Practical Symmetric On-line Encryption

Anonymous

Submission to FSE 2003

Abstract. This paper addresses the security of symmetric cryptosystems in the blockwise adversarial model. At Crypto 2002, Joux *et al.* have proposed a new kind of attackers against several symmetric encryption schemes. In this paper, we first show a generic technique to thwart blockwise adversaries for a specific class of encryption schemes. It consists in delaying the output of the ciphertext block. Then we also provide the first security proof for the CFB encryption scheme, which is naturally immune against such attackers.

Keywords: Symmetric encryption scheme, blockwise adversary, chosen plaintext attacks.

1 Introduction

Modes of operation are well-known techniques to encrypt messages longer than the block length of a cipher. The message is first cut into blocks and the mode of operation allows to securely encrypt the blocks. The resulting construction is called an encryption scheme. Specific properties are achieved by some of these modes: self-synchronization, ensured by chained modes such as CBC and CFB [13], or efficient encryption throughput, ensured by parallelized modes such as ECB and OFB [13]. Two different techniques are mainly used to build these schemes. The first one directly outputs the block of the block cipher (ECB, CBC), sometimes after being xored with some mask values (as in IAPM [11], IACBC [11], OCB [14]). The second method uses the block cipher to generate random strings which are xored with the message blocks (CTR [2], OFB, CFB). In this paper we investigate the security of the classical modes of operations in a more realistic and practical scenario than previous studies.

In cryptography, security is usually defined by the combination of a *security goal* and an *adversarial model*. The security goal of an encryption scheme is privacy. Most of the recently proposed modes also ensure integrity in a single pass. However, we do not focus on this security goal in this paper, although providing both privacy and integrity at the same time is not an easy task. But, since integrity can be simply provided by adding a secure

Message Authentication Code to the ciphertext, we just focus on privacy in the following.

Informally speaking, privacy of an encryption scheme guarantees that, given a ciphertext, an adversary is not able to learn any information about the corresponding plaintext. Goldwasser and Micali have formalized this notion in [6] where it has been called the semantic security. An equivalent definition called indistinguishability of encryptions (IND) has also been more extensively studied in [2] for the symmetric encryption setting: given two equal length messages M_0 and M_1 chosen by the adversary and the encryption C of one of them, it is difficult for the adversary to distinguish whether C is the encryption of M_0 or M_1. In practical scenarii, adversary goals can be different from this theoretical notion of privacy. For example, the adversary can try to recover the secret key or to recover the plaintext underlying a given ciphertext. However, in a security point of view, if the scheme is secure under the IND security notion, key recovery or plaintext recovery cannot be achieved more efficiently by the adversary. It is worth noticing that a security proof for encryption mode is not an absolute proof of security. As it is often the case in cryptography, we can only make proofs by reduction, in the complexity theoretic sense, between the security of the scheme and the security of the block cipher used in the encryption scheme. In practice, such a proof shows that the mode achieves the security goal assuming the security of the underlying block cipher.

Orthogonally to the security goal, the adversarial model defines the adversary means in order to win the security goal. For a long time, the only considered adversarial models were known plaintext attacks, chosen plaintext attacks (CPA) or chosen ciphertext attacks (CCA), depending on the adversary abilities. Moreover, usually, it is implicitly assumed that messages M_0 and M_1 are sent to the encryption oracle as atomic entities. However, in the real world, the encryption module can be a cryptographic accelerator hardware or a smart card with limited memory. For example, in real-time applications, a movie represents a much too large message to be stored in the cryptographic modules. Thus, ciphertext blocks are output by the module before having received the whole message. In another context, during talks over the internet, some ciphertext blocks need to be sent before the end of the communication. Practical applications are thus far from the theoretical model. Recently, in [10], Joux, Martinet and Valette have proposed to change the adversary interactions with the encryption oracle to better model *on-line* symmetric encryption schemes. An on-line encryption scheme is a scheme which is able to output the ciphertext block $C[i]$ just after the introduction of the block $M[i]$, without having the knowledge of the whole message. Many modes of operation have this nice property. Therefore, from an attacker point of view, adversaries in the IND security game can adapt the message blocks $M_0[i]$ and $M_1[i]$ according to the previously received ciphertext blocks. The same notion concerning integrity on real-time applications has been used by Gennaro and Rohatgi[4].

The blockwise adversarial model, presented in [10], is used to break the IND-CPA security of some encryption schemes, provably secure in the standard model. For example, in order to encrypt a message $M = M[1]M[2]\dots M[\ell]$ with the CBC encryption mode, a random initial vector $C[0] = IV$ is chosen and for all $1 \leq i \leq \ell$, $C[i] = E(M[i] \oplus C[i-1])$. In [2], Bellare *et al.* have shown that in the standard model, the CBC encryption scheme is IND-CPA secure up to the encryption of $2^{n/2}$ blocks, where n denotes the length of the block cipher E. However, in [10], Joux *et al.* have shown that the CBC encryption mode cannot be IND secure in the blockwise adversarial model: only two-blocks messages M_0 and M_1 allow the adversary to win the semantic security game. Indeed, if the same input is given twice to the block cipher, the same result is output. Consequently in the IND security game, if the adversary knows the initial vector $C[0] = IV$ and the first ciphertext block $C[1]$, he can adaptively choose $M_0[2]$ as $C[1] \oplus C[0] \oplus M_0[1]$ and a random value for $M_1[2]$. Then, if the second ciphertext block $C[2]$ is such that $C[2] = C[1]$, then the ciphertext $C = C[0]C[1]C[2]$ is the encryption of M_0. Otherwise it is the encryption of M_1. This attack works since the adversary can adapt his message blocks according to the output blocks. In the standard model, as the messages are chosen before the ciphertext is returned by the oracle, the probability that a such collision occurs in the inputs of the block cipher is upperbounded by $\mu^2/2^n$, where μ denotes the number of encrypted blocks with the same key. While μ remains small enough, the probability is negligible and the mode of encryption is secure.

In [10], Joux *et al.* have only described attacks against symmetric and hybrid encryption schemes, in the indistinguishability sense. From a practical point of view, the blockwise attack on the CBC encryption scheme is as efficient as an attack on the ECB encryption scheme in the standard model. Indeed, for both the ECB mode in the standard model and the CBC mode in the blockwise model, the adversary knows inputs and outputs of the block cipher. For ECB, he can then adapt his messages to force a collision. For the CBC, he adapts the message blocks. As a consequence, in both cases a key recovery attack on the block cipher is possible. Such an attack only requires the encryption of some chosen plaintext blocks. For example, a dictionary attack on the block cipher can be mounted (see for example [12]). In this kind of attacks, the adversary has a table where he has precomputed the encryption of the plaintext block P

under all the keys. Therefore, if he knows the encryption of P under the key used in the block cipher, he just looks in his table to recover the secret key. Moreover, the time/memory tradeoff of Hellman [9] can be adapted to reduce the required memory of this attack. Therefore, blockwise attacks need to be taken into account in practical uses since attacks are not only theoretical but paves the way to more practical and serious attacks.

In this paper we study the security of some well known encryption mode against blockwise adversaries. In a first part we show how to secure the CBC encryption mode in the blockwise model. The countermeasure we propose simply consists in delaying the output blocks. This modification allows to prove this DCBC (Delayed CBC) encryption mode secure against blockwise adaptive adversaries, mounting chosen plaintext attacks. Furthermore, it can be applied to secure several modes of operations. In a second part, we show that the CFB (Ciphertext FeedBack) encryption mode is secure without any change in this new model. We also give a rigorous proof for the security of the DCBC and CFB modes, in appendices.

2 Preliminaries

2.1 On-line Symmetric Encryption

Since the call for proposition from the NIST, the interest for the security of modes of operation has been renewed. Several new schemes have been proposed such as OCB [14], IACBC [11, 7], XCBC [5]. Most of these new modes of operation have the advantage of providing integrity and confidentiality in a single pass and have been called *authenticated encryption schemes*. Confidentiality and integrity of encryption schemes are interesting since their combination allows to construct chosen-ciphertext secure schemes.

A symmetric encryption scheme $\mathcal{SE}$ is a triple of algorithm $(\mathcal{K}, \mathcal{E}, \mathcal{D})$. $\mathcal{K}$ is the probabilistic key generation algorithm, taking as input a security parameter k and outputting a key K. The encryption algorithm $\mathcal{E}$ takes as input a plaintext $M = M[1] \ldots M[\ell]$ where $M[i]$ is an n-bit string, and the key K of length k bits and outputs the corresponding ciphertext C. $\mathcal{E}$ can be deterministic or probabilistic. $\mathcal{E}$ is said *on-line* if the ciphertext block $C[i]$ does not depend on the plaintext blocks $M[i+1] \ldots M[\ell]$. The decryption algorithm $\mathcal{D}$ is a deterministic algorithm that takes as input a candidate ciphertext C and outputs either the corresponding plaintext M, or a predefined symbol $\perp$ meaning the ciphertext was invalid (with invalid length for example). For every message M, we have $\mathcal{D}_K(\mathcal{E}_K(M)) = M$. Usually such symmetric encryption schemes are based on block ciphers. A block cipher E_K is a secret key permutation taking as input a plaintext of size n bits and a k-bit key K, and outputting a n-bit ciphertext block.

2.2 Security notions

Security notion for a symmetric encryption scheme is usually viewed as indistinguishability of encryptions, against chosen plaintexts attacks. This security notion has been redefined as the left-or-right indistinguishability. We use this latter notion since it has been shown stronger than indistinguishability of encryptions in [2]. For this security notion, mainly studied in [2], the adversary has access to an encryption oracle he feeds with queries of the form (M_0^i, M_1^i), where M_0^i and M_1^i are two equal length messages, chosen adaptively according to the previous responses. The oracle, called a *Left-or-Right* oracle and denoted $\mathcal{LR}$, flips a bit b at the beginning of the game, and always encrypts messages M_b^i accordingly. The adversary goal is to correctly guess the bit b. A symmetric encryption scheme is said $\mathsf{LOR\text{-}CPA}$ secure, if any polynomial adversary can guess the bit b only with negligible advantage. Note that one can also define the security of an encryption scheme against chosen ciphertext attacks. In this scenario, the attacker has also access to a decryption oracle. However, we do not look at this security notion here. Indeed, most of the encryption modes are not secure against such adversaries and chosen plaintext attacks are sufficient to ensure confidentiality for a scheme. Thus, we focus here on the left-or-right indistinguishability against chosen plaintext attacks.

However, the recent attacks on some schemes, proved secure in this model, show that a new adversarial model has to be defined. The new kind of adversaries, introduced in [10], are adaptive during a query, according the previous blocks of ciphertext. The security model has to take into account these adversaries, realistic in an implementation point of view. In the blockwise model, the adversary is allowed to feed the oracle block by block. Thus adversaries are now more adaptive than in the standard model. Formally, the adversary is given access to a blockwise encryption *left-or-right* oracle: this oracle is queried with inputs of the form $(M_0^i[j], M_1^i[j])$, where $M_0^i[j]$ and $M_1^i[j]$ are two plaintext blocks. At the beginning of the game, this oracle flips at random a bit b. Then, if $b = 0$ it will always encrypt the left plaintext blocks, and otherwise, if $b = 1$ it will encrypt the right ones. The corresponding ciphertext blocks $C_b^i[j]$ are returned to the adversary, whose goal is to guess which message has been encrypted. The difference with the standard model is that here the queries are made on the fly: for each plaintext block received, the oracle outputs a ciphertext block. This better modelizes on-line encryption. Thus, it is natural to consider a new kind of interactions, induced by this model: since the adversary does not send the whole plaintext in a single query, so that he can adapt the next plaintext block according to the ciphertext he receives, one can also suppose that the adversary may interleave the queries. In this case, the attacker is able to query the oracle for the encryption of a new message, even if the previous encryption is not finished. This introduces *concurrent* queries. The blockwise encryption *left-or-right* oracle takes as input now the two plaintext blocks $M_0^i[j]$ and $M_1^i[j]$ along with the index i of the query. Thus the oracle encrypts $M_b^i[j]$ depending on the previous plaintext and ciphertext blocks of the query i. The security model is thus modified in depth and security of known schemes has to be carefully re-evaluated in this new model.

3 Blockwise Secure Encryption Schemes

In this section, we propose modes of encryption, inspired from classical ones, that enable to withstand blockwise adversaries. The choice of those modes has been made in order to show that some well-known and simple modes, such as CFB, and variants of others, such as CBC, are secure even against the powerful adversaries we consider. The complete security proof are given in appendix and we only summerize in this section the security results and there implications on the use of those modes of encryption.

3.1 A blockwise secure variant of the CBC: the Delayed CBC

The CBC mode of encryption, probably one of the most currently used in practical applications, suffers from strong weaknesses in the blockwise adversary model, as it has been shown in [10]. The main reason is that the security of modes of operation is closely related to the probability of collision in the inputs of the underlying block cipher. As shown by the attacks presented in [10], blockwise adversaries can choose the message blocks according to the previously revealed ciphertext blocks so that they can force such a collision. This kind of adversaries are realistic if the output blocks are gradually released outside the cryptographic component.

A simple countermeasure, we propose in order to prevent an adversary from having access to the previous ciphered block, is to delay the output by one single block. Consequently, an attacker can no longer adapt the message blocks.

More precisely, we slightly modify the encryption algorithm in such a way that the encryption module delays the output by one block, i.e., instead of outputting $C[i]$ just after the introduction of $M[i]$, $C[i]$ is output after the introduction of $M[i+1]$. This modification in the encryption process is efficient

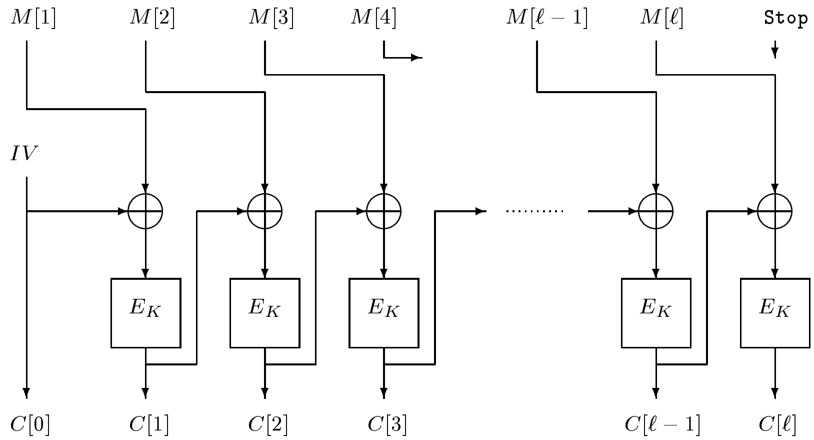

Fig. 1. The Delayed CBC encryption mode.

and does not require any modification of the scheme; ciphertexts produced by a device implementing the delayed CBC mode are compatible with those produced by standard ones.

A detailed description for this scheme, called Delayed CBC or simply DCBC, is given below and is also depicted in figure 1. We assume that each block is numbered from 1 to ℓ and that the end of the encryption is indicated by sending a special block $M[\ell+1] =$`stop`. We also assume that if the encryption algorithm does not have to output a block, it sends an acknowledgment "`Ack`". Of course, the index i is only given to simplify the description of the algorithm but in practice this counter should be handled by the encryption module. In other words, we do not consider attacks based on false values of i since they do not have any practical significance. We also note $IV \stackrel{R}{\leftarrow} \{0,1\}^n$ the random choice of an n-bit block IV.

Function $\mathcal{E} - \mathrm{DCBC}^f(K, M[i], i)$
If $i = 1$,
  $IV \stackrel{R}{\leftarrow} \{0,1\}^n$, $C[0] = IV$
  Return $C[0]$
Else If $M[i] =$`stop`
  Return $C[i-1]$
Else
  $C[i] = f(K, C[i-1] \oplus M[i])$
  Return $C[i-1]$

Function $\mathcal{D} - \mathrm{DCBC}^f(K, C[i], i)$
If $i = 0$,
  Return `Ack`
Else
  Return $C[i-1] \oplus f^{-1}(K, C[i])$

Note that the decryption process is unchanged compared to the standard CBC encryption mode. Indeed, there is no need to delay the output block in the decryption phase since the adversary is not given any access to a decryption oracle for chosen plaintext attacks. Furthermore, since the DCBC does not provide chosen ciphertext security, for both the standard and the blockwise model, the decryption does not need to be modified.

Blockwise Security of the DCBC Encryption Mode. In annex B, we analyze the security of the DCBC against blockwise concurrent adversaries mounting chosen plaintext attacks. Intuitively, it is easy to see that a blockwise adversary cannot adapt the plaintext blocks according to the previously returned ciphertext blocks since it does not know $C[i-1]$ when submitting $M[i]$. Furthermore, the knowledge of the previous blocks $C[0], \dots, C[i-2]$ does not help him to predict the i-th input $C[i-1] \oplus M[i]$ of the block cipher as long as the total number μ of encrypted blocks with the same key K is not too large. The security proof shows that the advantage of an adversary is at most increased by a term $\mu^2/2^n$. In other words, DCBC is provably secure in the blockwise model, assuming the security of the underlying block cipher, while the total number of encrypted blocks with the same key is much smaller than $2^{n/2}$.

It is important to note that this security bound is similar to the one obtained in the standard model for the CBC mode [2]. This means that the delayed CBC is as secure in the blockwise model as the classical CBC encryption scheme in the standard model.

Security of other schemes with the delay countermeasure. We have analyzed the security of the CBC, with one block output delay, against concurrent blockwise adversaries. It is worth noticing that such a modification on a scheme can be considered as a generic way to secure some particular encryption modes against blockwise adversary. Indeed, if we consider modes of operation where the encryption of the next input block only depends on the key and on the last ciphertext block such as HCBC or HPCBC [1] randomized with an initialization vector, the time delay modification can be applied. As for the DCBC, an adversary cannot adapt the next plaintext block to control the block cipher input. However, this countermeasure cannot be applied to secure any encryption modes. For example, if a scheme is such that a ciphertext block depends on the two previous ones, a one block delay could not be sufficient to secure the scheme against blockwise adaptive adversaries. This is also the case if the block cipher inputs are blinded with non independent mask values such as in IAPM [11], IACBC [11], XCBC [5] and OCB [14].

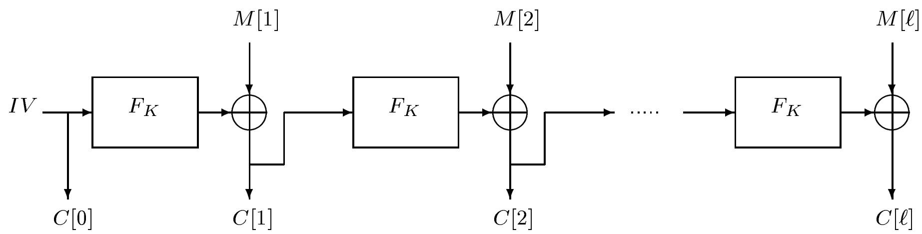

Fig. 2. The CFB encryption mode.

3.2 CFB encryption scheme

A review of the most classical modes of operation shows that one of them, the CFB mode, is naturally immune against blockwise attacks.

The CFB encryption mode is based on a function F, indexed by a key K. This function F does not need to be a permutation, i.e., does not need to be implemented using a block cipher. In the following, $F_K(.)$ will be denoted by $f(K,.)$. A detailed description for this scheme is given in the following and also depicted in figure 2, using the same conventions as for DCBC.

Function $\mathcal{E} - \mathrm{CFB}^f(K, M[i], i)$
If $i = 1$,
  $IV \stackrel{R}{\leftarrow} \{0,1\}^n$, $C[0] = IV$
  $C[1] = f(K, C[0]) \oplus M[1]$
  Return $C[0]$ and $C[1]$
Else
  $C[i] = f(K, C[i-1]) \oplus M[i]$
  Return $C[i]$

Function $\mathcal{D} - \mathrm{CFB}^f(K, C[i], i)$
If $i = 0$,
  Return `Ack`
Else
  Return $C[i] \oplus f(K, C[i-1])$

We insist on the fact that we have not modified the original CFB mode and that we only recall it in order to be complete.

Blockwise Security of the CFB Encryption Mode. In annex C, we analyze the security of the CFB against blockwise concurrent adversaries mounting chosen plaintext attacks. Intuitively, a blockwise adversary cannot adapt the plaintext blocks in order to force the input to the function f while the ciphertext blocks are all pairwise distinct. If no adaptive strategy is efficient, the inputs of f behave like random values and the system is secure until a collision at the output of this function occurs. If the total number μ of encrypted blocks with the same key K is not too large, i.e., much smaller than the square root of 2^n, this event only happens with negligible probability. The security proof formalizes those ideas and shows that the advantage of an adversary is at most increased by a term $\mu^2/2^n$, as for DCBC. In other words, the CFB mode is provably secure in the blockwise model, assuming the security of the underlying block cipher, while the total number of encrypted blocks with the same key is much smaller than $2^{n/2}$.

Such a bound is tight since practical attacks against the indistinguishability of the mode can be mounted if more than $2^{n/2}$ blocks are encrypted. In practice, notice that using 64-bit block ciphers such as DES or triple-DES, this bound of 2^{32} blocks could be quickly reached in some applications based on high speed networks.

We can use a block cipher rather than a pseudorandom function in the CFB mode as it is specified in [13]. Indeed, a secure block cipher behaves like a pseudorandom function up to the encryption of $2^{n/2}$ blocks.

Truncated CFB. We can also show that if the block cipher output is truncated to m bits ($m \leq n$), we denote by T^m this function, then the T^m family can be proven indistinguishable from a truly random function up to the encryption of $2^{n-(m/2)}$ blocks. This construction of pseudorandom function from pseudorandom permutation has been proposed by Hall *et al.* in [8] and has been proved by Bellare and Impagliazzo in [3]. The scheme resulting from the truncation of the output of a block cipher is called CFB-m. Consequently, the analysis performed in appendix C can also be applied to this scheme and gives a better security result. However, even this construction is only secure up to the encryption of $2^{n/2}$ blocks.

4 Conclusion and Open Problems

This article shows that for a special case of encryption schemes, called on-line encryption schemes, a new kind of adversary model has to be used in order to better model the adversary abilities. Moreover, we prove that some "old" encryption schemes prevent these adversaries more efficiently than newly presented schemes. Blockwise adversaries are important and must be taken into account in the referee process of the NIST concerning the modes of operation. For on-line applications, some "old" schemes, such as CFB, are more secure than the CBC encryption mode. We also prove a variant of a CBC on-line encryption immune against such adversary.

It seems however hard to prove the security of OCB and other CBC variants under the same assumptions as those used in the classical adversarial models. We left as an open problem these proofs.

References

1. M. Bellare, A. Boldyreva, L. Knudsen, and C. Namprempre. On-Line Ciphers and the Hash-CBC Construction. In J. Kilian, editor, *Advances in Cryptology – Crypto'01*, volume 2139 of *LNCS*, pages 292 – 309. Springer-Verlag, 2001.
2. M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *Proceedings of the 38th Symposium of Fundations of Computer Science*. IEEE, 1997.
3. M. Bellare and R. Impagliazzo. A tool for obtaining tighter security analysis of pseudorandom function based constructions, with applications to PRP $\rightarrow$ PRF conversion. February 1999, available at `http://www-cse.ucsd.edu/users/russell`.
4. R. Gennaro and P. Rohatgi. How to Sign Digital Streams. In B. Kaliski, editor, *Advances in Cryptology – Crypto'97*, volume 1294 of *LNCS*, pages 180 – 197. Springer-Verlag, 1997.
5. V.D. Gligor and P. Donescu. Fast Encryption and Authentication: XCBC and XECB Authentication Modes. In *Fast Software Encryption*, LNCS. Springer-Verlag, 2001.
6. S. Goldwasser and S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.
7. S. Halevi. An Observation regarding Jutla's modes of operation. Crytology ePrint archive, Report 2001/015, available at `http://eprint.iacr.org`.
8. C. Hall, D. Wagner, J. Kelsey, and B. Schneier. Building PRFs from PRPs. In H. Krawczyk, editor, *Advances in Cryptology – Crypto'98*, volume 1462 of *LNCS*, pages 370 – 389. Springer-Verlag, 1998.
9. M. E. Hellman. A Cryptanalytic Time-Memory Trade-Off. *IEEE Transactions on Information Theory*, IT-26(4):401 – 406, 1980.
10. A. Joux, G. Martinet, and F. Valette. Blockwise-Adaptive Attackers. Revisiting the (in)security of some provably secure Encryption Modes: CBC, GEM, IACBC. In M. Yung, editor, *Advances in Cryptology – Crypto'02*, volume 2442 of *LNCS*, pages 17 – 30. Springer-Verlag, Berlin, 2002.
11. C. Jutla. Encryption modes with almost free message integrity. In B. Pfitzmann, editor, *Advances in Cryptology – Eurocrypt'01*, volume 2045 of *LNCS*, pages 529 – 544. Springer-Verlag, 2001.
12. A. Menezes, P. van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.
13. NIST. FIPS PUB 81 - DES Modes of Operation, December 1980.
14. P. Rogaway, M. Bellare, J. Black, and T. Krovetz. OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption. In *Eighth ACM conference on Computer and Communications Security*. ACM Press, 2001.

A Preliminaries to the Security proofs

A.1 Notations

In the sequel, standard notations are used to denote probabilistic algorithms and experiments. If A is a probabilistic algorithm, then the result of running A on inputs $x_1, x_2, \ldots$ and coins r will be denoted by $A(x_1, x_2, \ldots; r)$. We let $y \leftarrow A(x_1, x_2, \ldots; r)$ denote the experiment of picking r at random and letting y be $A(x_1, x_2, \ldots; r)$. If S is a finite set then $x \leftarrow S$ is the operation of picking an element uniformly from S. If α is neither an algorithm nor a set then $x \leftarrow \alpha$ is a simple assignment statement. We say that y *can be output by* A if there is some r such that $A(x_1, x_2, \ldots; r) = y$. If $p(x_1, x_2, \ldots)$ is a predicate, the notation $\Pr[x_1 \leftarrow S; x_2 \leftarrow A(x_1, y_2, \ldots); \ldots : p(x_1, x_2, \ldots)]$ denotes the probability that $p(x_1, x_2, \ldots)$ is true after ordered execution of the listed experiments. Recall that a function $\varepsilon : \mathsf{N} \to \mathsf{R}$ is *negligible* if for every constant $c \geq 0$ there exists an integer k_c such that $\varepsilon(k) \leq k^{-c}$ for all $k \geq k_c$. The set of all functions from $\{0,1\}^m$ to $\{0,1\}^n$ is denoted by $\mathcal{R}^{m \to n}$. The set of all the permutations of $\{0,1\}^n$ is denoted by Perm_n.

A.2 Security Model

Security of a symmetric encryption scheme is viewed as indistinguishability of the ciphertexts, when considering chosen plaintext attacks. In this model, the adversary, denoted by $\mathcal{A}$ in the sequel, is given access to a blockwise concurrent encryption *left-or-right* oracle: this oracle is queried with inputs of the form $(M_0^i[j], M_1^i[j])$, where $M_0^i[j]$ and $M_1^i[j]$ are two plaintext blocks. At the beginning of the game, this oracle flips at random a bit b. Then, if $b = 0$ it will always encrypt $M_0^i[j]$, and otherwise, if $b = 1$ it will encrypt $M_1^i[j]$. The corresponding ciphertext block $C_b^i[j]$ is returned to the adversary, whose goal is to guess which message has been encrypted. Here the queries are made on the fly (for each plaintext block received, the oracle outputs a ciphertext block), and also concurrently (the adversary may interleave the queries). In this case, $\mathcal{A}$ is able to query the oracle for the encryption of messages, even if the previous encryption is not finished. This introduces *concurrent* queries. Thus, we define the encryption *left-or-right* oracle, denoted by $\mathcal{E}_K^{bl,c}(\mathcal{LR}(.,.,b,i))$, to take as input two plaintext blocks $M_0^i[j]$ and $M_1^i[j]$ along with the number i of the query, and encrypt $M_b^i[j]$. We now give the formal description of the attack scenario:

$$\begin{array}{l} \mathsf{Expt}_{\mathcal{SE},\mathcal{A}}^{\mathrm{lorc-bcpa}(b)}(k) \\ \quad K \stackrel{R}{\leftarrow} \mathcal{K}(k) \\ \quad d \leftarrow \mathcal{A}^{\mathcal{E}_K^{bl,c}(\mathcal{LR}(\cdot,\cdot,b,\cdot))} \\ \quad \text{Return } d \end{array}$$

The adversary advantage in winning the $\mathsf{LORC\text{-}BCPA}$ game is defined as:

$$\mathbf{Adv}_{\mathcal{SE},\mathcal{A}}^{\mathrm{lorc-bcpa}}(k) = \left| 2 \cdot \Pr[\mathsf{Expt}_{\mathcal{SE},\mathcal{A}}^{\mathrm{lorc-bcpa(b)}}(k) = 1] - 1 \right|$$

We define $\mathbf{Adv}_{\mathcal{SE}}^{\mathrm{lorc-bcpa}}(k, t, q, \mu) = \max_{A} \{\mathbf{Adv}_{\mathcal{SE},A}^{\mathrm{lorc-bcpa}}(k)\}$, where the maximum is over all legitimate A having time-complexity t, making to the concurrent oracles at most q encryption queries totaling μ blocks. A secret-key encryption scheme $\mathcal{SE}$ is said to be *lor-secure against concurrent blockwise adaptive chosen plaintext attack* ($\mathsf{LORC\text{-}BCPA}$), if for all polynomial-time probabilistic adversaries, the advantage in this guessing game is negligible as a function of the security parameter k. In this case, $\mathcal{SE}$ is said $\mathsf{LORC\text{-}BCPA}$ secure.

The security of a block cipher is viewed as the indistinguishability from random permutations, as defined for example in [2]. The attack scenario for the adversary is to distinguish the outputs of a permutation randomly chosen in Perm_n, from the outputs of a permutation randomly chosen in the family $\mathcal{P}$ of all permutations induced by a given block-cipher. The adversary advantage in winning this game is denoted by $\mathbf{Adv}_{\mathcal{P}}^{\mathrm{prp}}(k,t,q)$. Following the same idea, the security of a pseudo-random function f randomly chosen in a given family $\mathcal{F}$ of functions of input length m and output length n, is the indistinguishability from a random function of $\mathcal{R}^{m\rightarrow n}$. The attacker game is the same as above, except that permutations are replaced by functions. The adversary advantage in winning the game in denoted by $\mathbf{Adv}_{\mathcal{F}}^{\mathrm{prf}}(k,t,q)$.

B Security Proof for the DCBC encryption scheme

The following theorem gives the security bound for the DCBC encryption scheme, as defined in section 3.1.

Theorem 1. *Let $\mathcal{P}$ be a family of pseudorandom permutations of input and output length n where each permutation is indexed with a k-bit key. If the function f used in the DCBC mode is drawn at random in the family $\mathcal{P}$, then the DCBC encryption scheme is $\mathsf{LORC\text{-}BCPA}$ secure. Furthermore, for any t, q and $\mu < 2^{n-1}$, we have:*

$$\mathsf{Adv}_{\mathrm{DCBC}}^{\mathrm{lorc-bcpa}}(k,t,q,\mu) \leq 2 \cdot \mathsf{Adv}_{\mathcal{P}}^{\mathrm{prp}}(k,t,\mu) + \frac{\mu^2}{2^{n-1}}$$

Proof. The proof goes by contradiction. Assume for the sake of contradiction that there exists an adversary $\mathcal{A}$ against the DCBC encryption scheme with non-negligible advantage. From this adversary, we construct an attacker $\mathcal{B}$ that can distinguish the block cipher E_K used in the DCBC and chosen at random in the family $\mathcal{F}$ from a random permutation with non-negligible advantage. More precisely, the attacker $\mathcal{B}$ interacts with a permutation oracle that chooses a bit b and if $b = 1$, chooses f as a permutation in the set of all permutations Perm_n. Otherwise, if $b = 0$, it runs the key generation $\mathcal{K}(1^k)$, obtains a key $\mathcal{K}$ and sets f as $E_{\mathcal{K}}$. The goal of $\mathcal{B}$ is to guess the bit b with non-negligible advantage. To this end, $\mathcal{B}$ uses the adversary $\mathcal{A}$ and consequently $\mathcal{B}$ has to simulate the environment of the adversary $\mathcal{A}$.

First, $\mathcal{B}$ chooses a bit b' at random and runs $\mathcal{A}$. $\mathcal{B}$ has to concurrently answer the block encryption queries of the LORC game. When $\mathcal{A}$ submits pairs of input block $(M_0^i[j], M_1^i[j])$, $\mathcal{B}$ always encrypts the block $M_{b'}^i[j] \oplus C_{b'}^i[j-1]$ under the DCBC encryption mode thanks to the permutation oracle, yielding $C_{b'}^i[j]$, and returns $(C_{b'}^i[j-1])$ to $\mathcal{A}$. Finally, $\mathcal{A}$ will return a bit b'' and if $b' = b''$, then $\mathcal{B}$ returns $b^* = 0$, otherwise, $\mathcal{B}$ returns $b^* = 1$ to the oracle. The advantage of $\mathcal{A}$ in winning the LORC game is defined as: $\mathbf{Adv}_{DCBC,\mathcal{A}}^{\mathrm{lorc-bcpa}}(k) = 2 \cdot \Pr[b' = b'' | f \leftarrow E_K] - 1$.

It is easy to verify that the attacker $\mathcal{B}$ can simulate the concurrent lor-encryption oracle to adversary $\mathcal{A}$ since $\mathcal{B}$ has access to a permutation f and $\mathcal{B}$ can simulate the encryption mode of DCBC. The advantage for $\mathcal{B}$ in winning his game is defined as:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{P},\mathcal{B}}^{\mathrm{prp}}(k) &= |\Pr[b^* = 0 | b = 0] - \Pr[b^* = 0 | b = 1]| \\
&= |\Pr[b'' = b' | b = 0] - \Pr[b'' = b' | b = 1]| \\
&= \Pr[b'' = b' | f \leftarrow E_K] - \Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n] \\
&\geq \frac{1 + \mathbf{Adv}_{DCBC,\mathcal{A}}^{\mathrm{lorc-bcpa}}(k)}{2} - \Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n]
\end{aligned}$$

Let us now analyze $\Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n]$. We denote by D the event that all the inputs on the f permutation are distinct. Thus we have:

$$\begin{aligned}
\Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n] &= \Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n \wedge \mathsf{D}] \cdot \Pr[\mathsf{D}] \\
&\quad + \Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n \wedge \bar{\mathsf{D}}] \cdot \Pr[\bar{\mathsf{D}}] \\
&= 1/2 \cdot \big(1 - \Pr[\bar{\mathsf{D}}]\big) + \left(1 - \frac{1}{2^n}\right) \cdot \Pr[\bar{\mathsf{D}}]
\end{aligned}$$

This last equation comes from the fact that if f is a permutation chosen at random from the set of all permutations and no collision occurs, outputs of f are independent of the input blocks $M_0^i[j]$ and $M_1^i[j]$ and the adversary $\mathcal{A}$ has no advantage in winning the LORC game. Therefore, $\Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n \wedge \mathsf{D}] = \frac{1}{2}$. Otherwise, if a collision occurs, there exists i, i', j, j' such that $(i, j) \neq (i', j')$ and $C_{b'}^i[j] = C_{b'}^{i'}[j']$, and then since $\mathcal{A}$ knows all the plaintexts blocks (M_0^i, M_1^i) and the corresponding ciphertext blocks $C_{b'}^i$, he can decide whether $M_0^i[j] \oplus M_0^{i'}[j'] = C_{b'}^i[j-1] \oplus C_{b'}^{i'}[j'-1]$ or whether $M_1^i[j] \oplus M_1^{i'}[j'] = C_{b'}^i[j-1] \oplus C_{b'}^{i'}[j'-1]$. However, with probability $1/2^n$, we have $M_0^i[j] \oplus M_0^{i'}[j'] = M_1^i[j] \oplus M_1^{i'}[j']$ if (M_0^i, M_1^i) are chosen at random. Thus in any way $\mathcal{A}$ wins his game in this case and we have $\Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n \wedge \bar{\mathsf{D}}] \leq \left(1 - \frac{1}{2^n}\right)$. So, we get:

$$\Pr[b'' = b' | f \leftarrow \mathsf{Perm}_n] \leq \frac{1}{2} + \left(\frac{1}{2} - \frac{1}{2^n}\right) \cdot \Pr[\bar{\mathsf{D}}]$$

Now, let us bound the probability that a collision occurs. The following lemma shows that if μ is the number of encrypted blocks, then $\Pr[\bar{\mathsf{D}}] \leq \frac{\mu(\mu-1)}{2^{n-1}}$ provided $\mu \leq 2^{n-1}$. Consequently, the advantage of the attacker $\mathcal{B}$ is related to the advantage of the adversary $\mathcal{A}$:

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{prp}}_{\mathcal{P},\mathcal{B}}(k) &\geq \frac{1 + \mathbf{Adv}^{\mathrm{lorc-bcpa}}_{DCBC,\mathcal{A}}(k)}{2} - \left(\frac{1}{2} + \left(\frac{1}{2} - \frac{1}{2^n}\right) \cdot \Pr[\bar{\mathsf{D}}]\right) \\
&\geq \frac{\mathbf{Adv}^{\mathrm{lorc-bcpa}}_{DCBC,\mathcal{A}}(k)}{2} - \left(\frac{1}{2} - \frac{1}{2^n}\right) \cdot \Pr[\bar{\mathsf{D}}]
\end{aligned}$$

Consequently, we obtain

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{lorc-bcpa}}_{DCBC,\mathcal{A}}(k) &\leq 2 \cdot \mathbf{Adv}^{\mathrm{prp}}_{\mathcal{P},\mathcal{B}}(k) + \left(1 - \frac{1}{2^{n-1}}\right) \cdot \Pr[\bar{\mathsf{D}}] \\
&\leq 2 \cdot \mathbf{Adv}^{\mathrm{prp}}_{\mathcal{P},B}(k) + \left(1 - \frac{1}{2^{n-1}}\right) \cdot \frac{\mu(\mu-1)}{2^{n-1}}
\end{aligned}$$

and the theorem follows.

To conclude the proof, we have to prove the following lemma.

Lemma 1. $\Pr[\bar{\mathsf{D}}] \leq \frac{\mu(\mu-1)}{2^{n-1}}$.

Proof. We note that $\Pr[\bar{\mathsf{D}}] = \Pr[\mathtt{Coll}_\mu]$ where $\mathtt{Coll}_\mu$ denotes the event that a collision occurs on the input of the function f during the encryption of the μ blocks. Consequently,

$$\begin{aligned}
\Pr[\mathtt{Coll}_\mu] &= \Pr[\mathtt{Coll}_\mu \wedge \overline{\mathtt{Coll}_{\mu-1}}] + \Pr[\mathtt{Coll}_\mu \wedge \mathtt{Coll}_{\mu-1}] \\
&= \Pr[\mathtt{Coll}_\mu | \overline{\mathtt{Coll}_{\mu-1}}] \cdot \Pr[\overline{\mathtt{Coll}_{\mu-1}}] + \Pr[\mathtt{Coll}_{\mu-1}] \\
&\leq \Pr[\mathtt{Coll}_\mu | \overline{\mathtt{Coll}_{\mu-1}}] + \Pr[\mathtt{Coll}_{\mu-1}] \\
&\leq \sum_{k=1}^{k=\mu} \Pr[\mathtt{Coll}_k | \overline{\mathtt{Coll}_{k-1}}]
\end{aligned}$$

We are going to prove that $\Pr[\mathtt{Coll}_k | \overline{\mathtt{Coll}_{k-1}}] = \frac{2(k-1)}{2^n - (k-1)}$. This represents the probability that a collision occurs in the input of the function f at the kth block given that no collision appeared before. We have $\Pr[\texttt{Coll}_k \wedge \overline{\texttt{Coll}_{k-1}}] = \frac{2(k-1)}{2^n}$ since there is $(k-1)$ choices of picking one out of the $2(k-1)$ previous different values of $M_b^i[j] \oplus C^i[j-1]$ (as no collision occurs before the $(k-1)$th step). The factor 2 is needed to take into account that a collision can occur with the two messages M_0 and M_1, since if a collision occurs in one of the two cases, it allows the adversary to win the left-or-right indistinguishability game. The adversary cannot force a collision in the kth block: indeed, he does not know the output of the $(k-1)$th block and this output of the function f is independent of the $(k-1)$th input known by the adversary. Furthermore, there are 2^n different values of $M^i[j] \oplus C^i[j-1]$.

We also have $\Pr[\overline{\texttt{Coll}_{k-1}}] = \frac{2^n-(k-1)}{2^n}$ since there are $2^n-(k-1)$ different values for $M^i[j] \oplus C^i[j-1]$ out of the 2^n choices (f is a permutation). Consequently, for $k = 1, \ldots, \mu$, we get:

$$\Pr[\texttt{Coll}_k | \overline{\texttt{Coll}_{k-1}}] = \frac{2(k-1)/2^n}{[2^n-(k-1)]/2^n} = 2 \cdot \frac{2(k-1)}{2^n-(k-1)}$$

Thus, if $\mu \leq 2^{n-1}$,

$$\begin{aligned}
\Pr[\texttt{Coll}_\mu] &\leq \sum_{k=1}^{k=\mu} \Pr[\texttt{Coll}_k | \overline{\texttt{Coll}_{k-1}}] = \sum_{k=1}^{k=\mu} \frac{2(k-1)}{2^n-(k-1)} = \sum_{k=0}^{k=\mu-1} \frac{2k}{2^n-k} \\
&\leq \sum_{k=0}^{k=\mu-1} \frac{2k}{2^n-2^{n-1}} = \sum_{k=0}^{k=\mu-1} \frac{2k}{2^{n-1}} = \frac{\mu(\mu-1)}{2^{n-1}}
\end{aligned}$$

and the lemma is proved. □

C Security Proof for the CFB encryption mode

The following theorem gives the security bound for the CFB encryption scheme against concurrent blockwise adaptive adversaries.

Theorem 2 (Security of the CFB mode of operation). *Let $\mathcal{F}$ be a family of pseudorandom functions with input and output length n, where each function is indexed with a k-bit key. If the CFB encryption scheme is used with a function f chosen at random in the family $\mathcal{F}$, then, for every integers $t, q, \mu \geq 0$ such that $\mu \leq 2^{n-1}$, we have:*

$$\mathbf{Adv}_{\mathcal{SE}}^{\mathrm{lorc-bcpa}}(k,t,q,\mu) \leq 2 \cdot \mathbf{Adv}_{\mathcal{F}}^{\mathrm{prf}}(k,t,\mu) + \frac{\mu^2}{2^{n-1}}$$

Proof. We consider an adversary $\mathcal{A}$ against the CFB, trying to win the LORC-BCPA security game. We will show that this adversary can be turned into an adversary $\mathcal{B}$ trying to distinguish the function F_K from a random function chosen in $\mathcal{R}^{n \to n}$. The attack scenario for $\mathcal{A}$ is as defined in annex A.2. $\mathcal{B}$ has to simulate for the environment of $\mathcal{A}$, by using his own oracle. Indeed, $\mathcal{B}$ has access to an oracle $\mathcal{O}_f$, defined as follows: in the beginning of the game, $\mathcal{O}_f$ picks at random a bit b. If $b = 0$ then he chooses at random a key K for the function F and lets $f \leftarrow F_K$. Otherwise, if $b = 1$, then f is a random function chosen in the set $\mathcal{R}^{n \to n}$ of all the function from $\{0,1\}^n$ into $\{0,1\}^n$. $\mathcal{B}$ has to guess with non negligible advantage the bit b.

We now precisely describe how the adversary $\mathcal{B}$ answers the encryption queries made by $\mathcal{A}$. First, $\mathcal{B}$ picks at random a bit b'. $\mathcal{A}$ feeds his encryption oracle with queries of the form $(M_0^i[j], M_1^i[j])$, where $M_b^i[j]$ is the j-th block of the i-th query. Note that queries can be interleaved, so that some of the previous queries are not necessarily finished at this step. When $\mathcal{B}$ receives such a query and if $j = 1$, then $\mathcal{B}$ picks at random a value R_i, sends it to $\mathcal{O}_f$ and receives $f(R_i)$. If $j \neq 1$, then $\mathcal{B}$ transmits $C_{b'}^i[j-1]$ to $\mathcal{O}_f$ and receives $f(C_{b'}^i[j-1])$. Finally, $\mathcal{B}$ returns $C_{b'}^i[j] = M_{b'}^i[j] \oplus f(C_{b'}^i[j-1])$ or $C_{b'}^i[1] = M_{b'}^i[1] \oplus f(R_i)$ to $\mathcal{A}$, according to the value j. At the end of the game, $\mathcal{A}$ returns a bit b'' representing its guess for the bit b'. Then, $\mathcal{B}$ also outputs a bit b^* representing his guess for the bit b chosen by $\mathcal{O}_f$ and such that $b^* = 0$ if $b' = b''$, and $b^* = 1$ otherwise. We have to evaluate $\mathbf{Adv}_{\mathcal{F}}^{\mathrm{prf}}(k)$. We have:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{F}}^{\mathrm{prf}}(k) &= |\Pr[b^* = 0 | b = 0] - \Pr[b^* = 0 | b = 1]| \\
&= |\Pr[b' = b'' | f \leftarrow \mathcal{F}] - \Pr[b' = b'' | f \leftarrow \mathcal{R}^{n \to n}]| \\
&\geq \frac{1 + \mathbf{Adv}_{CFB,A}^{\mathrm{lorc-bcpa}}(k)}{2} - \Pr[b' = b'' | f \leftarrow \mathcal{R}^{n \to n}] \qquad (1)
\end{aligned}$$

Thus, $\mathbf{Adv}_{CFB,A}^{\mathrm{lorc-bcpa}}(k) \leq 2 \cdot \mathbf{Adv}_{\mathcal{F}}^{\mathrm{prf}}(k) + 2 \cdot \Pr[b' = b'' | f \leftarrow \mathcal{R}^{n \to n}] - 1$ and it remains to upperbound $\Pr[b' = b'' | f \leftarrow \mathcal{R}^{n \to n}]$.

As for the previous proof for the security of the DCBC encryption scheme, we will look at the collisions that can occur in the inputs of the function f. Indeed, if no such collision appears, then the advantage for the adversary $\mathcal{A}$ in winning his game equals 0. However if such a collision occurs then the adversary can easily detect it and consequently he can adapt the following plaintext block, to distinguish which of the messages is encrypted. Thus, in this case, the adversary wins the game. We denote by `Coll` the event that some collision appears on the inputs of the function f. So we have:

$$\begin{aligned}
\Pr[b' = b''|f \leftarrow \mathcal{R}^{n\to n}] &= \Pr[b' = b''|f \leftarrow \mathcal{R}^{n\to n} \wedge \mathtt{Coll}] \cdot \Pr[\mathtt{Coll}] \\
&\quad + \Pr[b' = b''|f \leftarrow \mathcal{R}^{n\to n} \wedge \overline{\mathtt{Coll}}] \cdot \Pr[\overline{\mathtt{Coll}}] \\
&\leq \Pr[\mathtt{Coll}] + \Pr[b' = b''|f \leftarrow \mathcal{R}^{n\to n} \wedge \overline{\mathtt{Coll}}] \\
&\leq \Pr[\mathtt{Coll}] + \frac{1}{2} \qquad (2)
\end{aligned}$$

The last inequality come from the fact that if no collision occurs on the input of the function f, where f is a function chosen at random in $\mathcal{R}^{n\to n}$, then the outputs of this function are random values, uniformly distributed in $\{0,1\}^n$ and independent of the previous values. Thus, the adversary cannot adapt the following message block, according to the previous ciphertext blocks. Thus, the random guess is the unique strategy for him to guess the bit b'.

We have now to evaluate $\Pr[\mathtt{Coll}]$. As before, we denote by $\mathtt{Coll}_k$ the probability that a collision occurs on the $(k-1)$-th input of the function f. We have: $\Pr[\mathtt{Coll}_k] = \Pr[\exists\ 0 \leq l < k \text{ s.t. } C^i_{b'}[l] = C^i_{b'}[k]]$, where $C^i_{b'}[0] = R_i$. Thus, we have:

$$\Pr[\mathtt{Coll}] = \sum_{k=1}^{\mu} \Pr[\mathtt{Coll}_k|\overline{\mathtt{Coll}_{k-1}}]$$

For sake of clarity, in the following we omit the bit b' and the index i representing the number of the queries. We remark that: $C[l] = C[k]$ if and only if $M[l] \oplus f(C[l-1]) = M[k] \oplus f(C[k-1])$. This last equation can be verified either at random, or if the adversary can choose $M[k]$ so that $M[k] = M[l] \oplus f(C[l-1]) \oplus f(C[k-1])$. However, since by assumption $C[k-1]$ does not collide with any of the previous ciphertext block, $f(C[k-1])$ has never been computed and is then a random value, uniformly distributed in $\{0,1\}^n$ and independent of the previous computed values. Thus, the adversary cannot guess to adapt $M[k]$ accordingly, except with negligible probability. Finally, we can write that for all $1 \leq k \leq \mu$: $\Pr[\exists 0 \leq l < k \text{ s.t. } C[l] = C[k] \mid \overline{\mathtt{Coll}_{k-1}}] \leq 2 \cdot \frac{k-1}{2^n}$. Indeed, there is at most $k-1$ choices for the value l, and two messages are queried. Thus, by summing up all the values k, we have:

$$\Pr[\mathtt{Coll}] \leq \frac{\mu^2}{2^{n-1}}$$

Finally, by replacing all the probabilities involved in equations 1 and 2, we obtain:

$$\mathbf{Adv}^{\mathrm{prf}}_{F}(k,t,\mu) \geq \frac{\mathbf{Adv}^{\mathrm{lorc-bcpa}}_{CFB,\mathcal{A}}(k,t,q,\mu)}{2} - \frac{\mu^2}{2^{n-1}}$$

and the theorem follows. □