Parallelizable Encryption Mode with Almost Free Message Integrity

Charanjit S. Jutla

IBM T. J. Watson Research Center,
Yorktown Heights, NY 10598-704

1 Introduction

In this document we propose a new mode of operation for symmetric key block cipher algorithms. The main feature distinguishing the proposed mode from existing modes is that along with providing confidentiality of the message, it also provides message integrity. In other words, the new mode is not just a mode of operation for encryption, but a mode of operation for authenticated encryption. As the title of the document suggests, the new mode achieves the additional property with little extra overhead, as will be explained below.

The new mode is also highly parallelizable. In fact, it has critical path of only two block cipher invocations. By one estimate, a hardware implementation of this mode on a single board (housing 1000 block cipher units) achieves terabits/sec (10^{12} bits/sec) of authenticated encryption. Moreover, there is no penalty for doing a serial implementation of this mode.

The new mode also comes with proofs of security, assuming that the underlying block ciphers are secure. For confidentiality, the mode achieves the same provable security bound as CBC. For authentication, the mode achieves the same provable security bound as CBC-MAC.

The new parallelizable mode removes chaining from the well known CBC mode, and instead does an input whitening (as well an output whitening) with a pairwise independent sequence. Thus, it becomes similar to the ECB mode. However, with the input whitening with the pairwise independent sequence the new mode has provable security similar to CBC (Note: ECB does not have security guarantees like CBC). Also, the output whitening with the pairwise independent sequence guarantees message integrity.

The pairwise independent sequence can be generated with little overhead. In fact, the input and output whitening sequence need only be pairwise differentially uniform, which is a weaker property than pairwise independence, as explained in the details below. The weaker pairwise differentially uniform sequence can be generated with even lesser overhead.

The parallelizable mode comes in two flavors. These flavors refer to how the pairwise differentially uniform sequence is generated. In one mode, we just use a pairwise independent sequence generated by a subset construction. In another mode, the pairwise differentially uniform sequence is generated by $(a*i)$ modulo a fixed prime number. There will be one standard prime number for each bit-size block cipher. Thus, for 64 bit block ciphers the prime could be $2^{64} - 257$. For 128 bit block ciphers, the prime could be $2^{128} - 159$.

The modes are described below in more detail.

We first give definitions of pairwise independence and related concepts. Then we describe the parallelizable mode using the *algebraic* construction $a * i$ modulo a fixed prime. Next, we describe the mode using only exclusive-or operations. In section 5, the different notions of security are defined. In section 6, we prove that the IAPM construction is secure for message integrity. We first start by proving the theorem for the construction in Fig 2 with t=1. In section 6.1 we give an alternative proof of this theorem inspired by Johan Håstad. In section 6.2 we extend the proof to arbitrary t. Then, in section 6.3 we prove the theorem for the construction in Fig 1, i.e.the IAPM mode using GFp. Finally, in section 7 we prove that the IAPM scheme is secure for message secrecy as well.

2 Definitions

Definition 1 (pair-wise independence) A sequence of uniformly distributed n-bit random numbers $s_1, s_2, ..., s_m$, is called *pair-wise independent* if for every pair $i, j, i \neq j$, and every pair of n bit constants c_1 and c_2, probability that $s_i = c_1$ and $s_j = c_2$ is 2^{-2n}.

Definition 2 (pair-wise differentially-uniform) A sequence of uniformly distributed n-bit random numbers $s_1, s_2, ..., s_m$, is called *pair-wise differentially-uniform* if for every pair $i, j, i \neq j$, and every n bit constant c, probability that $s_i \oplus s_j$ is c is 2^{-n}.

It is a fact that a pair-wise independent uniformly distributed sequence is also pair-wise differentially uniform.

Definition 3 (pair-wise differentially-uniform in GFp) A sequence of random numbers $s_1, s_2, ..., s_m$ uniformly distributed in GFp, is called *pair-wise differentially-uniform in* GFp if for every pair $i, j, i \neq j$, and every constant c in GFp, probability that $(s_i - s_j)$ mod p is c is $1/p$.

A sequence of m pair-wise independent numbers can be generated from about $\log m$ independent random numbers by a subset construction. The subset construction only involves exclusive-or operations.

A pair-wise independent sequence can also be generated by an algebraic construction in GFp, by using two independent random numbers a and b in GFp. The sequence is given by $s_i = (a + i * b)$ mod p.

A pair-wise differentially uniform in GFp sequence can be generated from only a single random number a in GFp by defining $s_i = (i * a)$ mod p.

3 Integrity Aware Parallelizable Mode (IAPM) using a prime number

Let n be the block size of the underlying block cipher. We will restrict our attention to $n = 128$ in this paper. If the block cipher requires keys of length

k, then this mode requires two independent keys of length k. Let these keys be called $K0$ and $K1$. From now on, we will use f_K to denote the encryption function under key K.

The message to be encrypted P, is divided into blocks of length n each. Let these blocks be $P_1, P_2, ..., P_{m-1}$. As in CBC, a random initial vector r of length n bits is chosen. The vector r need not be chosen randomly, as long as it is unique for each message. This random vector is used to generate a new random vector a using the block cipher and key K0, which in turn is used to prepare $m+1$ new pairwise differentially uniform vectors $S_0, S_1, ..., S_m$.

Let $p = 2^{128} - 159$. The number p is known to be a prime. This prime will be fixed for all invocations of this mode using block ciphers of block size 128 bit. For 64-bit ciphers $p = 2^{64} - 257$ is recommended.

Now, the sequence $S_0, S_1, ...S_m$ is generated by the following procedure:

```
*procedure* pairwise_diff_uniform_sequence(*in* $r, m, K0$; *out* $S$)
{
        $a = f_{K0}(r)$
        *if* $(a \geq (2^{128} - 159))$ $a = (a + 159) \bmod 2^{128}$
        $S_0 = a$
        *for* $j = 1$ to $m$ do
              $S_j = (S_{j-1} + a) \bmod 2^{128}$
              *if* $(a > S_j)$   $S_j = S_j + 159$
        *end for*
}
```

The condition $(a > S_j)$ is equivalent to 128-bit integer addition overflow in the previous step. Note that we do not reduce modulo p if $(S_{j-1} + a) < 2^{128}$, but we do compensate by 159 if $(S_{j-1} + a) \geq 2^{128}$, as in the latter case, $(S_{j-1} + a)$ mod p $= S_{j-1} + a - (2^{128} - 159) = (S_{j-1} + a - 2^{128}) + 159$.

In this mode, the input and output whitening is done by 128-bit integer addition. The ciphertext message $C =< C_0, C_1, ..., C_m >$ is generated as follows (see fig 1):

```
        $C_0 = r$
        *for* $i = 1$ to $m - 1$ do
              $M_i = (P_i + S_i) \bmod 2^{128}$
              $N_i = f_{K1}(M_i)$
              $C_i = (N_i + S_i) \bmod 2^{128}$
        *end for*
        checksum $= P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$
        $M_m = (\text{checksum} + S_m) \bmod 2^{128}$
        $N_m = f_{K1}(M_m)$
        $C_m = (N_m + S_0) \bmod 2^{128}$
```

Note that for computing the checksum we use xor instead of addition modulo 2^{128}. The scheme is secure even if the checksum is computed by a modulo 2^{128} sum, but for the standard we prefer that the checksum be computed by an xor-sum. Note that S_0 is used in the last step.

The above scheme is invertible. The inversion process yields blocks $P_1, P_2, ..., P_m$.

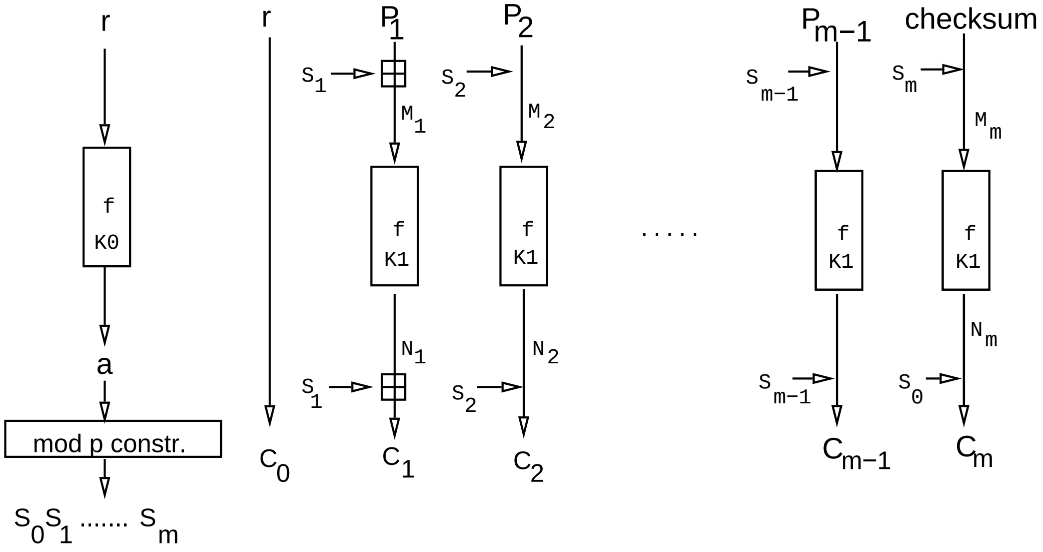

Fig. 1. Integrity Aware Parallelizable Mode (IAPM)

The decrypted plaintext is $< P_1, P_2, ..., P_{m-1} >$. Message integrity is verified by checking

$$P_m = P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$$

Here is the pseudo-code for decryption:

```
$r = C_0$
invoke pairwise_diff_uniform_sequence$(r, m, K0, S)$;
*for* $i = 1$ to $m - 1$ do
    $N_i = (C_i - S_i) \bmod 2^{128}$
    $M_i = f^{-1}_{K1}(N_i)$
    $P_i = (M_i - S_i) \bmod 2^{128}$
*end for*
checksum $= P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$
$N_m = (C_m - S_0) \bmod 2^{128}$
$M_m = f^{-1}_{K1}(N_m)$
$P_m = (M_m - S_m) \bmod 2^{128}$
Integrity $\equiv$ $(P_m ==$ checksum$)$
```

4 IAPM with only xor operations

The mode described above uses integer addition. We now describe a similar mode in which the only operations are block cipher invocations and exclusive-or operations. In particular, the pairwise differentially uniform sequence is generated using a subset construction. Actually, this sequence has the stronger property of pairwise independence. The subset construction is also optimized using Gray code (http://hissa.nist.gov/dads/HTML/graycode.html). The penalty one has to pay in this mode is that instead of generating one extra vector a as described in the previous section, one now generates about $\log m$ new vectors, where m is the number of blocks in the message to be encrypted.

As before the message P to be encrypted, is divided into blocks of length n each. Let these blocks be $P_1, P_2, ..., P_{m-1}$. The initial vector r is used to generate $t = \lceil \log(m+2) \rceil$ new vectors, which in turn are used to prepare $m+1$ new pairwise independent vectors $S_0, S_1, ..., S_m$.

The following pseudo-code is the proposed method of generating the sequence S.

```
*procedure* pairwise_independent_sequence(*in* $r, m, K0$; *out* $S$)
{
$W_0 = f_{K0}(r)$;
$S_0 = W_0$;
*for* $i = 1$ to $m$ do
        $j = i + 1$;
        $k = 0$;
        /* find the index of the least significant ON bit in $(i+1)$ */
        *while* $((j \& 1) == 0)$ do
                $k = k + 1$; $j = j >> 1$; /* increment $k$ and right shift */
        *end while*
        *if* $((j \oplus 1) == 0)$ /* if $(i+1)$ is a power of 2 */
                $W_k = f_{K0}(W_0 + k)$;
        $S_i = S_{i-1} \oplus W_k$;
*end for*
}
```

Note that S_i is obtained from S_{i-1} in just one XOR. The inner while loop condition is checked two times on average.

The ciphertext message $C =< C_0, C_1, ..., C_m >$ is generated as follows (see fig 2):

```
        $C_0 = r$
        *for* $i = 1$ to $m - 1$ do
                $M_i = (P_i \oplus S_i)$
                $N_i = f_{K1}(M_i)$
                $C_i = (N_i \oplus S_i)$
        *end for*
        checksum $= P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$
        $M_m =$ (checksum $\oplus S_m$)
        $N_m = f_{K1}(M_m)$
        $C_m = (N_m \oplus S_0)$
```

Again, note that S_0 is used in the last step. This pseudo-code is same as the one in the previous section except that all integer additions have been replaced by exclusive or operations.

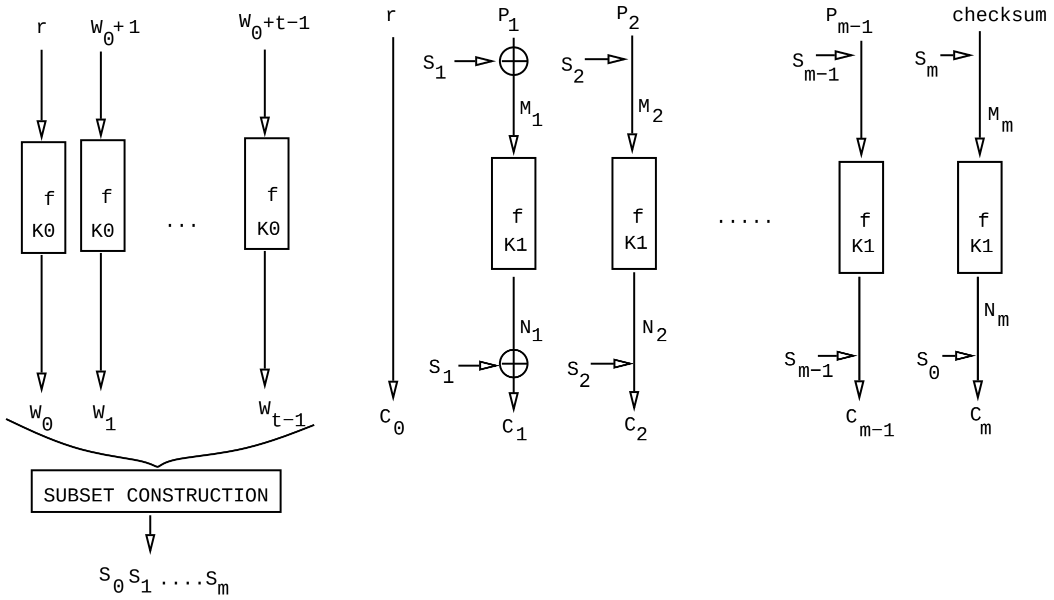

Fig. 2. IAPM with only xor operations

Here is the pseudo-code for decryption:

```
$r = C_0$
invoke pairwise_independent_sequence$(r, m, K0, S)$;
for $i = 1$ to $m - 1$ do
    $N_i = (C_i \oplus S_i)$
    $M_i = f_{K1}^{-1}(N_i)$
    $P_i = (M_i \oplus S_i)$
end for
checksum $= P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$
$N_m = (C_m \oplus S_0)$
$M_m = f_{K1}^{-1}(N_m)$
$P_m = (M_m \oplus S_m)$
Integrity $\equiv$ $(P_m ==$ checksum$)$
```

5 Encryption Schemes: Message Security with Integrity Awareness

We give definitions of schemes which explicitly define the notion of secrecy of the input message.

In addition, we also define the notion of message integrity. Moreover, we allow arbitrary length input messages (upto a certain bound).

Let Coins be the set of infinite binary strings. Let $l(n) = 2^{O(n)}$, and $w(n) = O(n)$. Let $\mathcal{N}$ be the natural numbers.

Definition A (probabilistic, symmetric, stateless) encryption scheme with message integrity consists of the following:

- **initialization:** All parties exchange information over private lines to establish a private key $x \in \{0,1\}^n$. All parties store x in their respective private memories, and $|x| = n$ is the security parameter.
- **message sending with integrity:**

$$\text{Let } E : \{0,1\}^n \times \text{Coins} \times \mathcal{N} \times \{0,1\}^{l(n)} \to \{0,1\}^{l(n)} \times \mathcal{N}$$
$$D : \{0,1\}^n \times \mathcal{N} \times \{0,1\}^{l(n)} \to \{0,1\}^{l(n)} \times \mathcal{N}$$
$$\text{MDC} : \mathcal{N} \times \{0,1\}^{l(n)} \to \{0,1\}^{w(n)}$$

be polynomial-times function ensembles. In E, the third argument is supposed to be the length of the plain-text, and E produces a pair consisting of cipher-text and its length. Similarly, in D the second argument is the length of the cipher-text. We will drop the length arguments when it is clear from context. The functions E and D have the property that for all $x \in \{0,1\}^n$, for all $P \in \{0,1\}^{l(n)}$, $c \in \text{Coins}$

$$D_x(E_x(c, P)) = P \| \text{MDC}(P)$$

We will usually drop the random argument to E as well, and just think of E as a probabilistic function ensemble.It is also conceivable that MDC may depend on Coins, cipher-text.

Definition (*Security under Find-then-Guess* [8], [2])

Let

$$A1 : \mathcal{N} \times Coins \times \{0,1\}^{l(n)} \to \{0,1\}^{l(n)}$$
$$A2 : Coins \times \{0,1\}^{l(n)} \to \{0,1\}^{l(n)} \times \{0,1\}^{l(n)}$$
$$A3 : \mathcal{N} \times Coins \times \{0,1\}^{l(n)} \times \{0,1\}^{l(n)} \to \{0,1\}^{l(n)}$$
$$A : Coins \times \{0,1\}^{l(n)} \to \{0,1\}$$

be adversaries.

The chosen plaintext attack works as follows:

(choose a private key) Randomly choose a private key x.

(chosen plaintext attack-I) Choose $R \in_{\mathcal{U}}$ Coins. For $j \in [1..l(n)]$, phase j works as follows. Let

$$C = (E_x(P^1), ..., E_x(P^{j-1}))$$

be the concatenation of the encryptions of the first $j-1$ message blocks (appropriately padded with zeroes). Then $P^j = A1(j, R, C)$. At the end of at most $p \leq l(n)$ phases, let $P = \langle P^1, ..., P^p \rangle$ be all the message blocks produced by $A1$, and let $C = \langle E_x(P^1), ..., E_x(P^p) \rangle$ be the encryption of P.

(choose a private message block) Let $\langle P'^0, P'^1 \rangle = A2(R, C)$ be the pair of message blocks produced by A2. It is required that neither P'^0 nor P'^1 is

among the message blocks in P. Also P'^0 and P'^1 should be of the same length. Choose $b \in_{\mathcal{U}} \{0,1\}$ privately, let $P' = P'^b$ be the privately chosen message, and let $C' = E_x(P')$ be the encryption of P'.
(chosen plaintext attack-II) the chosen plaintext attack is continued as in part I, resulting in another encrypted sequence D of an adaptively chosen plaintext sequence $P'' = \langle P^{p+2}, ..., P^{p+1+p'} \rangle$. We will extend C to denote the complete encrypted sequence $C = \langle E_x(P^1), ..., E_x(P^p), E_x(P'), E_x(P^{p+2}), ..., E_x(P^{p+1+p'}) \rangle$.
(predict the bit) The advantage of the adversary is

$$Adv_A = |\Pr[A(R,C) = b] - 1/2|$$

An encryption scheme is said to be (t, q, μ, ϵ)-secure against chosen plaintext attack if for any adversary A (including $A1, A2, A3$) which runs in time at most t and asks at most q queries of E_x, these totaling at most μ blocks, its advantage is at most ϵ.

The following notion of security is also called *integrity of ciphertext* ([4]).
Definition (*Message Integrity):* Consider an adversary A running in two stages. In the first stage (*find*) A asks r queries of the oracle E_x. Let the oracle replies be $C^1, ...C^r$. Subsequently, A produces a cipher-text C', different from each C^i, $i \in [1..r]$. Since D has length of the cipher-text as a parameter, the breakup of $D_x(C')$ as $P' \| P''$, where $|P''| = w(n)$, is well defined. The adversary's success probability is given by

$$\text{Succ} \stackrel{\text{def}}{=} Pr[\text{MDC}(P') = P'']$$

An encryption scheme is secure for message integrity if for any adversary A, A's success probability is negligible.

6 Message Integrity

In this section we show that the mode of operation IAPM in Fig 2 guarantees message integrity with high probability. We first restrict to the case where only one W is generated , i.e. W_0, as that brings out the main idea of the proof. In the next subsection we show how the proof easily generalizes to arbitrarily many W's. Finally, we show how these proofs also generalize to the mod p construction of Fig 1 (in which case only one W is generated anyway).

In the following theorem, we will assume that the block cipher (under a key $K1$) is a a random permutation F. We also assume that the t W's are generated using an independent random permutation G (for instance, using a different key $K2$ in a block cipher).

Let the adversary's queries in the first stage be $p^1, P^2, ...P^z$. We write p^1 in lower case, as for each adversary p^1 is fixed. All random variables will be denoted by upper case letters. Let the corresponding ciphertexts be $C^1, ..., C^z$. We will use C to denote the sequence of ciphertext messages $C^1, ..., C^z$. For all random variables corresponding to a block, we will use superscripts to denote the message number, and subscripts to denote blocks in a particular message. Thus C^i_j will be the random variable representing the jth block in ciphertext message i. More precisely, this variable should be written $C_j^i(F, G)$, as it is a function of the two permutations. However, we will drop the arguments when it is clear from context.

Let the adversary's query in the second stage be cipher-text C', different from all ciphertexts in the first stage. We will use primed variables to denote the variables in the second stage.

We will use W to denote the set of variables $\{W_j^i \ : i \in [1..z], j \in [1..t]\} \cup \{W_j', j \in [1..t]\}$. We will use S^i (S') to denote masks or "whitening" blocks generated using W^i (W' resp). Any method can be used to generate S^i from W^i, as long as S_j^i are pairwise differentially uniform. For a particular adversary, S_j^i is a function of permutation G and the initial vector, and hence should (more precisely) be written as $S_j^i(G, C_0^i(F, G))$ ($C_0^i(F, G)$ being the IV used to generate W_1^i). But, we will drop the arguments as it will be clear from context. For any constant r, we will denote by $S_j^i(r)$ the random variable $S_j^i(G, r)$.

The variables M and N are as in Fig 2. For example, $M_j^i = P_j^i \oplus S_j^i$.

We start with some informal observations to aid the reader in the eventual formal proof. Since the new ciphertext C' is different from all old ciphertexts, it must differ from each old ciphertext C^i in a least block number, say $d(i)$. For each C^i (except at most one C^k), the block number $d(i) = 0$, with high probability. In Lemma 3 we show that with high probability $N'_{d(k)}$ is different from all old N_j^i, and all other new N' blocks (except for a special case). Thus, $M'_{d(k)}$ is random. Then it follows (Theorem 1) that in either case the checksum is unlikely to validate.

We first prove the theorem for schemes in which the pairwise differentially uniform sequence is generated using only one W, i.e. $t = 1$. The general case is addressed in a later subsection.

Theorem 1. *Let A be an adversary attacking the message integrity of IAPM ($t = 1$) with random permutations F and G. Let A make at most z queries in the first stage, totaling at most μ blocks. Let $u = \mu + z$. Let v be the maximum number of blocks in the second stage. Then for adversary A,*

$$Succ < (2 * u^2 + z^2 + (z + 1)^2 + u + v + 2 + o(1)) * 2^{-n}$$

Proof:

In the first stage the adversary makes queries with a total of at most m plaintext messages (chosen adaptively). W.l.o.g. assume that the adversary actually makes exactly m total message queries in the first stage. Let L^i be the random variable representing the length of ciphertext C^i (i.e. the checksum block has index $L^i - 1$). Similarly, L' will denote the length of C'.

We prove that either the adversary forces the following event E0, or the event E1 happens with high probability. In either case the checksum validates with low probability.

The first event E0 is called deletion attempt, as the adversary in this case just truncates an original ciphertext, but retains the last block.

Event E0 (*deletion attempt*): There is an $i \in [1..z]$, such that $2 \leq L' < L^i$, and

$$(i)\ \ \forall j \in [0..L'-2] : C'_j = C^i_j$$

$$\text{and } (ii)\ \ C'_{L'-1} = C^i_{L^i-1}$$

Event E1 says that there is a block in the new ciphertext C', such that its N variable is different from all previous Ns (i.e. from original ciphertexts from the first stage), and also different from all other new Ns.

Event E1: there is an $x \in [1..L'-1]$ such that

$$(i)\ \ \forall s \in [1..z] \forall j \in [1..L^s - 1]\ \ : N'_x \neq N^s_j$$

$$\text{and } (ii)\ \ \forall j \in [1..L'-1], j \neq x\ \ : N'_x \neq N'_j$$

We next show that in both cases (i.e E0 or E1) the checksum validates with low probability.

For the case that E0 happens, we have (since $S' = S^i$ and $N'_{L'-1} = N^i_{L^i-1}$),

$$
\begin{aligned}
&(\sum_{j=1}^{L'-1} P'_j = 0)\ \wedge \text{E0} \\
&\Rightarrow \sum_{j=1}^{L'-2} (P^i_j) + M^i_{L^i-1} + S^i_{L'-1} = 0 \\
&\equiv \sum_{j=1}^{L'-2} (P^i_j) + \sum_{j=1}^{L^i-2} (P^i_j) + S^i_{L^i-1} + S^i_{L'-1} = 0
\end{aligned}
$$

Note that r^i can be chosen after P^i has been determined (as P^i is a deterministic function of $C^1, \ldots, C^{i-1}$), and hence the S^is are independent of P^i. Since the S^is are pairwise differentially uniform and $L' < L^i$, the above event happens with probability at most 2^{-n}.

For the case E1, by Lemma 2, the checksum validates with probability at most $1/(2^n - u - v)$

Thus the adversary's success probability is upper bounded by

$$\Pr[\neg(E0 \vee E1)] + \frac{1}{2^n - (u+v)} + \frac{1}{2^n}$$

which by Lemma 3 is at most

$$(u^2 + z^2 + u + v + 2) * 2^{-n} + (u^2 + (z+1)^2) * 2^{-n} + O(u+v) * 2^{-2n}$$

□

Lemma 2. $Pr[\sum_{j=1}^{L'-1} P'_j = 0 \mid E1] \leq \frac{1}{2^n - (u+v)}$

Proof: F being a random permutation, under E1, $F^{-1}(N'_x)$ can not take values already assigned to $F^{-1}(N^s_j)$, $s \in [1..z]$, $j \in [1..L^s - 1]$. Also, $F^{-1}(N'_x)$ can be chosen after $F^{-1}(N'_j)$ have been assigned values ($j \neq x$). Thus, under the condition that event E1 has happened we have that $M'_x = F^{-1}(N'_x)$ can take any of the other values, i.e. excluding the following (at most) $(\mu + z) + L' - 2$ values, with equal probability (independently of C, C', r^i, $i \in [1..z]$, G, and hence independently of W, and independent of E1 itself):

- values already taken by $M^s_1, ..., M^s_{L^s-1}$, for each s, and
- the values to be taken (or already fixed) by M'_j, $j \in [1..L' - 1]$, $j \neq x$.

Now, $\sum_{j=1}^{L'-1} P'_j = 0$ iff

$$F^{-1}(N'_x) \;=\; M'_x = \sum_{j=1, j\neq x}^{L'-1} (M'_j \oplus S'_j) \;\oplus S'_x$$

Given any value of the RHS, since the LHS can take (at least) $2^n - (u + v - 2)$ values, the probability of LHS being equal to RHS is at most $1/(2^n - (u + v))$. □

Lemma 3. *Let events E0,E1 be as in Theorem 1. Then,*

$$Prob[\neg(E0 \vee E1)] < (u^2 + z^2 + u + v) * 2^{-n} + (u^2 + (z + 1)^2) * 2^{-n}$$

Proof: We first calculate the probability of event $(E0 \vee E1)$ happening under the assumption that F and G are random functions (instead of random permutations). Since F (and G) is invoked only u times ($(z+1)$ times resp.), a standard argument shows that the error introduced in calculating the probability of event (E0 $\vee$ E1) is at most $(u^2 + (z + 1)^2) * 2^{-n}$.

We now consider an event, which says that all the M variables are different. The goal is to claim independence of the corresponding N variables, and hence the C variables. However, the situation is complicated by the fact that the condition that all the M^i_j variables for some i are different, may cause the variables $C^{i'}_j$, for $i' < i$, to be no more independent. However, a weaker statement can be proved by induction. To this end, consider the **event** E2(y), for $y \leq z$:

$$\forall i, i' \in [1..y], \forall j, j', j \in [1..L^i - 1], j' \in [1..L^{i'} - 1], (i, j) \neq (i', j') : \; (M^i_j \neq M^{i'}_{j'})$$

Event E2(z) will also be denoted by E2.

We also predicate on the event that all the initial variables C^i_0 are different. Let $E3$ be the **event** that

$$\forall i, j \in [1..z], i \neq j \;:\; C^i_0 \neq C^j_0$$

For $\overrightarrow{r} = r^1, ..., r^z$, all r^i different, let $E3(\overrightarrow{r})$ be the event that for all $i \in [1..z]$, $C^i_0 = r^i$.

Let $l()$ be the length of the first ciphertext (determined by the adversary). We will use constant c^i to denote strings of arbitrary block length. We will use c to denote the sequence $c^1, ..., c^z$. The function $|\cdot|$ is used below to represent length of a message in blocks. Given a sequence of ciphertext messages $c^1, ..., c^i$, $i \leq z$, let $l(c^1, ..., c^i)$ be the length of the $(i+1)$th ciphertext (which is determined by the adversary, and therefore is a deterministic function of $c^1, ...c^i$). Recall that each ciphertext includes the block C_0^i, which is just r^i under $E3(\overrightarrow{r})$. Also, since C' is a deterministic function of C, given $c^1, ..., c^z$ let the ciphertext in the second stage be c' with length l'. We have

$$Pr[\neg(E0 \vee E1) \ \wedge \ E2 \mid E3(\overrightarrow{r})\,] \ = \sum_{c^1:\,|c^1|=l()} \cdots \sum_{c^i:\,|c^i|=l(c^{i-1},...,c^1)} \cdots$$

$$\cdots \sum_{c^z:\,|c^z|=l(c^{m-1},...,c^1)} Pr[\neg(E0 \vee E1) \ \wedge \bigwedge_i C^i = c^i \ \wedge \ E2 \mid E3(\overrightarrow{r})\,] \qquad (1)$$

In this sum, if for some i, $c_0^i \neq r^i$, then the inside expression is zero. Also, if event E0 holds for c (which determines c'), then the inside expression above for that c is zero. So, from now on, we will assume that E0 does not hold for $C = c$. Then, the inside expression above becomes:

$$Pr[\neg(E0 \vee E1) \wedge \bigwedge_i C^i = c^i \wedge E2 \mid E3(\overrightarrow{r})]$$

$$\leq min_{x \in [1..l'-1]} \Bigg\{ \sum_{s \in [1..z], j \in [1..|c^s|-1]} Pr[(N'_x = N_j^s) \wedge \bigwedge_i C^i = c^i \wedge E2 \mid E3(\overrightarrow{r})]$$

$$+ \sum_{j \in [1..l'-1], j \neq x} Pr[(N'_x = N'_j) \wedge \bigwedge_i C^i = c^i \wedge E2 \mid E3(\overrightarrow{r})] \Bigg\}$$

For each s, j, we have $(N'_x = N_j^s)$ iff $(S'_{x^*} \oplus S_{j^*}^s) = (C'_x \oplus C_j^s)$, where $S'_{x^*}, S_{j^*}^s$ are the masks that are used for these ciphertext blocks. That is, $j^* = j$ if $j < |c^s| - 1$ and $j^* = 0$ otherwise, and similarly $x^* = x$ if $x < l' - 1$ and $x^* = 0$ otherwise (Similarly for $j \neq x$ we have $(N'_x = N'_j)$ iff $(S'_{x^*} \oplus S'_{j^*}) = (C'_x \oplus C'_j)$).

Since each of the summands in the expression above has a conjunct $C = c$ for some constant string c (and since the forged ciphertext C' is a function of C), it follows that each of the summands in the first sum can be written as $\Pr[(S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s) = c'_x \oplus c_j^s) \ \wedge C = c \wedge E2 \mid E3(\overrightarrow{r})]$. Note that $S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s)$ can in some cases be identically zero. As c is some constant string, then $c'_x \oplus c_j^s$ is also constant, and recall that the variables $S(c_0)$ depend only on the choice of G. Thus, each of these summands (if $S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s)$ is not identically zero) can be bounded by

$$\begin{aligned}
&\Pr[S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s) = c'_x \oplus c_j^s \ \wedge C = c \wedge E2 \mid E3(\overrightarrow{r})] \\
&= \ \Pr[C = c \wedge E2 \mid S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s) = c'_x \oplus c_j^s \ \wedge E3(\overrightarrow{r})] \\
&\ \ * \ \Pr[S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s) = c'_x \oplus c_j^s \mid E3(\overrightarrow{r})] \\
&\leq \ (2^{-n})^{\mu} \ * \ \Pr[S'_{x^*}(c'_0) \oplus S_{j^*}^s(c_0^s) = c'_x \oplus c_j^s \mid E3(\overrightarrow{r})]
\end{aligned}$$

where the last inequality follows by Claim 5 with $\mu = \sum_{i \in [1..z]}(l(c^{i-1}, \ldots, c^1) - 1)$. A similar inequality holds for the summands in the second sum (i.e. $N'_x =$

N'_j case). Thus, by Claim 4, the inside expression in equation (1) is at most $2^{-n\mu} * (u+v) * 2^{-n}$. Since we have $2^{n\mu}$ summands, it follows that

$$\Pr[\neg(E0 \vee E1) \wedge E2 \mid E3(\overrightarrow{r})] \leq (u+v) * 2^{-n}$$

Finally, we calculate $\Pr[\neg(E0 \vee E1)]$

$$\begin{aligned}
&\mathrm{P}r[\neg(E0 \vee E1)] \\
&\leq \mathrm{P}r[\neg(E0 \vee E1) \wedge E2 \mid E3] + \mathrm{P}r[\neg E2 \mid E3] + \mathrm{P}r[\neg E3] \\
&\leq \mathrm{P}r[\neg E3] + \\
&\quad \sum_{r^1,\ldots,r^z} ((\mathrm{P}r[\neg(E0 \vee E1) \wedge E2 \, |E3(\overrightarrow{r})] + \mathrm{P}r[\neg E2 \, |E3(\overrightarrow{r})]) * \mathrm{P}r[E3(\overrightarrow{r})|E3]) \\
&\leq z^2 * 2^{-n} + (u+v) * 2^{-n} + (u)^2 * 2^{-n}
\end{aligned}$$

where the last inequality follows by Claim 6. □

Claim 4: For each constant c (and its corresponding c') for which event E0 does not hold, and constant $\overrightarrow{r}$ with distinct values, there is an $x \in [1..l'-1]$ such that

(i) $\forall s \in [1..z] \forall j \in [1..|c^s|-1]$:
if $S'_{x^*}(c'_0) \oplus S^s_{j^*}(c^s_0)$ is identically zero then $c'_x \oplus c^s_j \neq 0$, otherwise

$$\Pr[S'_{x^*}(c'_0) \oplus S^s_{j^*}(c^s_0) = c'_x \oplus c^s_j \mid E3(\overrightarrow{r})] \leq 2^{-n},$$

(ii) $\forall j \in [1..|l'-1], j \neq x,$:

$$\Pr[S'_{x^*}(c'_0) \oplus S'_{j^*}(c^s_0) = c'_x \oplus c'_j \mid E3(\overrightarrow{r})] \leq 2^{-n}$$

Proof: These are the different cases (we will drop the argument from S^s and S' as it will be clear from context):

(**a**) (*New IV*) If for all $i \in [1..z]$, $c'_0 \neq r^i$, then we choose $x = 1$. In that case $N'_1 = N'_j$ is same as $C'_1 \oplus C'_j = S'_1 \oplus S'_{j^*}$, where $j^* = j$ if $j \neq (l'-1)$, and $j^* = 0$ otherwise. Thus, for $j \in [1..l'-1], j \neq x$, since S' is pairwise differentially uniform, probability of $(S'_1 \oplus S'_{j^*} = c'_1 \oplus c'_j)$ is 2^{-n} (even under $E3(\overrightarrow{r})$).

Similarly, $N'_1 = N^s_j$ is same as $C'_1 \oplus C^s_j = S'_1 \oplus S^s_{j^*}$, where $j^* = j$ if $j \neq |c^s|-1$, and $j^* = 0$ otherwise. Under event $E3(\overrightarrow{r})$, and the fact that c'_0 is different from all r^i, we have that $S'_1 \oplus S^s_{j^*}$ is uniformly distributed.

(**b**) There exists a k, $k \in [1..z]$ such that $c'_0 = r^k$. For all other $k' \in [1..z]$, $c'_0 \neq r^k$. Thus $S' = S^k$. We have several cases:

(b1) (*truncation attempt*) If c' is a truncation of c^k, then we let $x = l'-1$ which is the index of the last block of c'.

(b2) (*extension attempt*) If c' is an extension of c^k, then we let $x = |c^k|-1$ which is the index of the last block of c^k.

(b3) Otherwise, let x be the least index in which c' and c^k are different.

In all the cases (b1), (b2) and (b3), conjunct (ii) is handled as in (a).

In case (b1), $N'_x = N^s_j$ is same as $C'_{l'-1} \oplus S^k_0 = C^s_j \oplus S^s_{j^*}$, where $j^* = j$ if $j \neq |c^s| - 1$, and $j^* = 0$ otherwise. Now, for $s = k$, $j^* = 0$ (in which case $S'_0 \oplus S^s_j$ is identically zero), we have $c'_x \oplus c^s_j = c'_{l'-1} \oplus c^k_{|c^k|-1}$. This quantity is not zero, since E0 (the deletion attempt) doesn't hold for c. Otherwise, $S'_0 \oplus S^s_{j^*} = S^k_0 \oplus S^s_j$ is uniformly distributed.

In case (b2), $N'_x = N^s_j$ is same as $C'_{|c^k|-1} \oplus S^k_{|c^k|-1} = C^s_j \oplus S^s_{j^*}$, where $j^* = j$ if $j \neq |c^s| - 1$, and $j^* = 0$ otherwise. When $s = k$, j^* is never $|c^k| - 1$, and hence $S^k_{|c^k|-1} \oplus S^s_{j^*}$ is uniformly distributed.

In case (b3), $N'_x = N^s_j$ is same as $C'_x \oplus S^k_{x^*} = C^s_j \oplus S^s_{j^*}$, where $j^* = j$ if $j \neq |c^s| - 1$, and $j^* = 0$ otherwise, and $x^* = x$ if $x \neq (l' - 1)$, and $x^* = 0$ otherwise. If $s = k$, and $j^* = x^*$, then either $j^* = x^* = 0$, or $j = x$. In the latter case, $c'_x \oplus c^s_j = c'_x \oplus c^k_x$, which is non-zero as x is the index in which c' and c^k differ. In the former case, $j = |c^k| - 1$, and $x = (l' - 1)$. In this case, $c'_x \oplus c^s_j = c'_{l'-1} \oplus c^k_{|c^k|-1}$. If this quantity is zero, then since x $(= (l' - 1))$ was the least index in which c^k and c' differed, event E0 would hold for c, leading to a contradiction. In other cases, $S^k_{x^*} \oplus S^s_{j^*}$ is uniformly distributed. □

Recall that $E3(\overrightarrow{r})$ is the event that all C^i_0 are distinct (and set to $\overrightarrow{r}$).

Claim 5: Let l_1 be the length of the first ciphertext. Let $y \leq z$. For any constant lengths l_i $(i \in [2..y])$ and constant strings c^i, $(i \in [1..y], |c^i| = l_i)$, and any function G independent of F,

$$\Pr[\bigwedge_{i \in [1..y]} C^i = c^i \ \wedge \ E2(y) \mid G \ \wedge E3(\overrightarrow{r})] \ \leq (2^{-n})^{\mu}$$

where $\mu = \Sigma_{i \in [1..y]}(l^i - 1)$.

Proof: The above probability is zero unless for all $i \in [2..y]$, $l^i = l(c^1, ..., c^{i-1})$. From now on, we will assume that the l^i are indeed such.

We do induction over y, with base case $y = 0$.

The base case is vacuously true, as $\mu = 0$ and conditional probability of TRUE is 1.

Now assume that the lemma is true for y. We prove the lemma for $y + 1$. The explanation for the inequalities is given below the sequence of inequalities.

$$\begin{aligned}
&\Pr[\bigwedge_{i \in [1..y+1]} C^i = c^i \ \wedge E2(y+1) \mid G \wedge E3(\overrightarrow{r})] \\
&\leq \Pr[C^{y+1} = c^{y+1} \mid \bigwedge_{i \in [1..y]} C^i = c^i \ \wedge E2(y+1) \ \wedge \ G \wedge E3(\overrightarrow{r})] \\
&\quad * \Pr[\bigwedge_{i \in [1..y]} C^i = c^i \ \wedge \ E2(y+1) \mid G \wedge E3(\overrightarrow{r})] \\
&\leq (2^{-n})^{l^{y+1}-1} \ * \ \Pr[\bigwedge_{i \in [1..y]} C^i = c^i \ \wedge \ E2(y) \mid G \wedge E3(\overrightarrow{r})] \\
&\leq (2^{-n})^{\Sigma_{i \in [1..y]}(l^i - 1)}
\end{aligned}$$

The second inequality follows because under the condition $E2(y+1)$, all the M_j^{y+1} are different from the previous M, and hence the sequence of variables, for all $j \in [1..L^{y+1}-1]$, $F(M_j^{y+1})$ can take all possible $(2^n)^{(L^{y+1}-1)}$ values, independently of G, and $F(M_j^{\leq y})$, and hence also all ciphertext messages till index t. Hence, the sequence $C_j^{y+1} = F(M_j^{y+1}) \oplus S_j^{y+1}$ can take all possible values. Moreover, $L^{y+1} = l(c^1, ..., c^y) = l^{y+1}$.

The last inequality follows by induction. □

Claim 6: For every fixed $\overrightarrow{r}$ with distinct values,

$$Pr[\neg E2 \mid E3(\overrightarrow{r})] < u^2 * 2^{-n}$$

Proof: Recall that Event E2 is

$$\forall i, i' \in [1..z], \forall j, j', j \in [1..L^i], j' \in [1..L^{i'}], (i,j) \neq (i',j') : \ (M_j^i \neq M_{j'}^{i'})$$

Under $E3(\overrightarrow{r})$, we have

(a) The set of variables $\{W_1^i\}$, $i \in [1..z]$, are uniformly random and independent variables.

(b) For each i, the variable W_1^i is independent of all ciphertext messages $C^{i'}$, $i' < i$, and hence all plaintext messages $P^{i'}$, $i' \leq i$. This follows because W_1^i can be chosen after $C^{i'}$, $i' < i$ have been chosen.

Given $E3(\overrightarrow{r})$, the probability that event E2 does not happen is at most $(\Sigma_{i \in [1..z]} L^i)^2 * 2^{-n}$, which is at most $u^2 * 2^{-n}$. This is seen as follows:

$$\Pr[M_j^i = M_{j'}^{i'}] \ = \ \Pr[P_j^i \oplus S_j^i = P_{j'}^{i'} \oplus S_{j'}^{i'}] \ = \ \Pr[S_j^i = S_{j'}^{i'} \oplus P_j^i \oplus P_{j'}^{i'}]$$

Without loss of generality, let $i \geq i'$. Then from (b) above it follows that this probability is at most 2^{-n} (if $i = i'$, then we also use the fact that the sequence S is pairwise differentially uniform). □

6.1 Alternate Proof Sketch

In this section we give an alternate proof of Theorem 1 which was suggested by Johan Håstad.

We first expand the notation, and generalize event E2 to E6 as follows. Given $C = c$, and $G = g$, where c is a constant sequence of ciphertexts and g is a constant permutation the M values are fixed, because $M_j^i = P_j^i \oplus S_j^i$. The variable P_j^i is completely fixed by c, and S_j^i is fixed by $g(c_0^i)$'s. We will write $M_j^i(c,g)$ for this value of M_j^i. Similarly, for N_j^i. So, for any c and g, and $y \leq z$, define $\mathrm{E6}(y,c,g)$ to be

$$\begin{aligned}&\forall i, i' \in [1..y], \forall j, j', j \in [1..l^i - 1], j' \in [1..l^{i'} - 1], (i,j) \neq (i',j') :\\ &(M_j^i(c,g) \neq M_{j'}^{i'}(c,g)) \ \wedge \ (N_j^i(c,g) \neq N_{j'}^{i'}(c,g))\end{aligned}$$

Note that $E2$ as in the previous section, and $C = c$ and $G = g$ implies $\mathrm{E6}(z,c,g)$ as F is a permutation.

In the following lemma we assume that F and G are random permutations. This is different from lemma 3 in the previous section, where we had to first assume F to be a random function, and then add the error probability.

Lemma 4. *For every constant c, and for any permutation g such that $E6(z, c, g)$,*

$$Pr[G = g|C = c \ \wedge \ E6(z, c, G)] = \frac{Pr[G = g]}{Pr[E6(z, c, G)]}$$

Proof: Let U be the universe of G. Under the condition $C = c$ and E6(z, c, G) we show that every g such that E6(z, c, g) holds, is equally likely to be G. Since c is fixed, fixing G to g, fixes the N variables to a single value (with all N's different, for otherwise E6(z, c, g) wouldn't hold). This value of the N variables is not ruled out as all the M variables are different (by E6(z, c, G)), and F is a random permutation. Thus,

$$\begin{aligned} &\mathrm{P}r[G = g|C = c \ \wedge \ E6(z, c, G)] \\ &= \frac{1}{\#g : E6(z, c, g)} \\ &= \frac{1}{|U| * \mathrm{P}r[E6(z, c, G)]} \\ &= \frac{\mathrm{P}r[G = g]}{\mathrm{P}r[E6(z, c, G)]} \end{aligned}$$

□

The proof of lemma 3 in the previous section now changes where we bound the value of

$$\Pr[S'_{x^*}(c'_0) \oplus S^s_{j^*}(c^s_0) = c'_x \oplus c^s_j \ \wedge C = c \wedge E6 \mid E3(\overrightarrow{r})]$$

This can now be written as

$$\begin{aligned} &\Pr[S'_{x^*}(c'_0) \oplus S^s_{j^*}(c^s_0) = c'_x \oplus c^s_j \mid C = c \wedge E6(z, c, G) \ \wedge \ E3(\overrightarrow{r})] \\ * &\Pr[C = c \wedge E6(z, c, G) \mid E3(\overrightarrow{r})] \end{aligned}$$

The first factor is upper bounded by $2^{-n}/\Pr[E6(z, c, G)]$ by using the above lemma (all the different cases are handled as in claim 4). From equation (1), we then get

$$\Pr[\neg(E0 \vee E1) \wedge E6 \mid E3(\overrightarrow{r})] \leq (u + v) * 2^{-n}$$

Lemma 3 then follows by proving that $\Pr[\neg E6| E3(\overrightarrow{r})] < u^2 * 2^{-n}$ as in claim 6. Rest of theorem 1 is as before, with a slightly better bound as there is no error term corresponding to assuming F and G to be random functions.

6.2 General Case

We now prove the scheme IAPM ($t \geq 1$) secure for message integrity. Here F and G are independent random permutations.

Theorem 5. *Let A be an adversary attacking the message integrity of IAPM ($t \geq 1$) with random permutations F and G. Let A make at most z queries in the first stage, totaling at most μ blocks. Let $u = \mu + z$. Let v be the maximum number of blocks in the second stage. Then for adversary A,*

$$Succ < (2 * u^2 + 2tz^2 + tm + t^2(z+1)^2 + 3t(2z+1)(u+v) + 2 + o(1)) * 2^{-n}$$

Proof Sketch: We first calculate the adversary's success probability assuming that G is a random function. Then, the error introduced in the probability because of this approximation is at most $((t(z+1))^2 * 2^{-n})$.

The differences in the proof from that of Theorem 1 are (i) we can not assume a priori, that the sequence S^i is pairwise differentially uniform, (ii) E3($\overrightarrow{r}$) as defined in Lemma 3 does not imply that S^i is independent of S^j, for $i \neq j$, (iii) in proof of Theorem 1, the case of event E0 requires S^i to be pairwise differentially uniform, and (iv) in claim 4 case (a), $S'(c'_0)$ is not necessarily independent of all $S^i(r^i)$.

To this end, Event E3 is now defined to be the event that all entries in the following (multi-) set are different:

$$\{C^i_0, i \in [1..z]\} \cup \{G(C^i_0) + j - 1, i \in [1..z], j \in [1..t-1]\}$$

For $\overrightarrow{r} = r^1, ..., r^z$, all r^i different, let E3($\overrightarrow{r}$) be the event E3 and that for all $i \in [1..z]$, $C^i_0 = r^i$.

For $\overrightarrow{r}$=$r^1, ..., r^z$, all r^i different, Pr[$\neg$ E3($\overrightarrow{r}$)] $\leq (2tz^2 + tm) * 2^{-n}$

Under event E3, for all $i \in [1..z]$, the sequence S^i is pairwise differentially uniform, and is independent of S^j ($j \in [1..z]$, $j \neq i$). Now (in Theorem 1) the case of event E0 is also handled under the condition $E3(\overrightarrow{r})$.

In Claim 4, case (a) (i.e. New IV) now requires showing that $S'(c'_0)$ (with c'_0 different from all r^i) is independent of all $S^i(r^i)$ ($i \in [1..z]$).

Consider the following events (note that $W^i_1 = G(r^i)$):

$$\textbf{Event} E4 : \forall i \in [1..z], \forall j \in [1..t-1] \ : c'_0 \neq W^i_1 + j - 1$$

$$\textbf{Event}\ \text{E5:} \forall i \in [1..z] : |G(c'_0) - W^i_1| > t \ \wedge \ |G(c'_0) - r^i| > t \ \wedge \ |G(c'_0) - c'_0| > t$$

Now given that, for all $k \in [1..z], c'_0 \neq r^k$, and under event E4, it is the case that c'_0 has never been an oracle query to G, and thus Pr[$\neg$E5 | E4 $\wedge$ E3($\overrightarrow{r}$)] $< 2t(2z+1) * 2^{-n}$. Also, Pr[$\neg$ E4 | E3($\overrightarrow{r}$)] $\leq zt * 2^{-n}$.

Under events E4, E5 and E3($\overrightarrow{r}$), and c'_0 different from all r^i, $S'(c'_0)$ is indeed independent of previous $S^i(r^i)$, and is also pairwise differentially uniform. □

6.3 Modes using GFp

We now prove theorem 1 for the IAPM scheme as in Fig 1, i.e using the mod p construction.

Note that $a = f_{K0}(r)$ translates to $a^i = G(C_0^i)$ for all $i \in [1..z]$, and $a' = G(C_0')$, under the assumption that f_{K0} is modeled as a random function (the error introduced by considering G as a random function instead of a random permutation is as before). We now predicate our whole analysis on the condition that for all $i \in [1..z]$, $G(C_0^i) < p$, and $G(C_0') < p$. The probability of this not happening is at most $(z+1) * (2^n - p)/p$.

Given this condition, it follows that for all i, a^i, and also a' are uniformly distributed in GFp (as G is a random function).

We next show that for each i, j, S_j^i is uniformly distributed in GFp.

From now on we will drop i from the superscript. We will denote by S_j^* the intermediate value after execution of the first step in the for-loop, i.e. $S_j^* = (S_{j-1} + a) \, mod \, 2^n$. Thus, if $a > S_j^*$ then $S_j = S_j^* + (2^n - p)$, else $S_j = S_j^*$.

First we prove that there is no overflow in the last step of the for-loop ($S_j^* = S_j + 159$), i.e. while adding $(2^n - p)$.

If $(S_0 =) a < (2^n - p)$, then let t be the least j such that $S_j \geq (2^n - p)$, other-wise $t = 0$. Clearly, for $j \leq t$, the condition $(a > S_j^*)$ could not have been satisfied, as $(2^n - p)$ is much smaller than 2^{n-1}.

We next show by induction that for $j \geq t$, $S_j \geq (2^n - p)$. Clearly, for $j = t$ it is true by definition of t. If for some $j > t$, $(a \leq S_j^*)$, then $S_j = S_{j-1} + a$, hence by induction $S_j \geq (2^n - p)$. If for some $j > t$, $(a > S_j^*)$, then $S_j^* = S_{j-1} + a - 2^n$, which is less than p, as $a < p$ by design. Thus, there is no overflow while adding $(2^n - p)$, and hence $S_j > (2^n - p)$.

Claim 7: For every i, j, S_j^i is uniformly distributed in GFp.
Proof: Indeed, $S_j^i = a^i * (j+1) \, mod \, p$. Clearly, this is true for $j = 0$. Suppose it is true for $j - 1$, then we show that $S_j^i = a^i * (j+1) \, mod \, p$. Now, $(a > S_j)$ holds iff $(S_{j-1}^i + a^i) \geq 2^n$. So, suppose $(S_{j-1}^i + a^i) < 2^n$, then $S_j^i = S_{j-1}^i + a^i$, and hence $S_j^i = a^i * (j+1) \, mod \, p$, by induction. If $(S_{j-1}^i + a^i) \geq 2^n$ then, $S_j^i = (S_{j-1}^i + a^i) - 2^n + (2^n - p)$, since there is no overflow while adding $(2^n - p)$, and the claim follows. □

Claim 8: For each i, the sequence S_j^i is pairwise-differentially uniform in GFp.
Proof: Since, $S_j^i = a^i * (j+1) \, mod \, p$, and $S_{j'}^i = a^i * (j'+1) \, mod \, p$, $S_j^i - S_{j'}^i = a^i * (j - j') \, mod \, p$, and hence the claim follows. □

Claim 9: For any constant $c \in [0..2^n - 1]$, $\Pr[S_i - S_j = c \, mod \, 2^n] \leq 2/p$.
Proof:

Note that $S_i - S_j = c \, mod \, 2^n$ and $S_i \geq S_j$ implies $S_i - S_j = c \, mod \, p$. On the other hand, $S_i - S_j = c \, mod \, 2^n$ and $S_i < S_j$ implies $S_i - S_j = c - 2^n$, and hence $S_i - S_j = c - 2^n \, mod \, p$.

Thus,

$$Pr[S_i - S_j = c \; mod \; 2^n]$$

$$\begin{aligned}&= \Pr[S_i - S_j = c \ mod\ 2^n \ \wedge \ S_i \geq S_j] + \Pr[S_i - S_j = c \ mod\ 2^n \ \wedge \ S_i < S_j] \\ &\leq \Pr[S_i - S_j = c \ mod\ p] + \Pr[S_i - S_j = c - 2^n \ mod\ p] \\ &\leq 2/p\end{aligned}$$

where the last inequality follows by the previous claim. □

For modes of practical interest, the term $(z+1)*O(n)$ in the following theorem is really $(z+1)*2n$. For example, for 128 bit block ciphers, since $p = 2^{128} - 159$, this term is $(z+1)*159$.

Theorem 6. *Let A be an adversary attacking the message integrity of IAPM ($t = 1$) with the GFp construction (fig 1), with random permutations F and G. Let A make at most z queries in the first stage, totaling at most μ blocks. Let $u = \mu + z$. Let v be the maximum number of blocks in the second stage. Then for adversary A,*

$$Succ < (2*u^2 + z^2 + (z+1)^2 + u + v + 2 + o(1) + (z+1)*O(n))*2^{-n}$$

Proof: The proof is the same as the proof of theorem 1 except for a few differences. Firstly, as said earlier we predicate on the condition that for all $i \in [1..z]$, $G(C_0^i) \geq p$, and $G(C_0') \geq p$. The probability of this not happening is at most $(z+1)*(2^n - p)/p$, and that is an extra additive factor in the adversary's success probability.

We will use the following notation: $(X)_y$ will stand for X reduced modulo y, i.e. $(X)_y$ is the unique number in $[0..y-1]$ such that $X = (X)_y \ mod\, y$. Next in the proof of theorem 1, the case where E0 happens, now becomes (the big summations are xor-sums)

$$\begin{aligned}&(\sum_{j=1}^{L'-1} P_j' = 0) \ \wedge \mathrm{E0} \\ &\Rightarrow \sum_{j=1}^{L'-2} (P_j^i) \oplus (M_{L^i-1}^i - S_{L'-1}^i)_{2^n} = 0 \\ &\equiv \sum_{j=1}^{L'-2} (P_j^i) \oplus (\sum_{j=1}^{L^i-2} (P_j^i) + S_{L^i-1}^i - S_{L'-1}^i)_{2^n} = 0 \\ &\equiv (S_{L^i-1}^i - S_{L'-1}^i = \sum_{j=1}^{L'-2} (P_j^i) - \sum_{j=1}^{L^i-2} (P_j^i))\ mod\, 2^n\end{aligned}$$

This event happens with probability at most $2/p$ by claim 9.

Similarly lemma 2 now modifies as follows: $\sum_{j=1}^{L'-1} P_j' = 0$ iff

$$(M_x' - S_x')_{2^n} = \sum_{j=1, j \neq x}^{L'-1} (M_j' - S_j')_{2^n}$$

or

$$(M'_x = S'_x + \sum_{j=1, j\neq x}^{L'-1} (M'_j - S'_j)_{2^n})\, mod\, 2^n$$

The probability in lemma 2 remains as before.

In lemma 3, for each s, j, we now have $(N'_x = N^s_j)$ iff $(S'_{x^*} - S^s_{j^*}) = (C'_x - C^s_j)\, mod\, 2^n$, and thus by Claim 9 the probability bounds in claim 4 are in terms of $2/p$ instead of 2^{-n}. Similarly, the bound in claim 6 is now $u^2 * 2/p$.

Thus,

$$Succ < (u^2+z^2+(z+1)^2+1+o(1))*2^{-n}+(u^2+u+v+1)*2/p+(z+1)*(2^n-p)/p$$

However, since $2^n-p \approx n$, or $2^n-p = O(n)$, we have that $1/p < 2^{-n}+2n*2^{-2n}$. Thus replacing 2^{-n} by $1/p$ only adds a second order term to adversary's success probability.

□

7 Message Secrecy

We now prove security in the find-then-guess model, which implies that the IAPM scheme (both for fig 1 and fig 2) is secure for message secrecy.

Theorem 7. *Let A be a chosen plaintext attack adversary of the encryption scheme IAPM with random permutations F and G, making at most z queries, these totaling at most u blocks. Then*

$$Adv_A \leq (3u^2/2 + z^2) \cdot \frac{1}{2^n}$$

Proof:

We will calculate the probability of the adversary's success under the assumption that F and G are random functions. A standard argument shows that the error introduced in calculating the probability is at most $(u^2 + z^2) * 2^{-n-1}$.

As in the previous theorem, we will use subscripts to denote particular blocks in a message. We will use constants c^i, c', d^i to denote strings of arbitrary block length. Let the z queries be divided into p queries in the first phase, one query in the "choose" phase, and p' queries in the second phase. Thus $z = p + 1 + p'$. We will use c to denote the sequence $c^1, ..., c^z$.

Let $l()$ be the length of the first ciphertext (determined by the adversary). The function $|\cdot|$ is used below to represent length of a message in blocks. Given a sequence of ciphertext messages $c^1, ..., c^i$, $i \leq z$, let $l(c^1, ..., c^i)$ be the length of the $(i+1)$th ciphertext (which is determined by the adversary, and therefore is a deterministic function of $c^1, ...c^i$).

As in lemma 3, we consider the event E2, under which all the M variables are different. Similarly, we also predicate on the event that all the initial variables are different (event E3). Recall that the event E2(y) is that all the variables in the following multi-set are different:

$$\{M^i_j, i \in [1..y], j \in [1..L^i - 1]\}$$

Event E2(z) is also written as just E2. The event E3 now requires that all initial variables are different:

$$\{C_0^i, i \in [1..p]\} \cup \{C_0'\} \cup \{C_0^i, i \in [p+2..p+1+p']\}$$

Note that C^{p+1} is another name for C'.
We have,

$$\Pr[A(R,C) = b \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,] \ = \sum_{c^1:\,|c^1|=l()} \ldots \sum_{c^i:\,|c^i|=l(c^{i-1},\ldots,c^1)} \ldots$$
$$\ldots \sum_{c^z:\,|c^z|=l(c^{m-1},\ldots,c^1)} \Pr[A(R,C) = b \ \wedge C = c \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,]$$

If for some i, $c_0^i \neq r^i$, then the inside expression is zero.
The inside expression can be written as

$$\begin{aligned} & \Pr[A(R,C) = b \ \wedge C = c \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,] \\ & = \Pr[A(R,c) = 0 \ \wedge \ \bigwedge_{i \in [1..z]} C^i = c^i \ : \ \wedge b = 0 \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,] + \\ & \quad \Pr[A(R,c) = 1 \ \wedge \ \bigwedge_{i \in [1..z]} C^i = c^i \ : \ \wedge b = 1 \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,] \end{aligned}$$

where when $b = 0$, $C^{p+1} = C'$ is the encryption of P'^0, and when $b = 1$ it is the encryption of P'^1. Let's concentrate on the first summand.

$$\begin{aligned} & \Pr[A(R,c) = 0 \ \wedge \ \bigwedge_{i \in [1..z]} C^i = c^i \ \wedge b = 0 \ \wedge \ E2 \mid E3(\overrightarrow{r}) \,] \\ & = \Pr[A(R,c) = 0 \mid \bigwedge_{i \in [1..z]} C^i = c^i \ \wedge b = 0 \ \wedge \ E2 \ \wedge E3(\overrightarrow{r}) \,] * \\ & \quad \Pr[\bigwedge_{i \in [1..z]} C^i = c^i \ \wedge \ E2 \mid b = 0 \wedge E3(\overrightarrow{r}) \,] * Pr[b = 0] \\ & = \Pr[A(R,c) = 0] * \Pr[\bigwedge_{i \in [1..z]} C^i = c^i \ \wedge \ E2 \mid b = 0 \wedge E3(\overrightarrow{r}) \,] * Pr[b = 0] \end{aligned}$$

This quantity is upper bounded by

$$\frac{1}{2} * (2^{-n})^{\mu} * \Pr[A(R,c) = 0]$$

by Claim 5, and lower bounded by

$$\frac{1}{2} * (1 - \mu(\mu-1)/2 * 2^{-n}) * (2^{-n})^{\mu} * \Pr[A(R,c) = 0]$$

by Claim 10 below, where $\mu = \Sigma_{i \in [1..z]}(l^i - 1)$. Note that, both Claim 5 and Claim 10 hold regardless of whether $b = 0$ or $b = 1$.

Thus,

$$\frac{1}{2}*(1-\mu(\mu-1)/2*2^{-n})*(2^{-n})^{\mu} \le \Pr[A(R,C)=b \wedge C=c \wedge E2 \mid E3(\overrightarrow{r})] \le \frac{1}{2}*(2^{-n})^{\mu}$$

and hence,

$$\frac{1}{2}*(1-\mu(\mu-1)/2*2^{-n}) \le \mathrm{P}r[A(R,C)=b \wedge E2 \mid E3(\overrightarrow{r})] \le \frac{1}{2}$$

Thus by Claim 6, and $\Pr[\neg\ \mathrm{E3}] \le z^2 * 2^{-n-1}$, we have

$$|\Pr[A(R,C)=b]-\frac{1}{2}| \le (u^2+z^2/2)*2^{-n}$$

□

Claim 10: Let l_1 be the length of the first ciphertext. Let $y \le z$, and $j \in [0,1]$. For any constant lengths l_i $(i \in [2..y])$ and constant strings c^i, $(i \in [1..y]$, $|c^i| = l_i)$, such that for all $i \in [2..y]$, $l^i = l(c^1,...,c^{i-1})$,

$$\mathrm{P}r[\bigwedge_{i\in[1..y]} C^i = c^i \ \wedge\ E2(y)\ |b=j \wedge E3(\overrightarrow{r})]\ \ge (1-\mu(\mu-1)/2*2^{-n})*(2^{-n})^{\mu}$$

where $\mu = \Sigma_{i\in[1..y]}(l^i-1)$.
Proof:

We do induction over y, with base case $y=0$.
The base case is vacuously true, as $\mu = 0$ and conditional probability of TRUE is 1.
Now assume that the lemma is true for y. We prove the lemma for $y+1$. The explanation for the inequalities is given below the sequence of inequalities.

$$\begin{aligned}
&\mathrm{P}r[\bigwedge_{i\in[1..y+1]} C^i = c^i \ \wedge E2(y+1) \mid b=j \wedge E3(\overrightarrow{r})] \\
&= \mathrm{P}r[C^{y+1}=c^{y+1} \mid \bigwedge_{i\in[1..y]} C^i = c^i \ \wedge E2(y+1)\ \wedge\ b=j \wedge E3(\overrightarrow{r})] \\
&\quad * \mathrm{P}r[\bigwedge_{i\in[1..y]} C^i = c^i \ \wedge\ E2(y+1) \mid b=j \wedge E3(\overrightarrow{r})] \\
&= (2^{-n})^{l^{y+1}-1}\ *\ \mathrm{P}r[\bigwedge_{i\in[1..y]} C^i = c^i \ \wedge\ E2(y) \mid b=j \wedge E3(\overrightarrow{r})] \\
&\quad * \mathrm{P}r[E2(y+1) \mid \bigwedge_{i\in[1..y]} C^i = c^i \ \wedge\ E2(y) \wedge b=j \wedge E3(\overrightarrow{r})] \\
&\ge (2^{-n})^{l^{y+1}-1}\ * \mathrm{P}r[\bigwedge_{i\in[1..y]} C^i = c^i \ \wedge\ E2(y) \mid b=j \wedge E3(\overrightarrow{r})] \\
&\quad * (1-[(l^{y+1}-1)(l^{y+1}-2)/2+(l^{y+1}-1)*(\Sigma_{i\in[1..y]}(l^i-1))]*2^{-n})
\end{aligned}$$

and the claim follows by induction. The last inequality is seen as follows. Given the ciphertexts upto C^y, the plaintexts upto P^{y+1} are fixed. Also, given E2(y), E2($y+1$) is just the M values in message $y+1$ being different from each other and also different from all earlier M values. Given that S are pair-wise differentially uniform, the bound then follows by upper-bounding ¬E2($y+1$).

The probability of $C^{y+1} = c^{y+1}$ is calculated as in Claim 5. □

8 Acknowledgments

The author is extremely grateful to Shai Halevi and Pankaj Rohatgi for help with the proof of message integrity. The author also thanks J. Håstad for suggesting an alternative proof of message integrity.

References

1. ANSI X3.106, "American National Standard for Information Systems - Data Encryption Algorithm - Modes of Operation", American National Standards Institute, 1983.
2. M. Bellare, A. Desai, E. Jokiph, P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES Modes of OPeration", 38th IEEE FOCS, 1997
3. M. Bellare, J. Kilian, P. Rogaway, "The Security of Cipher Block Chaining", CRYPTO 94, LNCS 839, 1994
4. M. Bellare, C. Namprempre, "Authenticated Encryption: Relations among notions and analysis of the generic composition paradigm", Proc. Asiacrypt 2000, T. Okamoto ed., Springer Verlag 2000
5. Hugo Krawczyk, "LFSR-based Hashing and Authentication", Proc. Crypto 94. LNCS 839, 1994
6. ISO 8372, " Information processing - Modes of operation for a 64-bit block cipher algorithm", International Organization for Standardization, Geneva, Switzerland, 1987
7. ISO/IEC 9797, "Data cryptographic techniques - Data integrity mechanism using a cryptographic check function employing a block cipher algorithm", 1989
8. M. Luby, "Pseudorandomness and Cryptographic Applications", Princeton Computer Science Notes, Princeton Univ. Press, 1996
9. C.H. Meyer, S. M. Matyas, "Cryptography: A New Dimension in Computer Data Security", John Wiley and Sons, New York, 1982
10. National Bureau of Standards, NBS FIPS PUB 81, "DES modes of operation", U.S. Department of Commerce, 1980.
11. National Bureau of Standards, Data Encryption Standard, U.S. Department of Commerce, FIPS 46 (1977)
12. RFC 1510,"The Kerberos network authentication service (V5)", J. Kohl and B.C. Neuman, Sept 1993
13. Security Architecture for the Internet Protocol, RFC 2401, http://www.ietf.org/rfc/rfc2401.txt
14. The TLS Protocol, RFC2246, http://www.ietf.org/rfc/rfc2246.txt

This article was processed using the LaTeX macro package with LLNCS style