Encryption Modes with Almost Free Message Integrity

Charanjit S. Jutla
IBM T. J. Watson Research Center,
Yorktown Heights, NY 10598, USA

Abstract

We define a new mode of operation for block encryption which in addition to assuring confidentiality also assures message integrity. In contrast, previously for message integrity a separate pass was required to compute a cryptographic message authentication code (MAC). The new mode of operation, called Integrity Aware CBC (IACBC), requires a total of $m + \log m$ block encryptions on a plaintext of length m blocks. The well known CBC (cipher block chaining) mode requires m block encryptions. The second pass of computing the MAC essentially requires additional m block encryptions. We also show a lower bound of $\Omega(\log m)$ additional block encryptions for any reasonably modeled scheme which assures message integrity along with confidentiality.

1. Introduction

Symmetric key encryption has become an integral part of today's world of communication. It refers to the schemes and algorithms used to secretly communicate data over an insecure channel between parties sharing a secret key. It is also used in other scenarios like data storage.

There are two primary aspects of any security system: *confidentiality* and *authentication*. In its most prevalent form, confidentiality is attained by encryption of bulk digital data using *block ciphers*. The block ciphers (e.g. DES [4]), which are used to encrypt fixed length data, are used in various chaining modes to encrypt bulk data. One such mode of operation is cipher block chaining (CBC) ([1, 2, 3]). The security of CBC has been well studied [8].

Cipher block chaining of block ciphers is also used for authentication. The CBC-MAC (CBC Message Authentication Code) is an international standard [5]. The security of CBC MAC was demonstrated in [6]. Authentication in this setting is also called *Message Integrity*.

Despite similar names, the two CBC modes, one for encryption and the other for MAC are different, as in the latter the intermediate results of the computation of the MAC are kept secret. In fact in most standards (TSL, IPsec [12, 11]) and proprietary security systems, two different passes with two different keys, one each of the two modes is used to achieve both confidentiality and authentication.

Nevertheless, it is enticing to combine the two passes into one, that is in a single cipher block chaining pass, both confidentiality and authentication are assured. Many such attempts have been made, which essentially use a simple checksum or manipulation detection code (MDC) in the chaining mode ([9, 10, 13]). Unfortunately, all such previous schemes are susceptible to attacks (see e.g. [14], Appendix B).

In this paper, we present a new variant of CBC mode, which in a single pass achieves both confidentiality and authentication. To encrypt a message of length m blocks, it requires a total of $(m + \log m)$ block encryptions. All other operations are simple operations, like exclusive-or. To contrast this with the usual CBC mode, the encryption pass requires m block encryptions, and the MAC computation requires another m block encryptions.

We also show that there is indeed a matching lower bound to our mode of operation, in a reasonable model of computation. This also explains why all previous attempts which tried to attain both features together, without the extra $\log m$ cryptographic operations, have failed.

Our new mode of operation is also simple. A simpler (though not as efficient) version of the mode just requires a usual CBC encryption of the plaintext appended with the checksum (MDC), with a random initial vector r. As already mentioned, such a scheme is susceptible to message integrity attacks. However, if one "whitens" the complete output with a random sequence, the scheme becomes secure against message integrity attacks. Whitening just refers to xor-ing the output with a random sequence. The random sequence could be generated by running the block cipher on $r+1$, $r+2$, ... $r+m$ (but with a different shared key). This requires m additional cryptographic operations, and hence is no more efficient than generating a MAC.

The efficiency of the new mode comes from proving that the output whitening random sequence need only be pair-wise independent. In other words, if the output whitening sequence is s_1, s_2,...s_m, then each s_i is required to be random, but only pairwise-independent of the other entries. Such a sequence is easily generated by performing only $\log m$ cryptographic operations like block encryption.

The rest of the paper is organized as follows. Section 2 describes the new mode of operation. Section 3 gives definitions of random permutations, and formalizes the notions of security, for both confidentiality and message integrity. In section 4 we state the theorem for the security of the new mode of operation. In section 5 we prove that the new scheme is secure for message integrity. In section 6 we describe our model of computation for the lower bound, and prove the lower bound.

2. The New Mode of Operation - IACBC

In this section we describe the new mode of operation for encryption, which also guarantees message integrity. Although, the subsequent proofs generalize to various variants, we only present here a mode of operation which is most similar to CBC (cipher block chaining) mode of operation. We call this mode **IACBC** for *integrity aware cipher block chaining.*

Let n be the block size of the underlying block cipher (or pseudorandom permutation). If the block cipher requires keys of length k, then this mode of operation requires two independent keys of length k (however, see the end of this section for a further discussion). Let these keys be called $K0$ and $K1$. From now on, we will use f_x to denote the encryption function under key x. The same notation also holds for pseudorandom permutations.

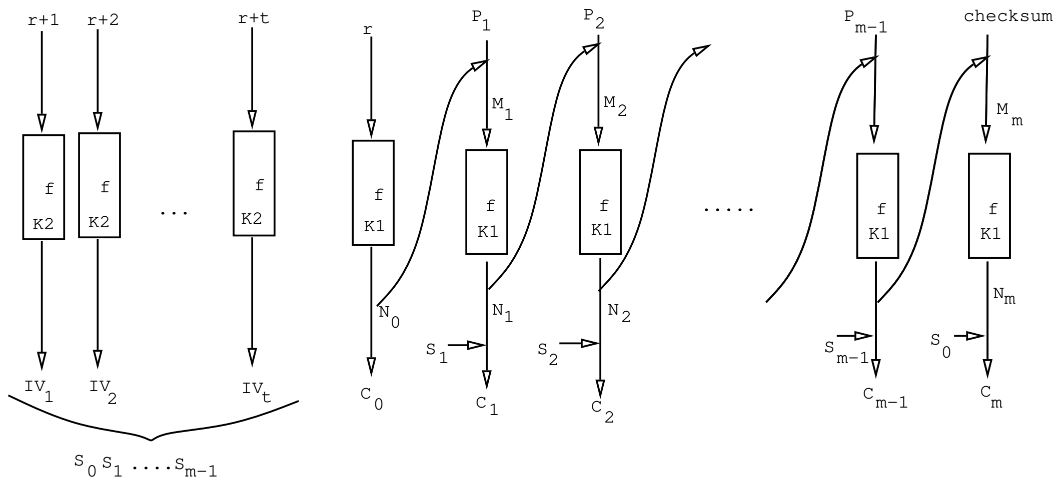

Figure 1: Encryption with Message Integrity (IACBC)

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 of length n (bits) is chosen. This random vector r is expanded into $t = \lceil \log m \rceil$ new random and independent vectors using the block cipher and key $K0$ as follows:

for $i = 1$ to t do
  $IV_i = f_{K0}(r + i)$
end for

The t random and independent vectors are used to prepare $2^t - 1$ new pair-wise independent random vectors $S_0, S_1, ..., S_{2^t-2}$. One way to do this is as follows:

for $i = 1$ to $2^t - 1$ do

  Let $< a_1, a_2, ...a_t >$ be the binary representation of i

  $S_{i-1} = \sum_{j=1}^{t} (a_j \cdot IV_j)$

end for

The summation in the for loop above is an xor-sum.

The ciphertext message $C = < C_0, C_1, ..., C_m >$ is generated as follows (see Figure 1):

$M_0 = r$

$N_0 = f_{K1}(M_0)$

$C_0 = N_0$

for $i = 1$ to $m - 1$ do

  $M_i = P_i \oplus N_{i-1}$

  $N_i = f_{K1}(M_i)$

  $C_i = N_i \oplus S_i$

end for

checksum $= \sum_{i=1}^{m-1} P_i$

$M_m =$ checksum $\oplus N_{m-1}$

$N_m = f_{K1}(M_m)$

$C_m = N_m \oplus S_0$

Again, the summation above is an xor-sum. Note that S_0 is used in the last step.

It is easy to see that that the above scheme is invertible. The inversion process yields blocks $P_1, P_2, ..., P_m$. The decrypted plaintext is $< P_1, P_2, ..., P_{m-1} >$. Message integrity is verified by checking $P_m = P_1 \oplus P_2 \oplus ... \oplus P_{m-1}$.

Note that, the random vectors $IV_1, ...IV_t$ could have been generated by a pseudorandom function (rather than pseudorandom permutation).

There is another way of generating the pairwise independent vectors $S_0, S_1, ..., S_{2^t-2}$. Instead of using the subset construction, one could use an algebraic construction, i.e. generate two random vectors IV_1, and IV_2, and then let $S_i = (IV_1 + IV_2 * i) \bmod p$, where p is a prime of appropriate size. For example, if the block cipher has block size 64 bits, p could be chosen to be $2^{64} - 257$. This leads to a fast implementation. Theorem 3 holds for this construction as well, as the main requirement there is for the output whitening sequence to be pairwise independent.

2.1 Parallelizable Mode

We now describe another mode, which is highly parallelizable. We call this mode the Integrity Aware Parallelizable Mode (IAPM). In a way, it is similar to the counter mode of encryption. However, IAPM also assures message integrity.

In this mode, there is no ciphertext chaining. Instead, the security of the scheme is obtained by "whitening" the input with the same pairwise independent sequence which is used to whiten the output, i.e., $S_0, S_1, ...S_{m-1}$. The proofs of security as given in section 4 and 5 for the IACBC scheme also work for IAPM.

The scheme is described in Figure 2.

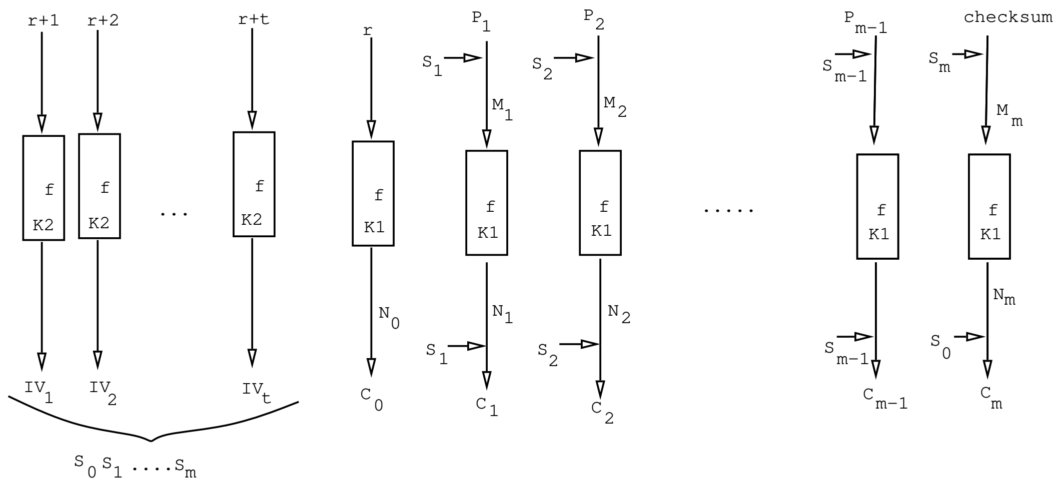

Figure 2: Parallelizable Encryption with Message Integrity (IAPM)

3. Preliminaries and Definitions

3.1 Random Permutation-like Functions

Definition (*Random Function*) A Random function is a function chosen randomly from $\{0,1\}^n \rightarrow \{0,1\}^l$. It could also be viewed as a random sequence (uniformly chosen) of length 2^n of l bit strings.

Definition (*Random Permutation*) A Random permutation is a function chosen randomly from class of permutations in $\{0,1\}^n \rightarrow \{0,1\}^n$. It could also be viewed as a random sequence chosen uniformly from the class of all 2^n length sequences of l bit strings, such that each l bit string is represented once in every sequence.

The following notion is new (i.e. non-standard). The new notion and the following theorem help simplify the proof of message integrity. It essentially separates the approximations in calculating the success probability that result from replacing random permutations by random functions in Theorem 3.

Definition (*Random Permutation-like Functions (RPF)*) A Random Permutation-like Function with parameter q is a pair of random functions $< f, g >$, with the following restriction

- For $i \in [1..q]$ define $\pi(i) = \min\{j : j \leq q \text{ and } f(j) = f(i)\}$
- if $j = f(i)$ for some $i \leq q$, j arbitrary, then $g(j) = \pi(i)$.

A permutation f can be viewed as a pair $< f, f^{-1} >$.

Theorem 1: Let $< F, G >$ be a random permutation-like function with parameter q. Let P be a random permutation. Consider an adversary which is allowed calls to a pair of oracles $< O_1, O_2 >$, with the restriction that it is only allowed to call O_1 on inputs $1, 2, ..., q$, whereas there is no restriction on calls to O_2. Any such adversary A that makes at most q total queries to a pair of oracles has probability at most $q^2/2^n$ of distinguishing $< F, G >$ from $< P, P^{-1} >$.

3.2 Encryption Schemes: Message Security with Integrity Awareness

The definitions of pseudorandom functions and permutations are not given here (see [7] for instance). We instead give definitions of schemes which explicitly define the notion of secrecy of the input message. Of course, pseudorandom permutations can be used to build encryption schemes which guarantee such message secrecy (see [7] for example).

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 awareness:**

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

$$D : \{0,1\}^n \times \mathcal{N} \times \{0,1\}^{l(n)} \rightarrow \{0,1\}^{l(n)+w(n)}$$
$$\text{MDC} : \mathcal{N} \times \{0,1\}^{l(n)} \rightarrow \{0,1\}^{w(n)}$$

be polynomial-times function ensembles. In E, the third argument is supposed to be the length of the plaintext. Similarly, in D the second argument is the length of the ciphertext. We will drop this argument when it is clear from context. The functions E and D have the property that for all $x \in \{0,1\}^n$, for all $m \in \{0,1\}^{l(n)}$, $c \in$ Coins

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

We will usually drop the random argument to E as well, and just think of E as a probabilistic function ensemble. We will also drop n when it is clear from context. Thus we will write l for $l(n)$ etc.

Definition (*Security under Find-then-Guess* [8, 7]) Consider an adversary A that runs in two stages. During the adversary's find stage he endeavors to come up with a pair of equal length messages, m^0, m^1, whose encryptions he wants to tell apart. He also retains some state information s. In the adversary's guess stage he is given a random ciphertext y for one of the plaintexts m^0, m^1, together with s. The adversary is said to "win" if he correctly identifies the plaintext.

An Encryption Scheme is said to be (t, q, μ, ϵ)-secure in the find-then-guess sense, if for any adversary A which runs in time at most t and asks at most q queries , these totaling at most μ bits,

$$Adv_A \stackrel{\text{def}}{=} 2 \cdot Pr[(m^0, m^1, s) \leftarrow A^{E_x(\cdot)}(\text{find});\ b \leftarrow \{0,1\};\ y \leftarrow E_x(m^b) :\ A^{E_x(\cdot)}(guess, y, s) = b] - 1 \ \leq\ \epsilon$$

Definition (*Integrity Awareness):* 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 ciphertext C, different from each C^i, $i \in [1..r]$. Since D has length of the ciphertext as a parameter, the breakup of $D_x(C)$ as $m \| m'$, where $|m'| = w(n)$, is well defined. The adversary's success probability is given by

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

4. Message Secrecy

We state the theorem for security under the Find-then-Guess notion of security. The proof follows standard techniques ([8, 7]), and will be given in the full version of the paper.

Theorem 2: Let A be an adversary attacking the IACBC encryption scheme (with f being a random function F) in the find-then-guess sense, making at most q queries, totaling at most μ bits. Then,

$$Adv_A \leq (\frac{\mu^2}{n^2} - \frac{\mu}{n}) \cdot \frac{1}{2^n}$$

5. Message Integrity

In this section we show that the mode of operation IACBC guarantees message integrity with high probability.

We start with some informal observations to aid the reader in the eventual formal proof. First thing to note is that since each encryption has a new random seed r, it does not help the adversary to have more than one pair of plaintext-ciphertext messages. Thus, essentially the problem of message integrity is the following. Given P^1, and corresponding C^1, can the adversary generate another C^2 different from C^1, such that on decryption the plaintext passes the integrity check.

We will take the following approach. We first restrict ourselves to the random permutation-like function model. That is, we model the block cipher by a random permutation-like function. Using Theorem 1 (section 3.1), one can show that the following theorem also holds for the random permutation model. Finally, yet another standard reduction shows that the theorem holds for pseudorandom permutations.

Theorem 3: Let A be an adversary attacking the IACBC encryption scheme with random permutation-like function $< F, G >$ making at most r queries in the first stage, totaling at most μ bits (where $\mu \leq qn$, q being the parameter of F). Then,

$$Succ < (\frac{\mu^2}{n^2}) \cdot \frac{1}{2^n}$$

Proof:

For sake of clarity, we assume that the adversary only has one query in the first stage with plaintext P of length m blocks and corresponding ciphertext C ($\mu = mn$).

In the first stage, we do a modification to the IACBC algorithm. The modified algorithm uses $F(\mu(i))$ instead of $F(M_i)$ for queries $F(M_i)$, where

$$\mu(i) = \min \{j : j \leq i \text{ and } M_j = M_i\}$$

Given that F is random, the behavior of the modified algorithm and the original algorithm is identical.

Its query in the second stage is with ciphertext $C' \neq C$. We will used primed variables to denote the variables in the second stage. For example, P'_m will denote the last decrypted block (if $C' = C'_0, ...C'_m$).

First note that, $r, IV_1, IV_2, ..., IV_t$ are uniformly random and independent variables. Also, they are all independent of P.

Now assume that P and r are such that

$$\forall i, j : \; M_i \neq M_j$$

This happens with high probability as in Theorem 2. This implies that $N_0, N_1, ..., N_m$ are uniformly random and independent variables. Also, they are all independent of $r, IV_1, IV_2, ..., IV_t$.

We first consider the case where the length of C' and C is same.
Let i be the smallest index in which C and C' differ. It is easy to see that $N_i \neq N'_i$.
The case $i = m$ is trivial, as $M'_m \neq M_m$ with high probability, and hence $P'_m \neq \sum_{i=1}^{m-1} P'_i = \sum_{i=1}^{m-1} P_i$.
Next, we consider the case $i \in [1..m-1]$. We first prove the following :

With high probability the following *does not* hold:

$$(1)\ \exists j : j = 0..m, N'_i = N_j$$

$$\text{or } (2)\ \exists j : j = 0..m, j \neq i, N'_i = N'_j$$

Now, $N'_i = C'_i \oplus S_i$, as $S'_i = S_i$, i being greater than zero.
Thus, for (1) to hold for a particular j would require

$$S_i \oplus S_j = C'_i \oplus C_j$$

But, $C_j = N_j \oplus S_j$ (for $j > 0$), and N_j is independent of S_j. Thus, C_j is independent of S_j. In fact, since $N_0, N_1, ..., N_m$ are independent of $IV_1, IV_2, ..., IV_t$, the whole of C is independent of $IV_i, IV_2, ..., IV_t$, and hence independent of each S_k (for any $k \in [0..m]$). We already know that P is independent of each S_k. Also, C' is completely determined by C and P, and hence C' is fixed independent of S_k (for any $k \in [0..m]$). Since $i \neq j$ (we already know that $N'_i \neq N_i$), $S_i \oplus S_j = S_k$, for some $k \in [0..m]$. Since, S_k is random and independent of C and C', the probability that $S_k = C'_i \oplus C_j$ is 2^{-n}. The case when $j = 0$ is proved similarly.

For case (2), for $j < i$, $N'_j = N_j$, and hence case (1) applies. For case (2) to hold for a particular $j > i$ would require

$$S_i \oplus S_j = C'_i \oplus C'_j$$

Again, as before, $C'_i \oplus C'_j$ is fixed completely independent of S_k (for any $k \in [0..m]$). And hence the probability is at most 2^{-n}.

Thus, the disjunction (1) or (2) holds with probability at most $2(m+1) * 2^{-n}$.

Now, we consider the case $i = 0$, i.e. $C'_0 \neq C_0 = N_0$. We show that with high probability, for all $j \in [1..m]$, $C'_0 \neq N_j$. We consider the individual event $N_j = C'_0$, or $S_j = C'_0 \oplus C_j$. Since, $N_0, N_1, ...N_m$ are independent of S_j, the whole of C is independent of S_j, and hence C' is also independent of S_j. Thus, $S_j = C'_0 \oplus C_j$ holds with probability 2^{-n}. Thus, with probability at most $m * 2^{-n}$, there exists a $j \in [1..m]$ such that $C'_0\ (= N'_0)\ = N_j$.

Thus, $M_0' = G(N_0')$ is a random variable independent of all previous variables. This implies, that with high probability, $IV_1', ...IV_t'$ are random and independent variables, independent of all previous variables $r, IV_1, ...IV_t, N_0, N_1, ...N_m$. Thus, with high probability $N_1' \neq N_1$, and now the previous case applies.

Thus, we have that with high probability, there is an $i \in [1..m-1]$ such that

$$(1)\ \forall j, j \in [0..m]\ : N_i' \neq N_j$$

$$\text{and } (2)\ \forall j, j \in [0..m], j \neq i\ : N_i' \neq N_j'$$

Thus, $M_i' = G(N_i')$ is a random variable independent of all of $r, r', IV_1, IV_1', ...IV_t, IV_t', N_0, N_0', ...N_m, N_m'$, and also independent of $P_1, P_2, ...P_{m-1}$, and all M_j' $(j \neq i)$.

Now,

$$P_m' = \sum_{j=1}^{m-1} P_j' = \sum_{j=1}^{m-1} (M_j' \oplus N_{j-1}) \text{ and } \mathrm{MDC}(P) = \sum_{j=1}^{m-1} P_j$$

Thus, the event we are interested in is

$$M_i' = \sum_{j=1}^{m-1} (P_j \oplus N_{j-1}) \oplus \sum_{j \neq i} M_j'$$

The LHS being independent of RHS, the probability of the event is 2^{-n}.

For the case when the lengths of C and C' are different, we just remind the reader that a designated set S_0 is used in the last block. □

6. Lower bound

In this section we show that the $\log m$ additional cryptographic operations in the IACBC scheme are essentially the least one has to do to assure message integrity along with message secrecy.

We consider the following model. We assume a fixed block size n for a block cipher (or random permutations or length preserving random functions). Any application of one of these will constitute one application of a cryptographic operation. The only other operations allowed are linear operations over $(\mathrm{GF}2)^n$, i.e. bit-wise exclusive-or. Of course, operations of testing whether an n bit quantity is zero is also allowed. Since, the scheme could be probabilistic, as IACBC is, we also allow v blocks of randomness, $r_1, ..., r_v$.

Let, the message to be encrypted be of size m blocks, i.e. mn bits. Call the input blocks $P_1, ..., P_m$. Let there be $m + k$ invocations of random functions, and let the inputs to these functions be $M_1, M_2, ..., M_{m+k}$. Similarly, let the outputs of these random functions be $N_1, N_2, ..., N_{m+k}$. Let, $C = C_1, C_2, ...C_{m+t}$ be a linear function of P's, r's, M's and N's. Here $0 \leq t \leq k$.

Our aim is to show that either the scheme is not secrecy secure, or it is not message integrity secure, or it is not invertible, or $k + v = \Omega(\log n)$. More formally, we would like the scheme to behave as a random function from mn bits to $(m+t)n$ bits. The scheme is not secrecy secure if an adversary can distinguish the scheme from such a random function with probability $\geq 1 - 2^{-n}$.

For message integrity, let there be $u > 0$ MDC functions $D_1, D_2, ..., D_u$. Without loss of generality (see below), assume that these are linear functions of r's, M's and N's, and they are linearly independent. The scheme is not message integrity secure, if given P and C, an adversary can produce a $C' \neq C$, such that on inversion, all the MDC functions evaluate to zero with high probability.

For invertibility, we assume the scheme has the following structure: There is a subset of N's which can be written as linear functions of just the C's. The corresponding M's then may lead to determination of some more M's, and hence N's. Using, these new M's and N's, a second subset of N's can be written as a linear combination of previously determined M's, N's and C, and so on. We are forced to take this approach, as by just allowing a system of equations with unique inverse is not enough. The unique inverse may exist but may not be efficiently computable. For example, $C_1 = M_1 \oplus N_1$, may have a unique inverse, but may be intractable to compute.

Due to the fact that C is completely determined by r's, M's, N's and P's, it follows from the above characterization that C can be expressed as linear expressions in only N's, M's and r's. For otherwise, the scheme is not secrecy secure (i.e. there is a linear relationship between only C's and P's). Similarly, P can be expressed as linear expressions in only N's, M's and r's. This justifies the above restriction on MDCs.

The proof of the lower bound is given in appendix A.

Acknowledgement

The author would like to thank Pankaj Rohatgi for help in simplifying the proof of message integrity, and in helping simplify the overall scheme. The author would also like to thank Don Coppersmith and Nick Howgrave-Graham for several helpful discussions.

References

[1] ANSI X3.106, "American National Standard for Information Systems - Data Encryption Algorithm - Modes of Operation", American National Standards Institute, 1983.

[2] ISO 8372, " Information processing - Modes of operation for a 64-bit block cipher algorithm", International Organization for Standardization, Geneva, Switzerland, 1987

[3] National Bureau of Standards, NBS FIPS PUB 81, "DES modes of operation", U.S. Department of Commerce, 1980.

[4] National Bureau of Standards, Data Encryption Standard, U.S. Department of Commerce, FIPS 46 (1977)

[5] ISO/IEC 9797, "Data cryptographic techniques - Data integrity mechanism using a cryptographic check function employing a block cipher algorithm", 1989

[6] M. Bellare, J. Kilian, P. Rogaway, "The Security of Cipher Block Chaining", CRYPTO 94, LNCS 839, 1994

[7] M. Luby, "Pseudorandomness and Cryptographic Applications", Princeton Computer Science Notes, Princeton Univ. Press, 1996

[8] 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

[9] RFC 1510,"The Kerberos network authentication service (V5)", J. Kohl and B.C. Neuman, Sept 1993

[10] C.H. Meyer, S. M. Matyas, "Cryptography: A New Dimension in Computer Data Security", John Wiley and Sons, New York, 1982

[11] Security Architecture for the Internet Protocol, RFC 2401, http://www.ietf.org/rfc/rfc2401.txt

[12] The TLS Protocol, RFC2246, http://www.ietf.org/rfc/rfc2246.txt

[13] V.D. Gligor, P.Donescu, "Integrity Aware PCBC Encryption Schemes", 7th Intl. Workshop on Security Protocols, Cambridge, LNCS, 1999

[14] S.G. Stubblebine and V.D. Gligor, "On message integrity in cryptographic protocols", Proceedings of the 1992 IEEE Computer Society Symposium on Research in Security and Privacy, 1992.

Appendix A

Here, we prove the lower bound on the number of block encryptions required in a scheme as modeled in Section 6.

Let

$$D_i = \sum_{j=1}^{m+k} (a_j^i \cdot M_j) \oplus \sum_{j=1}^{m+k} (b_j^i \cdot N_j) \oplus \sum_{j=1}^{v} (c_j^i \cdot r_j)$$

We say that N_i and N_j *resolve* if $N_i \oplus N_j$ can be written as a linear combination of only the C's and the P's. Similarly, for M_i and M_j.

Suppose there exists a pair $i, j, i \neq j, i, j \in [1..m+k]$ such that

1. N_i and N_j resolve
2. M_i and M_j resolve
3. For all $x \in [1..u]$, $a_i^x \oplus a_j^x = 0$, and $b_i^x \oplus b_j^x = 0$

Then, we show that an adversary can produce a new $C' \neq C$, such that all the MDC functions evaluate to zero. Note that, if there exists a C' such that

- $N_i' = N_j$
- $N_j' = N_i$
- for all other x, $N_x' = N_x$

then, we have a similar set of relations for M, and hence given (3), all the MDC functions would evaluate to zero.

Since C can be expressed only in terms of N's, M's and r's it is not difficult to come up with such a C'. Moreover, we have also assumed in our schemes, that a C' uniquely determines N', and M'.

Finally, we show that if $k+v$ is not $\Omega(\log n)$, then there exists a pair i, j satisfying (1), (2) and (3). Let

$$[P_1 ... P_m r_1 ... r_v N_1 ... N_{m+k}] \cdot \mathbf{B} \; = \; [C_1 ... C_m]$$

The rank of the matrix $\mathbf{B}$ is m. For a fixed P, let the resulting matrix be $\mathbf{B}'$, i.e.

$$[r_1 ... r_v N_1 ... N_{m+k}] \cdot \mathbf{B}' \; = \; [C_1 ... C_m]$$

The rank of the new matrix $\mathbf{B}'$ is still m, for otherwise we have a non-trivial linear relationship between C and P, and hence the scheme is not random. This implies that

$$[r_1...r_v N_1...N_{m+k}] \;=\; [f(C)] + (\mathrm{GF2})^n \cdot V_1 + ... + (\mathrm{GF2})^n \cdot V_{k+v}$$

where $f(C)$ is a set of linear functions of C's, and V_i are linearly-independent binary row-vectors. For a subset of N's with indices a set $J \subseteq [1..m+k]$ to be pair-wise independent thus requires $k+v \geq \log|J|$. In other words, there exists $i, j \in J, i \neq j$, N_i and N_j resolve if $k+v < \log|J|$. Stated differently, there is a set of size $|J| = (m+k)/2^{k+v}$ in which all pairs of N's resolve with each other. A similar statement holds for M's. Thus, there is a set of size $|J| = (m+k)/2^{2(k+v)}$ in which all pairs of N's resolve with each other, and all pairs of M's resolve with each other.

Similarly, a set of size $|J| = (m+k)/2^u$ has

$$\forall k \in [1..u],\ \forall i, j \in J:\ a_i^k \oplus a_j^k = 0$$

Combining these arguments, we get that there exists a pair satisfying (1), (2) and (3) if $2u+2(k+v) < \log n$.

To complete the proof, we show that $(k+v) \geq u$. We can write P's and D's as linear functions of r's, M's and C's (as discussed earlier N's can be replaced by r's, M's and C's). Thus, we have a matrix $\mathbf{A}$ such that

$$[C_1...C_m r_1...r_v M_1...M_{m+k}] \cdot \mathbf{A} \;=\; [P_1...P_m D_1...D_u]$$

The matrix $\mathbf{A}$ has rank at least $m+u$, for otherwise one would get a non-trivial linear relationship between D's and P's. In fact, for a fixed C, the rank of the resulting matrix $\mathbf{A}'$ is still at least $m+u$, for otherwise we would get a non-trivial linear relationship between D's, P's and C's. However, on a valid encryption, D's evaluate to zero. Thus, for valid encryptions we have a non-trivial linear relationship between the P's and the C's, which renders the encryption distinguishable from random. Thus, $m+k+v \geq m+u$.

□

Appendix B

A new mode of operation for combining confidentiality and authentication was recently described in [13]. The mode of operation is called IA-PCBC (Integrity Aware Plaintext Ciphertext Block Chaining). It was however shown by the author that the scheme is not secure for message integrity. We just remark here that the scheme was essentially as described in the model in Section 6. To encrypt a m blocks, only $m+2$ block encryptions are employed in IA-PCBC. The claimed security came from mixing addition over integers modulo 2^n, with exclusive-or operations. However, one can be approximated in terms of others= with reasonably high probability, and then the attack follows by the lower bound in Section 6.