{q} \Pr[T^i \not\sim r] \\
&\le \frac{q l_{\max} + q}{2^{n-2}} + \frac{q^3 l_{\max}^3}{2^{2n-2}} + \sum_{i=1}^{q} \frac{(i-1)^2 (3 l_{\max}^2 + 5 l_{\max} + 6)}{2^{2n-1}} \\
&\le \frac{q l_{\max} + q}{2^{n-2}} + \frac{2q^3 l_{\max}^3 + q^3 l_{\max}^2 + 2q^3 l_{\max} + 2q^3}{2^{2n-1}} \\
&= \epsilon,
\end{aligned}$$

which implies $\Pr[\mathcal{A}^{\text{3kf9}[P]} = 1] \ge \frac{N}{2^{qn}} \times (1 - \epsilon)$ by applying it to inequality (1).

Comparison

By the above analysis, we can get

$$\Pr[\mathcal{A}^R = 1] - \Pr[\mathcal{A}^{\text{3kf9}[P]} = 1] \le \frac{N}{2^{qn}} - \frac{N}{2^{qn}} \times (1 - \epsilon) \le \frac{N}{2^{qn}} \times \epsilon \le \epsilon.$$

On the other side, if we define Tset_0 and by similar analysis we can get

$$\Pr[\mathcal{A}^R = 0] - \Pr[\mathcal{A}^{\text{3kf9}[P]} = 0] \le \epsilon,$$

which implies $(1 - \Pr[\mathcal{A}^R = 1]) - (1 - \Pr[\mathcal{A}^{\text{3kf9}[P]} = 1]) \le \epsilon$. Thus we get $\Pr[\mathcal{A}^{\text{3kf9}[P]} = 1] - \Pr[\mathcal{A}^R = 1] \le \epsilon$.

Finally, we conclude

$$|\Pr[\mathcal{A}^{\text{3kf9}[P]} = 1] - \Pr[\mathcal{A}^R = 1]| \le \frac{q l_{\max} + q}{2^{n-2}} + \frac{2q^3 l_{\max}^3 + q^3 l_{\max}^2 + 2q^3 l_{\max} + 2q^3}{2^{2n-1}}.$$

□

Based on the main theorem, we can say that 3kf9[E] is a PRF if blockcipher E is secure. More precisely, we have

Theorem 2. *If blockcipher $E : \mathcal{K}_E \times \{0,1\}^n \to \{0,1\}^n$ is a PRP, then* 3kf9[E] *is a PRF for all adversaries, who make at most q queries, each of which is no longer than $l_{\max}$ blocks. That is,*

$$\text{Adv}^{\text{prf}}_{\text{3kf9}[E]}(t, q, \mu) \le \tfrac{q l_{\max} + q}{2^{n-2}} + \tfrac{2q^3 l_{\max}^3 + q^3 l_{\max}^2 + 2q^3 l_{\max} + 2q^3}{2^{2n-1}} + 3\text{Adv}^{\text{prp}}_{E}(t', q', \mu'),$$

where $t' = t + O(t)$, $q' \le q(l_{\max} + 1)$, and $\mu' \le \mu + qn$.

4 Key Lemma

The none occurrence of Case D implies the q pairs (Q^i, S^i) $(i = 1, 2, \cdots, q)$ are *free*. By "free", we mean for each $i \in [1, q]$, either Q^i is unique in its corresponding sequence $Q^1, Q^2, \cdots, Q^q$ or S^i is unique in its corresponding sequence

$S^1, S^2, \cdots, S^q$. This property is closely related to the newly appeared Cover Free notion [12], which says the q outputs $(N_1^i, N_2^i, \cdots, N_w^i)$ $(1 \leq i \leq q)$ from a cover-free function should satisfy the following property. For each i, there exists at least one $j \in [1, w]$ such that N_j^i is unique in its own subsequence $N_j^1, N_j^2, \cdots, N_j^q$. Unfortunately, the internal structure $\underline{\mathrm{CBC}}[P_1]$ can not satisfy the cover free property, when its outputs are made public. However, if adversaries can not get its internal states, $\underline{\mathrm{CBC}}[P_1]$ holds a similar property, as the following lemma says.

Lemma 1. *If P_1,P_2 and P_3 are independently random permutations from* $\mathrm{Perm}(n)$*, then for all computationally unbounded adversaries, who querying* $\mathrm{3kf9}[P]$ *no more than q times, with each query no longer than $l_{\max}$ blocks, the probability for internal states (Q^i, S^i) $(i = 1, 2, \cdots, q)$ to satisfy Case D is upper bounded by*

$$\Pr[\exists i \in [1, q] : \text{ Case D occurs}] \leq \tfrac{ql_{\max}+q}{2^{n-2}} + \tfrac{q^3 l_{\max}^3}{2^{2n-2}}.$$

In the following proof, we will prove an even stronger result. That is, all the pairs (Y_l^i, S_l^i) for $l = 1, 2, \cdots, L^i$ and $i = 1, 2, \cdots, q$ are free with this probability, excluding the trivial case that $(Y_l^i, S_l^i) = (Y_l^j, S_l^j)$ with $l \leq d$ for two different messages M^i and M^j, which after being padded are written as $M_1^i||M_2^i|| \cdots ||M_{L^i}^i$ and $M_1^j||M_2^j|| \cdots ||M_{L^j}^j$ and having common prefix $M_1^i||M_2^i|| \cdots ||M_d^i = M_1^j||M_2^j|| \cdots || M_d^j$ for some $d \leq \min\{L^i, L^j\}$. To do this, we check the process detail of $\underline{\mathrm{CBC}}[P_1]$ in dealing with the querying messages M^1, M^2, $\cdots$, M^q step by step, and record every Y_l^i and S_l^i for $l = 1, 2, \cdots, L^i$ and $i = 1, 2, \cdots, q$ with two sets YRange and SRange. By lazy sampling for P_1, we upper bound the probability for the events $Y_l^i \in$ YRange and $S_l^i \in$ SRange to occur at the same time, and in the end we sum up all these probabilities to get the final result.

Proof. For any q pairwise distinct queries $M^1, M^2, \cdots, M^q$, we use a program to show the process of $\underline{\mathrm{CBC}}[P_1]$ in dealing with them, as in Fig. 3. To better analyze the target probability, we do lazy sampling for P_1. Furthermore, we denote three flags `Zero`, `Cover` and `Bad`. `Zero` is used to identify whether there exists $Y_l^i = 0^n$, which may be easily used to undermine the freeness consistence of (Y_l^i, S_l^i) for $l = 1, 2, \cdots, L^i$ and $i = 1, 2, \cdots, q$. `Cover` is used directly to identify the freeness of (Y_l^i, S_l^i). Either [`Zero` = `True`] or [`Cover` = `True`] implies [`Bad` = `True`], so $\Pr[\exists i \in [1, q] :$ Case D occurs$] = \Pr[\texttt{Bad} = \texttt{True}] \leq \Pr[\texttt{Zero} = \texttt{True}] + \Pr[\texttt{Cover} = \texttt{True}]$.

Then, it is easy to get that $\Pr[\texttt{Zero} = \texttt{True}] \leq \sum_{j=1}^{q(l_{\max}+1)} \frac{1}{2^n-(j-1)} \leq \frac{q(l_{\max}+1)}{2^{n-1}}$, because for the q messages whose length is no more than $l_{\max}+1$ blocks after being padded, we do no more than $q(l_{\max}+1)$ lazy sampling for P_1, and in the j-th sampling for a new output Y, $\Pr[Y = 0^n] \leq \frac{1}{2^n-(j-1)}$. Here we use $q(l_{\max}+1) < 2^{n-1}$ to get the final bound.

To upper bound $\Pr[\texttt{Cover} = \texttt{True}]$ for all (Y_l^i, S_l^i), we will upper bound the probability for each lazy sampling that may result in the occurrence of $[Y_l^i \in \text{YRange} \wedge S_l^i \in \text{SRange}]$ with $l = 1, 2, \cdots, L^i$ and $i = 1, 2, \cdots, q$, and then sum up them. For better understanding the following analysis, we work on a simple case first (see Fig. 4 for an illustration), and then generalize it step by step.

```
Domain[$P_1$], Range[$P_1$], YRange, SRange $\leftarrow \phi$; Zero, Cover, Bad $\leftarrow$ False;
for $\mathcal{A}$'s $i$-th query $M^i \in \{0,1\}^*$, do
Ê   $M^i \leftarrow M^i||10^{n-1-|M^i| \bmod n}$; $M_1^i M_2^i \cdots M_{L^i}^i \leftarrow$ Partition($M^i$);
    $S_0^i \leftarrow 0^n$; $Y_0^i \leftarrow 0^n$;
    for $l \leftarrow 1$ to $L^i$ do
        $X_l^i \leftarrow Y_{l-1}^i \oplus M_l^i$;
        if $X_l^i \in$ Domain[$P_1$] then $Y_l^i \leftarrow P_1(X_l^i)$;
        else $Y_l^i \xleftarrow{\$} \{0,1\}^n \setminus$ Range[$P_1$];
            if $Y_l^i = 0^n$ then Zero $\leftarrow$ True; Bad $\leftarrow$ True; end if
            Range[$P_1$] $\leftarrow$ Range[$P_1$] $\cup \{Y_l^i\}$;
            Domain[$P_1$] $\leftarrow$ Domain[$P_1$] $\cup \{X_l^i\}$;
        end if
        $S_l^i \leftarrow S_{l-1}^i \oplus Y_l^i$;
        if $Y_l^i \in$ YRange and $S_l^i \in$ SRange and
          $\nexists j < i$ s.t. $M_1^i||M_2^i||\cdots||M_l^i = M_1^j||M_2^j||\cdots||M_l^j$
        then Cover $\leftarrow$ True; Bad $\leftarrow$ True;
        else YRange $\leftarrow$ YRange $\cup \{Y_l^i\}$; SRange $\leftarrow$ SRange $\cup \{S_l^i\}$;
        end if
    end for
```

Fig. 3. A program showing the process of CBC[P_1]

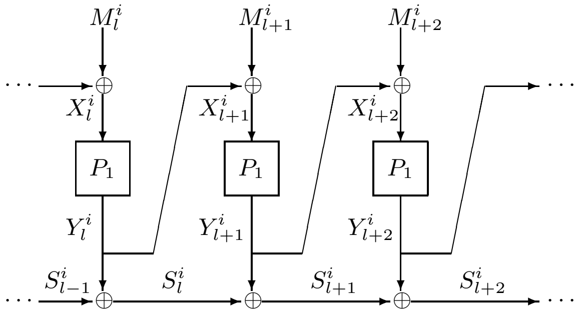

Fig. 4. An insight view on the internal structure of CBC[P_1]

4.1 The Most Common Case

For a new input $X_l^i \notin$ Domain[P_1], we will choose a value $Y_l^i \xleftarrow{\$} \{0,1\}^n \backslash$Range[$P_1$] by lazy sampling. Since Y_l^i is a new output, it is definite that (Y_l^i, S_l^i) is consistent with the previous pairs for freeness. However, if it happens that $X_{l+1}^i = Y_l^i \oplus M_{l+1}^i \in$ Domain[P_1], then event [$Y_{l+1}^i \in$ YRange] would occur, and the freeness consistence of pairs will rely only on the none occurrence of the event [$S_{l+1}^i \in$ SRange]. Consider the following two subcases:

1. $X_{l+1}^i = X_l^i$. This implies $Y_{l+1}^i = Y_l^i$ and $S_{l+1}^i = S_{l-1}^i$, and thus undermining the freeness consistence. The probability for this event to occur is no more

than $\Pr[X^i_{l+1} = X^i_l] = \Pr[Y^i_l = X^i_l \oplus M^i_{l+1}] \leq \frac{1}{2^n - \#\mathrm{Range}[P_1]} \leq \frac{1}{2^{n-1}}$, where we assume $\#\mathrm{Range}[P_1] < 2^{n-1}$.

2. $X^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\}$. This implies $Y^i_l \oplus M^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\}$, and so Y^i_l has no more than $\#\mathrm{Domain}[P_1] \setminus \{X^i_l\}$ choices. Choose any one such choice and fix Y^i_l, then $Y^i_{l+1} = P_1(X^i_{l+1}) = P_1(Y^i_l \oplus M^i_{l+1})$ would be fixed, so is $S^i_{l+1} = \sum_{c=1}^{l+1} Y^i_c$. On the other hand, the elements in SRange are $\sum_{c=1}^{d} Y^j_c$ $(1 \leq d \leq L^j,\ 1 \leq j \leq i-1)$ and $\sum_{c=1}^{d} Y^i_c$ $(1 \leq d \leq l)$. Then, event $[S^i_{l+1} \in \mathrm{SRange}]$ implies no more than #SRange equations, all of which can be written as linear combination of Y^a_b equals to linear combination of message blocks (i.e. $M^i_{l+1} \oplus M^a_{b+1}$ or 0^n) with $0 \leq b \leq L^a$, $1 \leq a \leq i-1$ or $0 \leq b \leq l-1$, $a = i$. Specially, note that Y^i_l is not included here because $X^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\}$ implies Y^i_l can be written as $M^i_{l+1} \oplus X^i_{l+1} = M^i_{l+1} \oplus \overline{Y} \oplus \overline{M}$, where $\overline{Y}$ and $\overline{M}$ appear in the previous (Y, S) pairs and queries respectively ($\overline{Y}$ may be 0^n if $b = 0$). Furthermore, notice that we have upper bounded $\Pr[Y = 0^n]$ by analyzing [`Zero` = `True`], so we can assume all Y^a_b $(b \geq 1)$ are non-zero values. Then, excluding the trivial case that two different messages would collide in their common prefix part, the possibility for each of these equations to hold is no more than $1/2^{n-1}$, because all Y^a_b $(b \geq 1)$ are chosen by the previous lazy samplings, from a space with roughly $2^n - \#\mathrm{Domain}[P_1] - \#\mathrm{Range}[P_1] - 1 \leq 2^{n-1}$ size. $2^n - \#\mathrm{Range}[P_1]$ is naturally understood, "1" is respect to 0^n, and "$\#\mathrm{Domain}[P_1]$" is respect to the number of bad points that may result in $Y^a_b \oplus M^a_{b+1} \in \mathrm{Domain}[P_1]$. So the linear combinations of Y^a_b has at least 2^{n-1} possible values, and their real values are hidden in the internal structure $\underline{\mathrm{CBC}}[P_1]$, not known by adversaries. So, in this subcase,

$$\begin{aligned}
&\Pr[Y^i_{l+1} \in \mathrm{Range}[P_1] \setminus \{Y^i_l\} \ \wedge\ S^i_{l+1} \in \mathrm{SRange}] \\
&= \Pr[X^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\} \ \wedge\ S^i_{l+1} \in \mathrm{SRange}] \\
&= \Pr[Y^i_l \oplus M^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\} \ \wedge\ S^i_{l+1} \in \mathrm{SRange}] \qquad (2)\\
&\leq \Pr[Y^i_l \oplus M^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\}] \times \Pr[S^i_{l+1} \in \mathrm{SRange}] \qquad (3)\\
&\leq \frac{\#\mathrm{Domain}[P_1] \setminus \{X^i_l\}}{2^n - \#\mathrm{Range}[P_1]} \times \frac{\#\mathrm{SRange}}{2^{n-1}} \\
&\leq \frac{(\#\mathrm{Domain}[P_1])^2}{2^{2n-2}},
\end{aligned}$$

Where we apply $\#\mathrm{Range}[P_1] < 2^{n-1}$. Notice that $P_1[X^i_l] = Y^i_l \xleftarrow{\$} \{0,1\}^n \setminus \mathrm{Range}[P_1]$ is a new lazy sampling, and $S^i_{l+1} \in \mathrm{SRange}$ is only related with previous lazy samplings ($X^i_{l+1} \in \mathrm{Domain}[P_1] \setminus \{X^i_l\}$ implying $Y^i_l = M^i_{l+1} \oplus \overline{Y} \oplus \overline{M}$ can be calculated by the previous pairs and queries), so the probability in (2) can be separated, thus we obtain inequality (3).

In this most common case, the probability for lazy sampling $P_1[X^i_l] = Y^i_l \xleftarrow{\$} \{0,1\}^n \setminus \mathrm{Range}[P_1]$ to undermine the freeness consistence is at most $\frac{1}{2^{n-1}} + \frac{(\#\mathrm{Domain}[P_1])^2}{2^{2n-2}}$.

4.2 Generalized Case 1

The above lazy sampling may further induce the occurrence of event $[X^i_{l+2} \in \text{Domain}[P_1]]$, so the previous analysis is not complete, and here we generalize it in this direction.

Suppose $P_1[X^i_l] = Y^i_l \overset{\$}{\leftarrow} \{0,1\}^n \setminus \text{Range}[P_1]$ induces series of occurrences, i.e. $[X^i_{l+1} \in \text{Domain}[P_1]]$, $[X^i_{l+2} \in \text{Domain}[P_1]]$, $\cdots$, $[X^i_{l+u-1} \in \text{Domain}[P_1]]$, with $u \leq L^i - l + 1$, let us consider the probability to undermine the freeness consistence. First, we have $\Pr[X^i_{l+1} = X^i_l] \leq \frac{1}{2^{n-1}}$ as before. Then, conditioned on $X^i_{l+1} \neq X^i_l$, those $u-1$ events imply $Y^i_l \oplus M^i_{l+1} \in \text{Domain}[P_1] \setminus \{X^i_{l+1}\}$ and $Y^i_{l+a} \oplus M^i_{l+a+1} \in \text{Domain}[P_1]$ for $1 \leq a \leq u-2$, and so Y^i_l has at most $\#\text{Domain}[P_1] \setminus \{X^i_{l+1}\}$ choices. Choose any one such choice and fix Y^i_l, then S^i_{l+a} $(0 \leq a \leq u-1)$ are also fixed. To keep freeness consistence, none of the events $[S_{l+1+a} \in \text{SRange} \cup \{S^i_l, S^i_{l+1}, \cdots, S^i_{l+a}\}]$ $(0 \leq a \leq u-2)$ should occur. These events imply no more than $(u-1)\#\text{SRange}+\frac{(u-1)(u-2)}{2}$ equations, and each has a probability of $1/2^{n-1}$ to occur, with similar reasons given in the most common case. So, here the probability for this lazy sampling to keep freeness consistence is upper bounded by $\frac{1}{2^{n-1}} + \frac{\#\text{Domain}[P_1]\setminus\{X^i_{l+1}\}}{2^n - \#\text{Range}[P_1]} \times \frac{(u-1)\#\text{SRange}+\frac{(u-1)(u-2)}{2}}{2^{n-1}} \leq \sum_{a=1}^{u}(\frac{1}{2^{n-1}} + \frac{(\#\text{Domain}[P_1]+a-1)^2}{2^{2n-2}})$. Notice that u is the number of invocations to P_1 related to lazy sampling $P_1[X^i_l] = Y^i_l \overset{\$}{\leftarrow} \{0,1\}^n \setminus \text{Range}[P_1]$.

4.3 Generalized Case 2

Since we assume adversaries can make any q pairwise distinct queries M^1, M^2, $\cdots$, M^q, it is possible that some queries share a common prefix. Here we generalize the probability for lazy sampling $P_1[X^i_l] = Y^i_l \overset{\$}{\leftarrow} \{0,1\}^n \setminus \text{Range}[P_1]$ to undermine the freeness consistence in this direction.

Without loss of generality, we assume M^i, M^{i+1}, $\cdots$, M^{i+v-1} share a common prefix (This can be reached by sorting the queries), and M^i_l is the last block in their prefix. If $X^{i+b}_{l+1} = Y^{i+b}_l \oplus M^{i+b}_{l+1} \notin \text{Domain}[P_1]$ for all $b \in [0, v-1]$, then Y^{i+b}_{l+1} can keep freeness consistence. However, if $\exists b \in [0, v-1]$ s.t. $X^{i+b}_{l+1} = Y^{i+b}_l \oplus M^{i+b}_{l+1} = X^{i+b}_l = X^i_l$, then the events $[Y^{i+b}_{l+1} = Y^{i+b}_l]$ and $[S^{i+b}_{l+1} = S^{i+b}_{l-1}]$ will occur, and thus undermine the freeness consistence. This probability is no more than $\Pr[\exists b \in [0, v-1], X^{i+b}_{l+1} = X^{i+b}_l] \leq \frac{v}{2^{n-1}}$. Based on its none occurrence, we focus on the probability of $[\exists b \in [0, v-1], X^{i+b}_{l+1} \in \text{Domain}[P_1] \setminus \{X^i_l\}]$. Note that some particular choices of Y^i_l may result in several $[X^{i+b}_{l+1} \in \text{Domain}[P_1] \setminus \{X^i_l\}]$ to occur at the same time, and the number of Y^i_l that induces v' such events is no more than $\#\text{Domain}[P_1]v/v'$. W.l.o.g. we assume $X^i_{l+1}, X^{i+1}_{l+1}, \cdots, X^{i+v'-1}_{l+1} \in \text{Domain}[P_1] \setminus \{X^i_l\}$ for some $v' \in [1, v]$. Choose any one such Y^i_l and fix it, then $Y^i_{l+1}, Y^{i+1}_{l+1}, \cdots, Y^{i+v'-1}_{l+1}$ would be fixed, so are $S^i_{l+1}, S^{i+1}_{l+1}, \cdots, S^{i+v'-1}_{l+1}$. The events $[S^{i+j}_{l+1} \in \text{SRange} \cup \{S^i_{l+1}, S^{i+1}_{l+1}, \cdots, S^{i+j-1}_{l+1}\}]$ $(0 \leq j \leq v'-1)$ imply no more than $v'\#\text{SRange} + \frac{v'(v'-1)}{2}$ equations, with probability $1/2^{n-1}$ to occur each. Then it is not hard to get the probability to keep freeness consistence in this case is no more than $\frac{v}{2^{n-1}} + \frac{\#\text{Domain}[P_1]v/v'}{2^n - \#\text{Range}[P_1]} \times \frac{v'\#\text{SRange}+\frac{v'(v'-1)}{2}}{2^{n-1}} \le \sum_{b=1}^{v}(\frac{1}{2^{n-1}} + \frac{(\#\text{Domain}[P_1]+b-1)^2}{2^{2n-2}})$. Notice that v is the number of invocations to P_1 related to lazy sampling $P_1[X_l^i] = Y_l^i \xleftarrow{\$} \{0,1\}^n \setminus \text{Range}[P_1]$.

4.4 The Most General Case

Based on the above, we generalize the most common case in two directions, as in Generalized case 1 and 2.

The analysis here is the same as that in Generalized case 2, until Y_l^i is fixed. and w.l.o.g. we assume $X_{l+1}^i, X_{l+1}^{i+1}, \cdots, X_{l+1}^{i+v'-1} \in \text{Domain}[P_1] \setminus \{X_l^i\}$ for some $v' \in [1, v]$ occurs. Then we take Generalized case 1 into account.

Suppose for X_{l+1}^{i+b} $(0 \le b \le v'-1)$, its following calls to P_1 $X_{l+2}^{i+b}, X_{l+3}^{i+b}, \cdots, X_{l+u[b]-1}^{i+b} \in \text{Domain}[P_1]$, with $u[b] \le L^{i+b} - l + 1$. Then $S_l^{i+b}, S_{l+1}^{i+b}, \cdots, S_{l+u[b]-1}^{i+b}$ can be fixed by Y_l^i. The events $[S_{l+a+1}^{i+b} \in \text{SRange} \cup \{S_l^{i+b}, S_{l+1}^{i+b}, \cdots, S_{l+a}^{i+b}\}]$ with $0 \le a \le u[b]-2$ and $0 \le b \le v'-1$ imply no more than $\sum_{w=1}^{s}(\#\text{SRange} + w - 1)$ equations ($s = \sum_{b=0}^{v'-1} u[b]$), with probability $1/2^{n-1}$ to occur each. Then we can get the probability for lazy sampling $P_1[X_l^i] = Y_l^i \xleftarrow{\$} \{0,1\}^n \setminus \text{Range}[P_1]$ to undermine the freeness consistence is at most $\frac{v}{2^{n-1}} + \frac{\#\text{Domain}[P_1]v/v'}{2^n - \#\text{Range}[P_1]} \times \frac{\sum_{w=1}^{s}(\#\text{SRange}+w-1)}{2^{n-1}} \le \sum_{w=1}^{s}(\frac{1}{2^{n-1}} + \frac{(\#\text{Domain}[P_1]+w-1)^2}{2^{2n-2}})$. Notice that $s = \sum_{b=0}^{v'-1} u[b]$ is the number of invocations to P_1 related to lazy sampling $P_1[X_l^i] = Y_l^i \xleftarrow{\$} \{0,1\}^n \setminus \text{Range}[P_1]$.

4.5 Summing Up

From the most common case to the most general case, we have observed that for every lazy sampling $P_1[X_l^i] = Y_l^i \xleftarrow{\$} \{0,1\}^n \setminus \text{Range}[P_1]$, its probability to undermine the freeness consistence is no more than $\sum_{w=1}^{s}(\frac{1}{2^{n-1}} + \frac{(\#\text{Domain}[P_1]+w-1)^2}{2^{2n-2}})$, where s is the number of invocations to P_1 related to this lazy sampling. Suppose in dealing with $M^1, M^2, \cdots, M^q$, we do z times lazy sampling in total, and the invocations to P_1 related to them are $s_1, s_2, \cdots, s_z$ respectively. Thus,

$$\begin{aligned}
\Pr[\texttt{Cover} = \texttt{True}] &\le \sum_{j=1}^{z} \Pr[\texttt{Cover} = \texttt{True} \text{ in lazy sampling } j] \\
&\le \sum_{j=1}^{z} \sum_{w=1}^{s_j} (\frac{1}{2^{n-1}} + \frac{(\#\text{Domain}[P_1] + w - 1)^2}{2^{2n-2}}) \\
&\le \frac{q(l_{\max}+1)}{2^{n-1}} + \sum_{w=1}^{q(l_{\max}+1)} \frac{(w-1)^2}{2^{2n-2}} \\
&\le \frac{ql_{\max}+q}{2^{n-1}} + \frac{q^3 l_{\max}^3}{2^{2n-2}},
\end{aligned}$$

where we apply $\sum_{j=1}^{z} s_j \leq q(l_{\max}+1)$ and note that #Domain$[P_1]$ is a variable growing from 0 to some value no larger than $q(l_{\max}+1)$, with lazy samplings.

At last, we get $\Pr[\exists i \in [1,q]:$ Case D occurs$] = \Pr[\texttt{Bad} = \texttt{True}] \leq \Pr[\texttt{Zero} = \texttt{True}] + \Pr[\texttt{Cover} = \texttt{True}] \leq \frac{q(l_{\max}+1)}{2^{n-1}} + \frac{ql_{\max}+q}{2^{n-1}} + \frac{q^3 l_{\max}^3}{2^{2n-2}} = \frac{ql_{\max}+q}{2^{n-2}} + \frac{q^3 l_{\max}^3}{2^{2n-2}}$. □

5 Some Suggestions

The key size in 3kf9 is three times of that for its underlying blockcipher, and this may be too large to be stored securely in some resource-restricted environments. For such cases, we give the following solutions:

1. Derive a master key $K \overset{\$}{\leftarrow} \{0,1\}^k$, and generate $K_i = E_K(\mathrm{Cst}_i)$ $(i = 1,2,3)$ with three different constants Cst_i. Then we need only to store the master key K securely. The security of the resulting scheme is still guaranteed by the PRP assumption on blockcipher E.
2. Derive $K_1 \overset{\$}{\leftarrow} \{0,1\}^k$, and generate $K_i = K_1 \oplus \mathrm{Cst}_i$ for $i = 2,3$, with two non-zero constants $\mathrm{Cst}_2, \mathrm{Cst}_3$. Then we need only to store K_1 securely. However, this solution requires blockcipher E should be a RK-PRP (pseudorandom against a kind of related-key attacks) [15].
   We warn that generating $K_2 = E_{K_1}(\mathrm{Cst}_2)$ and $K_3 = E_{K_1}(\mathrm{Cst}_3)$ may result in security flaws in 3kf9, because $E_K(K \oplus \cdot)$ may not reach pseudorandomness given E is a PRP [29].
3. Adopt a beyond-birthday-bound tweakable blockcipher TBC as the underlying primitive in 3kf9. Then, we can replace E_{K_1}, E_{K_2} and E_{K_3} by $\mathrm{TBC}_K^{T_1}$, $\mathrm{TBC}_K^{T_2}$ and $\mathrm{TBC}_K^{T_3}$, where T_1, T_2, T_3 are three public tweaks. Such a TBC has recently been introduced by Landecker, Shrimpton and Terashima [21], but the current TBC scheme still needs key size reducing.

Since CMAC has been widely used in practical applications [4], someone may want to use $\mathrm{CMAC}_{K_1}(\cdot) \oplus \mathrm{CMAC}_{K_2}(\cdot)$ to get a highly secure MAC. We note that the precise security of this proposal is still unclear [30], and it is rate-2, implying more power consumption and lower efficiency in serial implementations.

6 Conclusion

We propose a rate-1 CBC-based MAC 3kf9 with provable security beyond the birthday bound in this paper. 3kf9 is efficient for its rate-1 design, and highly-secure for its $O(\frac{l^3 q^3}{2^{2n}} + \frac{lq}{2^n})$ PRF bound. Moreover, 3kf9 is light in the sense that it needs only XOR operations besides blockcipher invocations, and thus it immediately turns into a lightweight MAC when equipped with a lightwight blockcipher. However, its key size seems to be too large in some particular environments, requiring further improvements therefore.

Acknowledgments. The authors would like to thank the anonymous referees at both FSE 2012 and Asiacrypt 2012 and the attendees at ASK 2011 for their valuable comments. Special thanks to Lei Wang for pointing out a flaw in an earlier proof, to Tetsu Iwata for some technical comments, and to Yuefei Sui for some editorial comments. Furthermore, this work is supported by the National Natural Science Foundation of China (No. 61272476, 91118006, 60903219 and 61202422), and the National Grand Fundamental Research 973 Program of China .

References

1. ISO/IEC 9797-1:1999. Information technology – Security Techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms Using a Block Cipher. Revised by ISO/IEC 9797-1:2011
2. Public Commnets, `http://csrc.nist.gov/groups/ST/toolkit/BCM/comments.html`
3. Requirements for SHA-3 by NIST, Federal Register vol. 72(212), `http://csrc.nist.gov/groups/ST/hash/sha-3/index.html`
4. Special Publication 800-38B. Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. National Institute of Standards and Technology, `http://csrc.nist.gov/groups/ST/toolkit/BCM/current_modes.html`
5. TS 33.105. 3G Security: Cryptographic Algorithm Requirements, `http://www.3gpp.org/ftp/Specs/html-info/33-series.htm`
6. TS 35.201. 3G Security: Specification of the 3GPP Confidentiality and Integrity Algorithms; Document 1: f8 and f9 Specifications, `http://www.3gpp.org/ftp/Specs/html-info/35-series.htm`
7. TS 35.202. 3G Security: Specification of the 3GPP Confidentiality and Integrity Algorithms; Document 2: Kasumi Specification, `http://www.3gpp.org/ftp/Specs/html-info/35-series.htm`
8. Bellare, M., Kilian, J., Rogaway, P.: The Security of Cipher Block Chaining. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 341–358. Springer, Heidelberg (1994)
9. Black, J., Rogaway, P.: CBC MACs for Arbitrary-Length Messages:The Three-Key Constructions. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 197–215. Springer, Heidelberg (2000)
10. Black, J., Rogaway, P.: A Block-Cipher Mode of Operation for Parallelizable Message Authentication. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 384–397. Springer, Heidelberg (2002)
11. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
12. Dodis, Y., Steinberger, J.: Domain Extension for MACs Beyond the Birthday Barrier. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 323–342. Springer, Heidelberg (2011)
13. Gilbert, H., Minier, M.: New Results on the Pseudorandomness of Some Blockcipher Constructions. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 248–266. Springer, Heidelberg (2002)

14. Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 222–239. Springer, Heidelberg (2011)
15. Iwata, T., Kohno, T.: New Security Proofs for the 3GPP Confidentiality and Integrity Algorithms. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 427–445. Springer, Heidelberg (2004)
16. Iwata, T., Kurosawa, K.: OMAC: One-Key CBC MAC. In: Johansson, T. (ed.) FSE 2003. LNCS, vol. 2887, pp. 129–153. Springer, Heidelberg (2003)
17. Iwata, T., Kurosawa, K.: On the Correctness of Security Proofs for the 3GPP Confidentiality and Integrity Algorithms. In: Paterson, K.G. (ed.) Cryptography and Coding 2003. LNCS, vol. 2898, pp. 306–318. Springer, Heidelberg (2003)
18. Jaulmes, É., Joux, A., Valette, F.: On the Security of Randomized CBC-MAC Beyond the Birthday Paradox Limit: A New Construction. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 237–251. Springer, Heidelberg (2002)
19. Joux, A., Poupard, G., Stern, J.: New Attacks against Standardized MACs. In: Johansson, T. (ed.) FSE 2003. LNCS, vol. 2887, pp. 170–181. Springer, Heidelberg (2003)
20. Knudsen, L.R., Mitchell, C.J.: Analysis of 3gpp-MAC and Two-key 3gpp-MAC. Discrete Applied Mathematics 128(1), 181–191 (2003)
21. Landecker, W., Shrimpton, T., Terashima, R.S.: Tweakable Blockciphers with Beyond Birthday-Bound Security. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 14–30. Springer, Heidelberg (2012)
22. Lucks, S.: The Sum of PRPs Is a Secure PRF. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 470–484. Springer, Heidelberg (2000)
23. Minematsu, K.: How to Thwart Birthday Attacks against MACs via Small Randomness. In: Hong, S., Iwata, T. (eds.) FSE 2010. LNCS, vol. 6147, pp. 230–249. Springer, Heidelberg (2010)
24. Nandi, M.: Fast and Secure CBC-Type MAC Algorithms. In: Dunkelman, O. (ed.) FSE 2009. LNCS, vol. 5665, pp. 375–393. Springer, Heidelberg (2009)
25. Patarin, J.: Pseudorandom Permutations Based on the DES Scheme. In: Cohen, G.D., Charpin, P. (eds.) EUROCODE 1990. LNCS, vol. 514, pp. 193–204. Springer, Heidelberg (1991)
26. Patarin, J.: The "Coefficients H" Technique. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 328–345. Springer, Heidelberg (2009)
27. Petrank, E., Rackoff, C.: CBC MAC for Real-Time Data Sources. J. Cryptology 13(3), 315–338 (2000)
28. Preneel, B., van Oorschot, P.C.: MDx-MAC and Building Fast MACs from Hash Functions. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 1–14. Springer, Heidelberg (1995)
29. Wang, P., Feng, D., Wu, W., Zhang, L.: On the Unprovable Security of 2-Key XCBC. In: Mu, Y., Susilo, W., Seberry, J. (eds.) ACISP 2008. LNCS, vol. 5107, pp. 230–238. Springer, Heidelberg (2008)
30. Yasuda, K.: The Sum of CBC MACs Is a Secure PRF. In: Pieprzyk, J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 366–381. Springer, Heidelberg (2010)
31. Yasuda, K.: A New Variant of PMAC: Beyond the Birthday Bound. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 596–609. Springer, Heidelberg (2011)