The Security of All-or-Nothing Encryption: Protecting against Exhaustive Key Search

Anand Desai

Department of Computer Science & Engineering,
University of California at San Diego,
9500 Gilman Drive, La Jolla, California 92093, USA.
adesai@cs.ucsd.edu

Abstract. We investigate the all-or-nothing encryption paradigm which was introduced by Rivest as a new mode of operation for block ciphers. The paradigm involves composing an all-or-nothing transform (AONT) with an ordinary encryption mode. The goal is to have secure encryption modes with the additional property that exhaustive key-search attacks on them are slowed down by a factor equal to the number of blocks in the ciphertext. We give a new notion concerned with the privacy of keys that provably captures this key-search resistance property. We suggest a new characterization of AONTs and establish that the resulting all-or-nothing encryption paradigm yields secure encryption modes that also meet this notion of key privacy. A consequence of our new characterization is that we get more efficient ways of instantiating the all-or-nothing encryption paradigm. We describe a simple block-cipher-based AONT and prove it secure in the Shannon Model of a block cipher. We also give attacks against alternate paradigms that were believed to have the above key-search resistance property.

1 Introduction

In this paper, we study all-or-nothing transforms in the context of the original application for which they were introduced by Rivest [20]. The goal is to increase the difficulty of an exhaustive key search on symmetric encryption schemes, while keeping the key size the same and not overly burdening the legitimate users.

Background and Motivation. Block ciphers, such as DES, can be vulnerable to exhaustive key-search attacks due to their relatively small key-sizes. The attacks on block ciphers also carry over to symmetric encryption schemes based on the block ciphers (hereafter called, *encryption modes*). One way to get better resistance to key-search attacks, is to use a longer key (either with the existing block cipher or with a next-generation block cipher such as AES). This, however, can be an expensive proposition, since it would necessitate changing the existing cryptographic hardware and software implementing these encryption modes. In some cases, the preferred approach might be to squeeze a little more security out of the existing encryption modes using some efficient pre-processing techniques.

M. Bellare (Ed.): CRYPTO 2000, LNCS 1880, pp. 359–375, 2000.

Rivest observed that with most of the popular encryption modes, it is possible to obtain one block of the message by decrypting *just one* block of the ciphertext. With the cipher-block-chaining mode (CBC) [18], for example, given any two consecutive blocks of ciphertext, it is possible to decrypt a single value and obtain one block of the message. Thus the time to check a candidate key is that of just one block cipher operation. Such modes are said to be *separable*. Rivest suggests designing *strongly non-separable* encryption modes. As defined in [20], strongly non-separable encryption means that it should be infeasible to determine even one message block (or any property of a particular message block) without decrypting *all* the ciphertext blocks.

The all-or-nothing encryption paradigm was suggested as a means to achieve strongly non-separable encryption modes. It involves using an *all-or-nothing transform* (AONT) as a pre-processing step to an ordinary encryption mode. As defined in [20], an AONT is an efficiently computable transformation, mapping sequences of blocks to sequences of blocks, with the following properties:

- Given the output sequence of blocks, one can easily obtain the original sequence of blocks.
- Given all but one of the output sequence of blocks, it is computationally infeasible to determine any function of any input block.

It is necessary that an AONT be randomized so that a chosen input does not yield a known output. Note that in spite of the privacy parallel in their definitions, an AONT is distinct from an encryption scheme. In particular, there is no secret-key information associated with an AONT. However, it is suggested that if the output of an AONT is encrypted, with say the codebook mode (ie. a secret-keyed block cipher applied block by block), then the resulting scheme will not only be secure as an encryption scheme but also be strongly non-separable.

We are interested in encryption modes wherein an exhaustive key-search is somehow dependent on the size of the ciphertext. This is the primary motivation for using strongly non-separable modes. The intuition is that brute-force searches on such encryption modes would be slowed down by a factor equal to the number of blocks in the ciphertext. But does strong non-separability really capture this property? A reason to believe otherwise is that the property we want is concerned more with the privacy of the underlying *key* than that of the data. Consider the (admittedly, contrived) example of an encryption mode that, in addition to the encrypted message blocks, always outputs a block that is the result of the underlying block cipher on the string of all 0s. Such a mode could turn out to be strongly non-separable although it clearly does not possess the property we desire: a key-search adversary can test any candidate key by decrypting just the block enciphering the 0 string. One could think of more subtle ways for some other "invariable information" about the key being leaked that would illustrate this point more forcefully. Strong non-separability does capture some strong (data-privacy) property, but that is not the one we are interested in. What we need here instead is a suitable notion of key-privacy. We want encryption modes that have this property, as well as the usual data-privacy ones.

Our Notions and Model. We give a notion, called non-separability of keys, that formalizes the inability of an adversary to gain *any* information about the underlying key, without "decrypting" *every* block of the ciphertext. The notion can be informally described through the following interactive protocol: an adversary A is first given two randomly selected keys a_0 and a_1. A then outputs a message x and gets back, based on a hidden bit b, the encryption y of x under a_b. We ask that it be infeasible for a hereafter "restricted" A to guess b correctly with probability significantly more than 0.5. The restriction we put on A is in limiting how it can use its knowledge of a_0 and a_1 in trying to guess b.

In order to make the above restriction meaningful, we describe our notion in the Shannon Model of a block cipher [21]. This model has been used in similar settings before [17,1]. Roughly speaking, the model instantiates an independent random permutation with every different key. We discuss the limitations of the model and their implications to our results in Section 7.

We show that our notion captures our desired key-search resistance property. That is, we prove that exhaustive key-search attacks on encryption modes secure in the non-separability of keys sense are slowed down by a factor equal to the number of blocks in the ciphertext. Our notion is orthogonal to the standard notions of data-privacy. In particular, the notion by itself does not imply security as an encryption scheme. It can, however, be used in conjunction with any notion of data-privacy to define a new encryption goal.

We want to justify the intuition that all-or-nothing encryption modes are secure encryption modes that also have the key-search resistance property. Recall that an all-or-nothing encryption mode is formed by composing an AONT with an ordinary encryption mode. The definition of an AONT from [20], however, is more of an intuitive nature than of sufficient rigor to establish any claims with it. One problem with the definition, as pointed out by Boyko [10], is that it speaks of information leaked about a *particular* message block. In our context, information leaked about the message as a whole, say the XOR of all the blocks, can be just as damaging. A formal characterization of AONTs was later given by Boyko [10]. He makes a case for defining an AONT with respect to any (and variable amount of) missing information, as opposed to a missing block. While this is certainly more general and probably necessary in some settings, we believe that in the context of designing efficient encryption modes with the key-search resistance property, a formalization with respect to a missing block is preferable. It turns out that even this weaker formalization is enough to realize our goal through the all-or-nothing encryption paradigm. An advantage of a weaker characterization of AONTs, as we will see later, is that we can build more efficient constructions that meet it. Our characterization of AONTs is tailored to their use in designing encryption modes that have the desired key-search resistance property.

Our Security Results. We establish that all-or-nothing encryption modes (using our definition of an AONT) are secure in the non-separability of keys sense as well as being secure against chosen-plaintext attack. Our analysis relates the security of the all-or-nothing encryption paradigm to the security of the underlying AONT in a precise and quantitative way.

We give an efficient block-cipher-based construction of an AONT. Our construction is a simplified version of Rivest's "package transform". The package transform may well have some stronger security properties than ours, but it turns out that even our simplified version is secure under our definition of an AONT. The proof of this is also in the Shannon Model of a block cipher. With this, we can now get all-or-nothing encryption modes that cost only two times the cost of normal CBC encryption, while with the package transform, the resulting modes had cost about three times the cost of CBC.

In addition, we give attacks against alternate paradigms believed to have the key-search resistance property. We show that a paradigm claimed to capture this property [7] does not actually do so. There seem to be several misconceptions about what it takes to capture this property. One of these is that symmetric encryption schemes secure against chosen-ciphertext attack or some even stronger (data-privacy) notion may already do so. We show otherwise by giving an attack on a scheme secure in the strongest data-privacy sense yet known.

Related Work. Rivest's all-or-nothing encryption is not the only way known to get more security out of a fixed number of key bits. Alternate approaches include DESX (an idea due to Rivest that was analyzed by Kilian and Rogaway [17]) and those favoring a long key set-up time, such as the method of Quisquater et al. [19]. These approaches do not incur the fixed penalty for every encrypted block that all-or-nothing encryption does, but unlike all-or-nothing encryption, they cannot work with existing encryption devices and software without changing the underlying encryption algorithm. In either case, as Rivest points out, the different approaches are complementary and can be easily combined.

Several approaches to the design of AONTs have been discussed by Rivest [20]. Our construction, like the package transform, happens to be based on a block cipher. The hash function based OAEP transform was proven secure in the Random Oracle Model by Boyko [10]. An information-theoretic treatment of a weaker form of AONTs has been given by Stinson [22]. Constructions based solely on the assumption of one-way functions have been given by Canetti et al. [12]. However, these are somewhat inefficient for practice. Applications of AONTs go beyond just the one considered in this work. They can be used to make fixed block-size encryption schemes more efficient [15], reduce communication requirements [20,14], and protect against partial key exposure [12].

2 Preliminaries

We use a standard notation for expressing probabilistic experiments and algorithms. Namely, if $A(\cdot, \cdot, \ldots)$ is a probabilistic algorithm then $a \leftarrow A(x_1, x_2, \ldots)$ denotes the experiment of running A on inputs $x_1, x_2, \ldots$ and letting a be the outcome, the probability being over the coins of A. Similarly, if A is a set then $a \leftarrow A$ denotes the experiment of selecting a point uniformly from A and assigning a this value.

BLOCK CIPHERS. For any integer $l \geq 1$ let P_l denote the space of all $(2^l)!$ permutations on l bits. A block cipher is a map $F : \{0,1\}^k \times \{0,1\}^l \mapsto \{0,1\}^l$. For every $a \in \{0,1\}^k$, $F(a,\cdot) \in P_l$. We define F_a by $F_a(x) = F(a,x)$. Let $\mathrm{BC}(k,l)$ denote the space of all block ciphers with parameters k and l as above.

We model F as an *ideal* block cipher in the sense of Shannon, in that F is drawn at random from $\mathrm{BC}(k,l)$. Given $F \in \mathrm{BC}(k,l)$, we define $F^{-1} \in \mathrm{BC}(k,l)$ by $F^{-1}(a,y) = F_a^{-1}(y)$ for $a \in \{0,1\}^k$. Note that in the experiments to follow there is no "fixed" cipher; we will refer to an ideal block cipher F, access to which will be via oracles for $F(\cdot,\cdot)$ and $F^{-1}(\cdot,\cdot)$.

ENCRYPTION MODES. Formally, an encryption mode based on a block cipher F is given by a triple of algorithms, $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, where

- $\mathcal{K}$, the *key generation algorithm*, is a probabilistic algorithm that takes a security parameter $k \in \mathsf{N}$ (provided in unary) and returns a key a specifying permutations F_a and F_a^{-1}.
- $\mathcal{E}$, the *encryption algorithm*, is a probabilistic or stateful algorithm that takes permutations F_a and F_a^{-1} (as oracles) and a message $x \in \{0,1\}^*$ to produce a ciphertext y.
- $\mathcal{D}$, the *decryption algorithm*, is a deterministic algorithm which takes permutations F_a and F_a^{-1} (as oracles) and ciphertext y to produce either a message $x \in \{0,1\}^*$ or a special symbol $\perp$ to indicate that the ciphertext was invalid.

We require that for all a which can be output by $\mathcal{K}(1^k)$, for all $x \in \{0,1\}^*$, and for all y that can be output by $\mathcal{E}^{F_a,F_a^{-1}}(x)$, we have that $\mathcal{D}^{F_a,F_a^{-1}}(y) = x$. We also require that $\mathcal{K}$, $\mathcal{E}$ and $\mathcal{D}$ can be computed in polynomial time. As the notation indicates, the encryption and decryption algorithms are assumed to have oracle access to the permutations specified by the key a but do not receive the key a itself. This is the distinguishing feature of encryption modes over other types of symmetric encryption schemes.

NOTION OF SECURITY. We recall a notion of security against chosen-plaintext attack for symmetric encryption schemes, due to Bellare et al. [2], suitably modified for encryption modes in the Shannon Model of a block cipher. This itself is an adaptation to the private-key setting of the definition of "polynomial security" for public-key encryption given by Goldwasser and Micali [13].

Definition 1. [Indistinguishability of Encryptions] *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption mode. For an adversary A and $b = 0, 1$ define the experiment*

Experiment $\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A, b)$

$F \leftarrow \mathrm{BC}(k,l)$; $a \leftarrow \mathcal{K}(1^k)$; $(x_0, x_1, s) \leftarrow A^{F,F^{-1},\mathcal{E}^{F_a,F_a^{-1}}}(\mathsf{find})$;
$y \leftarrow \mathcal{E}^{F_a,F_a^{-1}}(x_b)$; $d \leftarrow A^{F,F^{-1},\mathcal{E}^{F_a,F_a^{-1}}}(\mathsf{guess}, y, s)$; return d.

Define the advantage of A and the advantage function of Π respectfully, as follows:

$$\mathsf{Adv}^{\mathrm{ind}}_{\Pi}(A) = \Pr\left[\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A,1) = 1\right] - \Pr\left[\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A,0) = 1\right]$$

$$\mathsf{Adv}^{\mathrm{ind}}_{\Pi}(t, m, p, q, \mu) = \max_{A}\{\mathsf{Adv}^{\mathrm{ind}}_{\Pi}(A)\}$$

where the maximum is over all A with "time-complexity" t, making at most p queries to F/F^{-1}, choosing $|x_0| = |x_1|$ such that $|y| = ml$ and making at most q queries to $\mathcal{E}^{F_a, F_a^{-1}}$, these totaling at most μ bits. ▌

Here the "time-complexity" is the worst case total execution time of experiment $\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A, b)$ plus the size of the code of A, in some fixed RAM model of computation. This convention is used for other definitions in this paper, as well. The notation $A^{F, F^{-1}, \mathcal{E}^{F_a, F_a^{-1}}}$ indicates an adversary A with access to an encryption oracle $\mathcal{E}^{F_a, F_a^{-1}}$ and oracles for F and F^{-1}. The encryption oracle is provided so as to model chosen plaintext attacks, while the F/F^{-1} oracles appear since we are working in the Shannon Model of a block cipher.

3 Non-separability of Keys

We give a notion of key-privacy to capture the requirement that *every* block of the ciphertext must be "decrypted" before *any* information about underlying key (including that the key may not be the "right" one) is known. This notion is formally captured through a game in which an adversary A is imagined to run in two stages. In the find stage, A is given two randomly selected keys a_0 and a_1, and is allowed to choose a message x along with some state information s. In the guess stage, it is given a random ciphertext y of the plaintext x, under one of the selected keys, along with the state information s. Let $m = \frac{|y|}{l}$ be the number of blocks in the challenge y. The adversary is given access to oracles for F and F^{-1} in both stages. In the guess stage, we impose a restriction that the adversary may make at most $(m-1)$ queries to $F_{a_0}/F^{-1}_{a_0}$ and at most $(m-1)$ queries to $F_{a_1}/F^{-1}_{a_1}$. The adversary "wins" if it correctly identifies which of the two selected keys was used to encrypt x in the challenge.

Definition 2. [Non-Separability of Keys] *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption mode. For an adversary A and $b = 0, 1$ define the experiment*

Experiment $\mathrm{Exp}^{\mathrm{nsk}}_{\Pi}(A, b)$
$F \leftarrow \mathrm{BC}(k, l)$; $(a_0, a_1) \leftarrow \mathcal{K}(1^k)$; $(x, s) \leftarrow A^{F, F^{-1}}(\mathsf{find}, a_0, a_1)$;
$y \leftarrow \mathcal{E}^{F_{a_b}, F^{-1}_{a_b}}(x)$; $d \leftarrow A^{F, F^{-1}}(\mathsf{guess}, y, s)$; return d.

Define the advantage of A and the advantage function of Π respectfully, as follows:

$$\mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(A) = \Pr\left[\mathrm{Exp}^{\mathrm{nsk}}_{\Pi}(A, 1) = 1\right] - \Pr\left[\mathrm{Exp}^{\mathrm{nsk}}_{\Pi}(A, 0) = 1\right]$$

$$\mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(t, m, p) = \max_{A}\{\mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(A)\}$$

where the maximum is over all A with time complexity t, making at most p queries to F/F^{-1} such that, for $m = \frac{|y|}{l}$, at most $(m-1)$ of these are to $F_{a_0}/F^{-1}_{a_0}$ and at most $(m-1)$ of these are to $F_{a_1}/F^{-1}_{a_1}$ in the guess stage. ▌

Note that this definition only captures a notion concerned with the privacy of the underlying key. It does not imply security as an encryption scheme. The notion can be used in conjunction with the data-privacy notions of encryption schemes. Indeed, it also makes sense to talk about the key-privacy of encryption modes that are secure under data-privacy notions that are stronger than the one captured by Definition 1.

Non-Separability of Keys Versus Key-Search. We show that security in the non-separability of keys sense implies that "key-search" attacks are slowed down by a factor proportional to the number of blocks in the ciphertext. Indeed this is the primary motivation of using encryption modes secure in the non-separability of keys sense. Thus this implication may be taken as evidence of having a "correct" definition in Definition 2.

In the key-search notion, we measure the success of an adversary A in guessing the underlying key a given a ciphertext y (of a plaintext x of its choice). The insecurity of an encryption mode in the key-search sense is given by the maximum success over all adversaries using similar resources.

Definition 3. [Key-Search] *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption mode. For an adversary A define the experiment*

Experiment $\mathrm{Exp}^{\mathrm{ks}}_{\Pi}(A)$
$F \leftarrow \mathrm{BC}(k, l)$; $a \leftarrow \mathcal{K}(1^k)$; $(x, s) \leftarrow A^{F, F^{-1}}(\mathsf{select})$; $y \leftarrow \mathcal{E}^{F_a, F_a^{-1}}(x)$;
$a' \leftarrow A^{F, F^{-1}}(\mathsf{predict}, y, s)$; *If $a' = a$ then $d \leftarrow 1$ else $d \leftarrow 0$;* return d.

Define the success of A and the success function of Π respectfully, as follows:

$$\mathsf{Succ}^{\mathrm{ks}}_{\Pi}(A) = \Pr\left[\mathrm{Exp}^{\mathrm{ks}}_{\Pi}(A) = 1\right]$$

$$\mathsf{Succ}^{\mathrm{ks}}_{\Pi}(t, m, p) = \max_{A}\{\mathsf{Succ}^{\mathrm{ks}}_{\Pi}(A)\}$$

where the maximum is over all A with time complexity t, making at most p queries to F/F^{-1} and choosing $|x|$ such that $|y| = ml$. ▌

Note that there are no restrictions (for any key) on how many of A's p queries to F/F^{-1} are in the predict stage.

Our first theorem establishes our claim about the implication. We emphasize that this result, and every other result (on encryption) in this work, are on *encryption modes*. In particular, we assume that the encryption and decryption algorithms can be described given just oracle access to permutations and do not need the (block-cipher) key specifying these permutations.

Theorem 1. [Non-Separability of Keys Slows Down Key-Search] *Suppose Π is an encryption mode using an ideal cipher of key length k and block length l. Then*

$$\mathsf{Succ}^{\mathrm{ks}}_{\Pi}(t, m, p) \leq \mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(t', m, p) + \left(2 \cdot \left\lfloor \frac{p}{m} \right\rfloor + 4\right) \cdot \frac{1}{2^k - 1}$$

where $t' = t + \mathcal{O}(k + ml + pl)$. ▌

The proof of Theorem 1 appears in the full version of this paper [11]. We sketch only the basic idea here. The proof uses a fairly standard contradiction argument. Assume B is an adversary in the key-search sense. We construct a non-separability of keys adversary A, that uses B and has the claimed complexity. A will run B using its oracles to answer B's queries and then make its guess based on how B behaves. A complication arises due to the fact that there is a restriction on the number of queries A can make with the two keys it is given in the find stage and that B is not subject to this restriction. We get around this by having A keep track of how many queries B makes using these keys. If B ever exceeds the amount A is restricted to for one of these keys, then A guesses that its challenge was encrypted under this key. We then show that the probability of a false positive is small.

We next give an interpretation of Theorem 1. Say Π is secure in the sense of Definition 2. Then we know that for reasonable values of t', m, p the value of the $\mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(t', m, p)$ is negligible. The theorem says that for a reasonable value of t we could expect $\mathsf{Succ}^{\mathrm{ks}}_{\Pi}(t, m, p)$ to be not much more than $\left(2 \cdot \left\lfloor \frac{p}{m} \right\rfloor + 4\right) \cdot \frac{1}{2^k - 1}$. This means that after p queries to F/F^{-1} there is roughly only a $(\frac{p}{m} \cdot 2^{-k})$ chance of finding the key. Contrast this with an encryption mode where each query to F/F^{-1} could potentially rule out a candidate key. Then we would expect an $(p \cdot 2^{-k})$ chance of finding the underlying key. Thus we have succeeded in reducing the success of a key-search attack by a factor of m, as promised. (The factor of 2 in the theorem comes about due to the scaling factors implicit in the advantage function of Definition 2.)

4 Separable Attacks

It is easy to check that none of the commonly used encryption modes, such as the cipher-block-chaining (CBC) mode and the counter mode (CTR) (see [2] for a description of these modes) have the key-search resistance property we desire. There seems to be a belief that some of the existing notions and schemes may already capture this property. We show that this is unlikely by giving attacks on some paradigms that cover a large number of "promising" candidates.

Encode-then-Encipher Encryption. The variable-input-length (VIL) enciphering paradigm has been suggested in [7] as a practical solution to the problem of "encrypting" messages of variable and arbitrary lengths to a ciphertext of the same length. (Since enciphering is deterministic, it cannot be considered to be secure encryption. However, as pointed out in [8], simply encoding messages with some randomness, prior to enciphering, is enough to guarantee security as an encryption scheme.) It is claimed in [7] that the VIL paradigm also provides a way to provably achieve the goal of exhaustive key-search being slowed down proportional to the length of the ciphertext. However, we show that this is *not* the case by describing a simple but effective attack on their VIL mode of operation. The attack is effective even when the messages are encoded before enciphering. We point out here (deferring details to the full version of this paper [11]) that even "super" VIL modes [9] would be susceptible to this attack.

We describe a simplified version of an example of a VIL mode given in [7]. The construction first computes a pseudorandom value by applying a CBC-MAC on the plaintext. In the second step, the counter mode is used to "encrypt" the plaintext using the pseudorandom value from the first step as the "counter". We now describe a simple attack on this example. Our attack exploits the fact that for messages longer than a few blocks, most of the blocks in the VIL mode are being encrypted in the CTR mode. The main ideas of the VIL mode are on how to pick the "counter" for the CTR mode and on how to format the last few blocks so as to enable message recoverability while still maintaining the length requirement. We observe that the attack is effective given any two blocks of a challenge ciphertext, and moreover, is independent of the "counter" value. Given, say, just $y_i = x_i \oplus F_a(r+i)$ and $y_j = x_j \oplus F_a(r+j)$, for some plaintext $x = x_1 \cdots x_n$, counter r and indices $1 \leq i < j \leq n$, there is a test for any candidate key a' that requires just two queries to $F^{-1}(\cdot,\cdot)$. The test is that the following relationship hold: $F^{-1}(a', y_j \oplus x_j) - F^{-1}(a', y_i \oplus x_i) = j - i$. This test can be carried out effectively in the VIL mode example and serves to show that this paradigm in general does not capture the goal of slowing down exhaustive key-search.

Authenticated Encryption. The most common misconception seems to be that some of the stronger notions of data-privacy or data-integrity for symmetric encryption capture the key-search resistance property that we do in Definition 2. We claim that all of these notions, however strong they may be, are orthogonal to our notion of key-privacy. We argue this for the case of authenticated encryption, which is one of the strongest notions of security considered in symmetric encryption. In particular, this notion implies other strong notions, including security against chosen-ciphertext attack. Informally described, authenticated encryption requires that it be infeasible for an adversary to get the receiver to accept as authentic a string C where the adversary has not already witnessed C. Formal definitions appear in [4,8,16] along with methods to construct such schemes. One of the generic methods shown to be secure in the authenticated encryption sense is the "encrypt-then-MAC" paradigm [4]. In this paradigm, a ciphertext is formed by encrypting the plaintext to a string C using a generic symmetric encryption scheme secure in the indistinguishability of encryptions sense and then appending to C the output of a MAC on C. Clearly, if the underlying generic encryption scheme used does not have the property captured by our key-privacy notion, then neither would the resulting authenticated encryption scheme.

5 All-or-Nothing Transforms

The notion of an all-or-nothing transform (AONT) was suggested by Rivest [20] to enable a paradigm for realizing encryption modes with the key-search resistance property. The paradigm consists of pre-processing a message with an AONT and encrypting the result by an "ordinary" encryption mode. We give a systematic treatment of AONTs in this section. The paradigm itself will be discussed in Section 6.

SYNTAX. Formally, the syntax of an un-keyed AONT is given by a pair of algorithms, $\Pi = (\mathcal{E}, \mathcal{D})$, where

- $\mathcal{E}$, the *encoding algorithm*, is a probabilistic algorithm that takes a message $x \in \{0,1\}^*$ to produce a pseudo-ciphertext y.
- $\mathcal{D}$, the *decoding algorithm*, is a deterministic algorithm which takes a pseudo-ciphertext y to produce either a message $x \in \{0,1\}^*$ or a special symbol $\perp$ to indicate that the pseudo-ciphertext was invalid.

We require that for all $x \in \{0,1\}^*$, and for all y that can be output by $\mathcal{E}(x)$, we have that $\mathcal{D}(y) = x$. We also require that $\mathcal{E}$ and $\mathcal{D}$ be polynomial-time computable.

NOTION OF SECURITY. We give a new definition of security for AONTs. A block-length l will be associated with an AONT. During the adversary's find stage it comes up with a message and some state information. The challenge is either a pseudo-ciphertext y_0 corresponding to the chosen plaintext x or a random string y_1 of the same length as y_0. In the guess stage, it is allowed to adaptively see *all but one* of the challenge blocks and guess whether the part of challenge it received corresponds to y_0 or y_1.

Definition 4. [All-Or-Nothing Transforms] *Let $\Pi = (\mathcal{E}, \mathcal{D})$ be an* AONT *of block length l. For an adversary A and $b = 0, 1$ define the experiment*

Experiment $\mathrm{Exp}^{\mathrm{aon}}_{\Pi}(A, b)$
$(x, s) \leftarrow A(\mathsf{find})$;
$y_0 \leftarrow \mathcal{E}(x)$;
$y_1 \leftarrow \{0,1\}^{|y_0|}$; *// ($y_b = y_b[1] \cdots y_b[m]$ where $|y_b[i]| = l$ for $i \in \{1, \cdots, m\}$)*
$d \leftarrow A^{\mathcal{Y}}(\mathsf{guess}, s)$; *// ($\mathcal{Y}$ takes an index $j \in \{1, \ldots, m\}$ and returns $y_b[j]$)*
return d.

Define the advantage of A and the advantage function of Π respectfully, as follows:

$$\mathsf{Adv}^{\mathrm{aon}}_{\Pi}(A) = \Pr\left[\mathrm{Exp}^{\mathrm{aon}}_{\Pi}(A, 1) = 1\right] - \Pr\left[\mathrm{Exp}^{\mathrm{aon}}_{\Pi}(A, 0) = 1\right]$$

$$\mathsf{Adv}^{\mathrm{aon}}_{\Pi}(t, m) = \max_{A}\{\mathsf{Adv}^{\mathrm{aon}}_{\Pi}(A)\}$$

where the maximum is over all A with time complexity t, choosing $|x|$ such that $|y_0| = ml$ and making at most $(m-1)$ queries to $\mathcal{Y}$. ▌

Our formalization differs from that given by Boyko [10] in some significant ways. We require that the missing information be a block as opposed to some variable number of bits anywhere in the output. This captures a weaker notion, but as argued earlier, this is not necessarily a drawback. A consequence of this is that we are able to design more efficient AONTs. Notice that, in our missing-block formalization, we ask for the indistinguishability of the AONT output (with a missing block) from a *random string* of the same length. This is in contrast to the typical "indistinguishability of outputs on two inputs" required by [10] or any of the indistinguishability-based notions of encryption. We give some intuition

for the need for this strengthening here. Consider a transform that added some known redundancy to every block (say, the first bit of every output block was always a 0). This alone would not make a transform insecure if we had used the "indistinguishability of outputs on two inputs" formulation for capturing all-or-nothingness, since the outputs on every input would have this same redundancy. However under our formulation we will find such a transform to be insecure since the random string would not necessarily have this redundancy. It turns out that if the all-or-nothing encryption paradigm is to have the key-search resistance property then such transforms cannot be considered to be secure as AONTs. Recall that the paradigm is to use an "ordinary" encryption mode on the output of an AONT. It is easy to see that it is essential that a key-search adversary "decrypting" one block of ciphertext should not be able to figure out if the decrypted value was the output of an AONT or not.

We have so far assumed the standard model, but we will often want to consider AONTs in some stronger model like the Random Oracle Model or the Shannon Model. Definition 4 can be suitably modified to accommodate these. For example, with the Shannon Model of an ideal block cipher F, we will assume that all parties concerned have access to F and F^{-1} oracles. The queries made to F/F^{-1} become a part of the definition. We will define $\mathsf{Adv}^{\mathrm{aon}}_{\Pi}(t, m, p)$ rather than $\mathsf{Adv}^{\mathrm{aon}}_{\Pi}(t, m)$, where in addition to the usual parameters, p is the maximum number of queries allowed to F/F^{-1}.

Construction. We give a construction based on the CTR mode of encryption. We describe the transform CTRT = ($\mathcal{E}$-CTRT, $\mathcal{D}$-CTRT) of block length l, using an ideal cipher F with key length k and block length l, where $k \leq l$. (This condition can be easily removed and is made here only for the sake of exposition.) The message x to be transformed is regarded as a sequence of l-bit blocks, $x = x[1] \ldots x[n]$ (padding is done first, if necessary). We define $\mathcal{E}\text{-CTRT}^{F,F^{-1}}(x)$ and $\mathcal{D}\text{-CTRT}^{F,F^{-1}}(x')$, as follows:

Algorithm $\mathcal{E}\text{-CTRT}^{F,F^{-1}}(x[1] \ldots x[n])$	Algorithm $\mathcal{D}\text{-CTRT}^{F,F^{-1}}(x')$
$K' \leftarrow \{0,1\}^l$	Parse x' as $x'[1] \ldots x'[n+1]$
$K = K' \bmod 2^k$ // ($\lvert K \rvert = k$)	$K' = x'[1] \oplus \cdots \oplus x'[n+1]$
for $i = 1, \ldots, n$ do	$K = K' \bmod 2^k$ // ($\lvert K \rvert = k$)
$\quad x'[i] = x[i] \oplus F_K(i)$	for $i = 1, \ldots, n$ do
$x'[n+1] = K' \oplus x'[1] \oplus \ldots \oplus x'[n]$	$\quad x[i] = x'[i] \oplus F_K(i)$
return $x'[1] \ldots x'[n+1]$	return $x[1] \ldots x[n]$

CTRT is a variant of Rivest's package transform [20] where one "pass" has been skipped altogether. Yet we find it to be secure in the sense of Definition 4.

Theorem 2. [Security of CTRT**]** *Suppose transform* CTRT *of block length l uses an ideal cipher of key length k and block length l (where $k \leq l$). Then for any t, m, p such that $m + p \leq 2^{k-1}$,*

$$\mathsf{Adv}^{\mathrm{aon}}_{\mathrm{CTRT}}(t, m, p) \leq \frac{m^2 + 8 \cdot p}{2^k}$$

The proof of this theorem appears in the full version of this paper [11]. We mention here only some of the key aspects of the proof. As long as at least one block of the output is missing, the key K used to "encrypt" the message blocks is information theoretically hidden. The main step in the analysis is to bound the probability of an adversary calling its oracles with key K. In doing this, we need to be particularly careful with the fact that we allow adversaries to be adaptive. Another issue that complicates matters is the injectivity of ideal ciphers. For example, we cannot conclude that if an adversary has never queried its oracles with key K, then its "challenges" must be indistinguishable to it. The injectivity makes certain conditions impossible with the "AONT-derived" challenge that are possible with the "random" challenge.

It is conceivable that the package transform of Rivest [20] is actually secure in the strong sense captured by Boyko [10]. CTRT, on the other hand, is clearly *insecure* in that strong sense. (Note that CTRT would also have been secure in the sense given by Rivest [20].) However, as we will see next, it turns out that CTRT is strong enough that when used to realize the all-or-nothing encryption paradigm, the resulting mode will have the properties we desire.

6 All-or-Nothing Encryption

The all-or-nothing encryption paradigm consists of composing an AONT with an "ordinary" encryption mode. We study the particular case when the ordinary encryption mode is the codebook (ie. ECB) mode. It is easy to see that the codebook mode by itself is not secure encryption. However it does have many advantages over some of the other modes. In particular, it is simple, efficient, length-preserving, and admits an efficient parallel implementation. Following [20], we will refer to an AONT followed by the codebook mode as the "all-or-nothing codebook mode". We will establish the security of the all-or-nothing codebook mode in the theorems to follow. Similar results can be derived when some other reasonable mode is used in place of the codebook mode.

The all-or-nothing codebook mode is first and foremost a secure encryption scheme. We establish this in the following theorem.

Theorem 3. [**Security in the** indistinguishability of encryptions **sense**] *Suppose $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is an all-or-nothing codebook mode using an ideal cipher of key length k and block length l and an all-or-nothing transform $\Pi' = (\mathcal{E}', \mathcal{D}')$ of block length l. Let T be the time to decode a ml bit string using $\mathcal{D}'$ and nl be the length of a decoded ml bit string. Then for any $n \geq 2$ and any p, q, μ,*

$$\mathsf{Adv}^{\mathrm{ind}}_{\Pi}(t, m, p, q, \mu) \leq 2m \cdot \mathsf{Adv}^{\mathrm{aon}}_{\Pi'}(t', m) + \frac{2mp}{2^k} + \frac{2m}{2^l}$$

where $t' = t + (\frac{\mu}{l} + m - 1) \cdot T + \mathcal{O}(ml + pl + \mu)$. ▌

The proof of this theorem appears in the full version of this paper [11]. The intuition behind the result is as follows. An AONT has the property that the chances

of a collision amongst the blocks of its output (even across multiple queries made by an adaptive adversary) is small. Thus when an AONT is composed with the codebook mode, we have that with high probability each block of the ciphertext is a result of having enciphered on a new point. Note that although the codebook mode is deterministic, the fact that the AONT itself is probabilistic makes the resulting all-or-nothing codebook mode probabilistic. The main part of the proof is in formalizing and establishing this property of AONTs. We show that if this property did not hold for some transform, then that transform could not be secure as an AONT.

We next show that the all-or-nothing codebook mode also has the desired key-search resistance property.

Theorem 4. [Security in the non-separability of keys **sense]** *Suppose* $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ *is an all-or-nothing codebook mode using an ideal cipher of key length* k *and block length* l *and an all-or-nothing transform* $\Pi' = (\mathcal{E}', \mathcal{D}')$ *of block length* l*. Then*

$$\mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(t, m, p) \leq 2m \cdot \mathsf{Adv}^{\mathrm{aon}}_{\Pi'}(t', m)$$

where $t' = t + \mathcal{O}(ml + pl)$.

Proof. We use a contradiction argument to prove this. Let B be an adversary in the non-separability of keys sense. We construct an all-or-nothing adversary A, that uses B and has the claimed complexity.

Since we are assuming an all-or-nothing adversary A that does not receive F/F^{-1} oracles, these must be simulated when running B. The cost of this simulation will appear in the time complexity of A.

We use the notation $(+, K, z)$ (respectively, $(-, K, z)$) to indicate a query to F (respectively, F^{-1}) with key K and a l-bit string z; the expected response being $F_K(z)$ (respectively, $F_K^{-1}(z)$).

For an integer m let $[m] = \{1, \cdots, m\}$. For an ml bit string z let $z = z[1] \cdots z[m]$ such that $|z[i]| = l$ for $i \in [m]$.

The adversary A using adversary B is given in Figure 1. The idea is the following: A first picks two keys a_0, a_1 to simulate the experiment underlying non-separability of keys for B. It runs B, answering all of its queries by simulating the oracles, until B ends its find stage by returning some string x. A returns x as the output of its own find stage. In its guess stage, A will pick a random bit d and an index j. A will then ask its oracle $\mathcal{Y}$ for all but the j-th challenge block. It then uses the key a_d to encipher these blocks to get all blocks of a string z other than $z[j]$. A assigns a random value to $z[j]$ and then runs B's guess stage with the challenge being z. It will simulate B's oracles as before, but halt if B ever asks the query $(-, a_d, z[j])$ or $(+, a_d, F_{a_d}^{-1}(z[j]))$. If B was halted, then A outputs a random bit as its guess. Otherwise, it checks to see if B was correct. A guesses that its challenge must have been "real" if B is correct and outputs a random bit otherwise.

Algorithm $A(\mathsf{find})$ $a_0, a_1 \leftarrow \mathcal{K}(1^k)$ run $B(\mathsf{find}, a_0, a_1)$ using SimBOr let (x, s) be the output of B $s' \leftarrow (s, x, a_0, a_1)$ return (x, s') Subroutine SimBOr if B makes a query $(+, K, u)$ then answer B with $F_K(u)$ if B makes a query $(-, K, u)$ then answer B with $F_K^{-1}(u)$	Algorithm $A^{\mathcal{Y}}(\mathsf{guess}, (s, x, a_0, a_1))$ $d \leftarrow \{0,1\}$ $j \leftarrow [m]$ for $(i \in [m]) \wedge (i \neq j)$ do $y[i] \leftarrow \mathcal{Y}(i)$ $z[i] \leftarrow F_{a_d}(y[i])$ $z[j] \leftarrow \{0,1\}^l$ run $B(\mathsf{guess}, z, s)$ using SimBOr until B makes a query $(-, a_d, z[j])$ or B makes a query $(+, a_d, F_{a_d}^{-1}(z[j]))$ or B halts if B halts then let d' be its output else $b' \leftarrow \{0,1\}$ if $d' = d$ then $b' \leftarrow 0$ else $b' \leftarrow \{0,1\}$ return b'

Fig. 1. An all-or-nothing adversary using a non-separability of keys adversary

From the description, we have that the time complexity $t' = t + \mathcal{O}(ml + pl)$.

Next we compute the advantage function. For $b \in \{0,1\}$ let Probability Space b be that of the following underlying experiment:

$$(x, s') \leftarrow A(\mathsf{find});\ y_0 \leftarrow \mathcal{E}'(x);\ y_1 \leftarrow \{0,1\}^{ml};\ \mathcal{Y}(i) = y_b[i] \text{ for } i \in [m] :$$

For $b \in \{0,1\}$ let $\Pr_b[\cdot]$ denote the probability under Probability Space b.

$$\mathsf{Adv}^{\mathrm{aon}}_{\Pi'}(A) \stackrel{\mathrm{def}}{=} \Pr_0[A^{\mathcal{Y}}(\mathsf{guess}, s') = 0] - \Pr_1[A^{\mathcal{Y}}(\mathsf{guess}, s') = 0]$$

Hereafter, we suppress the superscripts and parenthesized parts for clarity.

Let Fail be the event that B makes a query $(-, a_d, z[j])$ or $(+, a_d, F_{a_d}^{-1}(z[j]))$ where j is the index of the block in y that A does not receive, $z[j]$ is the random block picked by A and $d \in \{0,1\}$ is the bit that selects the key a_0 or a_1. We have

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{aon}}_{\Pi'}(A) &\stackrel{\mathrm{def}}{=} \Pr_0[A = 0] - \Pr_1[A = 0] \\
&= \Pr_0[A = 0 \mid \mathsf{Fail}] \cdot \Pr_0[\mathsf{Fail}] + \Pr_0\left[A = 0 \mid \overline{\mathsf{Fail}}\right] \cdot \Pr_0[\overline{\mathsf{Fail}}] - \\
&\quad \Pr_1[A = 0 \mid \mathsf{Fail}] \cdot \Pr_1[\mathsf{Fail}] - \Pr_1\left[A = 0 \mid \overline{\mathsf{Fail}}\right] \cdot \Pr_1[\overline{\mathsf{Fail}}]
\end{aligned}$$

Now from the description of A we have:

$$\begin{aligned}
\Pr_0[\mathsf{Fail}] &= \Pr_1[\mathsf{Fail}] = \frac{m-1}{m} \\
\Pr_0[A = 0 \mid \mathsf{Fail}] &= \Pr_1[A = 0 \mid \mathsf{Fail}] = 0.5 \\
\Pr_0\left[A = 0 \mid \overline{\mathsf{Fail}}\right] &= \Pr_0[B \text{ guesses correctly}] = 0.5 + 0.5 \cdot \mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(B) \\
\Pr_1\left[A = 0 \mid \overline{\mathsf{Fail}}\right] &= \Pr_1[B \text{ guesses correctly}] = 0.5
\end{aligned}$$

The derivation of these equalities is quite straightforward. The only one requiring explanation is the last one. To determine $\Pr_1[\, B \text{ guesses correctly}\,]$ we recall that in Probability Space 1 the challenge z that B receives is the codebook output with key a_d on a random string y_1. The probability we want is that of B guessing d correctly. Given that B does not see y_1, but just the output z under the codebook mode with an ideal cipher, it is easy to see that the probability is exactly as claimed. Continuing with the advantage, we have

$$\mathsf{Adv}^{\mathrm{aon}}_{\Pi'}(A) = \frac{1}{m} \cdot \left(\frac{1}{2} + \frac{1}{2} \cdot \mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(B) - \frac{1}{2} \right) \;=\; \frac{1}{2m} \cdot \mathsf{Adv}^{\mathrm{nsk}}_{\Pi}(B)$$

From this, we get the claimed relationship in the advantage functions. ▌

We have assumed in our treatment of the all-or-nothing encryption paradigm that the block size associated with the AONT is the same as that of the encryption mode following it. This assumption could be easily removed by making a few changes to our framework. However, in this form, it has certain implications when a block-cipher based construction, like CTRT, is used as the underlying AONT. We would require that the key space for the AONT block cipher be sufficiently large that brute force searches are infeasible. The block cipher used in the AONT does not have to be the same as that used to encrypt its output, though it certainly could be.

7 Comments and Open Problems

Our notions and proofs are in the Shannon Model of a block cipher. The model makes some strong assumptions and may be unsuitable for some of today's block ciphers with their delicate key schedules. This raises the concern that our results may not be telling us much about the "real-world". However, we claim that the results proven in this model are still meaningful since they permit "generic" attacks (ie. attacks that assume the underlying primitives to be "ideal"). In practice, most attacks disregard the cryptanalytic specifics of the block cipher anyway and instead treat it as a black-box transformation.

Our use of the Shannon Model for capturing non-separability of keys was driven by the need to correctly and usefully formalize the notion. It is hard to see how this could have been done in the standard model. In establishing our claims about all-or-nothing encryption modes, we used a definition of an AONT from the standard model. (Things would change very little if we had instead started with a definition from one of the other models.) However, to prove that our CTRT transform was secure as an AONT, we needed to work in the Shannon Model. It may be possible to prove such constructions in the standard model or perhaps some other weaker model, but this is something that is currently unknown. Note that we do know that there are AONTs that can be proven secure in the standard model [12]. However, for efficiency reasons, we do not consider these to be viable options in practice.

In the case of CTRT being used as the AONT, the cost of the resulting all-or-nothing codebook mode would be a factor of only two greater than CBC. From our results, we get that the resulting all-or-nothing codebook mode would be secure in the non-separability of keys sense, as well as being secure against chosen-plaintext attack. It would be interesting to see if all-or-nothing encryption modes (with some modifications, if required) could be shown to be secure against chosen-ciphertext attack.

Acknowledgements

I am indebted to Mihir Bellare for providing invaluable support and direction with this work. Many of the ideas found here are due to him. I would also like to thank Jee Hea Lee, Sara Miner and the CRYPTO 2000 program committee and reviewers for their very helpful comments.

The author was supported in part by Mihir Bellare's 1996 Packard Foundation Fellowship in Science and Engineering and NSF CAREER Award CCR-9624439.

References

1. W. Aiello, M. Bellare, G. Di Crescenzo and R. Venkatesan, "Security amplification by composition: The case of doubly-iterated, ideal ciphers," *Advances in Cryptology - Crypto '98*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.
2. M. Bellare, A. Desai, E. Jokipii and P. Rogaway, "A concrete security treatment of symmetric encryption," *Proceedings 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*, Lecture Notes in Computer Science 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," Report 2000/025, *Cryptology ePrint Archive*, `http://eprint.iacr.org/`, May 2000.
5. M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," *Proceedings of the 1st Annual Conference on Computer and Communications Security*, ACM, 1993.
6. M. Bellare and P. Rogaway, "Optimal asymmetric encryption," *Advances in Cryptology - Eurocrypt '94*, Lecture Notes in Computer Science Vol. 950, A. De Santis ed., Springer-Verlag, 1994
7. M. Bellare and P. Rogaway, "On the construction of variable-input-length ciphers," *Fast Software Encryption '99*, Lecture Notes in Computer Science Vol. 1636, L. Knudsen ed., Springer-Verlag, 1999.
8. M. Bellare and P. Rogaway, "Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography," Manuscript, December 1998, available from authors.
9. D. Bleichenbacher and A. Desai, "A construction of super-pseudorandom cipher," Manuscript, May 1999, available from authors.

10. V. Boyko, "On the security properties of OAEP as an all-or-nothing transform," *Advances in Cryptology - Crypto '99*, Lecture Notes in Computer Science Vol. 1666, M. Wiener ed., Springer-Verlag, 1999.
11. A. Desai, "The security of all-or-nothing encryption," Full version of this paper, available via: `http://www-cse.ucsd.edu/users/adesai/`.
12. R. Canetti, Y. Dodis, S. Halevi, E. Kushilevitz and A. Sahai, "Exposure-Resilient Cryptography: Constructions for the All-Or-Nothing Transform without Random Oracles," *Advances in Cryptology - Eurocrypt '00*, Lecture Notes in Computer Science Vol. 1807, B. Preneel ed., Springer-Verlag, 2000.
13. S. Goldwasser and S. Micali, "Probabilistic encryption," *J. of Computer and System Sciences*, Vol. 28, April 1984, pp. 270–299.
14. M. Jakobsson, J. Stern and M. Yung, "Scramble All, Encrypt Small," *Fast Software Encryption '99*, Lecture Notes in Computer Science Vol. 1636, L. Knudsen ed., Springer-Verlag, 1999.
15. D. Johnson, S. Matyas, and M. Peyravian, "Encryption of long blocks using a short-block encryption procedure," Submission to IEEE P1363a, available via: `http://grouper.ieee.org/groups/1363/contributions/peyrav.ps`, Nov. 1996.
16. J. Katz and M. Yung, "Unforgeable Encryption and Adaptively Secure Modes of Operation," *Fast Software Encryption '00*, Lecture Notes in Computer Science Vol. ??, B. Schneier ed., Springer-Verlag, 2000.
17. J. Kilian and P. Rogaway, "How to protect DES against exhaustive key search," *Advances in Cryptology - Crypto '96*, Lecture Notes in Computer Science Vol. 1109, N. Koblitz ed., Springer-Verlag, 1996.
18. National Bureau of Standards, NBS FIPS PUB 81, "DES modes of operation," U.S Department of Commerce, 1980.
19. J.-J. Quisquater, Y. Desmedt and M. Davio, "The importance of "good" key scheduling schemes (how to make a secure DES scheme with $\leq$ 48 bit keys)," *Advances in Cryptology - Crypto '85*, Lecture Notes in Computer Science Vol. 218, H. Williams ed., Springer-Verlag, 1985.
20. R. Rivest, "All-or-nothing encryption and the package transform," *Fast Software Encryption '97*, Lecture Notes in Computer Science Vol. 1267, E. Biham ed., Springer-Verlag, 1997.
21. C. Shannon, "Communication theory of secrecy systems," *Bell Systems Technical Journal*, Vol. 28, No. 4, 1949, pp. 656-715.
22. D. Stinson, "Something about all-or-nothing (transforms)," Manuscript. Available from: `http://www.cacr.math.uwaterloo.ca/~dstinson/`, June 1999.