A One-Pass Authenticated-Encryption Mode with No Blockcipher Inverse

No Author Given

No Institute Given

Abstract. We propose OCFB (Offset Cipher FeedBack) mode of operation for nonce-based authenticated encryption. OCFB is one-pass and rate-1, requiring only one blockcipher call per message block, like the OCB mode of operation developed by Rogaway *et al.* in ACM CCS 2001. Unlike OCB, OCFB is inherently sequential but utilizes only the forward encryption of the underlying blockcipher. There are a number of advantages arising from these features of OCFB. First, the decipherment process of OCFB is still fully parallelizable. Second, the code size of OCFB becomes small, especially with non-involutary blockciphers such as AES. Third, the security of OCFB can be proven under the weaker assumption of PRP (Pseudo-Random Permutation) about the blockcipher, rather than SPRP (Strong PRP). Finally, the design of OCFB is relatively simple, tweaking the blockcipher only with the input whitening operation and encrypting the last (incomplete) message block in a consistent way. Moreover, OCFB provides us with an effective, single-key solution for AEAD (Authenticated Encryption with Associated Data). Indeed, OCFB, when used for AEAD, requires a fewer blockcipher calls than the OCB+PMAC construction proposed by Rogaway in ACM CCS 2002.

Keywords: CFB, nonce, tweakable cipher, checksum MAC, AEAD.

1 Introduction

There exist three kinds of blockcipher modes of operation: Encryption, MAC (Message Authentication Code) and AE (Authenticated Encryption). Encryption modes provide privacy of data, MAC modes ensure integrity, and AE modes do both at the same time. The last type is the focus of this paper; we propose a new AE construction.

An AE mode can be realized via a generic composition of an encryption mode and a MAC scheme, as in, for example, Encrypt-then-MAC [1]. Unfortunately, such a simple combination would result in a two-key construction, generally using two independent blockcipher keys. Also, such a generic composition remains necessarily slow, requiring the time to run its encryption mode plus the time to run its MAC scheme.

Blockcipher-based AE modes can be made more efficient (and one-key) via dedicated designing. There are various methods for constructing faster (integrated) AE modes. One method is to use other primitives in combination with blockciphers. The GCM [18, 22] mode, for example, utilizes universal hashing for its authentication part, which can, depending on the environments, provide much higher performance than blockcipher-based MACs such as CMAC [11, 21], RMAC [13] and PMAC [4, 24]. Another method is to use a "partial" blockcipher. For example, ASC-1 [12] iterates a 4-round AES [19] for masking a plaintext and for creating a MAC tag, where the mode's security proof is based on a rather strong assumption about its key-schedule component.

In this paper we focus on the designs using just the whole blockcipher as the main underlying primitive. We make this choice because AE modes whose design is based entirely on the iteration of a blockcipher yield compact implementation and sound security proofs. That is, the code size of such a mode becomes only a bit larger than that of the blockcipher, and we can provide security proofs for such a construction under the sole (pseudo-randomness) assumption about the cipher. Examples include RPC [15], CCFB [17], XCBC [9], IAPM [14] and OCB [26, 24, 16]. Minor (non-blockcipher) operations used by these modes are the tweak increment, which is usually realized by a counter or by whitening update using LFSR or Gray code.

To gain maximum benefits, it would be better for blockcipher modes to make use of only the forward encryption of the cipher. Such modes yield even smaller codes, especially with a non-involutary blockcipher like AES [8, 20], and the security of the construction can be proven under the weaker assumption of PRP (Pseudo-Random Permutation) about the cipher rather than SPRP (Strong PRP). The great need for "small" blockcipher usage exists, becoming increasingly evident from a boom in "lightweight" algorithms, *e.g.*, PRESENT [6], KATAN [7] and LED [10], just a few being mentioned. Also, it would be safer to rely on the PRP assumption rather than on the SPRP, as indicated by the recent attacks on the full AES [3, 2, 5], many of which exploit ciphertext queries to the decryption oracle.

The purpose of the current work is to devise a one-pass, rate-1 (meaning one online blockcipher call to encrypt and authenticate one message block) AE mode that utilizes only the forward encryption of the underlying cipher. No previous construction achieves this goal—OCB is one-pass and rate-1 but uses the inverse cipher for its decipherment process; CCFB operates without the inverse cipher but is not rate-1.

Our Contributions. We start with the design of OCB. OCB can be considered as a combination of a tweaked ECB (Electric Code Book) encryption mode and a checksum MAC. We adopt CFB (Cipher FeedBack) instead of ECB to construct our OCFB (Offset CFB) mode of operation. As a result, in a way OCFB looks like a whitening version of CCFB.

Table 1 summarizes the main features of OCFB. OCFB becomes inherently sequential and certainly loses parallelizability in the encipherment process. However, besides being rate-1 and free of the inverse cipher, the OCFB mode enjoys the following additional, appealing features:

1. The decipherment process of OCFB remains fully parallelizable.

Table 1. Comparison of OCFB with OCB and with CCFB

scheme	reference	parallel	rate-1	inverse-free	assumption
OCB	[26, 24, 16]	Enc and dec	✓		SPRP
CCFB	[17]	Dec only		✓	PRP
OCFB	This paper	Dec only	✓	✓	PRP

Table 2. Efficiency for a-block header and m-block message

scheme	reference	# blockcipher calls	
		online	offline
OCB+PMAC	[23, 24, 16]	$a + m + 2$	1
OCFB+	This paper	$a + m + 1$	0

2. The design of OCFB becomes relatively simple, tweaking the blockcipher only with the input whitening operation and encrypting the last (possibly incomplete) message block in a consistent way.
3. The security proof of OCFB also becomes fairly simple and easy.

Moreover, OCFB provides us with an effective, single-key solution for AEAD (Authenticated Encryption with Associated Data). Indeed, OCFB+, the OCFB mode for AEAD, requires a fewer blockcipher calls than the OCB+PMAC construction [23]. See Table 2.

Roughly speaking, we can consider OCFB+ as being based on the technique called "nonce stealing" [23] and processing the data $A \| M$, where A denotes a header and M a message. OCFB+ becomes fully compatible with OCFB by putting the header A to be null. OCFB+ is fairly flexible, except that the header A must be ready before the message M gets encrypted. This should be acceptable in most situations except when A contains some information dependent on the ciphertext C (*i.e.*, the encrypted M), for example the CRC checksum of C.

The extra online blockcipher call in OCB+PMAC lets us treat such special cases, based on the technique called "ciphertext translation" [23]; OCB+PMAC can handle A and M independently, resolving the above problem. Also, the extra offline blockcipher call in OCB+PMAC allows us to reuse a header tag, speeding up the process when the same header A is repeated.

We could have equipped OCFB with these features at the cost of additional blockcipher calls. However, we rather minimize the number of blockcipher calls, taking into consideration its usage by resource-constrained devices.

Organization of the Paper. In the following section we begin by defining notation necessary for describing OCFB and security notions necessary for security proofs. In Sect. 3 we describe OCFB (without associated data) and prove its security. Section 4 does the same for OCFB+, the AEAD version of OCFB. We conclude the paper in Sect. 5.

2 Preliminaries

In this section we define our notation and security notions. We first describe nonce-based AE schemes and an adversarial model to define the security goal to be achieved. We then do the same for blockciphers. Lastly, we describe a method for constructing tweakable ciphers from ordinary blockciphers using finite-field operations.

2.1 Nonce-Based AE Schemes

The formalization given here is based on the previous work [23, 25]. A nonce-based AE scheme is a pair $(\mathcal{E}, \mathcal{E}^{-1})$ of algorithms. The encipherment algorithm $\mathcal{E}$ takes as its input a key $K \in \mathcal{K}$, a nonce $N \in \mathcal{N}$, a header $A \in \{0,1\}^*$ and a message $M \in \{0,1\}^*$ and outputs $(C, T) \leftarrow \mathcal{E}_K(N, A, M)$, where C is the ciphertext and $T \in \{0,1\}^\tau$ is the authentication tag. The decipherment algorithm $\mathcal{E}^{-1}$ takes as its input a key K, a nonce N, a header A, a ciphertext C and a tag T and outputs either the message $M \leftarrow \mathcal{E}_K^{-1}(N, A, C, T)$ or the invalid symbol $\perp \leftarrow \mathcal{E}_K^{-1}(N, A, C, T)$. Whenever we compute $(C, T) \leftarrow \mathcal{E}_K(N, A, M)$, we always obtain the original message as $M \leftarrow \mathcal{E}_K^{-1}(N, A, C, T)$. For simplicity we require $|M| = |C|$.

An adversary is an oracle machine $\mathcal{A}$ that outputs a bit. We write $\mathcal{A}^{\mathcal{O}(\cdots)}$ to denote the bit value output by $\mathcal{A}$ after its interaction with the oracle $\mathcal{O}(\cdots)$. Define an advantage function as

$$\mathrm{Adv}_{\mathcal{E}}^{\mathrm{nae}}(\mathcal{A}) := \left[\mathcal{A}^{\mathcal{E}_K(\cdots), \mathcal{E}_K^{-1}(\cdots)} = 1 \;\middle|\; K \xleftarrow{\$} \mathcal{K}\right] - \left[\mathcal{A}^{\$(\cdots), \perp(\cdots)} = 1\right],$$

where $K \xleftarrow{\$} \mathcal{K}$ is a uniform random sampling of K, $\$(\cdots)$ is a random oracle that returns a random string of $|M| + \tau$ bits and $\perp(\cdots)$ is a reject oracle that always returns the invalid symbol $\perp$. We demand that A never ask the same nonce value N twice to its encipherment oracle, repeat its queries or make trivial queries.

2.2 Blockciphers

Our underlying primitive is an n-bit blockcipher $E_K : \{0,1\}^n \to \{0,1\}^n$. Since OCFB and OCFB+ are single-key constructions, we use the same key space $K \in \mathcal{K}$ as above. For each key value $K \in \mathcal{K}$ the function E_K is a permutation over the set $\{0,1\}^n$. We let $\mathrm{Perm}(n)$ denote the set of all permutations over $\{0,1\}^n$. For simplicity we set the nonce space $\mathcal{N} = \{0,1\}^n$. We assume $\tau \leq n$.

For an adversary $\mathcal{A}$, define an advantage function as

$$\mathrm{Adv}_{E}^{\mathrm{prp}}(\mathcal{A}) := \left[\mathcal{A}^{E_K(\cdot)} = 1 \;\middle|\; K \xleftarrow{\$} \mathcal{K}\right] - \left[\mathcal{A}^{P(\cdot)} = 1 \;\middle|\; P \xleftarrow{\$} \mathrm{Perm}(n)\right],$$

where P is a random permutation over $\{0,1\}^n$. We also define $\mathrm{Adv}_E^{\mathrm{prp}}(t,q) := \max_{\mathcal{A}} \mathrm{Adv}_E^{\mathrm{prp}}(\mathcal{A})$, where max runs over all adversaries $\mathcal{A}$ whose running time is at most t, making at most q queries to its oracle. To measure the time complexity of $\mathcal{A}$ we fix a model of computation and a method of encoding. We write $\mathrm{Time}(E)$ for the time necessary to perform one computation of E_K.

Similarly, we define $\mathrm{Adv}_{\mathcal{E}}^{\mathrm{nae}}(t,q,\sigma) := \max_{\mathcal{A}} \mathrm{Adv}_{\mathcal{E}}^{\mathrm{nae}}(\mathcal{A})$, where max runs over all adversaries $\mathcal{A}$ whose running time is at most t, making at most q queries to its oracles, the total query complexity being at most σ blocks (one block is n bits).

2.3 Tweakable Ciphers Using Finite-Field Operations

We follow [24] for the framework of tweakable ciphers based on multiplication by '2' and '3' in the finite field. Recall that the set $\{0,1\}^n$ of bit strings can be considered as a set of integers $[0, 2^n-1] = \{0,1,\ldots,2^n-1\}$ through the mapping $a_{n-1}\cdots a_1a_0 \in \{0,1\}^n \leftrightarrow \sum_{i=0}^{n-1} a_i 2^i \in [0, 2^n-1]$ and also as the finite field $GF(2^n)$ having 2^n elements through the correspondence $a_{n-1}\cdots a_1a_0 \leftrightarrow a_{n-1}x^{n-1}+\cdots+a_1x+a_0 \in GF(2)[x]$, where x is a formal variable. To represent the field, we fix a primitive polynomial such as $f(x) = x^{64}+x^4+x^3+x+1$ for $n = 64$ and $f(x) = x^{128}+x^7+x^2+x+1$ for $n = 128$. With these primitive polynomials, the discrete logarithm of $3 = x+1$ in base $2 = x$ becomes "huge" [24], which implies that for the range $i \in \{0,1,\ldots,\ldots,2^{n/2}\}$ and $j \in \{0,1,2\}$, the set

$$\left\{2^i 3^j W\right\}_{i,j}$$

gives us a system of unique representatives from the set $\{0,1\}^n$. Here, W is any non-zero value in $\{0,1\}^n$, and of course multiplications are performed in the finite field.

We generate values W as $W \leftarrow E_K(N)$ with $N \in \mathcal{N}$. Given i,j,N, we define $E_K^{i,j,N}(X) := E_K(2^i 3^j W \oplus X)$ where $W \leftarrow E_K(N)$. To access the oracle $E_K^{i,j,N}(\cdot)$, an adversary $\mathcal{A}$ specifies the index (i,j,N) and makes a query X.

Now let us define the following advantage function:

$$\begin{aligned}\mathrm{Adv}_E^{\mathrm{twk}}(\mathcal{A}) := \big[\mathcal{A}^{\ldots,E_K^{i,j,N}(\cdot),\ldots} = 1 \,\big|\, K \xleftarrow{\$} \mathcal{K}\big]& \\ - \big[\mathcal{A}^{\ldots,F^{i,j,N}(\cdot),\ldots} = 1 \,\big|\, F^{i,j,N} \xleftarrow{\$} \mathrm{Func}(n)\big]&,\end{aligned}$$

where $\mathrm{Func}(n)$ is the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$ and $F^{i,j,N}$ is an independent random function $F^{i,j,N} \xleftarrow{\$} \mathrm{Func}(n)$. The quantity $\mathrm{Adv}_E^{\mathrm{twk}}(t,q)$ is also defined in the natural way. The following lemma is our basic tool:

Lemma 1. *We have*

$$\mathrm{Adv}_E^{\mathrm{twk}}(t,q) \le \mathrm{Adv}_E^{\mathrm{prp}}(t',2q) + \frac{3q^2}{2^n},$$

where t' is about $t + 2q\,\mathrm{Time}(E)$.

Proof. A proof can be found in [24]. □

```
Input: a nonce $N \in \{0,1\}^n$, a message $M \in \{0,1\}^*$
Output: a ciphertext $C \in \{0,1\}^*$, a tag $T \in \{0,1\}^\tau$
$W \leftarrow E_K(N)$; $\bigl(M[1], M[2], \ldots, M[m]\bigr) \leftarrow M$
$C[0] \leftarrow 10^{n-1}$                              // $10^{n-1} = 100\cdots0 \in \{0,1\}^n$
for $i = 1$ to $m$ do
    $X[i] \leftarrow C[i-1] \oplus 2^{i+1}W$          // Multiplication in the finite field
    $Y[i] \leftarrow E_K\bigl(X[i]\bigr)$; $C[i] \leftarrow M[i] \oplus \mathrm{trunc}\bigl(Y[i], |M[i]|\bigr)$
end
$C \leftarrow \bigl(C[1], C[2], \ldots, C[m]\bigr)$
if $|M[m]| = n$ then                         // $|M|$ is a positive multiple of $n$
    $\varSigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus M[m]$; $U \leftarrow 2^{m+1}3W$
else
    $\varSigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus (M[m] \| 10^*)$; $U \leftarrow 2^{m+1}3^2W$
end
$\tilde{T} \leftarrow E_K(\varSigma \oplus U)$; $T \leftarrow \mathrm{trunc}(\tilde{T}, \tau)$
return $(C, T)$
```

Fig. 1. The OCFB encipherment algorithm (no associated data)

3 OCFB (No Associated Data)

In this section we describe the basic OCFB mode with no associated data and prove its security. OCFB corresponds with the OCFB+ algorithm when the header A is set null.

3.1 Specification of OCFB

The OCFB scheme utilizes a blockcipher $E_K : \{0,1\}^n \to \{0,1\}^n$ as its underlying component. See Fig. 1 for the definition of the encipherment algorithm, and Fig. 2 for decipherment. We also give a graphical representation of the encipherment algorithm in Fig. 3.

By $\bigl(M[1], M[2], \ldots, M[m]\bigr) \leftarrow M$ we mean partitioning a message $M \in \{0,1\}^*$ into n-bit blocks, so that we have $\bigl|M[i]\bigr| = n$ except the last block for which we have $1 \le \bigl|M[m]\bigr| \le n$ (Only when M is the null string we have $\bigl|M[m]\bigr| = \bigl|M[1]\bigr| = 0$). The initial value $C[0]$ is set to $10^{n-1} \in \{0,1\}^n$ for compatibility with OCFB+. The truncation function $\mathrm{trunc}(X, \alpha)$ denotes the leftmost α bits of the string X. The padding $M[m]\|10^*$ means appending a bit 1 and then a necessary number of zeros till the entire string becomes n-bit long.

3.2 Security Proofs of OCFB

Now we prove the security of OCFB. The following is the main result of OCFB's security:

```
Input: a nonce $N \in \{0,1\}^n$, a ciphertext $C \in \{0,1\}^*$, a tag $T \in \{0,1\}^\tau$
Output: a message $M \in \{0,1\}^*$ or the invalid symbol $\perp$
$W \leftarrow E_K(N)$; $\bigl(C[1], C[2], \ldots, C[m]\bigr) \leftarrow C$; $C[0] \leftarrow 10^{n-1}$
for $i = 1$ to $m$ do
    $X[i] \leftarrow C[i-1] \oplus 2^{i+1}W$; $Y[i] \leftarrow E_K\bigl(X[i]\bigr)$; $M[i] \leftarrow C[i] \oplus \mathrm{trunc}\bigl(Y[i], |C[i]|\bigr)$
end
if $|M[m]| = n$ then                              // $|C|$ is a positive multiple of $n$
    $\Sigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus M[m]$; $U \leftarrow 2^{m+1}3W$
else
    $\Sigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus (M[m] \| 10^*)$; $U \leftarrow 2^{m+1}3^2W$
end
$\tilde{T} \leftarrow E_K(\Sigma \oplus U)$; $T' \leftarrow \mathrm{trunc}(\tilde{T}, \tau)$
if $T = T'$ then
    return $(M[1], M[2], \ldots, M[m])$
else
    return $\perp$
end
```

Fig. 2. The OCFB decipherment algorithm (no associated data)

Theorem 1. *If the underlying blockcipher E is a secure PRP, then the OCFB mode $\mathcal{E}$ using E as its underlying primitive is a secure AE scheme. Specifically, we have*

$$\mathrm{Adv}_{\mathcal{E}}^{\mathrm{nae}}(t, q, \sigma) \leq \mathrm{Adv}_{E}^{\mathrm{prp}}(t', 2\sigma + 2q) + \frac{3(\sigma + q)^2}{2^n} + \frac{2q}{2^\tau},$$

where t' is about $t + (2\sigma + 2q)\,\mathrm{Time}(E)$.

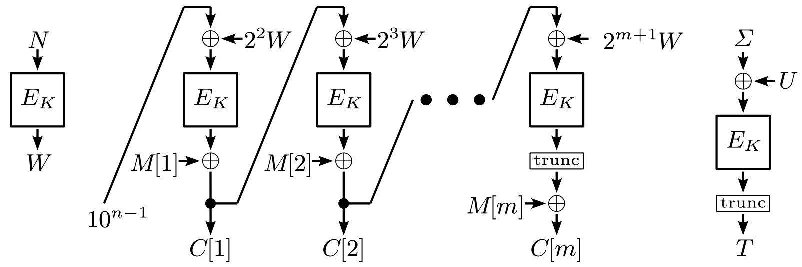
Proof. Let $\mathcal{A}$ denote an adversary attacking $\mathcal{E}$, whose running time is at most t, making at most q queries, the total query length being at most σ blocks. We begin by replacing blockcipher calls with independent random functions according to the whitening values. By Lemma 1 this counts $\mathrm{Adv}_{E}^{\mathrm{prp}}(t', 2\sigma + 2q) + 3(\sigma + q)^2/2^n$.

Now it is immediate that the encipherment oracle $\mathcal{E}(\cdots)$ behaves exactly like the random oracle $\$(\cdots)$, because the nonce value N changes every time $\mathcal{A}$ makes its queries. That is, the values (C, T) are computed using fresh random functions $F^{i,j,N}$ at each query.

So it remains to prove that the decipherment oracle $\mathcal{E}^{-1}(\cdots)$ behaves almost exactly like the reject oracle $\perp(\cdots)$. Let (N, C, T) denote $\mathcal{A}$'s query to the decipherment oracle, and we would like to evaluate the probability that at this query the $\mathcal{E}^{-1}$ oracle (now with independent random functions $F^{i,j,N}$) returns M rather than $\perp$.

Case *N* new: In this case we have a fresh random function outputting a tag T', so the probability that $T = T'$ occurs is at most $1/2^\tau$.

Case *N* old: This means that $\mathcal{A}$ has already made a query (N, M^*), for some $M^* \in \{0,1\}^n$, to the encipherment oracle. We must have $(C, T) \neq (C^*, T^*)$,



Let $\Sigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus M[m]$, $U \leftarrow 2^{m+1}3W$ if $|M[m]| = n$

Let $\Sigma \leftarrow M[1] \oplus M[2] \oplus \cdots \oplus (M[m] \| 10^*)$, $U \leftarrow 2^{m+1}3^2W$ otherwise

Fig. 3. Illustration of OCFB (no associated data)

which implies that $C \neq C^*$. If C and C^* differ in some "early" blocks, then it means that we have a random variable Σ, so the probability that $\Sigma = \Sigma^*$ holds is at most $1/2^n$. If C and C^* differ only in the last blocks, or if $\Sigma \neq \Sigma^*$, then the probability that $T = T^*$ holds is at most $1/2^\tau$. Overall, the probability that $T = T^*$ holds in the case N old is at most $2/2^\tau$.

Therefore, the probability that $\mathcal{E}^{-1}$ oracle (with random $F^{i,j,N}$) returns M at each query is at most $2/2^\tau$, which means that the overall probability across $\mathcal{A}$'s queries is at most $2q/2^\tau$. This proves the desired bound. □

4 OCFB+ for AEAD

Now we introduce OCFB+ which can handle associated data. OCFB+ is a generalized version of OCFB in that it coincides with OCFB when the associated data is null. After giving the description of OCFB+, we also prove its security.

4.1 Specification of OCFB+

Just like the OCFB mode, OCFB+ also iterates a blockcipher $E_K : \{0,1\}^n \to \{0,1\}^n$ as its underlying primitive. Figure 4 describes the encipherment algorithm of OCFB+, whereas Fig. 5 does decipherment. A graphical representation of the encipherment algorithm of OCFB+ is given in Fig. 6.

OCFB+ now accepts a header $A \in \{0,1\}^*$, which is processed via a CFB-like mode of operation with nonce-dependent whitening values. The result is used to construct the initial value $C[0]$ (to be more precise, the encrypted value $Y[1]$ to be xored with $M[1]$) for encrypting the message M. The rest is the same as OCFB, including the creation of a MAC tag T.

4.2 Security Proofs of OCFB+

We can prove the security of OCFB+ similarly to OCFB. We obtain the following result:

```
**Input**: a nonce $N \in \{0,1\}^n$, a header $A \in \{0,1\}^*$, a message $M \in \{0,1\}^*$
**Output**: a ciphertext $C \in \{0,1\}^*$, a tag $T \in \{0,1\}^\tau$
$W \leftarrow E_K(N)$; $\big(A[1], A[2], \ldots, A[a]\big) \leftarrow A$; $\big(M[1], M[2], \ldots, M[m]\big) \leftarrow M$
$Y[1] \leftarrow 0^n$
for $i = 2$ to $a$ do
    $X[i] \leftarrow Y[i-1] \oplus A[i-1] \oplus W$; $Y[i] \leftarrow E_K\big(X[i]\big)$
end
if $|A[a]| = n$ then                    // $|A|$ is a positive multiple of $n$
    $C[0] \leftarrow Y[a] \oplus A[a]$; $V \leftarrow 2W$
else
    $C[0] \leftarrow Y[a] \oplus (A[a] \| 10^*)$; $V \leftarrow 2^2 W$
end
$X[1] \leftarrow C[0] \oplus V$; $Y[1] \leftarrow E_K\big(X[1]\big)$; $C[1] \leftarrow M[1] \oplus \mathrm{trunc}\big(Y[1], |M[1]|\big)$
for $i = 2$ to $m$ do
    $X[i] \leftarrow C[i-1] \oplus 2^{i+1} W$; $Y[i] \leftarrow E_K\big(X[i]\big)$
    $C[i] \leftarrow M[i] \oplus \mathrm{trunc}\big(Y[i], |M[i]|\big)$
end
$C \leftarrow \big(C[1], C[2], \ldots, C[m]\big)$
...
```

Fig. 4. The OCFB+ encipherment algorithm (AEAD)

Theorem 2. *If the underlying blockcipher E is a secure PRP, then the OCFB+ mode $\mathcal{E}$ using E as its underlying primitive is a secure AEAD scheme. Specifically, we have*

$$\mathrm{Adv}_{\mathcal{E}}^{\mathrm{nae}}(t, q, \sigma) \leq \mathrm{Adv}_{E}^{\mathrm{prp}}(t', 2\sigma) + \frac{4\sigma^2}{2^n} + \frac{0.5q^2}{2^n} + \frac{5q}{2^\tau},$$

where t' is about $t + 2\sigma\,\mathrm{Time}(E)$.

Proof. The proof is very similar to that of OCFB. We need just a bit more of case analysis due to the presence of a header A.

Again, let $\mathcal{A}$ denote an adversary, whose running time is at most t, making at most q queries, the total query length being at most σ blocks (Note that header lengths contribute to σ). Replace blockcipher calls with random functions $F^{i,j,N}$. By Lemma 1 this costs us $\mathrm{Adv}_{E}^{\mathrm{prp}}(t', 2\sigma) + 3\sigma^2/2^n$.

We observe that the encipherment oracle $\mathcal{E}(\cdots)$ behaves exactly like the random oracle $\$(\cdots)$; the nonce value N changes every time $\mathcal{A}$ makes its queries, and the values (C, T) are computed using fresh random functions $F^{i,j,N}$ at each query.

Just like before, it remains to prove that the decipherment oracle $\mathcal{E}^{-1}(\cdots)$ behaves almost exactly like the reject oracle $\perp(\cdots)$. So again, let (N, A, C, T) denote $\mathcal{A}$'s query to the $\mathcal{E}^{-1}$ oracle, and we would like to evaluate the probability that at this query the $\mathcal{E}^{-1}$ oracle (now with independent random functions $F^{i,j,N}$) returns M rather than $\perp$.

```
Input: a nonce $N \in \{0,1\}^n$, a header $A \in \{0,1\}^*$, a ciphertext $C \in \{0,1\}^*$,
       a tag $T \in \{0,1\}^\tau$
Output: a message $M \in \{0,1\}^*$ or the invalid symbol $\perp$
$W \leftarrow E_K(N)$; $\big(A[1], A[2], \ldots, A[a]\big) \leftarrow A$; $\big(C[1], C[2], \ldots, C[m]\big) \leftarrow C$
$Y[1] \leftarrow 0^n$
for $i = 2$ to $a$ do
    $X[i] \leftarrow Y[i-1] \oplus A[i-1] \oplus W$; $Y[i] \leftarrow E_K\big(X[i]\big)$
end
if $|A[a]| = n$ then                          // $|A|$ is a positive multiple of $n$
    $C[0] \leftarrow Y[a] \oplus A[a]$; $V \leftarrow 2W$
else
    $C[0] \leftarrow Y[a] \oplus (A[a] \| 10^*)$; $V \leftarrow 2^2 W$
end
$X[1] \leftarrow C[0] \oplus V$; $Y[1] \leftarrow E_K\big(X[1]\big)$; $M[1] \leftarrow C[1] \oplus \mathrm{trunc}\big(Y[1], |C[1]|\big)$
for $i = 2$ to $m$ do
    $X[i] \leftarrow C[i-1] \oplus 2^{i+1} W$; $Y[i] \leftarrow E_K\big(X[i]\big)$; $M[i] \leftarrow C[i] \oplus \mathrm{trunc}\big(Y[i], |C[i]|\big)$
end
if $|M[m]| = n$ then                          // $|C|$ is a positive multiple of $n$
    ...
else
    ...
end
...
```

Fig. 5. The OCFB+ decipherment algorithm (AEAD)

Case *N* new: We have a fresh random function outputting a tag T', so the probability of this case is at most $1/2^\tau$. The overall probability of this event across $\mathcal{A}$'s queries can be bounded by $q/2^\tau$.

Case *N* old: This means that $\mathcal{A}$ has already made a query (N, A^*, M^*), for some $A^*, M^* \in \{0,1\}^n$, to the encipherment oracle $\mathcal{E}(\cdots)$. We further divide this case.

Case $\boldsymbol{A = A^*}$: We must have $C \neq C^*$. Similarly to the case of OCFB, we can show that the probability that $T = T^*$ holds in this case is at most $2/2^\tau$, and the overall probability of this event is bounded by $2q/2^\tau$.

Case $\boldsymbol{A \neq A^*}$: First, we may have a collision, between A and A^*, at the value $Y[1]$ $(=Y^*[1])$ to be xored with $M[1]$ (or with $M^*[1]$). The overall probability of this event can be bounded by $\binom{\sigma}{2}/2^n + \binom{q}{2}/2^n \leq 0.5(\sigma^2 + q^2)/2^n$. We may have a collision $C[0] = C^*[0]$, whose overall probability is at most $\binom{\sigma}{2}/2^n \leq 0.5\sigma^2/2^n$. If no such collisions occur, then the overall probability that $T = T'$ holds is at most $2q/2^\tau$.

Therefore, the overall probability that the $\mathcal{E}^{-1}$ oracle (with random $F^{i,j,N}$) returns M across $\mathcal{A}$'s queries is at most $q/2^\tau + 2q/2^\tau + 0.5(\sigma^2 + q^2)/2^n + 0.5\sigma^2/2^n + 2q/2^\tau = \sigma^2/2^n + 0.5q^2/2^n + 5q/2^\tau$. This proves the desired bound. □

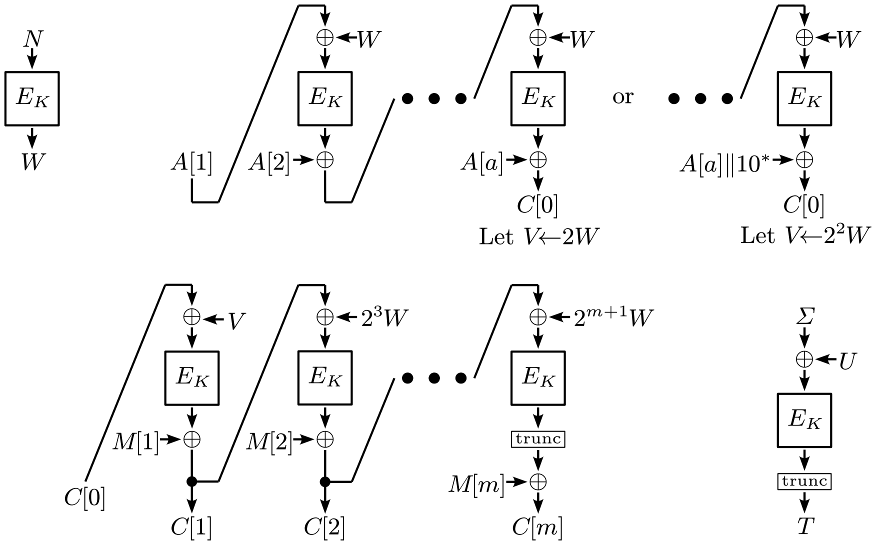


Fig. 6. Illustration of OCFB+ for AEAD

5 Concluding Remarks

We have introduced one-pass, rate-1 AE schemes OCFB and OCFB+. OCFB and OCFB+ do not require the blockcipher inverse for their decipherment processes. The big disadvantage of OCFB/OCFB+ is that the encipherment process cannot be parallelized. It remains open if we can construct a fully-parallelizable rate-1 AE scheme which operates with no blockcipher inverse.

References

1. Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Tatsuaki Okamoto, editor, *ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 531–545. Springer, 2000.
2. Alex Biryukov and Dmitry Khovratovich. Related-key cryptanalysis of the full AES-192 and AES-256. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 1–18. Springer, 2009.
3. Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolic. Distinguisher and related-key attack on the full AES-256. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 231–249. Springer, 2009.
4. John Black and Phillip Rogaway. A block-cipher mode of operation for parallelizable message authentication. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 384–397. Springer, 2002.
5. Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 344–371. Springer, 2011.

6. Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *CHES 2007*, volume 4727 of *LNCS*, pages 450–466. Springer, 2007.
7. Christophe De Cannière, Orr Dunkelman, and Miroslav Knezevic. KATAN and KTANTAN—a family of small and efficient hardware-oriented block ciphers. In Christophe Clavier and Kris Gaj, editors, *CHES 2009*, volume 5747 of *LNCS*, pages 272–288. Springer, 2009.
8. Joan Daemen and Vincent Rijmen. The block cipher Rijndael. In Jean-Jacques Quisquater and Bruce Schneier, editors, *CARDIS 1998*, volume 1820 of *LNCS*, pages 277–284. Springer, 2000.
9. Virgil D. Gligor and Pompiliu Donescu. Fast encryption and authentication: XCBC encryption and XECB authentication modes. In Mitsuru Matsui, editor, *FSE 2001*, volume 2355 of *LNCS*, pages 92–108. Springer, 2001.
10. Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *CHES 2011*, volume 6917 of *LNCS*, pages 326–341. Springer, 2011.
11. Tetsu Iwata and Kaoru Kurosawa. OMAC: One-key CBC MAC. In Thomas Johansson, editor, *FSE 2003*, volume 2887 of *LNCS*, pages 129–153. Springer, 2003.
12. Goce Jakimoski and Samant Khajuria. ASC-1: An authenticated encryption stream cipher. In *SAC 2011*, LNCS. Springer, 2012.
13. Éliane Jaulmes, Antoine Joux, and Frédéric Valette. On the security of randomized CBC-MAC beyond the birthday paradox limit: A new construction. In Joan Daemen and Vincent Rijmen, editors, *FSE 2002*, volume 2365 of *LNCS*, pages 237–251. Springer, 2002.
14. Charanjit S. Jutla. Encryption modes with almost free message integrity. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 529–544. Springer, 2001.
15. Jonathan Katz and Moti Yung. Unforgeable encryption and chosen ciphertext secure modes of operation. In Bruce Schneier, editor, *FSE 2000*, volume 1978 of *LNCS*, pages 284–299. Springer, 2001.
16. Ted Krovetz and Phillip Rogaway. The software performance of authenticated-encryption modes. In Antoine Joux, editor, *FSE 2011*, volume 6733 of *LNCS*, pages 306–327. Springer, 2011.
17. Stefan Lucks. Two-pass authenticated encryption faster than generic composition. In Henri Gilbert and Helena Handschuh, editors, *FSE 2005*, volume 3557 of *LNCS*, pages 284–298. Springer, 2005.
18. David A. McGrew and John Viega. The security and performance of the Galois/Counter Mode (GCM) of operation. In Anne Canteaut and Kapalee Viswanathan, editors, *INDOCRYPT 2004*, volume 3348 of *LNCS*, pages 343–355. Springer, 2004.
19. Kazuhiko Minematsu and Yukiyasu Tsunoo. Provably secure MACs from differentially-uniform permutations and AES-based implementations. In Matthew J. B. Robshaw, editor, *FSE 2006*, volume 4047 of *LNCS*, pages 226–241. Springer, 2006.
20. NIST. Advanced Encryption Standard (AES). FIPS 197, 2001.
21. NIST. Recommendation for block cipher modes of operation: The CMAC mode for authentication. SP 800-38B, 2005.

22. NIST. Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) for confidentiality and authentication. SP 800-38D, 2007.
23. Phillip Rogaway. Authenticated-encryption with associated-data. In Vijayalakshmi Atluri, editor, *ACM CCS 2002*, pages 98–107. ACM, 2002.
24. Phillip Rogaway. Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In Pil Joong Lee, editor, *ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 16–31. Springer, 2004.
25. Phillip Rogaway. Nonce-based symmetric encryption. In Bimal K. Roy and Willi Meier, editors, *FSE 2004*, volume 3017 of *LNCS*, pages 348–359. Springer, 2004.
26. Phillip Rogaway, Mihir Bellare, John Black, and Ted Krovetz. OCB: A block-cipher mode of operation for efficient authenticated encryption. In Michael K. Reiter and Pierangela Samarati, editors, *ACM CCS 2001*, pages 196–205. ACM, 2001.