EFFICIENT ALL-OR-NOTHING ENCRYPTION USING CTR MODE

Robert P. McEvoy and Colin C. Murphy
Department of Electrical & Electronic Engineering,
University College Cork, Ireland.

Keywords: All-or-Nothing Transforms, CTRT, CTR Mode, Efficient Symmetric-Key Encryption, MANETs.

Abstract: All-or-Nothing Encryption is a useful technique which can heighten the security of block ciphers. It can also be used to design faster symmetric-key cryptosystems, by decreasing the number of required encryption operations at run-time. An open problem in the literature regards the speed of all-or-nothing encryption, which we address in this paper by combining two techniques from the literature, forming a new all-or-nothing mode of operation. Trade-offs in the implementation of this design are considered, and theoretical proofs of security are provided.

1 INTRODUCTION

All-or-Nothing Transforms were originally proposed by Rivest, as a method to hinder brute-force key search attacks on block ciphers such as DES (Rivest, 1997). Essentially, All-or-Nothing Encryption consists of two stages. By applying an All-or-Nothing Transform (AONT) to a plaintext message, a 'pseudo-message' is formed. This pre-processing stage is not considered encryption, however, as the AONT does not utilise a secret key. An All-or-Nothing Encryption (AONE) mode is formed when the AONT output is encrypted using a symmetric block cipher. The resulting cryptosystem has the property that a brute-force attacker must decrypt *all* of the ciphertext blocks when testing each key. Hence, an exhaustive key-search attack on an AONE mode is slowed, in proportion to the number of blocks in the ciphertext.

However, modern symmetric-key cryptosystems are based on block ciphers with longer key lengths than DES, such as AES (minimum 128 bits). An attack on AES-128 is beyond the capability of modern computers, and it is believed that 128-bit symmetric keys will be secure until after the year 2030 (ECRYPT, 2006).

A second, more currently relevant application of AONTs relates to increasing the efficiency of block-cipher based encryption schemes (Johnson et al., 1996). In this scheme, an AONT is applied to a plaintext message as above, but only some (as opposed to all) of the pseudo-message blocks are subsequently encrypted. The AONT mixes the plaintext in such a way that all of the pseudo-message blocks are required in order to invert the transform and regain the plaintext. Therefore, fewer secret-key encryptions and decryptions are required.

In order for this 'efficient encryption' application of AONTs to be worthwhile, the AONT used must itself be quickly and efficiently computable. In (Rivest, 1997), an AONE mode was presented where the time penalty for encryption was a factor of 3 greater than standard CBC mode encryption, along with an open question as to whether this latency could be improved. Desai (Desai, 2000) proposed an AONT construction which reduced the relative penalty to a factor of 2. In this paper, a new AONE mode is considered for the first time, combining Desai's AONT and the well-known CTR mode of encryption. We show that when our design is used, the penalty for AONE can, under certain circumstances, be reduced to the negligible cost of just one xor operation.

The rest of this paper is organised as follows. The following section defines AONTs, and surveys related work. In Section 3 we present the new all-or-nothing encryption mode, analyse its efficiency, and consider some applications where it would be beneficial. Section 4 focuses on the provable security of the scheme, and conclusions are given in Section 5.

2 ALL-OR-NOTHING TRANSFORMS

2.1 Definitions

Informally, an all-or-nothing transform f maps a sequence of plaintext blocks $x_1, x_2, \ldots, x_n$ to a sequence of pseudo-message blocks, denoted

P. McEvoy R. and C. Murphy C. (2006).
EFFICIENT ALL-OR-NOTHING ENCRYPTION USING CTR MODE.
In *Proceedings of the International Conference on Security and Cryptography*, pages 237-245
DOI: 10.5220/0002098902370245

$y_1, y_2, \ldots, y_m$. As put forward in (Rivest, 1997), AONTs should possess four properties:

- Given the sequence of pseudo-message blocks, one can invert the transform to retrieve the plaintext blocks.
- Both the AONT and its inverse should be efficiently computable.
- All AONTs should be randomised, in order to avoid chosen-message and known-message attacks.
- Most importantly, if any one of the pseudo-message blocks is unknown, it should be computationally infeasible to invert the AONT, or determine any function of any plaintext block.

Since the original proposal, several other authors have published formal AONT definitions in the literature. The definitions differ in the strength of the security notions which each author uses.

Desai (Desai, 2000) defined a new notion of security for AONTs, called 'non-separability of keys', which captures the requirement for an adversary to decrypt every block of ciphertext before gaining any information about the underlying key. In addition, Desai required his AONTs to be secure in the 'indistinguishability of encryptions' sense, which is a notion defined in (Bellare et al., 1997). Furthermore, the AONT pseudo-message should be indistinguishable from a random string. We adopt these notions of security in our constructions.

2.2 Applications

In this paper, we focus on the use of AONTs for efficient encryption in symmetric-key block cipher cryptosystems. This is discussed further in Section 3. Another application of AONTs has already been mentioned in Section 1, namely the original application of impeding brute-force key search attacks. In their respective theses, Boyko (Boyko, 2000) and Dodis (Dodis, 2000) both give excellent surveys of potential AONT applications. These include remotely keyed encryption, gap secret sharing and protecting against partial key exposure. Subsequent publications have proposed the use of AONTs in electronic voting protocols (Kiong and Samsudin, 2003), 'multiple encryption' schemes (Zhang et al., 2004b), and secure file deletion schemes (Peterson et al., 2005).

2.3 Constructions

This paper deals specifically with the CTR-Transform (CTRT) of (Desai, 2000), which is described in Section 3, where it is used in our proposed efficient symmetric-key AON encryption scheme. Several other candidate constructions for AONTs have been proposed in the literature. These are listed below, with references for the interested reader.

- Package Transform (Rivest, 1997)
- OAEP (Boyko, 2000)
- Exposure-Resilient Function-based Transforms (Dodis et al., 2001)
- Quasigroup-based AONTs (Marnas et al., 2003)
- 'Extended-Indistinguishable' AONTs (Zhang et al., 2004a)
- Error-Correcting Code-based AONTs (Byers et al., 2006)

3 CTRT-CTR ENCRYPTION

This section introduces our new mode of all-or-nothing encryption, which we call 'CTRT-CTR'. In essence, it combines Desai's CTRT AONT (Desai, 2000) with the popular CTR mode of block cipher encryption (Lipmaa et al., 2000). Indeed, it is surprising that this mode has not been suggested already in the literature. We show that CTRT-CTR affords fast AON encryption, that is not attainable with other AONTs or encryption modes discussed in the literature to date.

3.1 Package Transform

Before examining the CTR Transform, it is instructive to first consider Rivest's original AONT proposal, the 'Package Transform' (Rivest, 1997). Since the package transform is based on a block cipher F, the plaintext message x being processed is broken up into fixed-sized blocks of data, labelled $x_1, x_2, \ldots, x_n$. A random key K' is chosen for the AONT, from a large enough keyspace to deter a brute force attack. Note that this key is not a 'secret key', and it does not have to be shared explicitly with the recipient of the message via public-key cryptography. The pseudo-message y is calculated as follows:

$$y_i = x_i \oplus F_{K'}(i) \qquad (1)$$

for $i = 1, \ldots, n$, where $F_{K'}(\cdot)$ denotes encryption with the block cipher F using the key K', and $\oplus$ denotes bit-wise xor. To complete the package transform, a further pseudo-message block y_{n+1} is added:

$$y_{n+1} = K' \oplus h_1 \oplus h_2 \oplus \cdots \oplus h_n \qquad (2)$$

where

$$h_i = F_{K_0}(y_i \oplus i) \qquad (3)$$

for $i = 1, \ldots, n$, and K_0 is a fixed, publically known key. In effect, the final pseudo-message block comprises a hash of all the previous pseudo-message blocks, xored with the random key K'. To invert the transform, the receiver simply calculates the random key, and uses it to decrypt the first n pseudo-message blocks:

$$K' = y_{n+1} \oplus h_1 \oplus \cdots \oplus h_n \quad (4)$$
$$x_i = y_i \oplus F_{K'}(i) \quad (5)$$

for $i = 1, \ldots, n$. Rivest observed that if the package transform was combined with an ordinary encryption mode such as CBC, the time to encrypt would increase threefold, as three passes through the block cipher encryption would be required. Rivest set the task of reducing this latency as an open problem, which Desai solved using the CTR transform.

3.2 CTR Transform

In the CTR Transform, as in the case of the package transform, a random key K' is chosen, and the plaintext is transformed via:

$$y_i = x_i \oplus F_{K'}(i) \quad (6)$$

The final pseudo-message block is given by:

$$y_{n+1} = K' \oplus y_1 \oplus y_2 \oplus \cdots \oplus y_n \quad (7)$$

Clearly, CTRT operates in the same fashion as the package transform, with one block cipher encryption stage omitted. In fact, Equations (6) and (7) are equivalent to Equations (1) and (2) with $h_i = y_i$. The final pseudo-message block is composed of the random key K' xored with all of the previous pseudo-message blocks. Intuitively, this may seem less secure than the package transform, yet it is proven secure within the model and definitions used in (Desai, 2000). Using CTRT, Desai reduced the cost of all-or-nothing encryption to just a factor of two greater than CBC mode, hence answering the open problem of (Rivest, 1997).

3.3 CTRT-CTR Mode

In the CTR mode of encryption (Lipmaa et al., 2000), the sender maintains an l-bit counter ctr, where l is the block length of the underlying cipher F. The value of ctr can be transmitted in the clear to the receiver when sending the ciphertext. A plaintext $x_1, x_2, \ldots, x_n$ is encrypted with a shared secret key K to a ciphertext $z_1, z_2, \ldots, z_n$ according to:

$$z_i = x_i \oplus F_K(ctr + i) \quad (8)$$

for $i = 1, \ldots, n$. Decryption is performed on the receiver side according to:

$$x_i = z_i \oplus F_K(ctr + i) \quad (9)$$

for $i = 1, \ldots, n$.

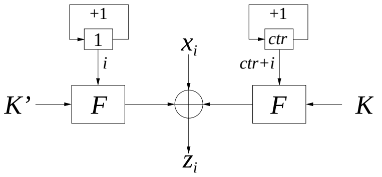

Figure 1: CTRT-CTR Mode Encryption, Equation (11).

Here we propose combining the CTR mode of encryption with the CTRT transform, via:

$$z_i = y_i \oplus F_K(ctr + i) \quad (10)$$
$$= x_i \oplus F_{K'}(i) \oplus F_K(ctr + i) \quad (11)$$

for $i = 1, \ldots, n$. For the final ciphertext block:

$$z_{n+1} = K' \oplus y_1 \oplus \cdots \oplus y_n \oplus F_K(ctr + n + 1) \quad (12)$$

To decrypt a message encrypted with CTRT-CTR, the receiver first calculates pseudo-message blocks:

$$y_i = z_i \oplus F_K(ctr + i) \quad (13)$$

for $i = 1, \ldots, n$, and uses their xor to uncover the random key K':

$$K' = z_{n+1} \oplus F_K(ctr + n + 1) \oplus \bigoplus_{i=1}^{n} y_i \quad (14)$$

A second pass is then required, where K' is used to retrieve the plaintext message blocks x_i:

$$x_i = y_i \oplus F_{K'}(i) \quad (15)$$

CTRT-CTR encryption is illustrated for block i ($i \neq n + 1$) in Figure 1.

3.4 Latency Considerations

The attraction of CTR-CTRT lies in its suitability for pre-calculation and parallelisation. With reference to Equation (11), it can be seen that all of the variables to be encrypted by F are potentially known by the sender *before* the plaintext block x_i becomes available. Therefore, the sender can use idle clock cycles to pre-compute $F_{K'}(i) \oplus F_K(ctr + i)$. When x_i becomes available, it can be quickly encrypted on-the-fly using a single l-bit xor. Therefore, given the necessary processing/memory resources, CTRT-CTR can acutely reduce the run-time cost of AON encryption.

This saving is possible because the CTRT and CTR mode do not operate directly on the plaintext blocks (other than with xor), and each ciphertext block is independent of the others. To the best of our knowledge, no other AONT or block cipher encryption mode in the literature possesses this property. Of course, not all implementations of AONE systems will have the capacity for full pre-calculation and storage of the $F_{K'}(i) \oplus F_K(ctr + i)$ 'one-time pad'. Below we present four implementation scenarios, beginning with the fastest. These are summarised in Figure 2.

(1) Full pre-processing: In this scenario, we assume that the sender A has knowledge of the secret key K before the plaintext x becomes available. A computes $F_{K'}(i)$ and $F_K(ctr+i)$ during idle clock cycles, requiring $l(2n+1)$ bits of memory storage. The run-time latency for encryption is negligible (i.e. one l-bit xor).

(2) Partial pre-processing: Here we assume that A does not know K before x becomes available. Because A chooses K', the $F_{K'}(i)$ values (i.e. the AONT part) can be calculated in advance, but the $F_K(ctr + i)$ must be computed during run-time. The latency in this scenario is the same as for ordinary CBC mode encryption, and nl bits of storage are required.

(3) Online parallel processing: In this case, A has no capability (or perhaps desire) for storage, but has two block cipher encryption engines at its disposal. These may be dedicated hardware cores, or microprocessors configured to perform F efficiently. This scheme runs at the same speed as ordinary CBC encryption, but requires no storage.

(4) Online processing: In this worst case, A performs no pre-processing, and has one F-processor available to it. Then the cost is the same as Desai's proposed CTRT-ECB scheme, i.e. twice that of CBC encryption. Memory of nl bits is also required to store intermediate results.

We note that the savings from using CTRT-CTR mode are forthcoming only on the transmission side. Receiver B must perform decryption in two stages, as described in Equations (13)-(15) (Equation (14) can be calculated cumulatively to save memory). This penalty is no worse than that for a receiver decrypting Desai's CTRT-ECB mode, however.

3.5 Efficient All-or-Nothing Encryption

CTRT-CTR mode can be further enhanced using the 'efficient encryption' method (United States Patent 5,870,470) introduced in Section 1, and pioneered in (Johnson et al., 1996). This technique takes advantage of the fact that an AONT cannot be inverted unless all of the pseudo-message blocks are known. Knowledge of some of the pseudo-message blocks should not reveal any information about the plaintext. Therefore, it should be possible to encrypt only a subset (as opposed to all) of the pseudo-message blocks, and still maintain the same level of security. The provable security of this remarkable scheme is investigated in Section 4.

The efficient CTRT-CTR encryption scheme would proceed as follows. Sender A applies the CTRT to the plaintext, as described by Equations (6) and (7). A then chooses a subset of the pseudo-message blocks to encrypt, whose indices are given by $\mathcal{S}$, where $\#\mathcal{S} = r$, $1 \le r \le n$. Of course, $\mathcal{S}$ must be made public so that receiver B can decrypt the ciphertext. To form the ciphertext z, A computes:

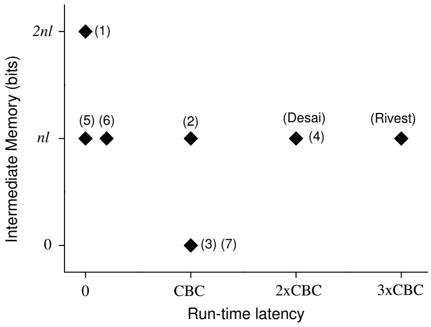

Figure 2: Performance Trade-offs with AONE.

$$z_i = \begin{cases} y_i \oplus F_K(ctr + i) & \text{for } i \in \mathcal{S} \\ y_i & \text{for } i \notin \mathcal{S} \end{cases} \tag{16}$$

The efficient encryption method allows even faster implementations of CTRT-CTR. Assuming $r = 1$ (i.e. just one pseudo-message block is encrypted), the scenarios described in Section 3.4 become:

(5) Full pre-processing: Run-time latency of one xor (negligible), $l(n + 1)$ bits of memory.

(6) Partial pre-processing: Run-time latency of one xor and one encryption, nl bits of memory.

(7) Online processing: Run-time latency approximately that of ordinary CBC encryption (as $r << n$), regardless of multiple processors. No extra memory required.

Figure 2 summarises the latency/memory trade-off associated with implementing CTRT-CTR in software or hardware. Fast, efficient all-or-nothing encryption is clearly attainable if the application justifies the added processing and storage costs. The best results are obtained by combining CTRT-CTR (or CTRT in any encryption mode) with the efficient encryption system of (Johnson et al., 1996). Fewer encryption operations are necessary, therefore the system power consumption is lower. However, the performance bonus must be traded off against the costs associated with using this patented method.

3.6 Applications

CTRT-CTR encryption would clearly be beneficial in applications where fast on-the-fly AON encryption is required, such as secure mobile telecommunications networks or MANETs (Mobile Ad-Hoc NETworks). MANETs are mainly used in highly mobile and hostile environments, where data confidentiality is important. Typical examples of nodes in a MANET include units where processing power and battery life are limited, such as backpack radios, handheld devices, and vehicle computers (Berman, 2005). Constrained environments such as these are well suited to secure, efficient all-or-nothing encryption schemes such as CTRT-CTR.

CTRT-CTR is also especially useful in protocols where the existing encryption mechanism is a block cipher (e.g. AES), as the encryption software/hardware can be re-used to perform the AONT, deeming implementation of additional algorithms unnecessary.

4 PROOFS OF SECURITY

The provable security of the proposed CTRT-CTR scheme is now considered. We also analyse how the security changes when we incorporate CTRT-CTR and Johnson et al.'s efficient encryption technique. For convenience, the definitions and notation used in our theorems rely heavily on those used by (Desai, 2000). In particular, we employ the framework used in Theorem 3 and Lemma 14 of that work, and extend those results.

4.1 Existing Security Results

Desai proved his results in the Shannon model ('ideal model') of a block cipher (Shannon, 1949). In essence, the Shannon model states that each new key to the block cipher defines an independent random permutation $F_k(\cdot)$. Desai defined a notion of security for AONTs, based on indistinguishability from a random string. The experiment (or 'game') used to capture this notion is as follows:

AONT Let $\Pi' = (\mathcal{E}', \mathcal{D}')$ be an AONT of block length l. For adversary A and bit b = 0 or 1 define $\mathrm{Exp}^{\mathrm{aon}}_{\Pi'}(A, b)$ as:

```
$(x, s) \leftarrow A(\text{find});$
$y_0 \leftarrow \mathcal{E}'(x);$          \\ All-or-nothing Transform
$y_1 \leftarrow \{0,1\}^{|y_0|};$        \\ Random $|y_0|$-bit string
$d \leftarrow A^{\mathcal{Y}}(\text{guess}, s);$
return $d$.
```

The advantage function of Π' is defined as:

$$\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(t, m) = \max_A\{\Pr[\mathrm{Exp}^{\mathrm{aon}}_{\Pi'}(A,0) = 0] - \Pr[\mathrm{Exp}^{\mathrm{aon}}_{\Pi'}(A,1) = 0] \quad \} \tag{17}$$

where A runs with time complexity t, and $|y_0| = |y_1| = ml$ (i.e. m l-bit blocks). The notation $A^{\mathcal{Y}}$ means that A has access to an oracle $\mathcal{Y}$, taking an index $j \in \{1, \ldots, m\}$ and returning $y_b[j]$, where $y_b = y_b[1], \ldots, y_b[m]$. At most m queries are allowed to $\mathcal{Y}$ during the guess stage. In addition, s denotes state information generated by the results of A's queries.

Based on any $m - 1$ blocks of the challenge, adversary A has to decide whether the challenge is a real AONT output, or a randomly chosen string. Equation (17) measures A's maximum probability of success in this game, which should be negligible.

Indistinguishability of encryptions (Bellare et al., 1997) is an important security notion for block cipher encryption schemes, described below:

IND Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption mode. For adversary A and bit b = 0 or 1 define $\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A, b)$ as:

```
$F \leftarrow \mathrm{BC}(k, l);$          \\ Choose block cipher at random
$a \leftarrow \mathcal{K}(1^k);$           \\ Choose key of length k
$(x_0, x_1, s) \leftarrow A^{F, F^{-1}, \mathcal{E}^{F_a, F_a^{-1}}}(\text{find});$
$y \leftarrow \mathcal{E}^{F_a, F_a^{-1}}(x_b);$    \\ Apply encryption mode
$d \leftarrow A^{F, F^{-1}, \mathcal{E}^{F_a, F_a^{-1}}}(\text{guess}, y, s);$
return $d$.
```

The advantage function of Π is defined as:

$$\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(t, m, p, q, \mu) = \max_A\{\Pr[\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A,1) = 1] - \Pr[\mathrm{Exp}^{\mathrm{ind}}_{\Pi}(A,0) = 1] \quad \} \tag{18}$$

where A runs with time complexity t, p is the maximum number of queries allowed to F/F^{-1}, $|y| = ml$, and q is the maximum number of queries allowed to $\mathcal{E}^{F_a, F_a^{-1}}$, these totalling at most μ bits.

In this game, A is given a ciphertext, and must determine which of two plaintexts it corresponds to. If the encryption mode is secure in the **IND** sense, A's success probability should be negligible. Theorem 3 of (Desai, 2000) quantifies this success probability when the encryption mode is an all-or-nothing ECB mode. He proved that if $n \geq 2$:

$$\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(t, m, p, q, \mu) \leq 2m\,\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(t', m) + \frac{2mp}{2^k} + \frac{2m}{2^l} \tag{19}$$

where n is the number of l-bit plaintext blocks, $t' = t + (\frac{\mu}{l} + m - 1).T + \mathcal{O}(ml + pl + \mu)$, and T is the time taken to decode a ml-bit string using $\mathcal{D}'$.

In the specific case where the AONT is CTRT, Desai also proved that for $k \leq l$ and $m + p \leq 2^{k-1}$, then:

$$\mathrm{Adv}^{\mathrm{aon}}_{\mathrm{CTRT}}(t, m, p) \leq \frac{m^2 + 8p}{2^k} \tag{20}$$

where k is the key length of the underlying block cipher, and t, m and p are as defined above.

4.2 Security of AONT-CTR Encryption

We extend the result of Equation (19) to evaluate the **IND**-security of the CTR mode of all-or-nothing encryption. Note that all m pseudo-message blocks are encrypted using CTR mode in this case. The security of the 'efficient AON encryption' scheme of

Section 3.5 is considered in Section 4.3.

Theorem 1 Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an all-or-nothing CTR mode in the Shannon model, using AONT $\Pi' = (\mathcal{E}', \mathcal{D}')$. Then for $n \geq 2$:

$$\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(t, m, p, q, \mu) \leq 2m\, \mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(t', m) + \frac{2p}{2^k} + \frac{2m}{2^l} \tag{21}$$

where $t' = t + (\frac{\mu}{l} + m - 1).T + \mathcal{O}(ml + pl + \mu)$ and all other variables are as defined in Section 4.1.

Proof A new game **COLL** is defined as follows:

COLL Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an AONT-CTR mode. For adversary A define $\mathrm{Exp}^{\mathrm{coll}}_{\Pi}(A)$ as:

```
01  $a \leftarrow \mathcal{K}(1^k)$;          \\ Choose key of length k
02  $(x, s, \mathrm{Elist}_F) \leftarrow A^{F, F^{-1}, \mathcal{E}^{F_a, F_a^{-1}}}(\mathrm{find})$;
03  $y \leftarrow \mathcal{E}^{F_a, F_a^{-1}}(x)$;     \\ Apply AONT-CTR
04  $\mathrm{Elist}_G \leftarrow A^{F, F^{-1}, \mathcal{E}^{F_a, F_a^{-1}}}(\mathrm{guess}, y, s)$;
05  for $i, j \in [m]$ {
06    if$(i \neq j \wedge y[i] = y[j]) \vee (y[i] \in \mathrm{Elist}_F \cup \mathrm{Elist}_G)$
07      then $d \leftarrow 1$, else $d \leftarrow 0$; }
08  return $d$.
```

The success function of Π is defined as:

$$\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(t, m, p, q, \mu) = \max_{A}\{\Pr[\mathrm{Exp}^{\mathrm{coll}}_{\Pi}(A) = 1]\} \tag{22}$$

where A runs with time complexity t, $[m]$ denotes the set $\{1, \ldots, m\}$, and all other variables are as defined in Section 4.1. $\wedge$ and $\vee$ denote logical AND and OR, respectively. Elist_F and Elist_G ('Encryption lists') contain the answers to A's $\mathcal{E}^{F_a, F_a^{-1}}$ queries during the find and guess stages.

This **COLL** experiment measures the probability of collision between ciphertext blocks, across queries. We proceed to relate $\mathrm{Succ}^{\mathrm{coll}}_{\Pi}$ to $\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}$ by a contradiction argument. Later, we will relate $\mathrm{Succ}^{\mathrm{coll}}_{\Pi}$ to $\mathrm{Adv}^{\mathrm{ind}}_{\Pi}$, and arrive at the theorem statement by substitution.

Consider an **AONT** adversary A, built using a **COLL** adversary B. Because A has no oracles other than $\mathcal{Y}$, it must simulate B's oracles. The $A(\mathrm{find})$ algorithm (i.e. **AONT** line 01) becomes:

$\mathbf{A}(\mathbf{find})$:

```
01  $a \leftarrow \mathcal{K}(1^k)$;     \\ Choose key of length k
02  $Tlist \leftarrow \{\}$;     \\ Initialise Tlist
03  $(x, s) \leftarrow B(\mathrm{find})$;
04  $s' \leftarrow (s, x, a, \mathrm{Tlist})$;
05  return $(x, s')$.
```

where Tlist is similar to B's Elist, except that it holds the pseudo-message blocks returned from queries to $\mathcal{E}^{F_a, F_a^{-1}}$, and the corresponding block indices. $A^{\mathcal{Y}}(\mathrm{guess}, s)$ (i.e. **AONT** line 04) becomes:

$\mathbf{A}^{\mathcal{Y}}(\mathrm{guess}, (\mathbf{s}, \mathbf{x}, \mathbf{a}, \mathbf{Tlist}))$:

```
01  $j \leftarrow [m]$;          \\ Choose index j at random
02  for $i \in [m] \wedge i \neq j$ {
03    $y[i] \leftarrow \mathcal{Y}(i)$;     \\ Retrieve challenge block i
04    $\mathrm{Tlist} \leftarrow \mathrm{Tlist} \cup \{(y[i], i)\}$;
05    $z[i] \leftarrow y[i] \oplus F_a(ctr + i)$; }
06  for $i \in [|\,\mathrm{Tlist}\,|]$ {
07    $(block, index) \leftarrow \mathrm{Tlist}[i]$;
08    $\mathrm{Elist}[i] \leftarrow block \oplus F_a(ctr + index)$; }
09  $z[j] \leftarrow \{0,1\}^l \setminus \mathrm{Elist}$;
10  $\mathrm{Tlist} \leftarrow \mathrm{Tlist} \cup B(\mathrm{guess}, z, s)$;
11  for $i \in [|\,\mathrm{Tlist}\,|]$ {
12    $y[j] \leftarrow \mathrm{Tlist}[i]$;
13    $x_i \leftarrow \mathcal{D}'(y)$;     \\ Inverse AONT
14    if $(x_i == x)\{d \leftarrow 0$; return $d\}\}$
15  $d \leftarrow 1$; return $d$.
```

In lines 01 to 05, $A^{\mathcal{Y}}(\mathrm{guess})$ forms all but one block of B's challenge ciphertext z, based on the $m - 1$ blocks of its own challenge y that it receives from $\mathcal{Y}$. For the missing block $z[j]$, A must choose a random string, but has to ensure that B has not seen this block already in its find stage, otherwise a collision would be guaranteed. This requirement is met by lines 06 to 09. Finally in lines 11 to 15, A searches incrementally through Tlist, inserting a block from Tlist in place of its missing $y[i]$ block, and inverting. If the result of the transform inversion equals the original plaintext message x, then A concludes that the challenge y was real (as opposed to random).

We now calculate $\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(A)$. For simplicity, we use $\Pr_b[A = 0]$ to denote $\Pr[\mathrm{Exp}^{\mathrm{aon}}_{\Pi'}(A, b) = 0]$.

$$\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(A) = \Pr_0[A = 0] - \Pr_1[A = 0] \tag{23}$$

Define the event **C** to be the event that the missing pseudo-message block $y[j]$ is on the Tlist. Expanding the first term in Equation (23):

$$\Pr_0[A = 0] = \Pr_0[A = 0|\mathbf{C}].\Pr_0[\mathbf{C}] + \Pr_0[A = 0|\bar{\mathbf{C}}].\Pr_0[\bar{\mathbf{C}}] \tag{24}$$

From the description of $A^{\mathcal{Y}}(\mathrm{guess})$, $\Pr_0[A = 0|\mathbf{C}]$ is clearly equal to unity. This implies:

$$\Pr_0[A = 0] \geq 1.\Pr_0[\mathbf{C}] \tag{25}$$

Intuitively, one would expect $\Pr_0[\mathbf{C}]$ to be lower bounded by $\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B)$ via:

$$\Pr_0[\mathbf{C}] \geq \tfrac{1}{m}\, \mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B)$$

since $\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B)$ gives the probability of collision with *one* of the blocks of y, and **C** implies collision with $y[j]$. However, we must also take into account that A's challenge to B is incorrectly formed, as block $z[j]$ is randomly chosen. There is a chance that B, through its queries to F/F^{-1}, will realise this, causing it to abort the game. It is in evaluating this probability that our proof differs from that of Desai.

We now examine how B could discover the anomaly. Consider the case where B queries F with a block of its choice, key k_1 and index j, and receives the block $z[j]$ in return. B can then test key k_1 on some known plaintext/ciphertext, but since $k_1 \neq a$, the known plaintext encrypted under k_1 will not equal the known ciphertext. B will realise that $z[j]$ was not the result of an encryption with key a, and abort the game.

The number of different block cipher permutations in CTR mode is $m.2^k$. For each permutation, there exists some string that will encrypt to give $z[j]$, which would cause B to abort. Therefore, the maximum probability of B choosing one such query (and choosing index = j) is $\frac{p}{m.2^k}$ (recall p is the maximum number of queries allowed to F/F^{-1}). Taking this into account, $\mathrm{Pr}_0[\mathbf{C}]$ is lower bounded by:

$$\mathrm{Pr}_0[\mathbf{C}] \geq \tfrac{1}{m}\,\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B) - \tfrac{p}{m.2^k} \tag{26}$$

Combining Equations (25) and (26):

$$\mathrm{Pr}_0[A=0] \geq \tfrac{1}{m}\,\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B) - \tfrac{p}{m.2^k} \tag{27}$$

The next step is to upper bound the second term in Equation (23), $\mathrm{Pr}_1[A=0]$. This is the ‘false probability’ of success, i.e. the probability that A will return $d=0$ when the challenge y was, in fact, random. From the code for $A(\mathrm{guess})$, it is clear that this can only occur if there is (at least) one other AONT output y_1 which also decodes to x. The probability of this event is independent of the encryption mode in which the AONT mode is used, therefore we can reuse Desai’s result:

$$\mathrm{Pr}_1[A=0] \leq \tfrac{1}{2^l} \tag{28}$$

for $n \geq 2$. Substituting Equations (27) and (28) into Equation (23), and rearranging, gives:

$$\begin{aligned}\mathrm{Adv}^{\mathrm{coll}}_{\Pi}(t,m,p,q,\mu) \leq m\,\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(t',m)+\\ p/2^k + m/2^l\end{aligned} \tag{29}$$

where $t' = t + (\frac{\mu}{l} + m - 1).T + \mathcal{O}(ml + pl + \mu)$, which is the complexity of running A.

The penultimate step is to consider an adversary A in the IND sense, and relate $\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(A)$ to $\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(A)$. Due to space restrictions, we do not present the argument here, but note that it is the same as that of (Desai, 2000), as it is mode-independent. We state:

$$\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(A) \leq 2.\,\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(A) \tag{30}$$

Finally, combining Equations (29) and (30) gives Equation (21), the theorem result. □

Comparing the result of Theorem 1 with Equation (19), it can be seen that AONT-ECB mode and AONT-CTR mode achieve similar levels of security, in the indistinguishability of encryptions sense. The adversarial advantage is, in fact, smaller in AONT-CTR mode, due to the larger set of permutations opened up by the inclusion of the block index in the encryption operation. Therefore, AONT-CTR is more secure than AONT-ECB, although neither is *in*secure.

4.3 Security of Efficient AONT-CTR Encryption

We now consider the scenario where an AONT is applied to the plaintext, and r out of the m pseudo-message blocks are encrypted. Intuitively, one would assume that this scheme would somehow be less secure than if all the blocks were encrypted, but we show that this is not the case.

Theorem 2 Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an efficient all-or-nothing CTR mode in the Shannon model, using AONT $\Pi' = (\mathcal{E}', \mathcal{D}')$, and encrypting r blocks of the pseudo-message. Then, for $n \geq 2$:

$$\begin{aligned}\mathrm{Adv}^{\mathrm{ind}}_{\Pi}(t,m,p,q,\mu) \leq 2m\,\mathrm{Adv}^{\mathrm{aon}}_{\Pi'}(t',m)+\\ (\tfrac{r}{m})\tfrac{2p}{2^k} + \tfrac{2m}{2^l}\end{aligned} \tag{31}$$

where $t' = t + (\frac{\mu}{l} + m - 1).T + \mathcal{O}(ml + pl + \mu)$ and all other variables are as defined in Section 4.1. A publically known set $\mathcal{S}$ holds the indices of the pseudo-message blocks that are encrypted.

Proof We use framework of Theorem 1, with the algorithms suitably modified. Specifically, lines `05` and `08` of $A(\mathrm{guess})$ must be modified to:

```
04     ...
05a    if $i \in \mathcal{S}$
05b        $z[i] \leftarrow y[i] \oplus F_a(ctr + i)$;
05c    else
05d        $z[i] \leftarrow y[i]$;
06     ...
07     ...
08a    if $index \in \mathcal{S}$
08b        $\mathrm{Elist}[i] \leftarrow block \oplus F_a(ctr + index)$;
08c    else
08d        $\mathrm{Elist}[i] \leftarrow block$;
09     ...
```

The proof continues in a similar manner, but $\mathrm{Pr}_0[\mathbf{C}]$ must be re-evaluated, where **C** is the event that the missing pseudo-message block $y[j]$ is on B’s Tlist. We argue that:

$$\mathrm{Pr}_0[\mathbf{C}] \geq \tfrac{1}{m}\,\mathrm{Succ}^{\mathrm{coll}}_{\Pi}(B) - (\tfrac{r}{m})\tfrac{p}{m.2^k} \tag{32}$$

Recall that the $p/(m.2^k)$ term arose in Theorem 1 as the probability that B would realise that block $z[j]$ had been fabricated. B can only come to this conclusion if $z[j]$ is one of the encrypted blocks, i.e. if $j \in \mathcal{S}$. If $j \notin \mathcal{S}$, B cannot perform the same check, as $z[j]$ is an AONT output block, and is not related to the other challenge blocks by the hidden key a. Therefore, the probability of $z[j] \in \mathcal{S}$ and B aborting is $(r/m)(p/m.2^k)$, since index j is chosen at random.

Combining Equations (23), (25), (32), (28) and (30) gives Equation (31), the theorem result. □

Note that when $r = m$, Equation (31) reduces to the result of Theorem 1, as expected. As $r \rightarrow 1$, the upper bound on the adversary's **IND** advantage gets smaller. This seems counter-intuitive, as encrypting fewer blocks would suggest a less secure scheme. However, we note that the r/m term in the theorem results from algorithm B realising that the challenge block $z[j]$ is incorrectly formed, and that this realisation should indeed be less likely when there are fewer encrypted blocks against which to compare. As $r \rightarrow 1$, the scheme's security becomes closer to the security of the AONT.

A similar encryption scheme was considered in (Bellare and Boldyreva, 2000), whereby an AONT is applied to a message, and the first pseudo-message block is encrypted via 'chaffing and winnowing'. The authors proved that this scheme is semantically secure, if the underlying cipher is semantically secure. This paper provides an different proof, where we work in the Shannon model of the block cipher, encrypt using CTR-mode, and allow the number of encrypted blocks r to vary.

5 CONCLUSIONS

In this paper, we proposed a new mode of all-or-nothing encryption, called CTRT-CTR. In doing so, we answered an open problem from the literature regarding the speed of all-or-nothing encryption. We proposed using CTRT-CTR in an efficient AON encryption mode, to further reduce power and memory overheads. Trade-offs between on-line encryption speed and memory were identified. The scheme would be beneficial in applications such as MANETs where low-power, secure run-time encryption is required. The proposed schemes were proven secure in the Shannon model of a block cipher.

Future work will investigate if it is possible to achieve secure all-or-nothing encryption with a lower total workload (both on-line and off-line) than CTRT-CTR, whilst still maintaining low latency.

ACKNOWLEDGEMENTS

The authors would like to thank Emanuel Popovici for his helpful discussions regarding the proofs of security. This research was supported by the Embark Initiative, operated by the Irish Research Council for Science, Engineering and Technology (IRCSET).

REFERENCES

Bellare, M. and Boldyreva, A. (2000). The Security of Chaffing and Winnowing. In *ASIACRYPT'00*, volume 1976 of *Lecture Notes in Computer Science*, pages 517–530. Springer.

Bellare, M., Desai, A., Jokipii, E., and Rogaway, P. (1997). A Concrete Security Treatment of Symmetric Encryption. In *FOCS'97*, pages 394–403.

Berman, V. (2005). Enhancing Data Security in Mobile Ad Hoc Networks via Multipath Routing and Directional Transmission. Master's thesis, University of California, Davis.

Boyko, V. (2000). *On All-or-Nothing Transforms and Password Authenticated Key Exchange Protocols*. PhD thesis, Massachusetts Institute of Technology.

Byers, J., Considine, J., Itkis, G., Cheng, M. C., and Yeung, A. (2006). Securing bulk content almost for free. *Journal of Computer Communications, Special Issue on Internet Security*, 29:290–290.

Desai, A. (2000). The Security of All-or-Nothing Encryption (Extended Abstract). In *CRYPTO'00*, volume 1880 of *Lecture Notes in Computer Science*, pages 359-375. Springer. Full version retrieved online, June 2006. `http://www.cs.ucsd.edu/users/adesai/`.

Dodis, Y. (2000). *Exposure-Resilient Cryptography*. PhD thesis, Massachusetts Institute of Technology.

Dodis, Y., Sahai, A., and Smith, A. (2001). On Perfect and Adaptive Security in Exposure-Resilient Cryptography. In *EUROCRYPT'01*, volume 2045 of *Lecture Notes in Computer Science*, pages 301–324. Springer.

ECRYPT (2006). ECRYPT Yearly Report on Algorithms and Keysizes (2005). http://www.ecrypt.eu.org.

Johnson, D., Matyas, S., and Peyravian, M. (1996). Encryption of Long Blocks Using a Short-Block Encryption Procedure. Submitted for inclusion in the IEEE P1363a standard.

Kiong, N. C. and Samsudin, A. (2003). A Concrete Security Treatment of Symmetric Encryption. In *APCC'03*, volume 2, pages 838–843.

Lipmaa, H., Rogaway, P., and Wagner, D. (2000). CTR-Mode Encryption. Comments to NIST concerning AES Modes of Operation.

Marnas, S. I., Angelis, L., and Bleris, G. L. (2003). All-Or-Nothing Transforms Using Quasigroups. In *Proc. of 1st Balkan Conference on Informatics*, pages 183–191.

Peterson, Z. N. J., Burns, R. C., Herring, J., Stubblefield, A., and Rubin, A. D. (2005). Secure Deletion for a Versioning File System. In *FAST'05*. USENIX.

Rivest, R. (1997). All-or-Nothing Encryption and The Package Transform. In *FSE'97, 4th International Workshop on Fast Software Encryption*. Springer.

Shannon, C. E. (1949). Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656–715.

Zhang, R., Hanaoka, G., and Imai, H. (2004a). On the Security of Cryptosystems with All-or-Nothing Transform. In *ACNS'04*, volume 3089 of *Lecture Notes in Computer Science*, pages 76–90. Springer.

Zhang, R., Hanaoka, G., Shikata, J., and Imai, H. (2004b). On the Security of Multiple Encryption or CCA-security+CCA-security=CCA-security? In *PKC '04*, volume 2947 of *Lecture Notes in Computer Science*, pages 360–374. Springer.