# Security analysis of the Milenage-construction based on a PRF

Alexander Maximov[1]* and Mats Näslund[2]†

[1] Ericsson Research, Lund, Sweden
[2] KTH, Royal Institute of Technology, Stockholm, Sweden

**Abstract.** This paper analyses the security of the so-called Milenage construction, developed by ETSI SAGE, when it is based on a non-one-to-one pseudo-random function (PRF) rather than a one-to-one pseudo-random permutation (PRP). It is shown that Milenage based on an $n$-bit random function and producing $t$ $n$-bit outputs, is indistinguishable from a random $tn$-bit function up to $q = O(2^{n/2}/t)$ queries. We also extend the existing security proof for PRP-based Milenage due to Gilbert by incorporating also the Milenage message authentication function in the proof.

**Keywords:** Milenage · PRF · security proof

## 1 Introduction

The 2G mobile systems defined two algorithms used for authentication (computing the authentication response), $A_3$, and a ciphering key-derivation function, $A_8$. For 3G, a new framework was defined, encompassing the following functions which are to be implemented in a network-side authentication server, and in the USIM[1], the latter residing in the phone:

- $f_1$: MAC-function, integrity protecting the authentication parameters. Most notably, the random challenge and a sequence number, SQN, is integrity protected.
- $f_2$: Computing the authentication response (corresponding to $A_3$).
- $f_3$: Ciphering key derivation function (corresponding to $A_8$).
- $f_4$: Integrity key derivation function.
- $f_5$: Anonymity key derivation function, protecting the sequence number and thereby providing some strengthened untraceability of the subscriber.

Additionally, there are two functions whose usage is conditional in the sense that they are only used in an error situation where the network's and USIM's values for the sequence number are out of synch. Specifically, a function $f_5^*$ provides confidentiality for the USIM-provided sequence number and $f_1^*$ provides integrity protection of a re-synch message (comprising the protected sequence number). The same functions are defined also for usage in the 4G and 5G standards.

The $f$-functions are not really concrete cryptographic algorithms, but rather an API-specification for such algorithms and the algorithms in practice need to be instantiated

* alexander.maximov@ericsson.com

† matsna@kth.se

[1] Universal Subscriber Identity Module

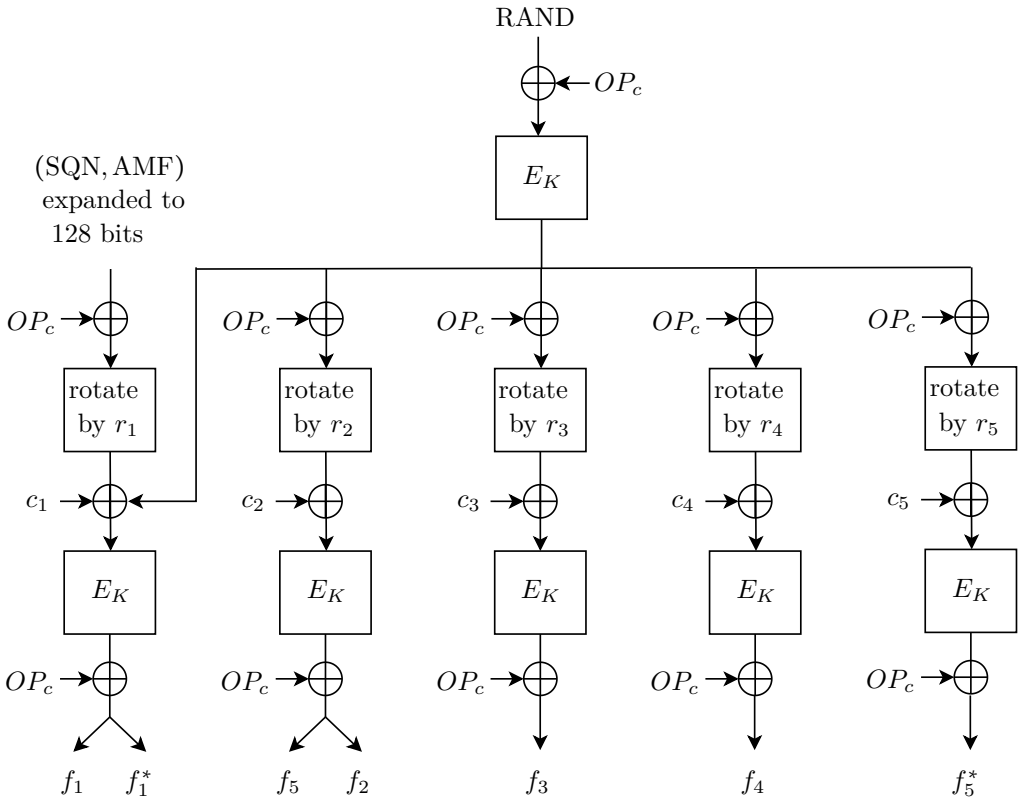


**Figure 1:** Milenage-128 instantiation of $f_1, f_2, \ldots$. $OP_c$ is derived from an operator-selectable parameter and the $c_i$-values are selectable constants with certain distinctness requirements.

with some concrete crypto primitives. The same held for $A_3, A_8$ and the lack of standardised, but not well analysed instantiations is likely a reason for some examples of insecure implementations, [AR98]. For this reason, ETSI SAGE defined an example algorithm set for $f_1$ to $f_5$, known as the *Milenage* algorithm set, [3GPa, 3GPb, SAG]. Also this more concrete algorithm-set is still just a framework, since it proposes that any suitable (secure) cryptographic core (e.g. a hash function or block cipher) can be "plugged in" as $E_K$, see Figure 1.

However, SAGE also specified a concrete instantiation of this framework using AES with 128-bit keys as the core $E_K$. In fact, at the time of specifying Milenage, NIST had not yet made the final selection for AES, so the core used was actually defined to be the block cipher Rijndael, which only later was adopted as the AES. In [Gil03] a formal security proof for $f_2, \ldots, f_5$ was given by Gilbert, assuming the core approximates a pseudo-random permutation (PRP).

Interestingly, while the technical documents [3GPa, 3GPb] as mentioned states that a non-permutation core can be used, there has so far not existed any security proof for that setting, and this is the scope of this paper. One might perhaps expect that a general pseudo-random function (PRF) could give somewhat better quantitative bounds than a PRP. However, in this specific case, the security bound that we can demonstrate for PRF-based instantiations is essentially the same as for the PRP case, i.e. that they are secure for attackers making up to $q = O(2^{n/2}/t)$ queries, where $n$ is the output block size of the core and $t$ is the number of $f$-functions (i.e. $t = 5$ for the concrete case). We also extend

the proof from [Gil03] by including the function $f_1$ in the proof.

We shall in general not deal further with $f_1^*, f_5^*$ and only analyse the algorithms under normal circumstances when no synch error occurs, i.e. when only $f_1$ to $f_5$ are used. It should be noted though that the security proof given in this paper automatically covers also the security of $f_5^*$, i.e. when both $f_5, f_5^*$ are exposed to an attacker.

### Related work

There is by now a vast number of papers devoted to the study of (pseudo)randomness properties of cryptographic constructions based on block ciphers and hash functions. A seminal work in the area is that of Luby and Rackoff, [LR98], which studies the security of the well-known Feistel-construction with 3 and 4 rounds. A number of later generalisations of that work (adding more rounds) has appeared, e.g. [Pat03].

The Milenage construction can be seen as a *mode of operation* of a block cipher. The randomness properties of the modes cipher block chaining (CBC), counter mode (CTR), and output feedback mode (OFB), was first analysed by Bellare et al., [BDJR97]. Later works have also included an associated message authentication function, e.g. the study of the Galois Counter Mode and CTR with CBC-MAC mode by McGrew and Viega, respectively Johnsson, [MV04, Joh03].

Of particular relevance to the present work is the aforementioned result by Gilbert that proves security for the basic Milenage construction (excluding the $f_1$-MAC function) when based on a PRP, [Gil03]. That work makes use of a very general result by Patarin, [Pat91a, Pat91b], which will be used also in our analysis.

Most of the constructions that do not iterate the cryptographic primitive more than 2 or 3 times, can usually be proven to be secure against attackers allowed to make up to $O(2^{n/2})$ queries to an oracle, where $n$ is the number of output bits. By adding more rounds (as in [Pat03]), or, by using the stronger security notion of *indifferentiability* due to Maurer et al., [MRH04], some works have been able to push these bounds higher, approaching $2^n$ oracle-queries, e.g. [BNR21].

## 2 Preliminaries

In the present document, $\oplus$ denotes bitwise XOR of binary strings in $I_n \stackrel{\text{def}}{=} \{0,1\}^n$. For $\mathcal{D}$, a probability distribution over some set $S$, $x \in_{\mathcal{D}} S$ refers to an element drawn from $S$ according to $\mathcal{D}$, where $x \in_{\mathcal{U}} S$ is used when we consider the uniform distribution over $S$. If $S$ is a set, $|S|$ is the cardinality of $S$ and $S^q$ denotes the Cartesian product of $q$ copies of $S$. A vector ($q$-tuple) representing an element of $S^q$, is written using boldface: $\mathbf{x} = (x^1, \ldots, x^q) \in S^q$.

Given $\mathbf{x} = (x^1, \ldots, x^q) \in (I_n)^q$, $\mathbf{y} = (y^1, \ldots, y^q) \in (I_m)^q$, we use $\mathbf{x} \mapsto^f \mathbf{y}$ to denote the event that a randomly chosen $f : I_n \mapsto I_m$ maps each component of $\mathbf{x}$ to the corresponding component of $\mathbf{y}$, i.e. the event that

$$\left(f(x^1) = y^1\right) \ \wedge \ \left(f(x^2) = y^2\right) \ \wedge \quad \cdots \quad \wedge \ \left(f(x^q) = y^q\right).$$

(This is in [Gil03] referred to as a $q$-ary transition event.)

The notation $\mathcal{F}_{n,m}$ is used for the set of all functions $\{f : I_n \mapsto I_m\}$. We will also study functions of two inputs and use $\mathcal{F}_{n_1 \times n_2, m}$ to denote $\{f : I_{n_1} \times I_{n_2} \mapsto I_m\}$.

A *distinguishing algorithm* for functions $F, G$, drawn from $\mathcal{F}_{n,m}$ according to probability distributions $\mathcal{D}_{\mathcal{F}}, \mathcal{D}_{\mathcal{G}}$, respectively, is a probabilistic algorithm, $\mathcal{A}$, that has a black-box access to an oracle $\mathcal{O}_H$, where $H$ is either $F$ or $G$, allowing $\mathcal{A}(\mathcal{O}_H)$ to obtain[2] the values

[2] If the distribution $\mathcal{D}_{\mathcal{F}}$ is efficiently sampleable, $\mathcal{A}$ can obviously also obtain values of $F'(x)$ for $F' \in_{\mathcal{D}_{\mathcal{F}}} \mathcal{F}_{n,m}$ of its own choosing.

of $H(x)$ for chosen values $x \in I_n$. Let $\mathcal{A}(\mathcal{O}_F), \mathcal{A}(\mathcal{O}_G) \in \{0,1\}$ be the output distribution of $\mathcal{A}$ when it has access to the corresponding oracle. The *distinguishing advantage* of $\mathcal{A}$ is defined to be

$$\mathrm{ADV}_{F\in_{\mathcal{D}_\mathcal{F}}\mathcal{F}_{n,m},G\in_{\mathcal{D}_\mathcal{G}}\mathcal{F}_{n,m}}(\mathcal{A}) \stackrel{\text{def}}{=} |\Pr[\mathcal{A}(\mathcal{O}_F)=1]-\Pr[\mathcal{A}(\mathcal{O}_G)=1]|,$$

the probability taken over the choices of $F, G$ and the internal random choices[3] of $\mathcal{A}$. For the purpose of this paper, $\mathcal{D}_\mathcal{F}, \mathcal{D}_\mathcal{G}$ will both be the uniform distribution over two (distinct) subsets of $\mathcal{F}_{n,m}$. When these subsets are clear from the context we simply write $\mathrm{ADV}_{F,G}(\mathcal{A})$. We similarly define

$$\mathrm{ADV}_{F,G}(q) \stackrel{\text{def}}{=} \max_{\mathcal{A}} \mathrm{ADV}_{F,G}(\mathcal{A}),$$

where the maximum is taken over all $\mathcal{A}$ making at most $q$ queries to the provided oracle. When we are considering distinguishers that are resource upper-bounded by some value $R$, we write $\mathrm{ADV}_{F,G}(q,R)$. For the case when $\mathcal{D}_\mathcal{G}$ is the uniform distribution over the entire set $\mathcal{F}_{n,m}$ (corresponding to choosing $G$ perfectly at random), we use the notation $\mathrm{ADV}_{F,\mathrm{RF}(n,m)}(\mathcal{A})$, $\mathrm{ADV}_{F,\mathrm{RF}(n,m)}(q)$, respectively.

# 3 Milenage specification

Figure 2 shows an overview of the Milenage construction that we analyse. Comparing to Figure 1, one will notice that the (configurable) data-rotations of the inputs [4] to the second layer $f$-functions have been removed to simplify analysis. For the complete understanding of the analysis below it is important to note that

- The input AMF||SQN [5] is padded to the left. In particular, the three leftmost bits are always zero-bits.
- The three leftmost bits of the constants $c_i$, $i = 1, \ldots, 5$, contain the binary encoding of the integer $i$. This is not enforced in the original specification [3GPb], but simplifies our analysis.

This implies that for a fixed input RAND, inputs to the second layer $f$-functions are always pairwise distinct.

In the sequel we shall restrict ourselves to only study the properties of the joint distribution of the output functions $f_1, f_2, \ldots, f_5$, or, more generally, the output distribution of $t$ distinct $f_i$-functions, $i = 1, 2, \ldots, t$ having the same structure as $f_2, \ldots, f_5$ above. Thus, these $t$ functions are defined entirely by the values of the constants $c_1, c_2, \ldots, c_t$ used to "offset" the inputs and where in analogy to the above, the $\lceil \log_2(t) \rceil$ leftmost bits of each $c_i$ are assumed to encode the integer value $i$.

For any $f \in \mathcal{F}_{n,n}$, define $\mathrm{Milenage}^f_{n,t}(x)$, for $x \in I_n$, as the corresponding mapping from $I_n$ to $I_{nt}$.

# 4 Security of Milenage with a PRF

We now turn to analysing the security of the Milenage construction based on an $n$-bit random function, i.e. a function $f$ drawn at random from $\mathcal{F}_{n,n}$, and we first exclude the $f_1$-function from analysis (this function will be included later).

[3] By standard arguments, one can without loss of generality consider only deterministic algorithms.

[4] The input data is rotated by $r_1, \ldots, r_5$ bits, and an operator may select these rotation parameters $r_i$ and the constants $c_i$ in Milenage. However, these parameters must follow certain criteria, e.g. to ensure that the inputs to the second layer $E_K$ are pairwise distinct.

[5] AMF=Authentication Management Field (16 bits); and SQN=Sequence Number (48 bits), see [3GPb] for more details.

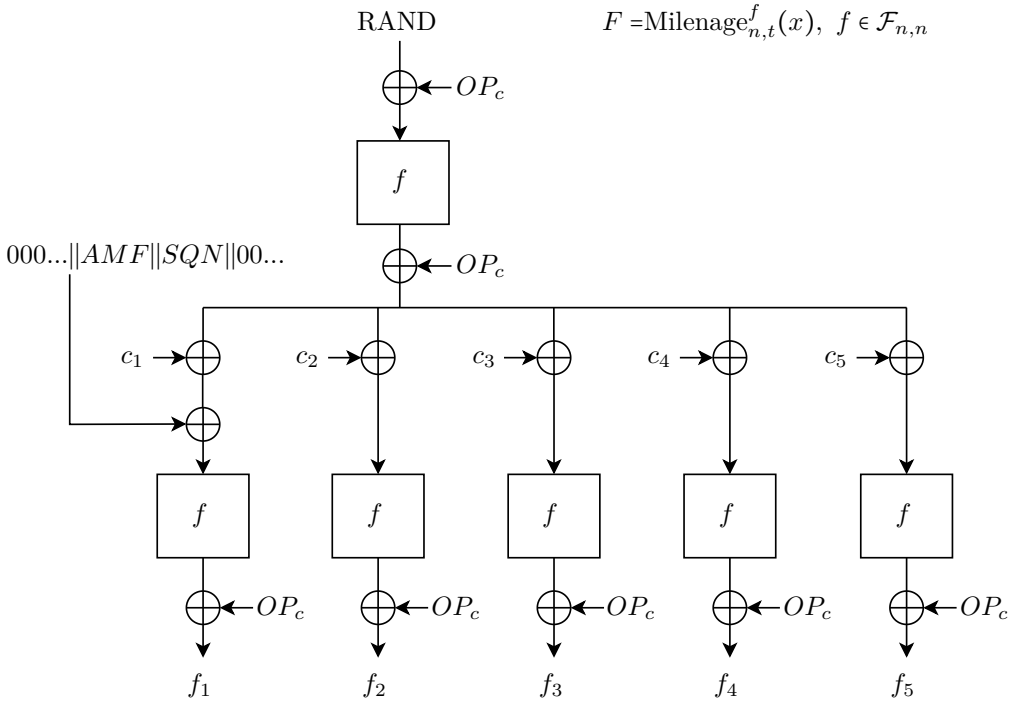


**Figure 2:** Milenage-256.

## 4.1 Proof outline

We shall analyse $\mathrm{ADV}_{\mathrm{Milenage}^f_{n,t}(x),\mathrm{RF}_{n,nt}}(q)$, the maximum possible advantage, in distinguishing Milenage$^f_{n,t}(x)$ from a function drawn at random from $\mathcal{F}_{n,tn}$ when $f$ is a (perfect) random function. Given such bound, $\epsilon_0$ say, we can by standard arguments conclude that when $f$ is instead selected as a PRF, $f'$, indistinguishable from a perfect random function from $\mathcal{F}_{n,n}$ within some $\epsilon'_0$, the ability to distinguish Milenage$^{f'}_{n,t}(x)$ from a random function must still be upper bounded by $\epsilon_0 + \epsilon'_0$. We focus on the former, "ideal" case fist.

The first part of the proof follows the structure of the existing proof for Milenage, based on a PRP, provided in [Gil03], and studies the security of all $f_i$-functions, except the $f_1$ (MAC) function. This proof in turn is based on the results by Patarin [Pat91a, Pat91b, **?**], and in order to apply these results, it is necessary to analyse the probabilities of collisions among *input* values to $f$, occurring during Milenage computations, and collisions among *output* values of $f$ during the same process. This is the case also here, but when $f$ is a random function, rather than a random permutation, two main differences need to be handled:

- Let $X = \{x^i \mid i = 1, 2, \ldots, q\}$ be the inputs queried by the attacker (i.e. corresponding to the RAND-values). Since we can without loss of generality assume that all $x^i$ are pairwise distinct, this means that when $f$ is a permutation, the outputs of the first "layer" (corresponding to the intermediate values $\{y^i\}$ in Figure 3) are automatically also pairwise distinct. This will not be the case when $f$ is not one-to-one, and this could be important[6] since two colliding $y^i = y^j$ will directly imply that also $z^i_k = z^j_k$

[6]However, as it turns out, there is actually another set of collisions that define the main limitation on the security: collisions among the *inputs* to the second layer PRF. This is due to the fact that these collisions are a factor $t^2$ more likely to occur, and collision-freeness in this set implies collision-freeness among $y$-values.

for all $k = 1, 2, \ldots, t$. This occurs after $q \sim 2^{n/2}$ queries, whereas a random function mapping into $I_{tn}$ would exhibit such collisions only after $q \sim 2^{nt/2}$ queries. On the other hand, considering also that we work with functions mapping $I_n \mapsto I_{tn}$, after $q = 2^n$ queries, an attacker will have been able to build a complete table of all possible outputs, which renders the function insecure for future usage, even if the function is otherwise perfectly random. Thus, the maximal number of allowable queries for a PRF mapping $I_n \mapsto I_{tn}$ is (in theory) the minimum of $2^n$ and $2^{tn/2}$. However, for Milenage, the maximum allowable number of queries is, due to colliding intermediate $y$-values as above, limited to $2^{n/2}$. This is thus an unavoidable upper bound on the maximum attainable security-level of the Milenage construction. So, the proof must now account for possible collisions among $y^i$ values.

- On the other hand, when $f$ is not a permutation, we can be more "liberal" about allowing collisions among individual output $z_k^i$ values. That is, if we encounter some values $z_k^i = z_l^j$ this is not a concern in the present case, since such collisions *should* occur naturally.

As mentioned, we shall rely on the following result:

**Theorem 1** (Patarin [Pat91a])**.** *Let $F \in_{\mathcal{D}} \mathcal{F}_{n,m}$ for some distribution $\mathcal{D}$ and let $q$ be an integer. Denote by $X$ the subset of $(I_n)^q$ containing all the $q$-tuples $\mathbf{x} = (x^1, \ldots, x^q)$ of pairwise distinct elements. If there exists a subset $Z \subseteq (I_m)^q$ and two positive real numbers $\epsilon_1, \epsilon_2$ such that*

1. *$|Z| \geq (1 - \epsilon_1)|I_m|^q$, and*
2. *$\forall \mathbf{x} \in X, \forall \mathbf{z} \in Z$, $\Pr_{F \in_{\mathcal{D}} \mathcal{F}_{n,m}}[\mathbf{x} \mapsto^F \mathbf{z}] \geq (1 - \epsilon_2)\frac{1}{|I_m|^q}$,*

*then* $\mathrm{ADV}_{F,\mathrm{RF}(n,m)}(q) \leq \epsilon_1 + \epsilon_2$.

Figure 3 shows the two cases which are placed as a challenge on the adversary to distinguish between: $\mathrm{Milenage}_{n,t}^f$, for randomly chosen $f$, or, a random function from $\mathcal{F}_{n,tn}$. The highlighted values $x^j, y^j, z_i^j$ as discussed above will be used in the proof below.

Note that the dependency on $x^j, y^j, z_i^j$ coming from $OP_c$ can be neglected since it merely translates all values by the same amount and has no impact on the probability distribution of these values.

After studying Milenage, restricted by omitting the $f_1$ (MAC) function, we return to study the full Milenage in Section 4.3.

## 4.2 Security of Milenage without the MAC-function

### 4.2.1 Ideal case

Our main theorem is the following

**Theorem 2.** *Let $F = \mathrm{Milenage}_{n,t}^f$, $t \geq 2$ and $f \in_{\mathcal{U}} \mathcal{F}_{n,n}$ be as in the left half of Figure 3 with pairwise distinct constants $c_i$, $i = 1, 2, \ldots, t$. Then,*

$$\mathrm{ADV}_{F,\mathrm{RF}(n,tn)}(q) \quad \leq \quad \frac{2t^2q^2}{2^n}.$$

*Proof.* Let $X \subset (I_n)^q$ be the subset consisting of all $q$-tuples of pairwise distinct values from $I_n$. We aim to apply Theorem 1, and to this end it is necessary to find a suitable set $Z$. We simply define $Z = (I_{nt})^q$ (without the pairwise distinctness condition as in [Gil03]). This immediately gives us the $\epsilon_1$ of Theorem 1 as $\epsilon_1 = 0$. It remains to find a bound on $\epsilon_2$ such that $\forall \mathbf{x} \in X, \forall \mathbf{z} \in Z$,

$$\Pr[\mathbf{x} \mapsto^F \mathbf{z}] \quad \geq \quad (1 - \epsilon_2)2^{-nqt},$$

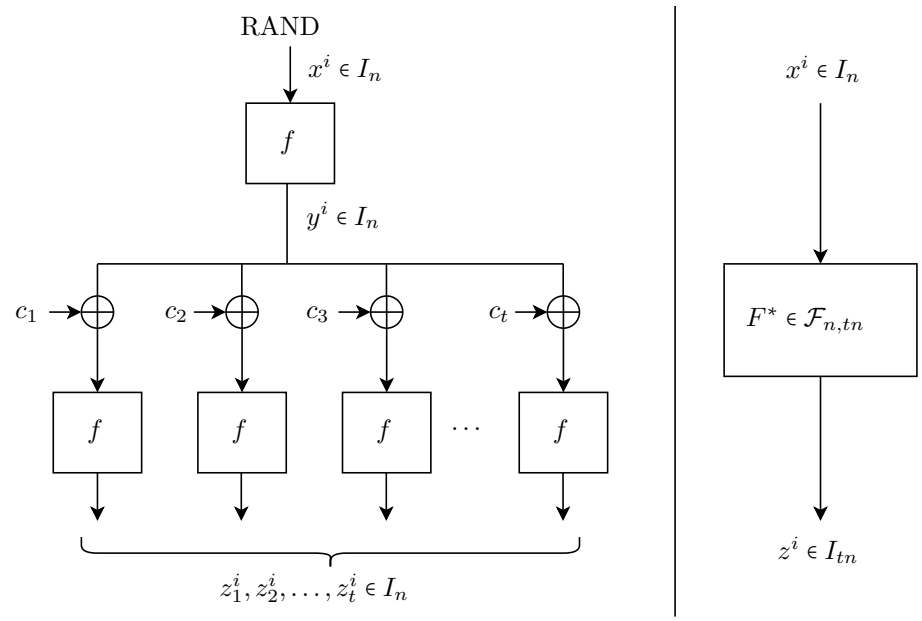


**Figure 3:** The two cases for the adversary to distinguish between.

where the probability is taken over the choices of $f$ used in instantiating $F = \text{Milenage}^f_{n,t}$.

With reference to Figure 3, we can as in [Gil03] condition the probability that a certain $\mathbf{x}$ maps to a certain $\mathbf{z}$ according to the intermediate values $\mathbf{y}$ as:

$$\begin{aligned}
\Pr[\mathbf{x} \mapsto^F \mathbf{z}] &= \sum_{\mathbf{y}\in Y} \Pr[\mathbf{x} \mapsto^f \mathbf{y} \wedge \forall i \in [1..q], \forall k \in [1..t] : (y^i \ \oplus c_k) \mapsto^f z^i_k] \\
&= \sum_{\mathbf{y}\in Y} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \ \oplus c_k) \mapsto^f z^i_k \mid \mathbf{x} \mapsto^f \mathbf{y}] \\
&\quad \cdot \Pr[\mathbf{x} \mapsto^f \mathbf{y}] \\
&= \frac{1}{2^{nq}} \sum_{\mathbf{y}\in Y} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \ \oplus c_k) \mapsto^f z^i_k \mid \mathbf{x} \mapsto^f \mathbf{y}], \qquad (1)
\end{aligned}$$

where $Y = (I_n)^q$. But since we wish to avoid collisions among intermediate $y$-values, we consider only the (proper) $Y' \subset Y$ consisting of pairwise distinct values.

Continuing from Eq. (1) we now obtain a lower bound:

$$\Pr[\mathbf{x} \mapsto^F \mathbf{z}] \geq \frac{1}{2^{nq}} \sum_{\mathbf{y}\in Y'} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \ \oplus c_k) \mapsto^f z^i_k \mid \mathbf{x} \mapsto^f \mathbf{y}]. \qquad (2)$$

In fact, let us restrict $Y'$ further to the subset $Y''_{\mathbf{x}} \subseteq Y'$ consisting of those $\mathbf{y} \in Y'$ that additionally (besides distinctness) also satisfy:

(a) $\forall i, j \in [1..q], \forall k \in [1..t]$: $x^i \neq y^j \oplus c_k$, and

(b) $\forall i, j \in [1..q], \forall k, l \in [1..t]$: $y^i \oplus c_k \neq y^j \oplus c_l$.

For simplicity of notation, we suppress the dependence on $\mathbf{x}$ and just write $Y''$ from here on. We need to lower bound the size of $Y''$, which we postpone for the moment.

Returning to Eq. (2), if we instead sum only over $\mathbf{y} \in Y''$ we obtain the lower bound

$$\begin{aligned}\Pr[\mathbf{x} \mapsto^F \mathbf{z}] &\geq 2^{-nq} \sum_{\mathbf{y}\in Y'} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \oplus c_k) \mapsto^f z_k^i \mid \mathbf{x} \mapsto^f \mathbf{y}] \\ &\geq 2^{-nq} \sum_{\mathbf{y}\in Y''} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \oplus c_k) \mapsto^f z_k^i \mid \mathbf{x} \mapsto^f \mathbf{y}]. \quad (3)\end{aligned}$$

But for $\mathbf{y} \in Y''$, due to the condition (a), the events $(y^i \oplus c_k) \mapsto^f z_k^i$ are completely independent from the event $\mathbf{x} \mapsto^f \mathbf{y}$ since the input sets $\{x^i\}$ and $\{y^i \oplus c_k\}$ are disjoint. Therefore

$$\Pr[\mathbf{x} \mapsto^F \mathbf{z}] \geq 2^{-nq} \sum_{\mathbf{y}\in Y''} \Pr[\forall i \in [1..q], \forall k \in [1..t] : (y^i \oplus c_k) \mapsto^f z_k^i]. \quad (4)$$

Furthermore, for $y \in Y''$, due to condition (b), the events $(y^i \oplus c_k) \mapsto^f z_k^i$ are also completely independent for distinct $(i,k)$-pairs. Therefore

$$\begin{aligned}\Pr[\mathbf{x} \mapsto^F \mathbf{z}] &\geq 2^{-nq} \sum_{\mathbf{y}\in Y''} \prod_{i=1}^{q} \prod_{k=1}^{t} \Pr[(y^i \oplus c_k) \mapsto^f z_k^i] \\ &= 2^{-nq} 2^{-nqt} |Y''| = 2^{-nq(t+1)} |Y''|. \quad (5)\end{aligned}$$

So to conclude, we need to lower-bound the size of $Y''$ and we start by lower-bounding the size of the super-set $Y'$.

First, we claim that the size of $Y'$ must be at least

$$|Y'| \geq \left(1 - \frac{q^2}{2^{n+1}}\right)|Y| = \left(1 - \frac{q^2}{2^{n+1}}\right)2^{qn}. \quad (6)$$

This follows since for a random subset $\mathbf{y} \in Y$ of size $q$, the *expected* number of collisions in $\mathbf{y}$ is $\frac{q(q-1)}{2} \cdot 2^{-n}$. Therefore, the probability that there is one or more collisions in such $\mathbf{y}$ is by Markov's inequality[7] at most $\frac{q(q-1)}{2} \cdot 2^{-n}$ so the probability of a collision-free subset must be greater than $(1 - \frac{q^2}{2^{n+1}})$. (If $|Y'|$ was smaller than the claimed bound, this would contradict the probability of obtaining collision-free subsets.)

We now lower bound the size of $Y''$ as a subset of $Y'$, by considering the restrictions imposed by conditions (a) and (b).

(a): The number of $\mathbf{y} \in Y'$ that fail to meet (a) for fixed $(i,j,k)$ is $\frac{|Y'|}{2^n}$. Since there are $q^2t$ distinct $(i,j,k)$, the fraction of $\mathbf{y} \in Y'$ failing (a) is at most $\frac{q^2t}{2^n}$.

(b): We divide the analysis into three cases. First, for $i = j$ there can be no $\mathbf{y} \in Y'$ failing this criteria, due to the distinctness of $c_k \neq c_l$. Similarly, for $k = l$ there cannot be any such $\mathbf{y}$ either, simply due to the fact that $\mathbf{y} \in Y'$ rules out that $y^i = y^j$. Finally, fix $i \neq j$ and $k \neq l$. The number of $\mathbf{y} \in Y'$ such that $y^i = y^j \oplus c_k \oplus c_l = y^j \oplus \Delta$, for any $\Delta \neq 0$, is $\frac{|Y'|}{2^n - 1}$. There are $q(q-1)/2$ pairs of $(i,j)$ and $t(t-1)/2$ pairs $(k,l)$ to consider so the fraction of $\mathbf{y} \in Y'$ not meeting (b) is therefore strictly less than

$$\frac{t^2q^2}{4(2^n - 1)} < \frac{t^2q^2}{2^{n+1}}.$$

In summary, the fraction of $\mathbf{y} \in Y'$ failing either (a) or (b) is upper-bounded by

$$\frac{q^2t}{2^n} + \frac{t^2q^2}{2^{n+1}} \leq \frac{t^2q^2}{2^n}. \quad (7)$$

[7] Markov's inequality states that for a random variable $R$, $\Pr[R \geq a] \leq \mathbb{E}[R]/a$, which we here apply with $a = 1$.

The bound Eq. (7) implies that the size of $Y''$ is at least a $(1-\frac{t^2q^2}{2^n})$-fraction of the size of $Y'$. Thus, combining this with the bound Eq. (6) on the size of $|Y'|$ we get

$$\begin{aligned}|Y''| &\geq \left(1-\frac{t^2q^2}{2^n}\right)\left(1-\frac{q^2}{2^{n+1}}\right)2^{qn} = \left(1-\frac{t^2q^2}{2^n}-\frac{q^2}{2^{n+1}}+\frac{t^2q^4}{2^{2n+1}}\right)2^{qn} \\ &> \left(1-\frac{2t^2q^2}{2^n}\right)2^{qn}.\end{aligned}$$

Putting this back into Eq. (5) we obtain

$$\begin{aligned}\Pr[\mathbf{x}\mapsto^F \mathbf{z}] &\geq 2^{-nq(t+1)}|Y''| \geq 2^{-nq(t+1)}\left(1-\frac{2t^2q^2}{2^n}\right)2^{qn} \\ &= \left(1-\frac{2t^2q^2}{2^n}\right)2^{-nqt} = \left(1-\frac{2t^2q^2}{2^n}\right)|Z|,\end{aligned}$$

so we may take $\epsilon_2 = \frac{2t^2q^2}{2^n}$. □

Returning to the Footnote 6, we now see that introducing the intermediate set $Y'$, capturing uniqueness among the intermediate $y$-values, was not really necessary. Although a collision among pairs of these $y$-values would limit the security, the important set of collisions are those captured by condition (b), since these are a factor $t^2$ more likely to occur. In fact, note that if there existed $y^i, y^j \in Y''$ with $y^i = y^j$ for $i \neq j$, then we would also have $y^i \oplus c_k = y^j \oplus c_k$ which would be captured by (b). In Theorem 4 below, a slightly different proof methodology is used which could probably be used also here and thereby slightly improve the bound on $q$. Nevertheless, we decided to keep the proof above as-is that also demonstrates a different approaches.

### 4.2.2 Going from the ideal case to a PRF

In a real instantiation of Milenage, $f$ will not be an ideal random function, but rather some concrete *pseudo-random function*, assumed to be indistinguishable from a random function within some advantage $\epsilon$. Also in this case we can obtain an upper bound on distinguishability, expressed in terms of $\epsilon$ and the bound established in Theorem 2.

**Theorem 3.** *Let $F' \subset \mathcal{F}_{n,n}$, e.g. a subset defined by a fixed, key-dependent construct for which it holds that*

$$\mathrm{ADV}_{F',\mathcal{F}_{n,n}}((t+1)q, R+q(t+1)) < \epsilon,$$

*for some $t \geq 2$. Let $M$ be the distribution obtained by instantiating* $\mathrm{Milenage}^{f'}_{n,t}(x)$ *by functions $f'$, drawn uniformly at random from $F'$. Then it holds that*

$$\mathrm{ADV}_{M,\mathcal{F}_{n,tn}}(q,R) \leq \epsilon + \frac{2t^2q^2}{2^n}.$$

*Proof.* Take any distinguisher $\mathcal{A}$ using at most $q$ queries and resources at most $R$ and assume for contradiction that

$$\mathrm{ADV}_{M,\mathcal{F}_{n,tn}}(\mathcal{A}) > \epsilon + \frac{2t^2q^2}{2^n}.$$

Let $p, p^*$ be the probabilities that $\mathcal{A}$ answers "1" on inputs from $M, \mathcal{F}_{n,tn}$ respectively, so that by the assumption $|p-p^*| > \epsilon + \frac{2t^2q^2}{2^n}$.

Consider the distinguisher $\mathcal{A}'$, aiming to distinguish between $F', \mathcal{F}_{n,n}$, obtained as follows: $\mathcal{A}'$ simply launches $\mathcal{A}$, and whenever $\mathcal{A}$ queries some input $x$, $\mathcal{A}'$ queries its own oracle, obtaining $y = f(x)$, then it queries its oracle again $t$ times, obtaining $z_k = f(y \oplus c_k)$

for $k = 1, 2, \ldots, t$, and then returns $(z_1, \ldots, z_t)$ as response to $\mathcal{A}$ (the values $y$ and $z_i$ are as in Figure 3.) Observe that $\mathcal{A}'$ in this way returns exactly $\text{Milenage}^f_{n,t}(x)$ where $f$ is either from $F'$, or, from $\mathcal{F}_{n,n}$. Finally, when $\mathcal{A}$ gives its response, $\mathcal{A}'$ returns the same value.

Observe that if $\mathcal{A}$ uses $q$ queries, then $\mathcal{A}'$ uses $q(t+1)$ queries. Further, if $\mathcal{A}$ uses resources at most $R$, then $\mathcal{A}'$ uses at most $q(t+1)$ additional operations (deriving the input queries for its own oracle[8]).

Note that $\Pr[\mathcal{A}' = 1 \mid f \in F'] = p$. If we define

$$p' \quad = \quad \Pr[\mathcal{A}' = 1 \mid f \in \mathcal{F}_{n,n}] \quad = \quad \Pr[\mathcal{A} = 1 \mid f \in \mathcal{F}_{n,n}],$$

we can on one hand see that $|p - p'|$ is precisely the value of $\text{ADV}_{F',\mathcal{F}_{n,n}}(\mathcal{A})$. But on the other hand, we know from Theorem 2 that $|p' - p^*| \leq \frac{2t^2q^2}{2^n}$. So, due to the triangle inequality, we must have $|p - p^*| \leq |p - p'| + |p' - p^*|$, which is then impossible unless $|p - p'| \geq \epsilon$, contradicting the assumption in the theorem statement. □

## 4.3 Including $f_1$ in the proof

The proof in Theorem 2 omits the Milenage function corresponding to the message authentication code (obtained by the $f_1$ output function). This function is different from the others since it has an additional input which, in an adversarial setting, could be selected by an attacker. As can be seen in Figure 1 and Figure 2, $f_1$ takes an additional input in the form of (SQN, AMF).

When $f_1$ is excluded (or modelled as any other $f$-functions), one can without loss of generality assume that all inputs (corresponding to the RAND-input) are distinct. When $f_1$ is added in scope, there is a trivial distinguishing attack using only two queries of the form (RAND, $\text{SQN}_1$, $\text{AMF}_1$) and (RAND, $\text{SQN}_2$, $\text{AMF}_2$), with ($\text{SQN}_1$, $\text{AMF}_1$) $\neq$ ($\text{SQN}_2$, $\text{AMF}_2$). For this reason, the model used to analyse also $f_1$ must require that the attacker is not allowed to query on the same RAND twice. In practice, a Milenage implementation would also refuse to output all $f$-functions if the same SQN is used more than once, but we do not require this for our analysis. Thus, the structure of the function analysed is changed to that of the Figure 4.

We model the special input to $f_1$ as an attacker-selectable string $u \in I_{n-d}$ for some $d \geq \lceil \log_2(t) \rceil$, corresponding to the actual (SQN, AMF) input. Although an analysis might be possible without this requirement, it simplifies the analysis. Specifically, the $d$ most significant bits of $c_i$ are used to encode a unique instance-index, which can be used to ensure that the inputs to the second layer of PRFs are always distinct, for a given input $x$. Thus, the attacker should not be allowed to influence these $d$ bits by his/her choice of $u^i$.

**Theorem 4.** *Let $F = \text{Milenage}^f_{n,t}(u, x)$, $t \geq 2$ and $f \in_{\mathcal{U}} \mathcal{F}_{n,n}$ be as in Figure 4 with constants $c_i$, $i = 1, 2, \ldots, t$, that are pairwise distinct in the $d$ leftmost bits. Then, for any distinguisher making at most $q \leq 2^{n/2}/t$ queries $(u^i, x^i)$ for pairwise distinct $x^i$:*

$$\text{ADV}_{F,\text{RF}((n-d)\times n, tn)}(q) \quad \leq \quad \frac{q^2t^2}{2^n}.$$

The proof proceeds similarly to Theorem 2 but needs to be more careful in the analysis of the influence from the $u^i$ values. Note that the bound is actually slightly better than in Theorem 2, but the bound of that theorem could as noted probably be slightly improved.

*Proof.* Let the sets $X, Y, Z$ be defined as in the proof of Theorem 2. We intend again to apply Theorem 1 so we also here immediately have $\epsilon_1 = 0$, following from the definition

[8] Computing an XOR is treated as taking constant time.

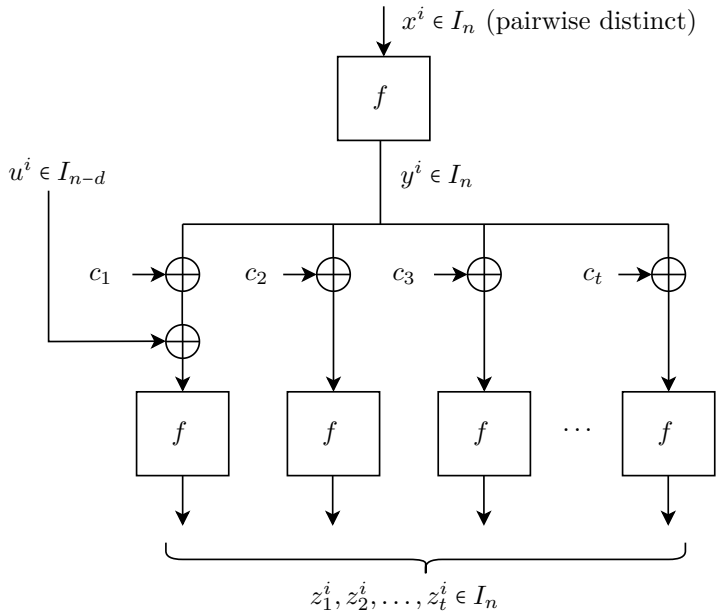


**Figure 4:** Including the "real" $f_1$ function.

of $Z$ and the main task remaining is to analyse what value we can obtain for $\epsilon_2$ such that $\forall \mathbf{x} \in X, \forall \mathbf{z} \in Z$,

$$\Pr[\mathbf{x} \mapsto^F \mathbf{z}] \quad \geq \quad (1-\epsilon_2)2^{-nqt}.$$

The restrictions on the intermediate-value set $Y$ to a subset $Y'$ become slightly different since we need to account for the special format of the inputs to $f_1$ comprising the attacker-selectable values $u^i \in I_{n-d}$.

Let $\mathbf{y} = (y^1, \ldots, y^q) \in Y$ be a set of intermediate values that have been produced after the $q$ queries. We say that $\mathbf{y}$ is *safe* if it satisfies the following conditions:

(A) $\forall i, j \in [1..q], \forall k \in [2..t]$: $x^i \neq y^j \oplus c_k$;

(B) $\forall i, j \in [1..q]$ : $x^i \neq y^j \oplus u^j \oplus c_1$;

(C) $\forall i, j \in [1..q]$, $\forall k, l \in [2..t]$ and $(i,k) \neq (j,l)$: $y^i \oplus c_k \neq y^j \oplus c_l$;

(D) $\forall i, j \in [1..q]$, $\forall k \in [2..t]$: $y^i \oplus c_k \neq y^j \oplus u^j \oplus c_1$;

(E) $\forall i, j \in [1..q]$ and $i \neq j$: $y^i \oplus u^i \oplus c_1 \neq y^j \oplus u^j \oplus c_1$.

The new restrictions to capture inclusion of $f_1$ are: B (which mirrors A), D and E (both mirroring C).

Let $Y'_{\mathbf{x},\mathbf{u}} \subseteq Y$ be the set of all safe sets as defined above. For simplicity, from here on, we again suppress the dependence of $\mathbf{x}, \mathbf{u}$ and write only $Y'$. We are interested in the size $|Y'|$, which is a function of $q$ and $t$ (where the value of $t$ is fixed by design) and we define this number $S(q)$. Consider a specific safe set $\mathbf{y} = (y^1, \ldots, y^q) \in Y'$. We can view $\mathbf{y}$ as having been incrementally built during the query process. Having selected the $i$ first values, a certain number of $y^{i+1}$ values are selectable such that the extended set will remain safe. There are $2^n - e(i)$ such choices for $y^{i+1}$, where $e(i)$ denotes the number of choices that needs to be *evicted* due to the conditions (A)-(E). Recursively, we see that $S(i+1) = S(i)(2^n - e(i))$. By defining $S(0) = 1$, we have that in general:

$$S(q) \quad = \quad \prod_{i=0}^{q-1} (2^n - e(i)). \tag{8}$$

Let $e_\chi(i)$ be the number of choices for $y^{i+1}$ evicted specifically by condition $\chi \in \{(\mathrm{A}),\ldots,(\mathrm{E})\}$. Since one and the same choice for $y^{i+1}$ could be evicted by more than one of (A)-(E), it will certainly hold that $e(i) \le \sum_\chi e_\chi(i)$, and therefore

$$S(q) \;\ge\; \prod_{i=0}^{q-1}(2^n - \sum_\chi e_\chi(i)). \tag{9}$$

We make the following claims, the proofs of which are postponed.

**Claim 1.** *For all* $i \in [0..q-1]$, $e_A(i) \le q(t-1)$;

**Claim 2.** *For all* $i \in [0..q-1]$, $e_B(i) = q$;

**Claim 3.** *For all* $i \in [0..q-1]$, $e_C(i) \le \frac{i(t^2-3t+4)}{2}$;

**Claim 4.** *For all* $i \in [0..q-1]$, $e_D(i) \le i(t-1)$;

**Claim 5.** *For all* $i \in [0..q-1]$, $e_E(i) \le i$.

By these claims

$$e(i) \;\le\; q(t-1) + q + \frac{i(t^2-3t+4)}{2} + i(t-1) + i \;=\; qt + i\frac{t^2-t+4}{2}. \tag{10}$$

Note that when $q \le 2^{n/2}/t$, the factors in Eq. (8) and Eq. (9) are all positive. By Eq. (9) and Eq. (10) we then get:

$$\begin{aligned} S(q) \;&\ge\; \prod_{i=0}^{q-1}\left(2^n - qt - i\frac{t^2-t+4}{2}\right) \;=\; 2^{nq}\prod_{i=0}^{q-1}\left(1 - 2^{-n}\left(qt + i\frac{t^2-t+4}{2}\right)\right) \\ &\ge\; 2^{nq}\left(1 - 2^{-n}\sum_{i=0}^{q-1}\left(qt + i\frac{t^2-t+4}{2}\right)\right) \;=\; 2^{nq}\left(1 - 2^{-n}\left(q^2t + \frac{q(q-1)}{2}\cdot\frac{t^2-t+4}{2}\right)\right) \\ &>\; 2^{nq}\left(1 - \frac{q^2}{2^n}\cdot\left(t + \frac{t^2-t+4}{4}\right)\right) \;>\; 2^{nq}\left(1 - \frac{q^2t^2}{2^n}\right), \end{aligned} \tag{11}$$

where the last inequality follows from $t \ge 2$. We now proceed as in the proof of Theorem 2, summing over only the safe intermediate set $Y'$:

$$\begin{aligned} \Pr[\mathbf{x} \mapsto^F \mathbf{z}] \;\ge\; \sum_{\mathbf{y}\in Y'} \Pr[\forall i \in [1..q], \forall k \in [2..t] :\; &((y^i \oplus c_k) \mapsto^f z_k^i) \\ &\wedge (y^i \oplus u^i \oplus c_1) \mapsto^f z_1^i) \mid \mathbf{x} \mapsto^f \mathbf{y}] \cdot \Pr[\mathbf{x} \mapsto^f \mathbf{y}], \end{aligned} \tag{12}$$

where $\Pr[\mathbf{x} \mapsto^f \mathbf{y}] = 2^{-nq}$. However, due to conditions (A) and (B), both events $(y^i \oplus c_k) \mapsto^f z_k^i$ and $(y^i \oplus u^i \oplus c_1) \mapsto^f z_1^i$ are independent from the event $\mathbf{x} \mapsto^f \mathbf{y}$ so that

$$\begin{aligned} \Pr[\mathbf{x} \mapsto^F \mathbf{z}] \;\ge\; 2^{-nq} \sum_{\mathbf{y}\in Y'} \Pr[\forall i \in [1..q], \forall k \in [2..t] :& \\ ((y^i \oplus c_k) \mapsto^f z_k^i) \wedge (y^i \oplus u^i \oplus c_1) \mapsto^f z_1^i)]&. \end{aligned} \tag{13}$$

Furthermore, due to condition (C), the events in the set $\{(y^i \oplus c_k) \mapsto^f z_k^i\}$ are independent for distinct $(i,k)$, and due to condition (E), the events $\{(y^i \oplus u^i \oplus c_1) \mapsto^f z_1^i\}$ are also independent for distinct $i$. Finally, due to condition (D), for any $(i,j,k)$ the event $(y^i \oplus c_k) \mapsto^f z_k^i$ is independent from the event $(y^j \oplus u^j \oplus c_1) \mapsto^f z_1^j$. To conclude,

$$\begin{aligned} \Pr[\mathbf{x} \mapsto^F \mathbf{z}] \;&\ge\; 2^{-nq} \sum_{\mathbf{y}\in Y'} \prod_{i=1}^{q}\left(\Pr[(y^i \oplus u^i \oplus c_1) \mapsto^f z_1^i] \cdot \prod_{k=2}^{t} \Pr[(y^i \oplus c_k) \mapsto^f z_k^i]\right) \\ &=\; 2^{-nq}2^{-nqt}|Y'| \;=\; 2^{-nq(t+1)}|Y'|. \end{aligned} \tag{14}$$

Substituting the lower bound on $|Y'|$ from Eq. (11):

$$\Pr[\mathbf{x} \mapsto^F \mathbf{z}] \quad \geq \quad 2^{-nq(t+1)} 2^{nq} \left(1 - \frac{q^2 t^2}{2^n}\right) \quad = \quad 2^{-nqt} \left(1 - \frac{q^2 t^2}{2^n}\right),$$

so $\epsilon_2 = \frac{q^2 t^2}{2^n}$ as claimed. □

What remains is to complete the proof of claims 1 to 5. Observe that the adversary may only select *distinct* inputs $(x^1, \ldots, x^q)$, while there is no restriction on the choices of $(u^1, \ldots, u^q)$. Also note that for distinct $x$ we must get a sequence of values of $y$ that is unknown to the adversary. Therefore, when adding a new $y^{i+1}$ to the set of $(y^1, \ldots, y^i)$ in the proofs below, we account that whatever values of $x^{1..q}$ the adversary selects in the end there will be exactly $q$ *distinct* input values. Since the adversary does not know the mapping between distinct $x$ and the values of $y$, then the selection of the values of $x$ and $u$ can be done either all together or one-by-one.

*Proof of Claim 1.* When adding a new value of $y^{i+1}$ to the set $(y^1, \ldots, y^i)$, we require that $y^{i+1} \neq x^j \oplus c_k$ for $j \in [1..q]$ and $k \in [2..t]$. For each $i$, there are therefore always at most $q(t-1)$ values to avoid, defined by the $q$ distinct $x^j$-values, combined with the $t-1$ distinct constants of $c_k$, i.e. $e_A(i) \leq q(t-1)$. □

*Proof of Claim 2.* We here need to evict values that satisfy $x^j = y^{i+1} \oplus u^{i+1} \oplus c_1$ for $j \in [1..q]$, which corresponds to the requirement $y^{i+1} \neq x^j \oplus u^{i+1} \oplus c_1$. Note that $u^{i+1}$ and $c_1$ are already fixed and all values of $x$ are distinct, thus we need to evict exactly $q$ values. Thus, $e_B(i) = q$. □

*Proof of Claim 3.* The next selected $y^{i+1}$ must not be such that $y^{i+1} \oplus c_k = y^j \oplus c_l$ for any $j \in [1..i+1]$, $k, l \in [2..t]$ and $(i+1, k) \neq (j, l)$. When $j = i+1$ we get the condition $y^{i+1} \oplus c_k \neq y^{i+1} \oplus c_l$ always holds since $c_k \neq c_l$ for $k \neq l$, thus we can limit our scope to $j \in [1..i]$. Note that if $y^{i+1} \oplus c_k = y^j \oplus c_l$, then $y^{i+1} \oplus c_l = y^j \oplus c_k$, so we can further limit to the case $k \leq l$. When $k = l$, we only require that $y^{i+1} \neq y^j$, thus evicting $i$ elements. For other cases where $k < l$, each selection of $y^{i+1}$ evicts at most $(t-1)(t-2)/2$ other elements for each $j$, so

$$e_C(i) \quad \leq \quad i\frac{(t-1)(t-2)}{2} + i \quad = \quad i\frac{t^2 - 3t + 4}{2}.$$

The set of evicted values does not need to have unique values, but collisions only means that fewer values would be evicted. Thus the upper bound holds. □

*Proof of Claim 4.* In this case the concern when selecting $y^{i+1}$ is related to collisions of type $y^{i+1} \oplus c_k = y^j \oplus u^j \oplus c_1$, for $j \in [1..i+1]$ and $k \in [2..t]$.

For the case $j = i+1$ we note that the adversary may have selected any values of $u^j$, but still we get that $(u^j \oplus c_1 \oplus c_k) \neq 0$ since any pair of $c_k, c_l$ for $k \neq l$ differ in the upper $d$ bits, where the attacker has no influence. Thus, the condition $y^{i+1} \oplus c_k \neq y^{i+1} \oplus u^{i+1} \oplus c_1$ always holds and we limit our scope to $j \in [1..i]$. Then for each $j$, the number of elements that we need to exclude is at most $(t-1)$, therefore we get $e_D(i) \leq i(t-1)$. □

*Proof of Claim 5.* Collisions of type $y^{i+1} \oplus u^{i+1} \oplus c_1 = y^j \oplus u^j \oplus c_1$ $j \in [1..i]$ are equivalent to $y^{i+1} = (y^j \oplus u^j) \oplus u^{i+1}$. Since the adversary does not know the previous values of $y$ (which are internal values before the final call to the PRF), she cannot select the next $u^{i+1}$ to guarantee a collision, but such a collision may still occur by chance. Thus, for any fixed $u^{i+1}$ the number of evicted choices for $y^{i+1}$ is at most $i$, i.e. $e_E(i) \leq i$. □

## 5 Summary and conclusion

We have shown that replacing the block-cipher core defined in the current Milenage specification by a (non-one-to-one) PRF is safe in the sense that the security against distinguishing attacks remain quantitatively true. We also included the $f_1$ MAC-function in the scope of the proof. This could stimulate new attractive implementation options based e.g. on hash functions. One rationale for this could be to increase the block size from 128 to 256 bits (or more) since few block ciphers with larger block size than 128 bits exist.

## References

[3GPa] 3GPP. TS 35.205. Specification of the Milenage Algorithm Set, Document 1: General.

[3GPb] 3GPP. TS 35.206. Specification of the Milenage Algorithm Set, Document 2: Algorithm Specification.

[AR98] Smartcard Developer Association and ISAAC Security Research. GSM Cloning, April 1998. http://www.isaac.cs.berkeley.edu/isaac/gsm-faq.html.

[BDJR97] Mihir Bellare, Anand Desai, Eric Jokipii, and Phillip Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *38th FOCS*, pages 394–403. IEEE Computer Society Press, October 1997. https://doi.org/10.1109/SFCS.1997.646128.

[BKR00] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The Security of the Cipher Block Chaining Message Authentication Code. *Journal of Computer and System Sciences*, 61(3):362–399, 2000. https://doi.org/10.1006/jcss.1999.1694.

[BNR21] Ritam Bhaumik, Mridul Nandi, and Anik Raychaudhuri. Improved indifferentiability security proof for 3-round tweakable Luby–Rackoff. *Designs, Codes and Cryptography*, 89:2255–2281, October 2021. https://doi.org/10.1007/s10623-021-00913-4.

[Gil03] Henri Gilbert. The Security of "One-Block-to-Many" Modes of Operation. In Thomas Johansson, editor, *FSE 2003*, volume 2887 of *LNCS*, pages 376–395. Springer, Heidelberg, February 2003. https://doi.org/10.1007/978-3-540-39887-5_27.

[Joh03] Jakob Johnsson. On the Security of CTR + CBC-MAC. In Mitsuru Matsui and Robert J. Zuccherato, editors, *SAC 2003*, volume 3006 of *LNCS*, pages 76–93. Springer, Heidelberg, August 2003. https://doi.org/10.1007/3-540-36492-7_7.

[LR98] Michael Luby and Charles Rackoff. How to construct pseudorandom permutations and pseudorandom functions. *SIAM Journal on Computing*, 17:373–386, April 1998. https://doi.org/10.1137/0217022.

[MRH04] Ueli Maurer, Renato Renner, and Clemens Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 21–39. Springer, Heidelberg, February 2004. https://doi.org/10.1007/978-3-540-24638-1_2.

[MV04] David Mcgrew and John Viega. The Security and Performance of the Galois/Counter Mode (GCM) of Operation. In Anne Canteaut and Kapalee Viswanathan, editors, *INDOCRYPT 2004*, volume 3348 of *LNCS*, pages 343–355. Springer, Heidelberg, December 2004. https://doi.org/10.1007/978-3-540-30556-9_27.

[Pat91a] Jacques Patarin. Étude des Générateurs de Permutations Basés sur le Schéma du D.E.S. Ph.D. Thesis, Université Paris IV, November 1991.

[Pat91b] Jacques Patarin. New Results on Pseudorandom Permutation Generators Based on the DES Scheme. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 301–312. Springer, Heidelberg, August 1991. https://doi.org/10.1007/3-540-46766-1_25.

[Pat03] Jacques Patarin. Luby-Rackoff: 7 rounds are enough for $2^{n(1-\epsilon)}$ security. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 513–529. Springer, Heidelberg, August 2003. https://doi.org/10.1007/978-3-540-45146-4_30.

[SAG] ETSI SAGE. TR 133 909. Report on the Design and Evaluation of the Milenage Algorithm Set.

# A MAC security

The function $f_1$ is a MAC function on the pair of message blocks $M_1 = x^i$ and $M_2 = u^i \oplus c_1$ defined as:

$$\text{MAC} \quad = \quad f_K(f_K(M_1) \oplus M_2),$$

where $f_K$ is the underlying (keyed) PRF drawn from $\mathcal{F}_{n,n}$ when the key $K$ is fixed. This is a well-known prefix-free CBC-MAC construct, but based on a PRF, instead of a PRP that is commonly used for instantiation of $f_K$ (e.g., AES-based).

Bellare et al., see Section 3 in [BKR00], showed that the CBC-MAC transform applied to a PRF yields a provably secure PRF, and prove the following results (Theorems 3.1 and 3.2 in [BKR00]):

$$\begin{aligned} \text{ADV}^{\text{dist}}_{\text{CBC-MAC}^m(\mathcal{F}_{n,n}),\mathcal{F}_{mn,n}} &\leq 1.5 \cdot \frac{q^2m^2}{2^n}, \\ \text{ADV}^{\text{prf}}_{\text{CBC-MAC}^m(f_K)}(q,T) &\leq \text{ADV}^{\text{prf}}_{f_K}(mq, T + O(mqn)) + 1.5 \cdot \frac{q^2m^2}{2^n}, \end{aligned}$$

where $m$ is the number of the message blocks, which is $m = 2$ in our case, $q$ is the number of queries, and $T$ is the computational time. This proves that the MAC generated by $f_1$ is secure up to $q = O(2^{n/2})$ queries. This is also inline with the results of Theorem 4: if there was a distinguisher for $f_1$ (alone), then that distinguisher could be used to build a distinguisher for $(f_1, \ldots, f_t)$.

Yet another question is whether the forgery of $f_1$ has a better advantage when the outputs from other function, $f_2, \ldots, f_t$ are also known. But this can be seen as the query-process of the forgery-algorithm getting $t-1$ additional CBC-MACs in a single query on 2-block messages, namely $\{(M_1, M_2)\} = \{(x^i, u^i \oplus c_1), (x^i, c_2), \ldots, (x^i, c_t)\}$, resulting in $t$ "MACs" $z^i_1, \ldots, z^i_t$. Thus, the security level of $q = O(2^{n/2})$ queries for the stand-alone function $f_1$ is still bounded by $q = O(2^{n/2}/t)$ when a single query results in $t$ "MACs" produced by $f_1, \ldots, f_t$. However, the freedom of the attacker is even smaller in this case, since the attacker cannot pick the $t$ messages to be queried independently from each other so that the actual security might be better.