On-line Encryption Schemes: New Security Notions and Constructions

ALEXANDRA BOLDYREVA NUT TAESOMBUT

November 18, 2003

Dept. of Computer Science & Engineering, University of California, San Diego
9500 Gilman Drive, La Jolla, CA 92093, USA.
Email: {aboldyre,ntaesomb}@cs.ucsd.edu
URL: http://www-cse.ucsd.edu/users/{aboldyre,ntaesomb}

Abstract

We investigate new strong security notions for *on-line* symmetric encryption schemes, which are the schemes whose encryption and decryption algorithms operate "on-the-fly" and in one pass, namely can compute and return an output block given only the key, the current input block and the previous input and output blocks. We define the strongest achievable notion of privacy which takes into account both chosen-ciphertext attacks and the recently introduced blockwise-adaptive [14, 11] attacks. We show that all the schemes shown to be secure against blockwise-adaptive chosen-plaintext attacks are subject to blockwise-adaptive chosen-ciphertext attacks. We present an on-line encryption scheme which is provably secure under our notion. It uses any strong on-line cipher, the primitive introduced in [1]. We finally discuss the notion of authenticated on-line schemes and provide a secure construction.

Keywords: Symmetric encryption, on-line encryption, proofs of security.

Contents

1 Introduction

On-line encryption schemes. In this work we investigate strong security notions for on-line symmetric encryption schemes and analyze various constructions.

The on-line property of a function requires that each output block depend only on the key, the previous input and output blocks and the current input block. We say that an encryption scheme is on-line if for each key its encryption and decryption algorithms are *both* on-line[1]. We provide a more formal definition of on-line encryption schemes in Section 3. The on-line property allows schemes to be used in applications where encryption and decryption should perform "on-the-fly", in one pass, i.e. an output block should be returned as soon as the next input block is received. Example situations include the use of encryption modes in SSH protocol or when implemented in tamper-proof devices with low memory. Most of the existing symmetric encryption schemes, e.g. CBC, CTR modes, are on-line [2].

Privacy of on-line schemes. In the model for the standard notion of privacy (indistinguishability) against chosen-plaintext attacks (IND-CPA) [2] an adversary runs in two stages. In both stages it is given a symmetric encryption oracle. During the first stage the adversary has to output two equal-length messages. At the last stage it gets a challenge ciphertext, which is an encryption of one of the two messages. It wins if it can distinguish which message was encrypted.

As has been recently observed by Joux, Martinet and Valette in [14], the standard notion for IND-CPA security treats encryption and decryption as "atomic" operations and, therefore, is not strong enough for on-line encryption schemes, whose algorithms can be executed in the on-line ("non-atomic") manner[3]. For the same reason the existing proofs of security done in the standard model cannot guarantee security of an on-line scheme with on-line execution. To model the privacy notion of schemes with on-line encryption, [14] takes into account so-called "blockwise-adaptive attacks", when an adversary is allowed to generate two equal-length messages block-by-block, each time seeing the corresponding portion of the challenge ciphertext and being able to "adapt" the next blocks of the messages as a function of currently received ciphertext blocks.

The authors noted that blockwise-adaptive attacks do not seem to apply for fully parallelizable "not chained" encryption schemes such as CTR mode but showed that some popular chained encryption schemes, including CBC, are in fact insecure against such attacks. The notion of security of on-line encryption schemes against blockwise-adaptive chosen-plaintext attacks (IND-BLK-CPA) has been defined more formally by Fouque, Martinet and Poupard in [11]. They prove that CFB encryption mode [17] and DCBC, the modification to CBC scheme, are IND-BLK-CPA.

But nowadays preserving privacy only against chosen-plaintext attacks is not considered sufficient; it is desirable to achieve stronger notions of privacy, such as security against chosen-ciphertext attacks. The standard notion of privacy against chosen-ciphertext attacks (IND-CCA) is similar to the one against chosen-plaintext attacks (IND-CPA), except that the adversary is also given a decryption oracle with a restriction of not querying it on a challenge ciphertext. IND-CCA notion seems insufficient for on-line schemes since it does not allow blockwise-adaptive attacks. Security of on-line schemes against blockwise-adaptive chosen-ciphertext attacks has not been previously investigated. This is the focus of our work.

[1] Note that in the literature on-line encryption schemes require *only* the encryption algorithm to be on-line.

[2] These modes can also be executed in non on-line manner, outputting the whole plaintext (resp. ciphertext) only after the encryption (resp. decryption) algorithm completes.

[3] If an on-line scheme is implemented only for "atomic" (not on-line execution), then the standard notions of security are appropriate. In this paper when we refer to an on-line scheme we always assume the possibility of "non-atomic" execution of both encryption and decryption algorithms of the scheme (when the output blocks are returned on-the-fly.)

Related work. Very recently in their independent work Fouque et al. [10] investigated the notions of secure authenticated encryption for applications using low memory highly-secure devices such as smart cards together with insecure but powerful host devices. The applications have been topics of previous research known as "remotely-keyed encryption" (RKE) and "remotely-keyed authenticated encryption" (RKAE). The works in these areas include [6, 16, 7, 13, 8]. The neat scheme proposed in [10] allows a smart card to partially decrypt a ciphertext produced by any secure authenticated encryption scheme (most of which have on-line encryption algorithms) in on-line manner while preserving privacy and authenticity. However, the host needs to perform a second pass over the data to complete the decryption (this phase is keyless). Hence, their scheme (as well as the schemes of [6, 16, 7, 13, 8]) does not fall under our definition of an on-line scheme, since their decryption algorithm is not done in one-pass.

The authors of [10] attempt to define the notion of privacy against blockwise-adaptive chosen-ciphertext attacks. While their notion is appropriate for schemes with on-line encryption, it is unachievable for schemes with on-line decryption, as we show below. Therefore, a new security notion is needed for analysis of on-line encryption schemes beyond chosen-plaintext attacks.

Towards a stronger notion of privacy for on-line schemes. As we mentioned above we seek for an appropriate notion of privacy under blockwise-adaptive chosen-ciphertext attacks for on-line encryption schemes and secure constructions. It turns out that finding the answers for these interesting theoretical and practically important questions requires some extra care.

For example, the approach taken by Fouque et al. [10] was to strengthen the IND-BLK-CPA notion defined by [14, 11] by giving the adversary the particular decryption oracle which operates block-by-block, namely, takes ciphertext blocks and returns the corresponding plaintext blocks "on-the-fly". Indeed, this is the first definition that comes to mind. However, we point out that giving such a decryption oracle is equivalent to giving the adversary the standard decryption oracle. This can be realized by noting that unlike encryption, the decryption algorithm is always deterministic and any "blockwise-adaptive" queries $C[1], C[2], \ldots$ to the particular decryption oracle can be replaced by queries $C[1], C[1]||C[2], \ldots$ to the standard decryption oracle.

Moreover, we claim that extending the IND-BLK-CPA notion with chosen-ciphertext attacks by giving the adversary the decryption oracle (which is equivalent to the notion proposed in [10]) is too strong for analyzing on-line encryption schemes. We show that by noting that no on-line encryption scheme can be secure even under the standard IND-CCA notion. We justify this claim in Section 3 by presenting a simple adversary which does not use any blockwise-adaptive attacks. Therefore, the standard IND-CCA notion and, moreover, its straightforward extension to blockwise-adaptive attacks are unachievable for on-line schemes.

We seek a strongest *achievable* privacy notion for on-line encryption schemes. In Section 3 we provide such a notion. The model takes into account both blockwise-adaptive chosen-plaintext attacks using the ideas from [14, 11] and a class of chosen-ciphertext attacks by giving the adversary the *special* decryption oracle. The queries to this special decryption oracle are answered in such a way that the notion is achievable. Please refer to Definition 3.1 for the details of this security notion, which we call IND-BLK-CCA. One may argue that our notion does not exactly capture the whole class of chosen-ciphertext attacks. This is true, but one should realize that any scheme with on-line encryption and decryption execution is *always* subject to some chosen-ciphertext attacks as we claimed above and therefore the designers and implementors of on-line encryption schemes should not target for IND-CCA security. This does not mean, however, that IND-BLK-CPA security will be sufficient. Our notion captures the best possible security for schemes with on-line execution.

Attacks. In Section 4 we show that the schemes proved to resist blockwise-adaptive chosen-plaintext attacks (shown in [11] to be IND-BLK-CPA secure) are subject to blockwise-adaptive chosen-ciphertext attacks. Namely, we show that CFB and DCBC encryption modes are not IND-BLK-CCA secure. These attacks are very simple and are similar to the well-known IND-CCA attacks on these schemes. The result is not surprising since these schemes were not designed to resist such powerful attacks.

IND-BLK-CCA SECURE CONSTRUCTION. If one does not have a goal of having an on-line symmetric encryption scheme, it is easy to achieve the IND-CCA security. Having an IND-CPA secure symmetric encryption scheme $\mathcal{SE}$, e.g. CBC, and a message authentication code F strongly unforgeable against chosen-message attack (SUF-CMA secure), e.g. XCBC [5] or OMAC [12][4], one can obtain an IND-CCA secure scheme via "encrypt-then-MAC" paradigm [4]. Namely, in order to encrypt a message, first use the encryption algorithm of $\mathcal{SE}$ to get a ciphertext C and then apply the tag generation algorithm of F to C to get the tag τ. The resulting ciphertext is $C\|\tau$. In order to decrypt a ciphertext C', parse it as $C''\|\tau''$, verify the tag τ'', using the verification algorithm of F, and if it valid, decrypt C'' using the decryption algorithm of $\mathcal{SE}$ and return the result, otherwise, return a special symbol $\perp$. However, this "encrypt-then-MAC" paradigm does not allow to preserve the on-line property of the given encryption scheme. More precisely, the on-line property of decryption becomes violated: the resulting decryption algorithm needs the whole ciphertext to verify the tag. Even though a block of the *potential* output can be computed "on-the-fly", the decision whether it can be output cannot be made until the tag is verified. The same problem holds for specific authenticated encryption schemes such as, for example, OCB authenticated scheme [18], IACBC, IAPM [15] and XCBC [9], which are not constructed using the generic "encrypt-then-MAC" paradigm. They provide both privacy (IND-CPA security) and authenticity (INT-CTXT security) and, hence IND-CCA security [4] [5]. However, their decryption algorithm is not on-line, before being able to output the message, it first needs to verify the checksum, which depends on all the blocks of the message.

We look for an on-line encryption scheme provably secure against blockwise-adaptive chosen-ciphertext attacks. As a building block we use on-line ciphers, the notion introduced and analyzed by Bellare, Boldyreva, Namprempre and Knudsen in [1]. On-line ciphers are on-line, length-preserving permutations. Secure on-line ciphers are the ones which closely approximate truly random on-line length-preserving permutations. The authors [1] formally define the security notion for on-line ciphers against chosen-ciphertext attacks and present the construction called HPCBC, which is secure against chosen-ciphertext attacks assuming the underlying block cipher and the hash function are secure (the construction uses one block cipher and one keyed hash function invocation per input block). On-line ciphers are deterministic permutations and therefore are not even IND-CPA secure. The authors discuss how on-line ciphers can be used to obtain IND-CPA and INT-CTXT secure schemes. As we discussed above and as independently noted in [10], this does not immediately guarantee IND-BLK-CCA security.

In Section 5 we present our second main result. We give an on-line encryption scheme which is constructed using any on-line cipher. We prove it IND-BLK-CCA secure assuming the underlying cipher is a strong pseudorandom on-line permutation. Theorem 5.2 states the concrete security result. We next present a specific scheme based on HPCBC on-line cipher.

AUTHENTICATED ON-LINE ENCRYPTION SCHEMES. As we discussed above, all known schemes that provide authenticity (INT-CTXT security) are not on-line. Moreover, the standard notion

[4]These MACs are shown to be weakly unforgeable (WUF-CMA secure), however, for stateless deterministic MACs these two notions are equivalent.

[5]It is shown in [4] that if an encryption scheme is secure against chosen-plaintext attacks (IND-CPA secure) and provides integrity of a ciphertext (INT-CTXT secure) then it is also secure against chosen-ciphertext attacks (IND-CCA secure).

of authenticity INT-CTXT is not appropriate to analyze on-line encryption schemes because it assumes that decryption is an atomic operation. An on-line decryption algorithm, however, needs to process the ciphertext and output the message block-by-block before the validity of the whole ciphertext can be verified. In Section 6 we define an appropriate notion of authenticity for on-line encryption schemes.

We next provide an encryption scheme which simultaneously provides authenticity and IND-BLK-CCA security. Again, it uses on-line ciphers. The scheme is very similar to the construction of an authenticated encryption scheme from [1]. We modify it so its decryption algorithm is on-line. We prove the security of our scheme assuming security of the underlying cipher against chosen-ciphertext attacks using the proofs of [1] and of Theorem 5.2. In this paper we define an appropriate notion of authenticity for on-line encryption schemes and provide an encryption scheme which simultaneously provides authenticity and IND-BLK-CCA security.

2 Preliminaries

NOTATION. A *string* is a member of $\{0,1\}^*$. Let $|X|$ denote the length of a string X. For strings X, Y let $X||Y$ denote their concatenation. The empty string is denoted ε. If $d \geq 1, n \geq 1$, are integers, then $D_{d,n}$ denotes the set of all strings whose length is a positive multiple of n bits and at most dn bits (we borrow some notation from [1] for convenience). If $X \in D_{d,n}$, then $X[i]$ denotes its ith block, meaning $X = X[1]|| \ldots ||X[l]$ where $l = |X|/n$ and $|X[i]| = n$ for all $i = 1, \ldots, l$. We will mostly consider functions with inputs and outputs in $D_{d,n}$, hence both can be viewed as sequences of n-bit-long blocks.

CIPHERS AND ON-LINE CIPHERS. Informally, a cipher is a keyed deterministic length-preserving permutation. We recall the formal definitions presented in [1]. Let $F\colon \mathit{Keys}(F) \times \mathit{Dom}(F) \to \mathit{Rng}(F)$ be a function family, where $\mathit{Keys}(F)$ is the *key space* of F; $\mathit{Dom}(F)$ is the *domain* of F; and $\mathit{Rng}(F)$ is the *range* of F. F is a *cipher* if $\mathit{Dom}(F) = \mathit{Rng}(F)$ and each instance $F(K, \cdot)$ of F is a length-preserving permutation. If F is a cipher, then F^{-1} is the *inverse cipher*, defined by $F^{-1}(K, x) = F(K, \cdot)^{-1}(x)$ for all $K \in \mathit{Keys}(F)$ and $x \in \mathit{Dom}(F)$.

A function $f\colon D_{d,n} \to D_{d,n}$ is n-on-line if the first block of the output depends only on the first block of the input and the i-th block of the output (for $i \geq 2$) is determined completely by the $(i-1)$-th block of the input, $(i-1)$-th block of the output and the i-th block of the input[6]. A cipher is n-on-line if each instance is an on-line function. It is shown in [1] that the inverse of an on-line cipher is itself n-on-line. For convenience we will often omit n in the "n-on-line" term.

SECURITY OF ON-LINE CIPHERS. Let $\mathsf{OPerm}_{d,n}$ denote the family of all n-on-line, length-preserving permutations on $D_{d,n}$. A "secure" on-line cipher is one that closely approximates $\mathsf{OPerm}_{d,n}$. Let $F\colon \mathit{Keys}(F) \times D_{d,n} \to D_{d,n}$ be a family of functions with domain and range $D_{d,n}$. Let A be a distinguisher with access to two oracles.

We define the advantage of A attacking an on-line, pseudorandom permutation (OPRP) F chosen-ciphertext attacks as

$$\mathbf{Adv}^{\text{oprp-cca}}_{F,A} \;=\; \Pr\left[\, g \stackrel{R}{\leftarrow} F \;:\; A^{g,g^{-1}} = 1 \,\right] - \Pr\left[\, g \stackrel{R}{\leftarrow} \mathsf{OPerm}_{d,n} \;:\; A^{g,g^{-1}} = 1 \,\right].$$

This captures the *advantage* of the distinguisher in the task of distinguishing a random instance of F from a random, length-preserving, n-on-line permutation on $D_{d,n}$. The distinguisher can query

[6] For convenience we use a slightly stronger definition, than the one proposed in [1]. All the results and constructions of [1] satisfy our definition as well.

the challenge instance and its inverse. For any $t, q_e, \mu_e, q_d, \mu_d$ we define the *advantage* of F as OPRP against chosen-ciphertext attacks as

$$\mathbf{Adv}_F^{\text{oprp-cca}}(t, q_e, \mu_e, q_d, \mu_d) = \max_A \left\{ \mathbf{Adv}_{F,A}^{\text{oprp-cca}} \right\},$$

where the maximum is over all A, having the time complexity t, making at most q_e queries to the challenge instance oracle, whose total length is at most μ_e, and making at most q_d queries to the challenge inverse oracle, whose total length is at most μ_d. An on-line cipher F is secure against chosen-ciphertext attacks (is a *strong* cipher) if the function $\mathbf{Adv}_F^{\text{oprp-cca}}()$ grows "slowly".

SYMMETRIC ENCRYPTION SCHEMES AND THEIR SECURITY. A symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms. The randomized key generation algorithm $\mathcal{K}$ returns a key K; we write $K \stackrel{R}{\leftarrow} \mathcal{K}$. Associated with each encryption scheme there is a *message space* MsgSp, from which messages are drawn. In our context it will be important to make explicit the random coins underlying the randomized encryption algorithm $\mathcal{E}$. On input a key K, a plaintext $M \in$ MsgSp, and coin tosses R the randomized *encryption* algorithm $\mathcal{E}$ returns the ciphertext C; we write $C \leftarrow \mathcal{E}_K(M; R)$. The notation $C \stackrel{R}{\leftarrow} \mathcal{E}_K(M)$ is a shorthand for $r \stackrel{R}{\leftarrow} \mathsf{Coins}_{\mathcal{E}}$; $C \leftarrow \mathcal{E}_K(M; R)$, where $\mathsf{Coins}_{\mathcal{E}}$ is the set from which randomness is drawn. The deterministic decryption algorithm $\mathcal{D}$ takes K and a ciphertext C and returns the message M or a symbol $\perp$; we write $M \leftarrow \mathcal{D}_K(C)$ or $\perp \leftarrow \mathcal{D}_K(C)$. We require that $\mathcal{D}_K(\mathcal{E}_K(M)) = M$ for all $M \in$ MsgSp.

An encryption scheme is said to provide privacy against chosen-plaintext attacks (be IND-CPA secure) if no adversary with reasonable resources can win with non-negligible probability the game defined in the following experiment. An adversary runs in two stages. In both stages it is given a symmetric encryption oracle. During the first stage the adversary has to output two equal-length messages. One of the messages is chosen at random. At the last stage the adversary gets a challenge ciphertext, which is an encryption of this message. The adversary wins if it can distinguish which message was encrypted.

A symmetric encryption scheme is said to provide privacy against chosen-ciphertext attacks (be IND-CCA secure) if no adversary with reasonable resources can win with non-negligible probability the game defined in the following experiment:

The experiment is the same as above but in addition the adversary in both stages is given a decryption oracle with a restriction of not querying it during the last stage on the challenge ciphertext.

3 On-line Encryption Schemes and Their Security.

3.1 On-line Encryption Schemes

Let d, n be integers. Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme with MsgSp $= D_{d,n}$. We will refer to n as the *block-length* of a message. We say that $\mathcal{SE}$ is a scheme with *n-on-line encryption* if for every key K, coins R and message M, $C[1]$ (where $C = \mathcal{E}_K(M; R)$) depends only on K and R; and $C[i]$ (for $i \geq 2$) depends only on $M[i-2], M[i-1], C[i-1]$, K and possibly, R[7]. Similarly, we say that $\mathcal{SE}$ is a scheme with *n-on-line decryption* if for every key K and ciphertext C, $M[i]$ (for $i \geq 1$) depends only on $C[i], C[i+1], K$ and possibly R. In this work we are interested in symmetric encryption schemes with n-on-line encryption and decryption. We call such a subclass *n-on-line* symmetric encryption schemes. It is easy to see that most of the popular block-cipher

[7]For an example of on-line encryption scheme, see Figure 1 in Section 5.2. The definition can be easily modified to include nonces, counters, etc. to satisfy any intuitively on-line scheme, e.g. OCB.

based encryption schemes (which are also called encryption modes) are on-line encryption schemes, e.g. CBC and CTR modes.

3.2 Privacy of On-line Encryption Schemes

We investigate stronger notions of security for on-line encryption schemes. As we mentioned in Section 1 we look for a notion of privacy which takes into account both blockwise-adaptive attacks and chosen-ciphertext attacks.

But let us first claim that even the standard notion of privacy against chosen-ciphertext attacks is unachievable for on-line encryption schemes, meaning no on-line encryption scheme can be IND-CCA. We prove the claim by presenting a simple adversary attacking a given on-line encryption scheme. It does not make use of any blockwise adaptive attacks, but simply employs the on-line property of the decryption algorithm. The adversary is given the decryption oracle. First, the adversary generates any two two-block messages with distinct first blocks. Given the challenge ciphertext, which is an encryption of one of the two messages, it removes its last block and queries the result to its decryption oracle. The query is different from the challenge and thus is being legitimate. Since the first block of the corresponding plaintext does not depend on the removed last block of the ciphertext due to the on-line property of decryption, the adversary will get back the first block of one of the messages it generated at the previous stage and hence can always win its game.

The same attack would obviously apply when we enhance the IND-CCA notion with blockwise-adaptive attacks. We want to define the strongest achievable notion of privacy for on-line schemes. Let us start with some intuition.

Towards a proper notion.As we discussed in the Introduction and above, in the applications which require on-line encryption schemes the standard notion of privacy against chosen-ciphertext attacks cannot be achieved. This means that there exist a particular type of chosen-ciphertext attacks which an on-line encryption scheme cannot resist, namely the ones where the chosen ciphertext has the same prefix as the challenge ciphertext. This does not mean, however, that the practitioners should not expect an on-line encryption scheme to resist various other chosen-ciphertext attacks. For example, assume an adversary wants to decrypt a ciphertext of some important message. Suppose that it flips a bit in each block of the ciphertext, makes the receiver decrypt the result and manages to see the corresponding message. It is highly undesirable that the adversary be able to infer some information on the original important message. This kind of attack is not captured by either of IND-CPA nor IND-BLK-CPA notions and as we show in the next section, even the schemes known to be IND-BLK-CPA secure are subject to this type of attacks. But there exist on-line encryption schemes which resist such attacks as well as blockwise-adaptive attacks. Therefore, such schemes would be desirable in on-line applications with stronger security requirements. Accordingly we devise a notion of security which captures all blockwise-adaptive chosen-plaintext attacks and in addition captures these type of chosen-ciphertext attacks.

The intuition behind the notion is as follows. As in IND-BLK-CPA notion the adversary gets an encryption oracle and outputs two equal-length messages block by block, each time receiving the new portion of the challenge ciphertext, which is an encryption of one of the messages. This captures blockwise-adaptive attacks since the adversary can adapt each next message-block pair depending on the previously received portions of the ciphertext. To capture the aforementioned class of chosen-ciphertext attacks we allow the adversary to obtain decryptions of ciphertexts of its choice unless the ciphertext has a common prefix with a challenge ciphertext. This captures the attack discussed above. If, however, there is a common prefix, say l first blocks, the adversary can only see the decrypted message starting from the $l+1$ block. We now define the notion in detail.

IND-BLK-CCA SECURITY. We use the ideas for the model of security against blockwise-adaptive chosen-plaintext attacks [14, 11] and will extend this model to capture on-line chosen-ciphertext attacks. An adversary A runs in several stages. In all stages A is given the encryption oracle and the *on-line decryption oracle*, which we define below. In the first stage find_1 the adversary outputs two messages $M_0[1], M_1[1]$ of a block-length (we will refer to such messages as to message-blocks), a state information s it wants to preserve and a flag *done* indicating whether these are the last message-blocks it wants to output (in which case *done* $= 1$). A challenge bit b and coin tosses R are chosen at random. In the next stage find_2 an encryption of $M_b[1]$ is computed using R and given to A along with s. A can continue outputting more message-block pairs $M_0[i], M_1[i]$ for $i = 2, \ldots$ which can be a function of previously seen information, and updating s. Each time A receives a current challenge ciphertext C which is an encryption of $M_b[1] || \ldots || M_b[i]$ using coins R. This process continues until A sets *done* $= 1$. At the last guess stage A is given the whole challenge ciphertext and s and has to guess the challenge bit b. Here is the formal definition.

Definition 3.1 Fix $d, n \in N$. Let $\mathcal{OSE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an on-line symmetric encryption scheme with MsgSp $= D_{d,n}$. Let A be an adversary that has access to two oracles. For $b \in \{0, 1\}$ define the following experiments:

Experiment $\mathbf{Exp}^{\text{ind-blk-cca-b}}_{\mathcal{OSE},A}$
  $K \stackrel{R}{\leftarrow} \mathcal{K}$
  $R \stackrel{R}{\leftarrow} \mathsf{Coins}_{\mathcal{E}}$
  $i \leftarrow 1$; $C \leftarrow \varepsilon$; $s \leftarrow \varepsilon$
  Repeat
    $(M_0[i], M_1[i], s, done) \leftarrow A^{\mathcal{E}_K(\cdot), \mathcal{OD}_K(\cdot)}(\mathsf{find}_i, C, s)$
    $C \leftarrow \mathcal{E}_K(M_b[1] \| M_b[2] \| \ldots \| M_b[i]; R)$
    $i \leftarrow i + 1$
  Until $done = 1$
  $\bar{b} \leftarrow A^{\mathcal{E}_K(\cdot), \mathcal{OD}_K(\cdot)}(\mathsf{guess}, C, s)$
  Return $\bar{b}$

It is mandated that A outputs two message blocks of block-length n in each find stage and the total number of find stages is less or equal to d. The on-line decryption oracle $\mathcal{OD}_K(\cdot)$ takes inputs of length ln for some $l \in \mathsf{N}$. In stage find_1 on input C' it returns $M = \mathcal{D}_K(C')$. At next stages it returns $M[k] || M[k+1] || \ldots || M[l]$, where $M = \mathcal{D}_K(C')$ and $1 \le k \le l$ is the smallest index such that $C'[k] \neq C[k]$ (where C is the challenge ciphertext given to A at this stage).

We define the *advantage* of the adversary A as follows:

$$\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OSE},A} = \Pr[\mathbf{Exp}^{\text{ind-blk-cca-0}}_{\mathcal{OSE},A} = 0] - \Pr[\mathbf{Exp}^{\text{ind-blk-cca-1}}_{\mathcal{OSE},A} = 0].$$

For any $t, \mu_t, q_e, \mu_e, q_d, \mu_d$ we define the *advantage* of $\mathcal{OSE}$ via

$$\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OSE}}(t, \mu_t, q_e, \mu_e, q_d, \mu_d) = \max_A \left\{ \mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OSE},A} \right\},$$

where the maximum is over all A having time-complexity t, outputting message blocks during all find stages of total length at most μ_t in all find stages, making at most q_e queries to the encryption oracle, whose total length is at most μ_e, and making at most q_d queries to the on-line decryption oracle, whose total length is at most μ_d.

We say that on-line symmetric encryption scheme $\mathcal{OSE}$ is *secure against blockwise-adaptive chosen-ciphertext attacks* (or simply IND-BLK-CCA secure) if the function $\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OSE}}()$ grows slowly. ▮

Let us briefly recall why we do not explicitly allow the adversary to query its on-line decryption oracle in "blockwise-adaptive" manner. The reason is that we could do it, but this would not add any power to the adversary. This is because unlike encryption, the decryption algorithm is deterministic. Therefore, to see the decryption of a ciphertext $C = C[1]\|C[2]\| \ldots \|C[l]$ block-by-block, our adversary can simply query its decryption oracle on $C[1]$, $C[1]\|C[2]$, ..., $C[1]\|C[2]\| \ldots \|C[l]$.

4 Analyses of Several On-line Encryption Schemes

It was suggested in [14] that the fully parallelizable encryption modes such as counter mode CTR are not subject to blockwise-adaptive chosen-plaintext attacks. It is not hard to see that CTR is subject to a straightforward blockwise-adaptive chosen-ciphertext attack. An adversary can just choose two distinct one-block messages, flip a bit in the challenge ciphertext, query the result to the on-line decryption oracle and it will get back the one of the two challenge messages with a flipped bit in it. Thus the adversary can always win its game.

In [11] the authors prove that $\mathcal{CFB}$ encryption scheme [17] and $\mathcal{DCBC}$, the modification of $\mathcal{CBC}$ encryption scheme, are on-line encryption schemes secure against chosen-plaintext blockwise-adaptive attacks. We show that these schemes are unfortunately insecure against blockwise-adaptive chosen-ciphertext attacks. This is consistent with the design goal of these schemes which did not include chosen-ciphertext attack resistance. The attacks are simple and are similar to the one on CTR mode. We define the schemes and present the attacks in Appendix A.

In the next section we propose schemes which are IND-BLK-CCA secure.

5 Proposed Schemes and Their Security

5.1 On-line-cipher-based On-line Encryption Schemes

It is suggested (without a proof) in [14] that the encryption scheme based on HPCBC on-line cipher [1] does not seem to be vulnerable to blockwise-adaptive chosen-plaintext attacks. We generalize, strengthen and formally justify this suggestion. We propose an encryption scheme based on any strong on-line cipher and formally prove that it resists not only blockwise-adaptive chosen-plaintext attacks but also blockwise-adaptive chosen-ciphertext ones.

It is shown in [1] that the use of on-line ciphers provide security against standard chosen-plaintext attacks, if the plaintext space has appropriate properties, namely, if the first block of a plaintext is a random string. We show that any secure strong on-line cipher applied to messages with a random first block (or to any message with prepended random block) is also an on-line symmetric encryption scheme secure against blockwise-adaptive chosen-ciphertext attacks.

Construction 5.1 Let n, d be integers, and let $F\colon\ Keys(F) \times D_{d,n} \to D_{d,n}$ be an on-line cipher. We associate to them the following symmetric encryption scheme $\mathcal{OCSE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$:

Algorithm $\mathcal{K}$	Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$
$K \stackrel{R}{\leftarrow} Keys(F)$	$R \stackrel{R}{\leftarrow} \{0,1\}^n$	$X \leftarrow F^{-1}(K, C)$
Return K	$X \leftarrow R\|M$	Parse X as $R\|M$ with $\lvert R\rvert = n$
	$C \leftarrow F(K, X)$	Return M
	Return C	

Given the fact that F is an on-line cipher, it is easy to see that $\mathcal{OCSE}$ is on-line encryption scheme. We want to show that it provides IND-BLK-CCA security when F is an on-line cipher secure against chosen-ciphertext attacks (a strong on-line cipher). The following theorem states the result.

Theorem 5.2 Let d, n be integers, and let $F\colon\ Keys(F) \times D_{d,n} \to D_{d,n}$ be an n-on-line cipher. Let $\mathcal{OCSE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the on-line symmetric encryption scheme associated to it as per Construction 5.1. Then

$$\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OCSE}}(t, \mu_t, q_e, \mu_e, q_d, \mu_d) \leq 2 \cdot \mathbf{Adv}^{\text{oprp-cca}}_{F}(t', q'_e, \mu'_e, q_d, \mu_d) + \frac{q_e + q_d}{2^{n-1}},$$

where

$$t' = t + O(q_e, q_d, \mu_t), q'_e = q_e + \frac{\mu_t}{2n}, \mu'_e = \mu_e + nq_e + \frac{\mu_t^2 + 6\mu_t}{8n} \ .$$

Proof: Let A be an adversary attacking the $\mathcal{OCSE}$ scheme against blockwise-adaptive chosen-ciphertext attacks. We construct a distinguisher B attacking the pseudorandomness of on-line cipher F against chosen-ciphertext attacks. B has access to two oracles: g, which is either a random instance of the on-line cipher F, or a random instance of $\mathsf{OPerm}_{d,n}$, and g^{-1}, the inverse of g. B runs A as a subroutine and has to answer its oracle queries and provide the necessary inputs. We present pseudocode for B followed by some explanations and the analysis.

Adversary $B^{g,g^{-1}}$
    $R \stackrel{R}{\leftarrow} \mathsf{Coins}_{\mathcal{E}}$; $b \stackrel{R}{\leftarrow} \{0,1\}$
    $i \leftarrow 1$; $C \leftarrow \varepsilon$; $s \leftarrow \varepsilon$
    Repeat
        Run A on input (find_i, C, s)
            When A outputs $(M_0[i], M_1[i], s, done)$, do
                $C \leftarrow g(R||M_b[1]\|M_b[2]\| \ldots \|M_b[i])$
                $i \leftarrow i + 1$
                Return (find_i, C, s) to A
            On A's encryption query M, do
                $R' \stackrel{R}{\leftarrow} \mathsf{Coins}_{\mathcal{E}}$
                $Y \leftarrow g(R'||M)$
                Return Y to A
            On A's on-line decryption query C', do
                $X \leftarrow g^{-1}(C')$, parse X as $R''||M'$ where $|R''| = n$
                $l \leftarrow |M'|/n$
                Return $(M'[k]||M'[k+1]|| \cdots ||M'[l])$ to A,
                where k is the smallest index such that $C'[k] \neq C[k]$
        Until $done = 1$
    Run A on input (guess, C, s) replying to its oracle queries as before
        until it halts and returns $\bar{b}$
    If $(\bar{b} = b)$ then return 1 else return 0 EndIf

B picks a random bit b and a random n-bit string R and runs the adversary A answering its oracle queries and providing all the necessary inputs for it. When A outputs two messages block-by-block during find stages, each time B returns the current challenge ciphertext. The challenge ciphertext is computed by prepending R to the message determined by the bit b, and querying it to B's own oracle g. Similarly, B uses its oracles g, g^{-1} to answer A's encryption and on-line decryption oracle queries. This is possible since encryption and decryption algorithms use only oracle access to $F(K, \cdot)$. A little extra care is required when answering A's on-line decryption oracle queries. According to the experiment $\mathbf{Exp}^{\text{ind-blk-cca-b}}_{\mathcal{OSE},A}$ defined in Definition 3.1, B returns to A only the message blocks starting from position at which the query block is different from the latest challenge ciphertext given to A. Finally, when A outputs its guess at the end of guess stage, B compares it with the bit b, and if they are equal, B concludes that g is a random instance of on-line cipher and outputs 1, otherwise, it outputs 0 to indicate its guess that g is a random on-line permutation.

We now proceed to the analysis. First consider the case when g is a random instance of the on-line cipher F. We claim that A's view in the simulated experiment is exactly as in experiment $\mathbf{Exp}^{\text{ind-blk-cca-b}}_{\mathcal{OSE},A}$. By the construction B will return 1 if and only if the adversary A can guess the random bit b correctly. Thus we have:

$$\begin{aligned}
\Pr\left[g \stackrel{R}{\leftarrow} F \ : \ B^{g,g^{-1}} = 1 \right] &= \frac{1}{2} \cdot \Pr[\mathbf{Exp}^{\text{ind-blk-cca-0}}_{\mathcal{OCSE},A} = 0] \\
&+ \frac{1}{2} \cdot (1 - \Pr[\mathbf{Exp}^{\text{ind-blk-cca-1}}_{\mathcal{OCSE},A} = 0]) \\
&= \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{OCSE},A}. \qquad (1)
\end{aligned}$$

Next consider the case when g is a random instance of $\mathsf{OPerm}_{d,n}$. Let Col denotes the event when some of the coins the adversary B tosses to answer A's encryption oracle queries collide with coins R used to compute the challenge ciphertext, or when the challenge ciphertext given to A in stage find_2 has a common prefix of at least one block with A's previous on-line decryption oracle queries.

We show that if Col does not happen then A's view is independent from the challenge bit b. Since encryption is done by prepending the random block to the message and applying g (which is a random on-line permutation) to the result, the challenge ciphertext is a random string. If Col does not occur, encryption oracle answers are computed by applying g to the messages with prepended random first blocks which are all distinct from the random block used to compute the challenge ciphertext. Hence similarly, since g is a random on-line permutation, the resulting ciphertexts are just random strings independent from the challenge bit b. Now consider the answers to the on-line decryption oracle queries A gets before it sees the first block of the challenge ciphertext. If Col does not occur, no such answer ciphertexts share the same prefix with the challenge ciphertext. Since g is a random on-line permutation, the oracle answers A gets are independent from the challenge ciphertext and bit b. The same is true for the on-line decryption oracle queries A makes after it sees some blocks of the challenge ciphertext, due to the way the on-line decryption oracle answers the queries. Therefore, if Col does not occur, the view of A in the simulated experiment is independent of the bit b and A cannot win with probability bigger than $1/2$. Finally we note that Col occurs with probability is at most $(q_e + q_d)/2^n$, which is the probability of one of the random n-bit strings colliding with the fixed $(q_e + q_d)$ strings.

Hence we have:

$$\begin{aligned}
\Pr\left[g \stackrel{R}{\leftarrow} \mathsf{OPerm}_{d,n} \ : \ B^{g,g^{-1}} = 1 \right] &\le \frac{1}{2} \cdot (1 - \Pr[\mathsf{Col}]) + \Pr[\mathsf{Col}] \\
&\le \frac{1}{2} + \frac{q_e + q_d}{2^n}. \qquad (2)
\end{aligned}$$

After subtracting Equation(1) and Equation(2) and taking maximum, we get the statement of the theorem.

We now check the resources used by the adversary B. If A makes at most q_d queries to the decryption oracle, whose total length is at most μ_d, B also makes at most q_d queries to the challenge inverse oracle, whose total length is at most μ_d. In addition, B will make a query to its challenge instance oracle if and only if A makes a query to the encryption oracle or outputs new message blocks in the find state. In the first case, because A makes at most q_e queries, whose total length is at most μ_e, B calls its challenge instance oracle at most q_e times. However, the total length of all of these queries is at most $\mu_e + nq_e$ because B prepends each of A's query with a random n-bit string R. In the second case, if A outputs the i^{th} pair of message block, B will make a query whose length is $(i+1)n$. When A outputs at most $\mu_t/2n$ pairs of block-length messages, B makes at most $\mu_t/2n$ queries, whose total length is

$$\sum_{1}^{\frac{\mu_t}{2n}} (i+1) \cdot n = \frac{\mu_t^2}{8n} + \frac{3\mu_t}{4n}.$$

To sum up, B makes at most $q_e + \mu_t/2n$ queries to the challenge instance oracle, whose total length is $\mu_e + nq_e + \mu_t^2/8n + 3\mu_t/4n$. Finally, B runs A, which is executed in time t, as a subroutine and all other operations in B can be bound by $O(q_e, q_d, \mu_t)$. ▮

5.2 HPCBC Encryption Scheme.

We present the $\mathcal{HPCBC}$ encryption scheme, based on the HPCBC on-line cipher proposed in [1]. According to Construction 5.1, the encryption algorithm applies the HPCBC on-line cipher to an input message prepended with a random block. Here are the details of the scheme.

Let $d, n \in \mathsf{N}$, let E: $Keys(E) \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher family and let H: $Keys(H) \times \{0,1\}^{2n} \to \{0,1\}^n$ be a family of functions. The message space of $\mathcal{HPCBC}$ is $D_{d,n}$ and the scheme consists of the following three algorithms:

Algorithm $\mathcal{K}$
  $K_1 \xleftarrow{R} Keys(E)$
  $K_2 \xleftarrow{R} Keys(H)$
  $K \leftarrow K_1 \| K_2$
  Return K

Algorithm $\mathcal{E}_K(M)$
  Parse K into $K_1 \| K_2$
  Parse M into n-bit blocks
    $M[1] \| \ldots \| M[l]$
  $R \xleftarrow{R} \{0,1\}^n$; $M[0] \leftarrow R$
  $T \leftarrow 0^{2n}$; $Y \leftarrow H(K_2, T)$
  $P \leftarrow Y \oplus M[0]$
  $C[1] \leftarrow E(K_1, P) \oplus Y$
  For $i = 2, \ldots, l+1$ do
    $T \leftarrow M[i-2] \| C[i-1]$
    $Y \leftarrow H(K_2, T)$
    $P \leftarrow Y \oplus M[i-1]$
    $C[i] \leftarrow E(K_1, P) \oplus Y$
  EndFor
  $C \leftarrow C[1] \| \ldots \| C[l+1]$
  Return C

Algorithm $\mathcal{D}_K(C)$
  Parse K into $K_1 \| K_2$
  Parse C into n-bit blocks
    $C[1] \| \ldots \| C[l+1]$
  $T \leftarrow 0^{2n}$; $Y \leftarrow H(K_2, T)$
  $P \leftarrow E^{-1}(K_1, (C[1] \oplus Y))$
  $M[0] \leftarrow Y \oplus P$
  For $i = 1, \ldots l$ do
    $T \leftarrow M[i-1] \| C[i]$
    $Y \leftarrow H(K_2, T)$
    $P \leftarrow E^{-1}(K_1, C[i+1] \oplus Y)$
    $M[i] \leftarrow Y \oplus P$
  EndFor
  $M \leftarrow M[1] \| \ldots \| M[l]$
  Return M

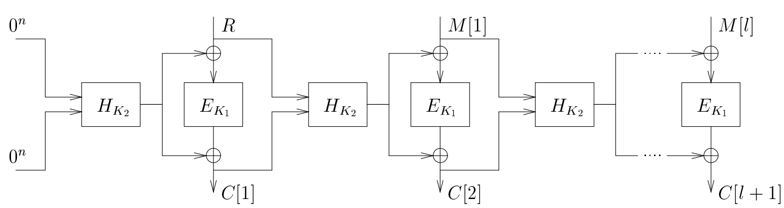

Figure 1: The HPCBC encryption scheme

The $\mathcal{HPCBC}$ encryption scheme is depicted in Figure 1, from which it is easy to see that the scheme is on-line. The scheme makes one use of a keyed hash function in addition to that of a block cipher per each input block. HPCBC cipher has been proved to be a strong on-line cipher (secure against chosen-ciphertext attacks) [1] assuming E is a strong PRP family and H is almost-xor-universal function family. Hence, it follows from Theorem 5.2 that $\mathcal{HPCBC}$ is IND-BLK-CCA secure. The concrete security result can be easily obtained from the result of [1] and from Theorem 5.2.

6 Authentication of On-line Symmetric Encryption Schemes

Authenticity of on-line encryption schemes. We first recall the standard notion of authenticity (integrity) of a symmetric encryption scheme [4, 1].

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. A ciphertext C is called valid if $\mathcal{D}_K(C) \neq \perp$. The encryption scheme is said to provide authenticity or ciphertext integrity (or be INT-CTXT secure) if any adversary with reasonable resources can be successful in the following experiment only with small probability. At first, the random key K is generated by $\mathcal{K}$. The adversary has access to the encryption oracle $\mathcal{E}_K(\cdot)$. It is successful if it can output a valid ciphertext C that was never returned by the encryption oracle.

We note that the above standard definition does not allow analysis of on-line encryption schemes. This is because it assumes that the decryption algorithm outputs the whole message only if it is valid, otherwise, the algorithm returns $\perp$. But an on-line decryption algorithm has to be able to output the message block-by-block before the validity check of the whole message can be performed.

At the same time it makes sense and is important to analyze authenticity of on-line encryption schemes. To facilitate this task we provide the security definition for authenticity of on-line encryption schemes, which is a slight modification of the standard definition. Namely, we modify only the definition of the validity of a ciphertext as follows.

We say that a ciphertext C is valid if the last block of $\mathcal{D}_K(C)$ is not $\perp$. The rest of the definition is analogous to the standard one. Note that to use this definition it should be possible for a decryptor to distinguish the last block of $\mathcal{D}_K(C)$, since the message blocks are output on-the-fly. One way to do this is for the encryption algorithm to output some special symbol, e.g. $\top$, after returning the last block of the message.

Authenticated IND-BLK-CCA secure scheme. As we mentioned in Section 1, the decryption algorithm of all of the known authenticated encryption schemes is not on-line. Here we suggest that a slightly modified Construction 9.3 from [1] simultaneously provides authenticity and privacy against blockwise-adaptive blockwise-adaptive attacks. We modified the encryption algorithm so that it appends $\top$ to the output ciphertext and the decryption algorithm so that it is on-line. Here is the scheme.

Construction 6.1 Let n, d be integers, and let $F\colon\ Keys(F) \times D_{d,n} \to D_{d,n}$ be an on-line cipher. We define on-line symmetric encryption scheme $\mathcal{AOCSE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ as follows:

Algorithm $\mathcal{K}$	Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$
$K \xleftarrow{R} Keys(F)$	$R \xleftarrow{R} \{0,1\}^n$	$C' \leftarrow$ all but the last block of C
Return K	$X \leftarrow R \Vert M \Vert R$	$X \leftarrow F^{-1}(K, C')$
	$C \leftarrow F(K, X) \Vert \top$	Return $M \Vert \top$ if the last block of X is R
	Return C	Else return $M \Vert \perp$ ▮

Note that since the on-line cipher F is on-line, the encryption and decryption algorithms are also on-line, even though for simplicity we chose the standard description.

Theorem 6.2 Let n, d be integers, and let $F\colon\ Keys(F) \times D_{d,n} \to D_{d,n}$ be an on-line cipher. Then $\mathcal{AOCSE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ as defined in Construction 6.1 is an IND-BLK-CCA and INT-CTXT secure on-line encryption scheme.

Proof: The authors of [1] prove that their scheme defined in Construction 9.3 is INT-CTXT secure. Their proof holds for $\mathcal{AOCSE}$ as well. As for IND-BLK-CCA security, the proof of Theorem 5.2 also holds for $\mathcal{AOCSE}$. ▌

7 Acknowledgements

We thank Mihir Bellare and Bogdan Warinschi for helpful discussions and the anonymous referees for their comments. Alexandra Boldyreva is supported in part by NSF Grant CCR-0098123 and NSF Grant ANR-0129617.

References

[1] M. Bellare, A. Boldyreva, L. Knudsen, C. Namprempre, "On-line Ciphers and the Hash-CBC Constructions," Available at `www.cse.ucsd.edu/users/mihir/papers/olc.html`. The abstract appeared in *Advances in Cryptology – Crypto '01*, LNCS Vol. 2139, J. Kilian ed., Springer-Verlag, 2001.

[2] M. Bellare, A. Desai, E. Jokipii and P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES Modes of Operation," *Proc. of the 38th Symposium on Foundations of Computer Science*, IEEE, 1997.

[3] M. Bellare, J. Kilian and P. Rogaway, "The Security of Cipher Block Chaining," *Advances in Cryptology – Crypto '94*, LNCS Vol. 839, Y. Desmedt ed., Springer-Verlag, 1994.

[4] M. Bellare and C. Namprempre, "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm," *Advances in Cryptology – ASIACRYPT '00*, LNCS Vol. 1976, T. Okamoto ed., Springer-Verlag, 2000.

[5] J. Black and P. Rogaway, "CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions," *Advances in Cryptology – Crypto '00*, LNCS Vol. 1880, M. Bellare ed., Springer-Verlag, 2000.

[6] M. Blaze, "High-bandwidth Encryption with Low-bandwidth Smart Cards," *Fast Software Encryption '96*, LNCS Vol. 1039, D. Gollmann ed., Springer-Verlag, 1996.

[7] M. Blaze, J. Feigenbaum and M. Naor, "A Formal Treatment of Remotely Keyed Encryption," *Advances in Cryptology –Eurocrypt '98*, LNCS Vol. 1403, K. Nyberg ed., Springer-Verlag, 1998.

[8] Y. Dodis and J.H. An, "Concealment and Its Applications to Authenticated Encryption," *Advances in Cryptology –Eurocrypt '03*, LNCS Vol. 2656 , E. Biham ed., Springer-Verlag, 2003.

[9] V. Gligor and P. Donescu, "Fast Encryption and Authentication: XCBC Encryption and XECB Authenticated Modes," *Fast Software Encryption '01*, LNCS Vol. 2355, M. Matsui ed., Springer-Verlag, 2001.

[10] P.-A. Fouque, A. Joux, G. Martinet and F. Valette, "Authenticated On-line Encryption", *Selected Areas in Cryptography Workshop, SAC '03*, 2003.

[11] P.-A. Fouque, G. Martinet, G. Poupard, "Practical Symmetric On-line Encryption", *Fast Software Encryption '03*, LNCS, T. Johansson Ed., Springer Verlag, 2003.

[12] T. IWATA AND K. KUROSAWA, "OMAC: One-Key CBC MAC," *Fast Software Encryption '03*, LNCS, T. Johansson Ed., Springer Verlag, 2003.

[13] M. JAKOBSSON, J. STERN AND M. YUNG, "Scramble All, Encrypt Small", *Fast Software Encryption '99*, LNCS Vol. 1636, L. Knudsen ed., Springer-Verlag, 1999.

[14] A. JOUX, G. MARTINET AND F. VALETTE, "Blockwise Adaptive Attackers - Revisiting the (In)security of Some Provably Secure Encryption Modes: CBC, GEM, IACBC", *Advances in Cryptology – Crypto '02*, LNCS Vol. 2442 , M. Yung ed., Springer-Verlag, 2002.

[15] C. JUTLA, "Encryption Modes With Almost Free Message Integrity", *Advances in Cryptology – Eurocrypt '01*, LNCS Vol. 2045, B. Pfitzmann ed., Springer-Verlag, 2001.

[16] S. LUCKS, "On the Security of Remotely Keyed Encryption," *Fast Software Encryption '97*, LNCS Vol. 1267, E. Biham ed., Springer-Verlag, 1997.

[17] NIST. FIPS PUB81 - DES MODES OF OPERATION, December 1980.

[18] P. ROGAWAY, M. BELLARE, J. BLACK, AND T. KROVETZ, "OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption," *Eighth ACM Conference on Computer and Communications Security (CCS-8)*, ACM Press, 2001.

A IND-BLK-CCA Attacks on CFB and DCBC Schemes

A.1 CFB On-line Encryption Scheme

First, we recall $\mathcal{CFB}$ encryption scheme. Let $d, n \in \mathsf{N}$, and let $E\colon\ Keys(E) \times \{0,1\}^n \to \{0,1\}^n$ be a family of functions. The message space of $\mathcal{CFB}$ is $D_{d,n}$ and the scheme's algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ are as follows:

Algorithm $\mathcal{K}$	Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$

$K \xleftarrow{R} Keys(E)$
Return K
	Parse M into n-bit blocks
	$M[1] \| \ldots \| M[l]$
	$C[1] \xleftarrow{R} \{0,1\}^n$
	For $i = 2, \ldots, l+1$ do
	$C[i] \leftarrow E(K, C[i-1]) \oplus M[i-1]$
	EndFor
	$C \leftarrow C[1] \| \ldots \| C[l+1]$
	Return C
		Parse C into n-bit blocks
		$C[1] \| \ldots \| C[l+1]$
		For $i = 1, \ldots, l$ do
		$M[i] \leftarrow E(K, C[i]) \oplus C[i+1]$
		EndFor
		$M \leftarrow M[1] \| \cdots \| M[l]$
		Return M

Above E can be a block cipher family, e.g. AES. Note that we presented the encryption and decryption algorithms the standard way, as taking the whole input processing it and returning the whole output. We do it only for clarity of notation. It is still not hard to see that the above scheme is an on-line encryption scheme and hence it can be implemented in "on-line" manner. We will use the same approach for the descriptions of other encryption schemes in this paper.

ATTACK ON THE CFB ON-LINE ENCRYPTION SCHEME. The following theorem shows that $\mathcal{CFB}$ is insecure against blockwise-adaptive chosen-ciphertext attacks for any underlying function family E.

Theorem A.1 Let $n \in \mathsf{N}$, and let $E\colon\ Keys(E) \times \{0,1\}^n \to \{0,1\}^n$ be a function family. Let $\mathcal{CFB} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the n-on-line encryption scheme as described above. Then,

$$\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{CFB}}(t, 2n, 0, 0, 1, 2n) = 1,$$

where $t = O(n)$.

Proof of Theorem A.1: In order to show that the $\mathcal{CFB}$ is not IND-BLK-CCA secure, we present an adversary A, which is given two oracles $\mathcal{E}_K(\cdot)$ and $\mathcal{OD}_K(\cdot)$:

Adversary $A^{\mathcal{E}_K(\cdot),\mathcal{OD}_K(\cdot)}(\mathsf{find}_1, \varepsilon, s)$
$M_0 \leftarrow 0^n; M_1 \leftarrow 1^n$
Output $(M_0, M_1, s, 1)$
Adversary $A^{\mathcal{E}_K(\cdot),\mathcal{OD}_K(\cdot)}(\mathsf{guess}, C[1]||C[2], s)$
$C' \leftarrow C[2] \oplus 1^n$
Make a query $(C[1]||C')$ to $\mathcal{OD}_K(\cdot)$ and let M' be the reply
If $M' = 1^n$ then return 0, otherwise return 1

We now analyze A. If $b = 0$, we have:

$$\begin{aligned} M' &= E(K, C[1]) \oplus C' = E(K, C[1]) \oplus C[2] \oplus 1^n \\ &= M_0 \oplus 1^n = 1^n. \end{aligned}$$

When $b = 1$ we have:

$$\begin{aligned} M' &= E(K, C[2]) \oplus C' = E(K, C[1]) \oplus C[2] \oplus 1^n \\ &= M_1 \oplus 1^n = 0^n \neq 1^n. \end{aligned}$$

Thus, A always guesses the challenge bit correctly and $\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{CFB},\mathcal{A}} = 1$. This implies the statement of the theorem. We now check the resources that A uses. The adversary outputs two n-bit messages, makes one query to its on-line decryption oracle of length $2n$, and does not make any encryption oracle queries. The time complexity t is thus linear in n. ∎

A.2 Delayed CBC On-line Encryption Scheme.

Here we describe the delayed-CBC on-line encryption scheme ($\mathcal{DCBC}$), which was first mentioned in [14] and more formally defined in [11]. We first recall the standard CBC encryption scheme and then discuss how it can be modified to provide $\mathcal{DCBC}$.

Let $d, n \in \mathsf{N}$, and let E: $Keys(E) \times \{0,1\}^n \to \{0,1\}^n$ be a family of permutations. The message space of CBC is $D_{d,n}$ and the scheme's algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ are as follows:

Algorithm $\mathcal{K}$	Algorithm $\mathcal{E}_K(M)$	Algorithm $\mathcal{D}_K(C)$

$K \stackrel{R}{\leftarrow} Keys(E)$
Return K
	Parse M into n-bit blocks
	$M[1]\|\|\ldots\|\|M[l]$
	$C[1] \stackrel{R}{\leftarrow} \{0,1\}^n$
	For $i = 2, \ldots, l+1$ do
	$C[i] \leftarrow E(K, (C[i-1] \oplus M[i-1]))$
	EndFor
	$C \leftarrow C[1]\|\|\ldots\|\|C[l+1]$
	Return C
		Parse C into n-bit blocks
		$C[1]\|\|\ldots\|\|C[l+1]$
		For $i = 1, \ldots, l$ do
		$M[i] \leftarrow E^{-1}(K, C[i+1]) \oplus C[i]$
		EndFor
		$M \leftarrow M[1]\|\|\cdots\|\|M[l]$
		Return M

Usually, E is a block cipher family, e.g. AES.

We now describe the delayed-CBC encryption scheme. All operations in $\mathcal{DCBC}$ are the same as those in the standard CBC encryption scheme, except that each output block of the encryption and decryption algorithms is delayed for a single block time before it is returned. The $(i-1)^{th}$ block of the corresponding ciphertext (resp. message) will be returned after the i^{th} block of the message (resp. ciphertext) has been computed. However, no delay is required for the last block. It is also easy to see that both $\mathcal{DCBC}$ is on-line encryption scheme.

ATTACK ON THE DCBC ON-LINE ENCRYPTION SCHEME. Even though the scheme has been proven secure against blockwise-adaptive chosen-plaintext attacks [11], the following theorem shows that $\mathcal{DCBC}$ is IND-BLK-CCA insecure for any underlying permutation family E.

Theorem A.2 Let $n \in \mathsf{N}$, and let E: $Keys(E) \times \{0,1\}^n \to \{0,1\}^n$ be a permutation family. Let $\mathcal{DCBC} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be the n-on-line encryption scheme as described above. Then,

$$\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{DCBC}}(t, 2n, 0, 0, 1, 2n) = 1,$$

where $t = O(n)$.

Proof of Theorem A.2: In order to show that $\mathcal{DCBC}$ is not IND-BLK-CCA secure, we present an adversary A, which is given two oracles $\mathcal{E}_K(\cdot)$ and $\mathcal{OD}_K(\cdot)$:

Adversary $A^{\mathcal{E}_K(\cdot), \mathcal{OD}_K(\cdot)}(\mathsf{find}_1, \varepsilon, s)$
  $M_0 \leftarrow 0^n; M_1 \leftarrow 1^n$
  Output $(M_0, M_1, s, 1)$
Adversary $A^{\mathcal{E}_K(\cdot), \mathcal{OD}_K(\cdot)}(\mathsf{guess}, C[1]||C[2]), s$
  $C' \leftarrow C[1] \oplus 1^n$
  Make a query $(C'||C[2])$ to $\mathcal{OD}_K(\cdot)$ and let M' be the reply
  If $M' = 1^n$ then return 0, otherwise return 1

We now proceed with the analysis of A. From the construction of the scheme, we have: $M' = E^{-1}(K, C[2]) \oplus C' = E^{-1}(K, C[2]) \oplus C[1] \oplus 1^n = M_b \oplus 1^n$. Therefore, the adversary can determine whether M' is the complement of M_0 or M_1 and easily guesses the challenge bit b. Thus, $\mathbf{Adv}^{\text{ind-blk-cca}}_{\mathcal{CFB},\mathcal{A}} = 1$, which implies the statement of the theorem.

We next check the resources A uses. The adversary outputs two n-bit messages, makes one query to its on-line decryption oracle of length $2n$, and does not make any encryption oracle query. Besides calling the oracles, all other operations of the adversary can be bound by $O(n)$. ▮