An observation regarding Jutla's modes of operation

Shai Halevi[*]

April 2, 2001

Abstract

Recently, Jutla suggested two new modes of operation for block ciphers. These modes build on traditional CBC and ECB modes, respectively, but add to them masking of the outputs and inputs. Jutla proved that these masking operations considerably strengthen CBC and ECB modes. In particular, together with a simple checksum, the modified modes ensure not only confidentiality, but also authenticity. Similar modes were also suggested by Gligor and Donescu and by Rogaway.

In Jutla's proposal (as well as in some of the other proposals), the masks themselves are derived from an IV via the same block cipher as used for the encryption (perhaps with a different key). In this work we note, however, that the function for deriving these masks need not be cryptographic at all. In particular, we prove that a universal hash function (a-la-Carter-Wegman) is sufficient for this purpose.

[*]IBM T. J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, USA,shaih@watson.ibm.com

1 Introduction

The area of modes of operations for block ciphers received much attention lately, partly due to an announcement by NIST that they are considering an update to their list of standardized modes. As part of this work, some new modes of operations were suggested, that have practical advantages over known modes such as CBC and ECB.

The purpose of some of these new modes is to ensure – in addition to secrecy – also the integrity of the encrypted string. Although there have been many proposals on how to do that in the past, they all either turned out to be flawed, or they required a computation of a MAC, separately from the encryption process. Recently, however, Jutla observed that with the addition of a simple checksum *and proper masking*, CBC and ECB modes can provide both secrecy and authentication [3]. Jutla proposed two new modes, called IACBC and IAPM, which are based on CBC and ECB modes, respectively.

Roughly speaking, these modes work as follows: When encrypting an L-block plaintext $P = P_1 \ldots P_L$, we first attach at the end of the message a simple checksum of all these blocks, $P_{L+1} = \sum_j P_j$. Then, we pick a new IV (which was never used before with the current key), and from this IV, we derive "random looking" masks $S_0, S_1 \ldots S_{L+1}$. For the IACBC mode, we now apply CBC encryption to the plaintext message (with the extra P_{L+1} block), and simply mask the $L+1$ output blocks by the masks $S_1, \ldots, S_L, S_0$ before outputting them as the ciphertext. For the IAPM mode we first mask the plaintext blocks using masks $S_1, \ldots, S_L, S_{L+1}$, then encrypt the result in ECB mode, and then mask again with $S_1, \ldots, S_L, S_0$ before outputting the ciphertext. Similar modes were also suggested by Gligor and Donescu (XCBC and XECB modes [2]) and by Rogaway (OCB mode [4]). Below we refer to such modes as "masked CBC" and "masked ECB" modes.

The "magic" of these new modes of operations is in the way they generate the masks S_j. Jutla observed that these masks need not be fully pseudorandom (and therefore can potentially be generated much faster than L applications of the underlying block cipher). Roughly speaking again, the security analyses of these modes rely on the following two properties of the masks:

(a) The masks that are used in different encryptions are independent.[1]
(b) Within each encryption, the masks that are used for different blocks are *pairwise independent*.[2]

The difference between full independence and pairwise independence is very important here. To get full independence, we must use the underlying block cipher, whereas to get pairwise-independence it is sufficient to use combinatorial (*non-cryptographic*) techniques. These modes of operation typically prescribe that the masks are generated by first "encrypting the IV" to get some pseudorandom values, and then using these pseudorandom values in a simple (and fast) combinatorial construction.

In this work we observe that Property (a) from above is not really needed. Namely, it is sufficient that all the masks be only "pairwise-independent". Hence, there is no need to use the block cipher at all in the generation of the masks.

1.1 Organization

In this note we only consider "masked ECB" modes. (The treatment for "masked CBC" modes is similar.) In Section 2 we define (our abstraction of) Jutla's IAPM mode of operation. We also

[1] More precisely, their distribution is indistinguishable from independent masks
[2] The property that is needed here is actually a bit weaker than pairwise independence. See details below.

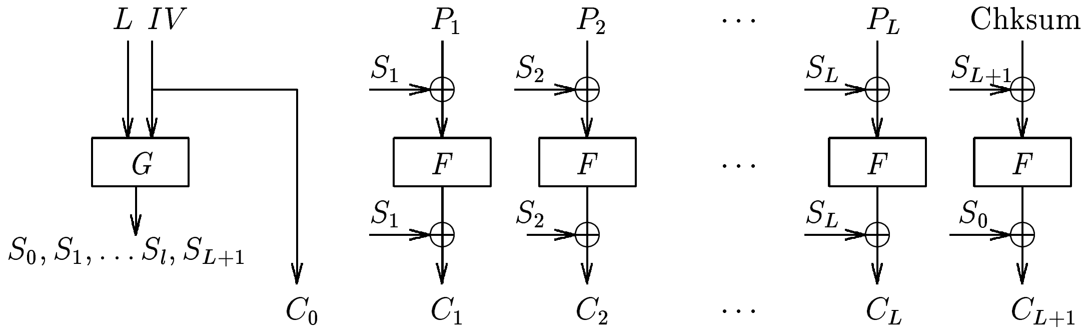

Figure 1: Jutla's "masked-ECB" mode of operation

re-cap the security definition that we seek to prove, define the property that we need from the masks, and briefly discuss some methods for implementing the mask generation. The technical "meat" of this note is in Section 3, where we prove security of this mode of operation, based on the abovementioned property of the masks. In Section 4 we discuss some variants of this mode of operation, and elaborate on the applicability of the analysis in this note to those variants.

2 The masked ECB modes of operation

The mode that we analyze in this note is almost identical to Jutla's IAPM, except that we abstracted away the specifics of the mask generation. The notations that we use are also similar to those used by Jutla [3].

For this mode, it is postulated that the encryptor and decryptor share an invertible pseudorandom permutation over $\{0,1\}^n$ (i.e., an n-bit block cipher), as well as a mask-generation function. The mask-generation function takes as input an n-bit IV string, and the length L of the plaintext, and generates $L+2$ n-bit masks. Below we denote the shared permutation by $F : \{0,1\}^n \to \{0,1\}^n$, and the mask-generating function by $G : \mathcal{N} \times \{0,1\}^n \to (\{0,1\}^n)^*$. (We assume for the most part that $G(L, IV)$ is independent of L.)

Below we use $\oplus$ to denote bit-wise exclusive-or between two n-bit strings, and when we write $\sum_j P_j$ we means the exclusive-or of the P_j's.

To encrypt an L-block plaintext $P = P_1 \ldots P_L$ (with $P_j \in \{0,1\}^n$), the encryptor first picks a new nonce, which we denote by IV. It is important that this nonce was never used before by the encryptor with the same F, G (and that it is chosen independently of F, G). The encryptor uses G to compute $\langle S_0, S_1, \ldots, S_{L+1}\rangle \leftarrow G(L, IV)$. The ciphertext $C = C_0, \ldots, C_{L+1}$ is computed by setting $C_0 = IV$, $C_j = S_j \oplus F(S_j \oplus P_j)$ for $i = 1 \ldots L$, and $C_{L+1} = S_0 \oplus F(S_{L+1} \oplus \sum_{i=1}^{L} P_j)$. This mode is depicted in Figure 1.

To decrypt a ciphertext of $L+2$ blocks, $C = C_0, C_1, \ldots, C_{L+1}$, the decryptor computes the masks $\langle S_0, S_1, \ldots, S_{L+1}\rangle \leftarrow G(L, C_0)$, then it recovers $P_j = S_j \oplus F^{-1}(S_j \oplus C_j)$ for $i = 1 \ldots L$, and $P_{L+1} = S_{L+1} \oplus F^{-1}(S_0 \oplus C_{L+1})$, and verifies that $P_{L+1} = \sum_{i=1}^{L} P_j$. If the check passes, the plaintext is $P_1 \ldots P_L$. Otherwise, the ciphertext is deemed invalid.

2.1 Security

Security for this mode of operation is defined as a combination of the properties *indistinguishability under chosen-plaintext attacks* and *integrity of ciphertext*. Here we only provide a brief overview of these notions. We refer the reader to, e.g., [1] for a formal presentation of these definitions (as well as a proof that together they imply indistinguishability under chosen-message-and-ciphertext attacks).

Secrecy. The secrecy requirement is the (usual) left-or-right notion. An adversary is given access to an "encryption oracle", that accepts pairs of plaintexts (both of the same length), and encrypts one of them. Either always the first plaintext, or always the second. The adversary is successful if it can guess which is the case, and the encryption is considered secure if feasible adversaries have only negligible advantage in guessing correctly (vs. a random guess).

Authenticity. An adversary is given access to an encryption oracle. It can query that oracle as much as it wants, and at the end it needs to produce a ciphertext that is different than all the ones produces by the oracle. In the sequel we call this ciphertext the "forged ciphertext". The adversary is successful if the "forged ciphertext" is valid, and the scheme is secure if any feasible adversary only has a negligible success probability.

2.2 The mask-generation function

The property we need is essentially just the usual notion of an *ϵ-xor-universal function* (see, e.g. [5], where this name was coined). The only difference is that our syntax is slightly different. We need another piece of notation here. For $j = 0, 1, \ldots$, we denote by $G_j(L, IV)$ the j'th mask generated by $G(L, IV)$. G_j is undefined when $j > L + 1$. Below we usually assume that $G_j(L, IV)$ is independent of L. That is, $G_j(L, IV) = G_j(L', IV)$, provided that both are defined. We call such function "L-independent". (We only need L for the "additional property" below.)

Definition 1 (Xor-Universal Functions) *Let n be an integer and let ϵ be some some real number $\epsilon \in [0, 1]$. A distribution over "L-independent" functions $G : \mathcal{N} \times \{0,1\}^n \to (\{0,1\}^n)^*$ is said to be ϵ-xor-universal if for any fixed $\Delta \in \{0,1\}^n$, and any two fixed tuples $(j, L, IV), (j', L', IV')$, s.t. $j \leq L + 1$, $j' \leq L' + 1$ and $(j, IV) \neq (j', IV')$,*

$$\Pr_G \left[G_{j'}(L', IV') \oplus G_j(L, IV) = \Delta \right] \leq \epsilon$$

Remark: an additional property. The security proof in Section 3 can be somewhat simplified, if we assume that the function $G_0(\cdots)$ is not "L-independent", but rather it is also xor-universal with respect to the L's. Namely, we still require that G_j is "L-independent" for $j > 0$, but for $j = 0$ we have the requirement that for any two fixed pairs $(L, IV) \neq (L', IV')$ we have $\Pr\left[G_0(L', IV') \oplus G_0(L, IV) = \Delta\right] \leq \epsilon$. During the proof, we mark by footnotes the places where this extra requirement can be used for simplification.

Implementing the mask-generation function. In many of the modes suggested in [3, 2, 4], the masks where generated more or less according to the following recipe: the encryptor and decryptor share a second (pseudo)random function F'. Then, on a given IV, they compute $r = F'(IV)$, and use r to form an xor-universal sequence. (Say, by setting $S_j = (j+1) \cdot r$ where the multiplication occurs in some field, and $(j+1)$ is viewed as some non-zero element in that field. In Rogaway's scheme [4] it is even suggested to set $S_j = r + (j+1) \cdot s$, where s can be thought of as part of the key.)

Such methods achieve a property similar to Definition 1, but in fact, they achieve much more than that. If F' is (pseudo)random, then the value $G_j(L^i, IV^i)$ is (pseudo)independent of all the sequences $G(L^{i'}, IV^{i'})$ for $i' \neq i$. Indeed, the security arguments in the abovementioned works take advantage of this stronger property of G. As it turns out, however, the property in Definition 1 is sufficient.

One way to get only our weaker property, is to use the following method, that does not involve cryptographic functions: Let $\bar{L}$ be an upper bound on the ciphertext length that can be handled by the encryption algorithm (say, $\bar{L} = 2^n$, so the plaintext can have at most $2^n - 2$ blocks). The parties share a random $n \times (\lceil \log \bar{L} \rceil + n)$ boolean matrix M, and the mask $S_j = G_j(L, IV)$ is computed as $S_j = M \cdot (j, IV)$, where (j, IV) is viewed as a boolean vector of length $\lceil \log \bar{L} \rceil + n$.

To get the additional property from above, we can slightly modify this construction: the matrix M is now of dimension $n \times (\lceil \log \bar{L} \rceil + 1 + n)$, masks S_j, $j = 1 \dots L+1$ are computed as $S_j = M \cdot (2j, IV)$, and the mask S_0 is computed as $S_0 = M \cdot (2L+1, IV)$.

3 Analysis

3.1 Integrity of ciphertexts

The structure of the analysis below generally follows the structure of Jutla's proof [3]. We also use similar notations. Throughout the analysis we sometimes denote some quantities by lowercase English letters, when we want to stress that these quantities are fixed, rather that being random variables. Otherwise we denote everything by uppercase English letters. In the analysis below, we assume that the permutation F is chosen uniformly at random from the space of all permutations over $\{0,1\}^n$. The transformation to pseudorandom permutation is standard. We also assume that the mask-generating function G is chosen independently of F, from an ϵ-xor-universal distribution, and that the IV's are all distinct and independent of F, G.

Fix an adversary A, and assume that A is deterministic (or else, fix also the randomness that A uses). Let m be an upper bound on the number of queries that A asks the encryption oracle, let u be an upper bound on the number of of blocks in all the ciphertexts that the encryption oracle returns (*not counting the IV's*). Let v be an upper bound on the number of blocks in the "forged ciphertext" (again, not counting the IV). To somewhat reduce notations, we assume (w.l.o.g.) that A always asks exactly m queries, whose answers total exactly u blocks. (This means that the total number of plaintext blocks in A's queries is exactly $u - m$, since each ciphertext has one block more than its plaintext, other than the IV.)

Theorem 2 *If F is a random permutation and G is an ϵ-xor-universal function, independent of F, then for any adversary A that asks exactly m plaintext queries, totalling exactly $u - m$ blocks, and produces a "forged ciphertext" of length at most v blocks, it holds that*

$$\Pr[A \textit{ produces a valid "forged ciphertext"}] < 2^{-n}\binom{u}{2} + \epsilon\left(\binom{u}{2} + u + v\right) + \frac{1}{2^n - u - v}$$

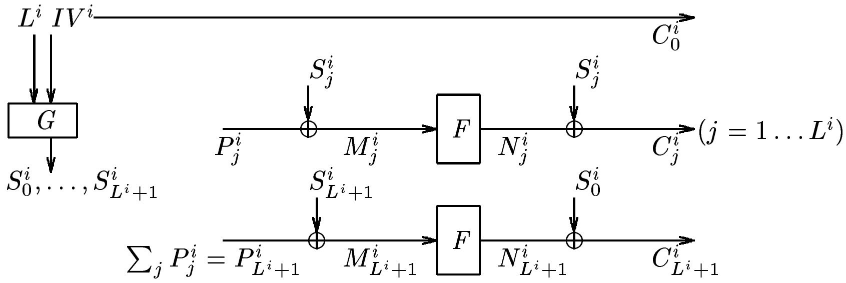

Figure 2: Notations used in the analysis

The probability is taken over the choice of F, G and the IV's.

Proof: We use the following notations: The length of the i'th plaintext query to the encryption oracle is denoted L^i, and the query itself is denoted $P^i = P_1^i \ldots P_{L^i}^i$. The checksum block is denoted $P_{L^i+1}^i = \sum_{j=1}^{L^i} P_j^i$. The IV in the i'th encryption is denoted by IV^i, the masks that are used in this encryption are denoted $S_0^i \ldots S_{L^i+1}^i$, with $S_j^i = G_j(L, IV^i)$. The masked blocks (to which ECB encryption is applied) are denoted $M_1^i \ldots M_{L^i+1}^i$, where $M_j^i = P_j^i \oplus S_j^i$. The blocks after the ECB encryption are denoted $N_1^i \ldots N_{L^i+1}^i$, where $N_j^i = F(M_j^i)$. Finally the ciphertext is denoted $C^i = C_0^i \ldots C_{L^i+1}^i$, where $C_0^i = IV^i$, $C_j^i = N_j^i \oplus S_j^i$ for $j = 1 \ldots L^i$, and $C_{L^i+1}^i = N_{L^i+1}^i \oplus N_0^i$. These notations are depicted in Figure 2.

Similarly, we denote the "forged ciphertext" by $C' = C'_0 \ldots C'_{L'+1}$, and the implied IV is denoted $IV' = C'_0$. The masks are denoted $S'_j = G_j(L', IV')$, the inputs to F^{-1} are $N'_j = C'_j \oplus S'_j$ for $j = 1 \ldots L'$ and $N'_{L'+1} = C'_{L'+1} \oplus S'_0$. The result of applying F^{-1} is denoted $M'_j = F^{-1}(N'_j)$, and the corresponding "plaintext blocks" are denoted $P'_j = M'_j \oplus S'_j$. Recall that the goal of the adversary here is to generate a ciphertext C' that is different from all the C^i's, and yet $P'_{L'+1} = \sum_{j=1}^{L'} P'_j$.

Let $\vec{c}$ be any fixed vector of u n-bit blocks, and let $\vec{iv}$ be a fixed vector of m IV's. We identify with the pair $\langle \vec{iv}, \vec{c} \rangle$ a run of the adversary A. In this run, A produces the plaintexts $P^1 \ldots P^m$, totalling $u - m$ blocks. For each L^i-block plaintext P^i, it gets back a ciphertext C^i with the IV being the next block from $\vec{iv}$, and the rest of the ciphertext being the next $L^i + 1$ blocks from $\vec{c}$. This run induces a unique parsing of $\vec{c}$ into m ciphertexts, and it also uniquely defines the corresponding vector of P^i's.[3] Notice that since A is deterministic, then P^i is a deterministic function of $C^1 \ldots C^{i-1}$.

Proof overview. The proof proceeds in two steps. Roughly speaking, in Lemma 3 we show that with overwhelming probability (over the choice of F and G), there exists at least one block N'_x which does not appear anywhere else. That is, $N'_x \neq N_j^i$ for all $i = 1 \ldots m,\ j = 1 \ldots L^i + 1$, and $N'_x \neq N'_j$ for all $j = 1 \ldots L' + 1,\ j \neq x$. This is because the mask generation function G is ϵ-xor-universal: To get $N'_x = N_j^i$ we must have $S_j^i \oplus S'_x = C_j^i \oplus C'_x$, which happens with probability at most ϵ. The formal proof below requires some care to deal with the adaptiveness of the adversary, and is otherwise just a case analysis, using this fact about the masks.

Call a block N'_x as above a *unique block*. In Lemma 6 we prove that if there exists such unique block,

[3]Potentially, this run may not be consistent with any choice of G, F. I.e., it could be that there is no such choice that for each of the P^i returns the corresponding C^i. In the analysis below we show that this does not happen, but for now we ignore this point.

then the probability of getting a valid ciphertext (i.e., $P'_{L'+1} = \sum_{j \le L'} P'_j$) is very low. Roughly, this is because there is only one value of $M'_x = F^{-1}(N'_x)$ that will make the ciphertext valid, but since N'_x does not appear anywhere else, then $F^{-1}(N'_x)$ can assume almost any value.

A CAVEAT. The above line of proof has a problem, though. In fact, there is a strategy for the adversary to ensure that there is no unique block. Specifically, this happens when the ciphertext C' is obtained by taking a previous ciphertext C^i, and removing from it all the blocks $C^i_{r+1}, \dots C^i_{L^i}$ for some r. That is, $C' = C^i_0, \dots, C^i_r, C^i_{L^i+1}$. It is not hard to show that in this case, all the N'_j's already appeared as N^i_j's (see Subcase (b4) in the proof of Claim 4(iii) below). However, it is also not hard to see that the probability of getting a valid ciphertext in this case is very low (see proofs of Claim 4(ii) and Lemma 6).

Thus, we have to somewhat modify the proof structure. In Lemma 3 we show that with overwhelming probability, one of two events happens: either C' is of the form above but the "forged plaintext" is invalid, or else there exists a unique block N'_x. In Lemma 6 we show that in the latter case, the probability of getting a valid ciphertext is very low.[4]

More notations. We denote by Succ the event in which the adversary is successful. We also denote by E0 the event in which C' is obtained as a truncation of some C^i as described above, and by E1 the event in which there is a unique block. By E2 we denote the event in which all the M^i_j's (i.e., the inputs to F) are distinct, and by E3 we denote a sub-event of E0, which roughly corresponds to the "forged ciphertext" being invalid. Formally, we have:

Succ: $C' \neq C^i$ for all $i \in \{1 \dots m\}$, and yet $P'_{L'+1} = \sum_{j=1}^{L'} P'_j$

Event E0: $\exists\, i \in \{1 \dots m\}, r \in \{0, \dots, L^i - 1\}$, s.t. $C' = C^i_0, \dots, C^i_r, C^i_{L^i+1}$

Event E1: $\exists\, x \in \{1 \dots L'+1\}$, s.t.
(a) $\forall\, i \in \{1 \dots m\},\ j \in \{1 \dots L^i + 1\},\ N'_x \neq N^i_j$, and
(b) $\forall j \in \{1 \dots L'+1\}$ s.t., $j \neq x,\quad N'_x \neq N'_j$

Event E2: $\forall i, i' \in \{1 \dots m\},\ j \in \{1 \dots L^i + 1\},\ j' \in \{1 \dots L^{i'} + 1\}$,
if $(i,j) \neq (i',j')$ then $M^i_j \neq M^{i'}_{j'}$

Event E3: $\exists\, i \in \{1 \dots m\}, r \in \{0, \dots, L^i - 1\}$, s.t. $C' = C^i_0, \dots, C^i_r, C^i_{L^i+1}$
but for these i, r, we have $S^i_{r+1} \oplus S^i_{L^i+1} \neq \left(\sum_{j=1}^{r} P^i_j\right) \oplus P^i_{L^i+1}$

We recall that any two fixed vectors $\langle \vec{iv}, \vec{c} \rangle$ define a run of the adversary A, which in turn induces a parsing of $\vec{c}$ into m ciphertexts, the corresponding m plaintexts (totalling $m-u$ blocks) and the "forged ciphertext" C'. To stress the fact that some quantities are uniquely determined by $\langle \vec{iv}, \vec{c} \rangle$, we sometimes denote them as a function of these vectors. For example, $C'(\vec{iv}, \vec{c})$ if the "forged ciphertext" that is induced by $\langle \vec{iv}, \vec{c} \rangle$, $P^i_j(\vec{iv}, \vec{c})$ is the j'th block in the i'th plaintext induced by $\langle \vec{iv}, \vec{c} \rangle$, etc. We say that *the pair* $\langle \vec{iv}, \vec{c} \rangle$ *is valid* if all the blocks in $\vec{iv}$ are distinct, and if the "forged ciphertext" $C'(\vec{iv}, \vec{c})$ is different than all the C^i's. Below we assume (w.l.o.g.) that only valid pairs $\langle \vec{iv}, \vec{c} \rangle$ happen in an execution of A with non-zero probability.

[4]This caveat does not arise if we assume that the function G has the additional property that is mentioned after Definition 1. Namely, under that extra assumption, there will always be a unique block (with high probability), regardless of the adversary's strategy.

Observe that since $\langle \vec{iv}, \vec{c}\rangle$ uniquely determine all the ciphertexts C^i and also the "forged ciphertext" C', it uniquely determines whether or not event E0 happens in the associated run of A. Slightly abusing notations, we write $\langle \vec{iv}, \vec{c}\rangle \in E0$ if event E0 happens in that run, and $\langle \vec{iv}, \vec{c}\rangle \notin E0$ otherwise. However, since $\langle \vec{iv}, \vec{c}\rangle$ do not uniquely determine the M^i_j's and N^i_j's (as those depend also on the masks, which in turn depend on the choice of G), then they do not determine whether or not events E1,E2,E3 happen in the associated run of A.

Now consider fixed $\langle \vec{iv}, \vec{c}\rangle$, and any fixed mask-generation function g. These together determine not only the actions of the adversary, but also all the M^i_j's, N^i_j's, and N'_j's. Therefore, they also determine whether or not events E1,E2,E3 happen in the associated run.[5] Again, to stress that some quantities are determined by $\langle \vec{iv}, \vec{c}, g\rangle$, we write these quantities as a function of $\langle \vec{iv}, \vec{c}, g\rangle$. For example, $M^i_j(\vec{iv}, \vec{c}, g)$, or $N'_j(\vec{iv}, \vec{c}, g)$. We also write $\langle \vec{iv}, \vec{c}, g\rangle \in E1$ (resp. $\in E2, E3$) if event E1 (resp. E2,E3) happens in that run, and otherwise $\langle \vec{iv}, \vec{c}, g\rangle \notin E1$ (resp. $\notin E2, E3$). We use expressions like $\langle \vec{iv}, \vec{c}, g\rangle \in E2 \cap (E1 \cup E3)$ in the obvious way.

We start with the main technical lemma, where we show that the event $(E1 \vee E3)$ occurs with overwhelming probability.

Lemma 3 $\Pr[\neg(E1 \vee E3)] \le 2^{-n}\binom{u}{2} + \epsilon(\binom{u}{2} + u + v)$.

Proof: Below we prove that $\Pr[\neg(E1 \vee E3)] \le \epsilon(\binom{u}{2} + u + v)$ when F is chosen as a *random function.* A standard argument shows that moving from random functions to random permutations adds at most $2^{-n}\binom{u}{2}$ to this probability.

Intuitively, Lemma 3 is true since: (a) Event E2 happens with very high probability, due to the properties of the mask-generation functions; (b) Conditioned on E2, the ciphertexts that the adversary sees are independent of the masks that are used; and (c) Therefore, the properties of G again ensure that with high probability, either at least one N'_x does not "collide" with any previous N^i_j (when E0 does not happens), or $S^i_{r+1}, S^i_{L^i+1}$ cause event E3 to happen (when E0 happens). The formal argument follows.

Claim 4 *(i) For any fixed, valid, pair* $\langle \vec{iv}, \vec{c}\rangle$, $\Pr_G\left[\langle \vec{iv}, \vec{c}, G\rangle \notin E2\right] \le \epsilon \cdot \binom{u}{2}$.

(ii) For any fixed, valid, pair $\langle \vec{iv}, \vec{c}\rangle \in E0$, $\Pr_G\left[\langle \vec{iv}, \vec{c}, G\rangle \notin E3\right] \le \epsilon$.

(iii) For any fixed, valid, pair $\langle \vec{iv}, \vec{c}\rangle \notin E0$, $\Pr_G\left[\langle \vec{iv}, \vec{c}, G\rangle \notin E1\right] \le \epsilon(u + v)$.

In all cases, the probability is taken over the choice of G, according to the distribution of the mask-generating function.

Proof: (i) Since all the IV's are distinct, then from $(i,j) \ne (i',j')$ we get that also $(j, IV^i) \ne (j', IV^{i'})$. For each $(i,j) \ne (i'j')$, we have $M^i_j = M^{i'}_{j'}$ if an only if $P^i_j \oplus S^i_j = P^{i'}_{j'} \oplus S^{i'}_{j'}$, which means that $G_j(L^i, IV^i) \oplus G_{j'}(L^{i'}, IV^{i'}) = P^i_j \oplus P^{i'}_{j'}$. But since $\vec{iv}, \vec{c}$ are fixed, then so are all the P^i_j's, and by the ϵ-xor universality of G, we get $\Pr_G[G_j(L^i, IV^i) \oplus G_{j'}(L^{i'}, IV^{i'}) = P^i_j \oplus P^{i'}_{j'}] \le \epsilon$. As there are exactly u plaintext blocks (including the checksums), we conclude that $\Pr_G[\langle \vec{c}, G\rangle \notin E2] \le \epsilon \cdot \binom{u}{2}$.

[5] Here it can certainly be the case that there is no choice of F that is consistent with these $\vec{iv}, \vec{c}, g$. I.e., no F maps all the M^i_j's to the corresponding N^i_j's. The analysis below, however, implies that for a random g, this rarely happens.

(ii) We note that $\langle \vec{iv}, \vec{c}\rangle$ uniquely define the indices i and r in the event E3, and also uniquely define P^i and IV^i. For event E3 *not to happen*, we must get

$$\left(\sum_{j=1}^{r} P_j^i\right) \oplus P_{L^i+1}^i = S_{r+1}^i \oplus S_{L^i+1}^i = G_{r+1}(L^i, IV^i) \oplus G_{L^i+1}(L^i, IV^i)$$

By the definition of E0, we have $r \leq L^i - 1$, and therefore, by the ϵ-xor-universality of G we get that the probability of the above equality holding is at most ϵ.

(iii) Bounding $\Pr_G[\langle \vec{iv}, \vec{c}, G\rangle \notin E1]$ requires some careful case analysis.

Case (a) is when the IV of the "forged ciphertext" is different than all the IV^i's (which are the blocks of $\vec{iv}$). In this case, for any $i \in \{1, \ldots, m\}$ and any $j \in \{1, \ldots, L^i\}$ we have

$$N_1' = N_j^i \quad \Rightarrow \quad S_1' \oplus C_1' = S_j^i \oplus C_j^i \quad \Rightarrow \quad G_j(L^i, IV^i) \oplus G_1(L', IV') = C_1' \oplus C_j^i$$

Since we assume that $IV' \neq IV^i$, then this happens with probability at most ϵ. The case of $j = L^i + 1$ is similar, except that here we need to get $G_0(L^i, IV^i) \oplus G_1(L', IV') = C_1' \oplus C_j^i$. Similarly, to get $N_j' = N_1'$ (for some $j > 1$) we need $G_j(L', IV') \oplus G_1(L', IV') = C_1' \oplus C_j'$, which again happens with probability at most ϵ.

The number of blocks that can "collide" with N_1' is u from the C^i's and $v - 1$ blocks from C', so the probability of collision in this case is at most $\epsilon(u + v - 1)$. Namely, for $\langle \vec{iv}, \vec{c}\rangle$ where $IV' \neq IV^i$ for all i, we can set $x = 1$, and we get $\Pr_G[\langle \vec{c}, G\rangle \notin E1] < \epsilon(u + v)$.

Case (b) is when for some i we have $IV' = IV^i$. (As the pair $\langle \vec{iv}, \vec{c}\rangle$ is valid, then this i is unique, and also $C' \neq C^i$.) Since $IV' = IV^i$, and since for a given G the masks only depend on the IV, we have $S_j^i = G_j(L^i, IV^i) = G_j(L', IV') = S_j'$ for all $j \leq \min(L^i, L') + 1$. Here we have a few sub-cases, depending on the relations between C' and C^i.

Subcase (b1) is when the first block in which C', C^i differ, is not the last block in either of them. In this case, let x be the index of the first block where they differ. Since $S_x^i = S_x'$, we get $N_x^i = C_x^i \oplus S_x^i \neq C_x' \oplus S_x' = N_x'$. For the other blocks in P^i, we have $N_j^i = N_x'$ if and only if $S_j^i \oplus S_x' = C_j^i \oplus C_x'$ (or $S_0^i \oplus S_x' = C_{L^i+1}^i \oplus C_x'$ for the last block). Since $j \neq x$ (and $x > 0$) each of these happen with probability at most ϵ.

Subcase (b2) is when C', C^i are of the same length, and they differ only in their last block. Here we set $x = L^i + 1 (= L' + 1)$. As in the previous subcase, since $S_0^i = S_0'$, we have $N_x^i = C_x^i \oplus S_0^i \neq C_x' \oplus S_0' = N_x'$. Also, for all $j \leq L^i (= L')$ we have $N_j^i = N_x'$ if and only if $S_j^i \oplus S_0' = C_j^i \oplus C_x'$, which happens with probability ϵ.

Subcase (b3) is when C' is longer than C^i (and they agree on all the blocks upto L^i – but perhaps not on block $L^i + 1$). Here we set $x = L^i + 1$, so we have $N_x^i = C_x^i \oplus S_0^i$ and $N_x' = C_x' \oplus S_x'$. Hence we get that $N_x^i = N_x'$ if and only if $S_0^i \oplus S_x' = C_x^i \oplus C_x'$, and for all $j < x$, $N_j^i = N_x'$ if $S_j^i \oplus S_x' = C_j^i \oplus C_x'$. Again, each of these happen with probability at most ϵ.

Subcase (b4) is when C^i is longer than C' (and they agree on all the blocks upto L' – but perhaps not on block $L' + 1$). Here we set $x = L' + 1$, $N_x' = C_x' \oplus S_0'$. For all $1 \leq j \leq L^i$, we have $N_j^i = C_j^i \oplus S_j^i$, so we still get $N_j^i = N_x'$ only if $S_j^i \oplus S_0' = C_j^i \oplus C_x'$, which happens with probability at most ϵ.

However, for $j = L^i + 1$ we have $N^i_j = C^i_{L^i+1} \oplus S^i_0$ and $N'_x = C'_x \oplus S'_0$, with $S^i_0 = S'_0$. Thus, if we had $C'_{L'+1} = C^i_{L^i+1}$ we would get $N'_{L'+1} = N^i_{L^i+1}$ too.[6] But recall that C' agrees with C^i on all the blocks $C'_0, \ldots, C'_{L'}$, so this is exactly the definition of event E0. Since we assume that $\langle \vec{iv}, \vec{c} \rangle \notin E0$, then it must be that $C'_{L'+1} \neq C^i_{L^i+1}$, and therefore also $N'_{L'+1} \neq N^i_{L^i+1}$.

As for the other ciphertexts $C^{i'}$ $i' \neq i$, and the other blocks in C', the same reasoning as for Case (a) holds here too. We conclude that in either case, we have $\Pr_G[\langle \vec{c}, G \rangle \notin E1] \leq \epsilon(u+v-1) < \epsilon(u+v)$. □

The last observation that we need for Lemma 3 is that conditioned on event E2, any ciphertext vector $\vec{c}$ has the same probability of occuring as any other vector. Below we denote the concatenation of all the ciphertexts $C^1 \ldots C^m$ *without the IV's* by $\vec{C}$, and the concatenation of all the IV's is denoted $\vec{IV}$. Then we have

Claim 5 *For any fixed $\langle \vec{iv}, \vec{c}, g \rangle \in E2$, it holds that $\Pr[\vec{C} = \vec{c} \mid \vec{IV} = \vec{iv},\ G = g] = 2^{-un}$, where the probability is takes over the choice of F as a random function over $\{0,1\}^n$.*

Proof: Since $\vec{iv}, \vec{c}$ and g are fixed, then so are all the M^i_j's and N^i_j's. We therefore get $\vec{C} = \vec{c}$ if and only if $F(M^i_j) = N^i_j$ for all (i,j) in the appropriate ranges. Since $\langle \vec{iv}, \vec{c}, g \rangle \in E2$, then the M^i_j's are all distinct (and there are u of them), and as F is random function, we have

$$\Pr\left[F(M^i_j(\vec{c},g)) = N^i_j(\vec{c},g) \text{ for all } i,j\right] = \prod_{i,j} \Pr\left[F(M^i_j(\vec{c},g)) = N^i_j(\vec{c},g)\right] = (2^{-n})^u$$

□

We now put everything together. Let $\vec{iv}$ be any fixed vector of m distinct IV's. Then,

$$\begin{aligned}
&\Pr\left[E2 \wedge (E1 \vee E3) \;\middle|\; \vec{IV} = \vec{iv}\right] \\
&= \sum_{\vec{c}} \Pr_{F,G}\left[\vec{C} = \vec{c} \;\wedge\; \langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap (E1 \cup E3) \;\middle|\; \vec{IV} = \vec{iv}\right] \\
&= \sum_{\vec{c}} \Bigg(\Pr_{G}\left[\langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap (E1 \cup E3) \;\middle|\; \vec{IV} = \vec{iv}\right] \\
&\qquad \cdot \Pr_{F,G}\left[\vec{C} = \vec{c} \;\middle|\; \vec{IV} = \vec{iv} \;\wedge\; \langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap (E1 \cup E3)\right] \Bigg) \\
&\overset{(a)}{=} \sum_{\vec{c}} \Pr_{G}\left[\langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap (E1 \cup E3)\right] \cdot 2^{-un} \\
&= 2^{-un} \cdot \Bigg(\sum_{\vec{c} \text{ s.t. } \langle \vec{iv}, \vec{c} \rangle \notin E0} \Pr_{G}\left[\langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap E1\right] + \sum_{\vec{c} \text{ s.t. } \langle \vec{iv}, \vec{c} \rangle \in E0} \Pr_{G}\left[\langle \vec{iv}, \vec{c}, G \rangle \in E2 \cap E3\right] \Bigg)
\end{aligned}$$

Equation (a) holds, since the IV's are chosen by the encryptor independently of the function G, and since by Claim 5 we have probability 2^{-un} for every individual tuple $\langle \vec{iv}, \vec{c}, g \rangle \in E2$.

Claim 4 implies that each individual summand in the last expression is no less than $1 - \epsilon(\binom{u}{2} + u + v)$, and since there are exactly 2^{un} such terms, we get

$$\Pr\left[E1 \vee E3 \;\middle|\; \vec{IV} = \vec{iv}\right] \geq \Pr\left[E2 \wedge (E1 \vee E3) \;\middle|\; \vec{IV} = \vec{iv}\right] \geq 1 - \epsilon \cdot (\tbinom{u}{2} + u + v)$$

[6] If G had the additional property from Section 2.2, then since the lengths of C', C^i are different, we would again get that the probability of $N'_{L'+1} = N^i_{L^i+1}$ is at most ϵ, and we won't need the additional argument about E0.

As this holds for every fixed vector $\vec{iv}$, the proof of Lemma 3 is complete. $\Box$

Next we show that conditioned on either E1 or E3, the chances of the event Succ are very slim.

Lemma 6 $\Pr[Succ \mid E1 \vee E3] \;\leq\; 1/(2^n - u - v)$.

Proof: We first show that $\Pr[Succ \mid E3] = 0$. Consider some fixed $\langle \vec{iv}, \vec{c}, g \rangle \in E3$. Recall that event E3 is a sub-event of E0, where the "forged ciphertext" is $C' = C_0^i, \ldots, C_r^i, C_{L^i+1}^i$ for some (unique) $i \in \{1, \ldots, m\}$ and some $r \in \{1, \ldots, L^i - 1\}$. Since the IV's are the same, $C_0' = C_0^i$, then the masks are also the same, namely $S_j' = S_j^i$ for all j. Therefore, for $j = 1, \ldots, r$, we have

$$C_j' = C_j^i \;\Rightarrow\; N_j' = N_j^i \;\Rightarrow\; M_j' = M_j^i \;\Rightarrow\; P_j' = P_j^i$$

For the last block we have

$$\begin{aligned} C_{r+1}' = C_{L^i+1}^i \quad &\Rightarrow \quad N_{r+1}' = C_{r+1}' \oplus S_0' = C_{L^i+1}^i \oplus S_0^i = N_{L^i+1}^i \\ &\Rightarrow \quad M_{r+1}' = M_{L^i+1}^i \\ &\Rightarrow \quad P_{r+1}' = M_{r+1}' \oplus S_{r+1}' = M_{L^i+1}^i \oplus S_{r+1}^i = P_{L^i+1}^i \oplus S_{r+1}^i \oplus S_{L^i+1}^i \end{aligned}$$

The "forged ciphertext" C' is valid only when $\sum_{j=1}^{r+1} P_j' = 0$, which we can write as $\left(\sum_{j=1}^{r} P_j^i\right) \oplus \left(P_{L^i+1}^i \oplus S_{r+1}^i \oplus S_{L^i+1}^i\right) = 0$. But this contradicts $\langle \vec{iv}, \vec{c}, g \rangle \in E3$.

We now show $\Pr[Succ \mid E1] \leq \frac{1}{2^n - u - v}$. We view an execution of the adversary A as follows: First the mask-generation function G and the IV's are chosen, and then the permutation F is chosen in an "on-line" fashion: Whenever we need to assign a value of F (or F^{-1}) at some new point, we choose it at random from all the values that were not yet assigned to any point. Conditioned on E1, the "forged ciphertext" induces a "unique block" N_x' that was not yet assigned an F^{-1} value. When the adversary outputs this "forged ciphertext", we first assign all the other F^{-1} values, and assign $F^{-1}(N_x')$ at the end. By the time we make this assignment, all the P_j''s except P_x' are already assigned some values, and S_x' is also assigned a value.

The "forged ciphertext" C' will be valid only if $\sum_{j=1}^{L'+1} P_j' = 0$. Since $P_x' = S_x' \oplus F^{-1}(N_x')$, we can re-write this condition as $F^{-1}(N_x') = S_x' \oplus \sum_{j \neq x} P_j'$. Since so far we assigned F or F^{-1} values at exactly $u + v - 1$ points, then

$$\Pr[Succ \mid E1] \;=\; \Pr\left[F^{-1}(N_x') = S_x' \oplus \sum_{j \neq x} P_j'\right] \;=\; \frac{1}{2^n - (u + v - 1)} \;<\; \frac{1}{2^n - u - v}$$

$\Box$

This completes the proof of Theorem 2, since

$$\Pr[Succ] \;\leq\; \Pr[\neg(E1 \vee E3)] + \Pr[Succ \mid E1 \vee E3] \;\leq\; 2^{-n}\binom{u}{2} + \epsilon\left(\binom{u}{2} + u + v\right) + \frac{1}{2^n - u - v}$$

$\Box$

3.2 Indistinguishability

Theorem 7 *If F is a random permutation and G is an ϵ-xor-universal function, independent of F, then for any adversary A that asks exactly m plaintext-pair queries, each entry of the pair totalling* *exactly $u-m$ blocks, it holds that*

$$\Big|\Pr[A \textit{ outputs 1} \mid \textit{the oracle encrypts always the 1st entry}]$$
$$- \Pr[A \textit{ outputs 1} \mid \textit{the oracle encrypts always the 2nd entry}]\Big| \;<\; (2^{-n}+2\epsilon)\binom{u}{2}$$

The probability is taken over the choice of F, G and the IV's.

Proof (sketch): This proof is somewhat similar to (but considerably simpler than) the proof of Theorem 2. We use similar notations, with the following exceptions: The "plaintext queries" of the adversary are now pairs (of the same length), which we denote by (P^i, Q^i). As before, we have $M^i_j = P^i_j \oplus S^i_j$, but now we also denote $(M')^i_j = Q^i_j \oplus S^i_j$. The N^i_j variables can now be set either as $N^i_j = F(M^i_j)$ or $N^i_j = F((M')^i_j)$, depending on which plaintext is being encrypted.

We also need to modify the event E2 from the proof of Theorem 2. In the modified event, not only are all the M^i_j's distinct, but all the $(M')^i_j$'s are also distinct. (We do not need the M^i_j to be different than the $(M')^i_j$'s though.) Similar to Claim 4(i), we can show that this event happens with probability at least $1-2\epsilon\binom{u}{2}$.

Conditioned on this modified E2, we can show that every ciphertext vector $\vec{c}$ has probability exactly 2^{-un}, regardless of which entry of the ciphertext pair is encrypted — assuming that F is a random function over $\{0,1\}^n$. This is essentially the same as Claim 5 above. Hence, conditioned on E2, the adversary has advantage zero when F is a random function, so its advantage when F is a random permutation is at most $2^{-n}\binom{u}{2}$.

We conclude that the total advantage of the adversary is at most $\Pr[\neg E2]+2^{-n}\binom{u}{2} \le (2^{-n}+2\epsilon)\binom{u}{2}$.
$\square$

4 Concluding remarks

In this note we proved that it is sufficient for the key-generation function in Jutla's construction to be ϵ-xor-universal. We note, however, that one may be able to show tighter bounds if G is stronger than just ϵ-xor-universal (for example, if the masks in different encryptions are completely independent). Also, some of the technicalities in the proof may be somewhat easier to handle if we make stronger assumptions about G. Below we briefly discuss some variations on the scheme that was analyzed in Section 3, and explain how the analysis here can be adapted to handle these variants.

G DEPENDS ON F. The analysis above assumed that G is independent of F. In some variants, however, it was suggested that G be computed from the IV using F itself (say, by setting $S_j = (j+1)\cdot F(IV)$). To handle such variants, one needs to extend "event E2" from the proof of Theorem 2: Not only do we need all the M^i_j's to be distinct, but all the inputs to F (including the IV's) should be distinct. It is easy to see that under this event, the masks are independent of the ciphertext vectors, and the rest of the proof follows.

G DEPENDS ON F^{-1}. In yet other variants, computing G (as a function of C_0) involves evaluating F^{-1}. For example, one of the variant that Jutla suggested, the masks are computed by choosing at random a string S, computing the masks from $F(S+1), F(S+2)\ldots$, and setting $C_0 = F(S)$.

To encompass this process within the framework of our analysis, we have to define the function G as $G(L, C_0) =$ some-function-of$(F(1 + F^{-1}(C_0)), F(2 + F^{-1}(C_0)), \dots)$.

The problem with this, is that in our analysis we prove Lemma 3 assuming that F is a random function rather than a permutation (specifically, we used this assumption in Claim 5). This is acceptable as long as we don't use F^{-1} during the run of A, but here it does not seem to work. This can sometimes be solved in an "ad hoc" manner, by considering an "alternative implementation" of G that does not involve F^{-1}, and showing that it produces a distribution close to the real implementation.

Another solution is to forgo going through random functions altogether. This is done as follows: We again extend "event E2", this time insisting not only that the inputs to F (i.e., the M_j^i's) be distinct, but also that the N_j^i's be distinct. Similarly to Claim 4, one can show that this event happens with probability of roughly $1 - 2\epsilon\binom{u}{2}$. Conditioned on this new event, we can now prove a claim similar to Claim 5, even when F is a random permutation. The argument here is that for any two fixed vectors $\vec{m}, \vec{n}$, each consisting of exactly u distinct blocks, the probability that a random permutation F maps $\vec{m}$ to $\vec{n}$ is exactly $(2^n - u)!/(2^n)!$, independently of the values of $\vec{m}, \vec{n}$ themselves. This lets us complete the proof of Lemma 3 without going via random functions at all.

THE "MASKED CBC" MODES. The proof of the "masked CBC" modes is nearly identical to the proof from Section 3 above: All the events are defined in exactly the same manner, and the reasoning is also the same. The only difference is with the blocks M_1^i: for a fixed pair $\langle \vec{iv}, \vec{c} \rangle$, the block M_1^i is just $iv^i \oplus P_1^i(\vec{iv}, \vec{c})$, so we cannot appeal to the properties of the function G to prove that it is different than other M_j^i blocks. Instead, we must require that IV^i itself is "unpredictable" given $IV^1 \dots IV^{i-1}$ and $C^1 \dots C^{i-1}$. That is, for every fixed setting of $iv^1, \dots, iv^i$ and $c^1 \dots c^{i-1}$, if must be that $\Pr[IV^i = iv^i \mid \forall i' < i, IV^{i'} = iv^{i'}, C^{i'} = c^{i'}] \leq \epsilon$. This can easily be shown when the IV's are computed as an application of F to some nonce. We note that the treatment of all the other M_j^i blocks remains as before, since for $j > 1$ we have $M_j^i = S_{j-1}^i \oplus C_{j-1}^i(\vec{iv}, \vec{c}) \oplus P_j^i(\vec{iv}, \vec{c})$.

DEALING WITH PARTIAL BLOCKS. One feature that is unique to Rogaway's OCB mode [4], is that it also handles "partial blocks". That is, the encrypted messages need not be of bit length which is a multiple of the block length. Although it is trivial to accomplish this feat using padding, OCB is unique in that it does not use padding. (That is, the length of the ciphertext equals that of the plaintext – not counting the IV and the tag.)

We note here that if we relax the "not padding" requirement, and instead require only that we do not pad messages whose length is already a multiple of the block length (so that we never introduce an additional encryption operation), then a very slight modification to Jutla's scheme can be used. Specifically, we process padded and unpadded messages the same way, except that we use a different mask for the last block. For example, we can use S_0 as before when the message is unpadded, but use $S_0 + \Delta$ when the message is padded, where Δ is a random quantity which is part of the key. It is not hard to see that the only part of the proof that is effected by this change is the treatment of event E1 (specifically, Claim 4(iii)). It is also not hard to see that this claim still holds for the modified scheme.

Acknowledgements. I thank Charanjit Jutla for many interesting conversations regarding his schemes.

References

[1] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In T. Okamoto, editor, *Advances in Cryptology, Asiacrypt 2000*, volume 1976 of *Lecture Notes in Computer Science*. Springer-Verlag, 2000.

[2] V. D. Gligor and P. Donescu. Fast encryption authentication: XCBC encryption and XECB authencation modes. presented in NIST's workshop on modes of operations, in October, 2000. See `http://csrc.nist.gov/encryption/modes/workshop1/`.

[3] C. Jutla. Encryption modes with almost free message integrity. In *to appear in EUROCRYPT'2001.* preliminiary version was presented in NIST's workshop on modes of operations, in October, 2000. See `http://csrc.nist.gov/encryption/modes/workshop1/`.

[4] P. Rogaway. OCB mode: Parallelizable authenticated encryption. presented in NIST's workshop on modes of operations, in October, 2000. See `http://csrc.nist.gov/encryption/modes/workshop1/`.

[5] P. Rogaway. Bucket hashing and its application to fast message authentication. In *Advances in Cryptology – CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 313–328. Springer-Verlag, 1995.