Cryptanalysis of Authenticated Encryption Modes for Wireless and Real-Time Systems

Alexander Bille and Elmar Tischhauser(✉)

Department of Mathematics and Computer Science, University of Marburg, Marburg, Germany
{bille,tischhauser}@informatik.uni-marburg.de

Abstract. Authenticated encryption (AE) plays a central role in building secure channels for wireless systems, with well-established AE schemes such as CCM or GCM being widely used in security protocols for wireless networks based on IEEE 802.11 (Wi-Fi), IEEE 802.15.4 (such as Zigbee), as well as LTE and 5G mobile networks. Having been proposed as general-purpose AE schemes, they leave optimization potential for new algorithms specifically designed for wireless applications. In this paper, we analyze the security of three such AE algorithm families, namely PFX, PFC and IAR, which were designed to guarantee confidentiality and authenticity in a single-pass process while reducing the number of block cipher calls and avoiding expensive operations like finite field multiplications. As such, they were proposed as alternatives to CCM or GCM for wireless systems, lightweight wireless sensor networks, and real-time wireless applications.

In this paper, we describe universal forgery attacks on all three algorithm families, allowing an adversary to compute valid ciphertexts and authentication tags for any message of their choice without knowledge of the secret key. All attacks only have linear complexity in the length of the target message and as such are entirely practical, essentially as fast as the encryption itself. Our attacks imply that the affected schemes should not be used in practice, despite their attractive performance characteristics.

Keywords: Symmetric cryptography · authenticated encryption · cryptanalysis · universal forgery attacks · wireless network security

1 Introduction

1.1 Motivation and Background

Wireless and mobile networks have become integral components of modern communication systems, playing a central role in connecting individuals, devices, and applications. Since it is common for such networks to handle sensitive and private information, ensuring secure communication to protect transmitted data against unauthorized access is of great importance.

K. Minematsu and M. Mimura (Eds.): IWSEC 2024, LNCS 14977, pp. 23–42, 2024.
https://doi.org/10.1007/978-981-97-7737-2_2

In order to achieve these security objectives, one usually uses authenticated encryption (AE) schemes, which provide both confidentiality and authenticity and integrity in one combined cryptographic primitive [4,24]. In many applications, e.g. secure software updates, healthcare IoT or smart grid management, data authentication is arguably even more important than confidentiality.

Authenticated encryption schemes can broadly be divided into two main categories: the generic composition [4] of an encryption scheme and a message authentication code (MAC), and dedicated constructions aimed at integrating both with more attractive performance or implementation characteristics. Many AE schemes are modes of operation for a block cipher, meaning they can be instantiated with any desired block cipher (for instance, the AES or GIFT [3]) as the underlying cryptographic primitive.

Authenticated encryption for wireless networks is implemented in the IEEE 802.11 (Wi-Fi) family of protocols. The WPA2 and WPA3 protocols employ CCM for confidentiality and integrity [15]. CCM, a mode of operation for block ciphers combining Counter (CTR) mode with CBC-MAC, requires two passes over the message and hence two block cipher calls per message block.

In the context of low-power and resource-constrained wireless sensor networks, the IEEE 802.15.4 standard, used in applications like Zigbee, also uses authenticated encryption in the form of CCM mode [16]. One main concern and design restriction in the context of wireless sensor networks is extending the operational lifespan of battery-powered devices.

In more recent wireless communication protocols such as Long-Term Evolution (LTE) and 5G networks as well as in WPA3, Galois/Counter Mode (GCM) [10,23] has gained prominence due to its parallelizable nature and efficiency for high-speed data transmission, both to secure user data and control plane signaling [11,15]. However, due to the use of large finite field multiplications in addition to block cipher calls, GCM is not particularly suited for resource-constrained environments.

Another particularly efficient scheme is the OCB mode [21,26,27], which is widely standardized [17,22] and in the final portfolio of the NIST-sponsored CAESAR competition [5]. It has the advantage of being a single-pass scheme, requiring only one block cipher call per message block and being completely parallelizable. Its patent status and large internal state however mean that OCB has not found as widespread use as one might expect. However, OCB has been considered in scenarios where minimizing overhead and achieving low-latency communication are critical, such as in real-time applications within mobile networks.

1.2 New AE Designs for Wireless and Real-Time Systems

Design constraints in wireless networks, including limited bandwidth, variable channel conditions, and power constraints, necessitate the careful selection of authenticated encryption schemes. The resulting trade-offs between security, computational efficiency, and energy consumption are central for the inclusion of these schemes in current and future wireless network protocols. As the landscape

of wireless communication evolves with emerging technologies like the Internet of Things (IoT) and 6G, it remains an active research topic to improve upon existing authenticated encryption schemes to better meet these specific design constraints.

One particular need for resource-constrained platforms is to minimize the amount of state (e.g., the number of keys or tweaks derived from the master key and nonce) and auxiliary routines (such as finite field multiplication) beyond simple block cipher calls. It is also important to achieve secure AE within a single pass over the data and ideally with only one block cipher call per message block.

These requirements have led to the proposal of several new AE schemes specifically designed for use in wireless and real-time systems. In this paper, we consider the PFX, PFC and IAR families of AE algorithms. PFX [13] is a family of authenticated encryption modes designed to achieve single-pass AE with only $n + 1$ block cipher calls for an n-block message. It relies on the idea of plaintext feedback and consists of three individual variants, plain PFX as the basic algorithm, and the two main new variants PFX-CTR and PFX-INC combining ideas from CTR mode and GCM and OCB, respectively. Its main application area are general-purpose wireless networks. The PFC [14] family of AE schemes follows similar design ideas as PFX, but is tailored towards more lightweight platforms such as wireless sensor networks and comes in two variants based on CTR and OCB mode. Finally, IAR [12] is family of two AE modes IAR-CTR and IAR-CFB developed for use in applications with real-time constraints. It caters for a maximum acceptable system delay by using multiple authentication tags.

All three families are designed to improve upon the state of the art in Wi-Fi security by providing superior performance characteristics compared to existing modes such as CCM or GCM. They are also accompanied by security proofs, meaning that they are designed to offer confidentiality and authenticity up to the standard birthday bound of $2^{n/2}$ provided the underlying n-bit block cipher is secure. They also have in common that they are based upon widely used and standardized secure building blocks such as the CTR, CFB and OCB modes of operation.

We finally note that all AE schemes discussed in this section depend on the uniqueness of a nonce for their security guarantees. The same holds for the standard AE schemes such as CCM, GCM, and OCB. All of our attacks respect this setting and never repeat nonces for queries with the same key.

1.3 Contributions

In this paper, we present universal forgery attacks on several authenticated encryption schemes proposed for wireless and real-time systems, in particular the PFX, PFC and IAR families of algorithms. These attacks allow the adversary to create valid ciphertexts and tags for any message of their choice without knowledge of the secret key in a chosen plaintext attack (CPA). We note that the CPA setting is the standard security model in symmetric cryptography, and

all schemes attacked in this paper actually come with a security proof in this model. Our attacks hence also invalidate these proofs.

The basic attack strategy is to simulate the calls to block cipher encryptions with the fixed but unknown key by auxiliary chosen plaintext queries. The results from these queries can then be used by the attacker to compute ciphertext and tag for an arbitrary message, resulting in universal forgery attacks. The complexity of our attacks is also very low, namely linear in the length of the target message of the forgery. This means that the effort to universally forge a message for these schemes is basically equivalent to the effort of actually carrying out the authenticated encryption algorithm with knowledge of the secret key.

Altogether, our attacks imply that the affected schemes do not provide the claimed security guarantees and, despite their attractive performance characteristics, should not be used in practice.

Outline of the Paper. We first describe the three algorithm families analyzed in this paper in Sects. 2 to 4. Section 5 outlines the attack model and the general strategy for the universal forgery attacks, then presents our attacks on the PFX, PFC and IAR families of authenticated encryption schemes. Section 6 concludes. A detailed description of our notation can be found in Appendix A.1.

2 The PFX Family of Authenticated Encryption Schemes

The scheme PFX and its advanced modes PFX-CTR, PFX-INC and PFX-CBC are authenticated encryption (AE) protocols designed by Hwang and Gope [13]. Their goal was to perform encryption and authentication with only $n + 1$ block encryption calls and in one natural single process (referred to as "authencryption"). The main idea of this family is the use of plaintext feedback as seen in Fig. 1. Each mode has two variants for certifying the integrity of the message. The first works with a so called indicator I which is a preshared value between sender and receiver. This indicator "may not be confidentia" [13] and therefore may be known to the adversary in an attack scenario. The second variant encrypts the last block with a second key K'. Since the basic version of PFX has some limitations compared to AE schemes such as CCM, its designers only recommend this mode for improved authenticity and integrity over conventional encryption-only modes such as CTR. For a full replacement of standard AE schemes, they propose three advanced modes building on PFX: PFX-CTR and PFX-CBC are a fusion of PFX with counter mode [6] and CBC mode [25], respectively, whereas PFX-INC is a fusion with schemes including a incrementing function. The authors mention to use the incremental interface of GCM, OCB, IAPM [19] or CWC [20] for their incremental function. Detailed algorithmic descriptions and illustrations for the encryption process of PFX, PFX-CTR and PFX-INC are provided in Algorithms 1 and 2 and Figs. 1, 3 and 4 in Sect. 5 for easier cross-reference with the attack procedures.

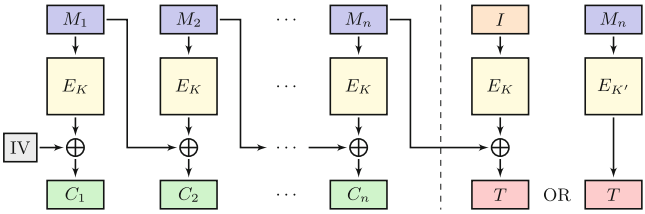


Fig. 1. The PFX authenticated encryption algorithm.

3 The PFC Family of Authenticated Encryption Algorithms

The modes PFC-CTR and PFC-OCB [14] are authenticated encryption schemes. Their motivation is to guarantee integrity and confidentiality with a small additional computation cost. Their schemes work with only $n + 2$ encryption block calls for an n-block message and no other expensive functions, aiming at resource-constrained platforms such as wireless sensor networks. The main idea consists of plaintext feedback, the truncation of block cipher outputs to some most significant bits and a double encryption for the tag. The scheme PFC-OCB is illustrated in Fig. 7 and is a fusion of the well-known OCB mode with the general framework of PFC. This variant follows the OCB standard quite closely, and as such is not affected by our analysis. When instantiated with a p-bit block cipher, the plaintext and ciphertext blocks are r bits long and the tag consists of ℓ bits with $r < \ell \leq p$. The tag is computed by a double encryption where the number of blocks (NOB) of the message is xored between the encryptions. The authors specifically propose the PFC schemes for use in the context of wireless sensor networks, Global Mobility Networks and cloud computing environments because of their attractive computational properties. Detailed algorithmic descriptions and illustrations for PFC-CTR are provided in Algorithm 3 and Fig. 5 in Sect. 5 alongside the corresponding attack procedures.

4 The IAR Family of Authenticated Encryption Schemes

The authenticated encryption modes IAR-CTR and IAR-CFB [12] have been developed for the use in real-time applications, in particular low-latency wireless real-time networks. As for the other families a major focus is the efficiency. Hence IAR-CTR and IAR-CFB use only $n + 2t$ and $n + t$ respectively many block cipher calls for an n-block message. Both modes are designed to cater for a system delay of t encryption blocks. This delay can be adjusted according to the time a message block is processed in a concrete application. The procedure of the IAR schemes can be separated in three parts, as illustrated in Fig. 2. The first part is t blocks long without the plaintext feedback. The input of the block cipher is not message dependent and could in principle be preprocessed. In the second part, the remaining message blocks are encrypted where the input

is xored with the t previous (zero padded) plaintexts. The last part creates the t authentication tags. Note that the IAR family use a p-bit block cipher, message blocks of r bits and t many ℓ-bit tags with $r < \ell \leq p$.

The first proposed mode IAR-CTR has its focus on the use of a counter and double encryption for the tags, similar to the PFC family. The second mode IAR-CFB makes use of ciphertext feedback after the initial t ciphertext blocks, meaning a ciphertext block is concatenated to the last one shifted by r bits in a ciphertext feedback shift register (or in other words, the $p - r$ least significant bits are taken). For the first shift operation (during the computation of C_{t+1}), the last counter value from the first part is used instead. A detailed description and illustration of the encryption algorithm for IAR-CTR and IAR-CFB can be found in Algorithm 4 and Figs. 2 and 6 in Sect. 5 alongside the corresponding attack procedures.

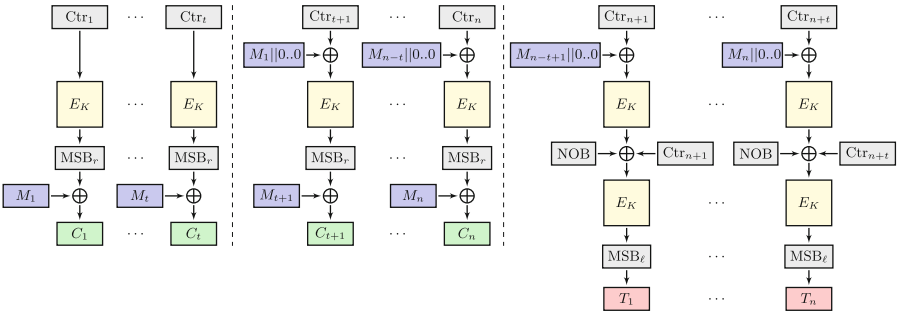


Fig. 2. The IAR-CTR authenticated encryption algorithm.

5 Attacks

In this section, we present several universal forgery attacks on the PFX, PFC and IAR families of authenticated encryption modes. All these attacks allow an adversary to produce valid ciphertexts and tags for an arbitrary message of their choice in a standard chosen plaintext attack setting. Their complexity is always at most linear in the length of the target message, which makes them completely practical, essentially as efficient as running the encryption algorithm itself on the same message.

5.1 Attack Model

We assume the adversary to be able to make chosen plaintext queries to the scheme with an unknown but fixed key K, which is the exact model used for the security proofs of PFX, PFC and IAR [12–14]. In detail, the rules for the adversary are the following: The adversary does not know the key(s). The adversary

can make chosen plaintext queries by asking for the encryption of some messages of their choice. Extra parameters such as IV or nonce may be set by the adversary. To respect the design constraints of the modes, the adversary is not allowed to ask for messages with repeating values for an extra parameter, e.g. the adversary may not ask for two message encryptions with the same nonce. In the indicator version of the modes of the PFX family, we assume that the indicator is not known. When making requests, the correct but secret indicator is used for the encryption.

The goal is to build a valid ciphertext-tag pair $C_1, \ldots, C_n, T$ for an arbitrary message $M = M_1, \ldots, M_n$ of our choice out of these queries. When making auxiliary chosen plaintext queries, these auxiliary messages need to be different from M itself. Such a procedure constitutes a *universal forgery attack* on the authenticated encryption scheme, since the adversary is able to forge arbitrary messages of their choice without knowledge of the secret key.

5.2 General Strategy

Most of our forgery attacks are based on the general approach of simulating valid encryptions through carefully crafted auxiliary queries, which we summarize in the following observation:

Observation 1. *Obtaining a block cipher oracle $E_K(X)$ through one or more requests to the authenticated encryption mode is equivalent to being able to perform the encryption and authentication process without knowing the secret key K since the subsequent operations then depend only on values known to the adversary.*

The algorithm which implements such a block cipher oracle through auxiliary chosen plaintext queries to the scheme is referred to as a *gadget* G which has the property that $G(X) = E_K(X)$ without knowledge of the secret key K.

In some schemes, the outputs of the block cipher calls are immediately truncated to the most b significant bits. In these cases we create a gadget simulating the combined process. We mark this property in the superscript of the gadget, i.e. $G^b(X) = \mathrm{MSB}_b(E_K(X))$.

5.3 Attack on PFX

We present a universal forgery attack on PFX in this section. Assume we want to forge the authenticated encryption $C_1, \ldots, C_n, T^*$ of the message $M = M_1, \ldots, M_n$ with initial value IV^*.

We use the strategy described in Observation 1. In the case of PFX, the inputs to the encryption query interface only consist of the IV and plaintext and, depending on the variant, the indicator. We create a gadget G_{PFX} simulating calls to the block cipher $E_K(\cdot)$ by auxiliary chosen plaintext queries to the PFX_K authenticated encryption scheme. To obtain $E_K(X)$ the gadget requests

$$\mathrm{PFX}_K^{\overline{\mathrm{IV}}}(X) = C_1', T'$$

with an arbitrary, unused $\overline{\text{IV}} \neq \text{IV}^*$. The block C_1' is one-block result and T' is the tag which is of no interest. Since $C_1' = E_K(X) \oplus \overline{\text{IV}}$ the gadget returns $G_{\text{PFX}}(X) := C_1' \oplus \overline{\text{IV}} = E_K(X)$.

Observation 2. *With G_{PFX} one can get $E_K(X)$ for an arbitrary X without knowledge of the key.*

Now we can forge the ciphertext of our message by the following algorithm.

1. Use G_{PFX} to obtain $E_K(M_1), \ldots, E_K(M_n)$.
2. Compute $C_i = E_K(M_i) \oplus M_{i-1}$ with $M_0 = \text{IV}^*$.
3. Request any j-block message $M' = \overline{M}_1, \ldots, \overline{M}_{j-1}, M_n$ with M_n as the last block. The tag T' of $\text{PFX}_K^{\text{IV}'}(M') = C_1', \ldots, C_j'$, T' fits the demands since
   (a) (indicator variant) $T' = E_K(I) \oplus M_n = T^*$.
   (b) (two key variant) $T' = E_{K'}(M_n) = T^*$.

The use of single-block auxiliary queries to PFX in the gadget means that the above method cannot be used for forging single block messages. For this special case, we can use an insertion variant which works for messages of length 1 (as well as also for longer messages). This variant requests

$$\text{PFX}_K^{\overline{\text{IV}}}(IV^*, M_1, \ldots, M_n) = C_0', C_1, \ldots, C_n, T^*$$

for some arbitrary $\overline{\text{IV}}$, obtaining all the necessary ciphertext blocks for our forgery. Only the block C_0' is of no use and is discarded. In both the indicator and the two-key variants of PFX, the token T^* is valid as shown in the last step of the above forgery algorithm.

Algorithm 1: Encryption $\text{PFX}_K^{\text{IV}}(M_1, \ldots, M_n)$

$C_1 \leftarrow E_K(M_1) \oplus \text{IV}$
for $i = 2$ *to* n **do**
  $C_i \leftarrow E_K(M_i) \oplus M_{i-1}$
if $I \neq NULL$ **then** `// indicator version`
  $T \leftarrow E_K(I) \oplus M_n$
else
  $T \leftarrow E_{K'}(M_n)$
return $C_1, \ldots, C_n, T$

5.4 Attack on PFX-CTR

We now describe a universal forgery attack on PFX-CTR. Assume we want to forge the encryption $C_1, \ldots, C_n, T^*$ of the message $M = M_1, \ldots, M_n$ with starting value for the counter SV^* and initial value IV^*. First note that Observation 1 also holds for PFX-CTR, we therefore create a gadget $G_{\text{PFX-CTR}}$ simulating $E_K(X)$. The use of a starting value gives us more possibilities for the gadget. To simulate $E_K(X)$ the gadget requests

$$\mathrm{PFX}_K^{\overline{\mathrm{SV}},\overline{\mathrm{IV}}}(X \oplus \overline{\mathrm{SV}}) = C_1', T'$$

with an unused $\overline{\mathrm{IV}} \neq \mathrm{IV}^*$ and an unused $\overline{\mathrm{SV}} \neq \mathrm{SV}^*$. The block C_1' is a one-block result and T' is the tag which is of no interest. Since $C_1' = E_K((X \oplus \overline{SV}) \oplus \overline{\mathrm{SV}}) \oplus \overline{\mathrm{IV}}$ the gadget returns $G_{\text{PFX-CTR}}(X) := C_1' \oplus \overline{\mathrm{IV}}$.

Observation 3. *For a given X the gadget $G_{PFC\text{-}CTR}$ returns $E_K(X)$ without knowledge of the key K.*

The universal forgery attack procedure is then as follows. Note that the indicator I is not known to the attacker and that $T^* = E_K(I \oplus \mathrm{Ctr}_{n+1})$ where $\mathrm{Ctr}_{n+1} = \mathrm{SV}^* + n$ is the counter value on I when M is encrypted.

1. Use $G_{\text{PFX-CTR}}$ to obtain $A_1, \ldots, A_n = E_K(M_1 \oplus \mathrm{SV}^*), E_K(M_2 \oplus (\mathrm{SV}^* + 1)), \ldots, E_K(M_n \oplus (\mathrm{SV}^* + n - 1))$.
2. Compute $C_i = A_i \oplus M_{i-1}$ with $M_0 = \mathrm{IV}^*$.
3. Request any j-block message $M' = \overline{M}_1, \ldots, \overline{M}_{j-1}, M_n$ with M_n as the last block and let $\mathrm{SV}' = \mathrm{SV}^* + n - j$. The tag T' of $\mathrm{PFX}_K^{\mathrm{SV}',\overline{\mathrm{IV}}}(M') = C_1', \ldots, C_j', T'$ with an arbitrary $\overline{\mathrm{IV}} \neq \mathrm{IV}^*$ fits the demands since
   (a) (indicator variant) the starting value $\mathrm{SV}^* + n - j$ is so chosen that the counter value for the indicator is $(\mathrm{SV}^* + n - j) + j = \mathrm{SV}^* + n$. Hence, $T' = E_K(I \oplus (\mathrm{SV}^* + n)) \oplus M_n = T^*$.
   (b) (two key variant) $T' = E_{K'}(M_n) = T^*$.

Due to the freedom provided by choosing the starting value, the same procedure also works for one block messages, so no special variant is required for this case as was necessary for PFX.

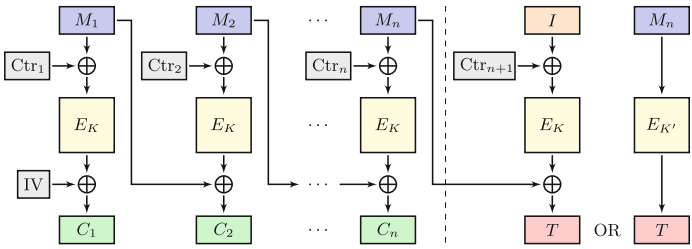


Fig. 3. The PFX-CTR authenticated encryption algorithm.

5.5 Attack on PFX-INC

In this section, we show a universal forgery attack on PFX-INC for the two key variant. Furthermore, we present a universal forgery attack when the indicator is known to the adversary, which is within the security model of PFX-INC. Since PFX-INC follows the pattern of increment-based schemes, we assume that a nonce is part of the scheme and incorporated into the first increment as indicated

Algorithm 2: Encryptions of PFX-CTR and PFX-INC

$\text{PFX-CTR}_K^{\text{SV,IV}}(M_1, \ldots, M_n)$:
$\text{Ctr}_1 \leftarrow \text{SV}$
$O_1 \leftarrow E_K(M_1 \oplus \text{Ctr}_1)$
$C_1 \leftarrow \text{IV} \oplus O_1$
for $i = 2$ *to* n **do**
  $\text{Ctr}_i \leftarrow \text{Ctr}_{i-1} + 1$
  $O_i \leftarrow E_K(M_i \oplus \text{Ctr}_i)$
  $C_i \leftarrow M_{i-1} \oplus O_i$
if $I \neq NULL$ **then** `// indicator version`
  $\text{Ctr}_{n+1} \leftarrow \text{Ctr}_n + 1$
  $O_{n+1} \leftarrow E_K(I \oplus \text{Ctr}_{n+1})$
  $T \leftarrow M_n \oplus O_{n+1}$
else
  $T \leftarrow E_{K'}(M_n)$
return $C_1, \ldots, C_n, T$

$\text{PFX-INC}_K^{\text{Nonce}}(M_1, \ldots, M_n)$:
$\Delta \leftarrow \text{Init}(\text{Nonce})$
$\Delta_1 \leftarrow \text{Inc}_1(\Delta)$
$O_1 \leftarrow E_K(M_1 \oplus \Delta_1)$
$C_1 \leftarrow O_1 \oplus \Delta_1$
for $i = 2$ *to* n **do**
  $\Delta_i \leftarrow \text{Inc}_i(\Delta)$
  $O_i \leftarrow E_K(M_i \oplus \Delta_i)$
  $C_i \leftarrow M_{i-1} \oplus O_i \oplus \Delta_i$
if $I \neq NULL$ **then** `// indicator version`
  $\Delta_{n+1} \leftarrow \text{Inc}_{n+1}(\Delta)$
  $O_{n+1} \leftarrow E_K(I \oplus \Delta_{n+1})$
  $T \leftarrow M_n \oplus O_{n+1} \oplus \Delta_{n+1}$
else
  $T \leftarrow E_{K'}(M_n)$
return $C_1, \ldots, C_n, T$

in Algorithm 2. First, assume we want to forge the encryption $C_1, \ldots, C_n, T^*$ of the message $M = M_1, \ldots, M_n$ with nonce N^* in the two key variant. Since the tag depends only on the last block, the adversary asks for a message with a doubled last block. This means that one can ask for

$$\text{PFX-INC}_K^{N^*}(M_1, \ldots, M_n, M_n) = C_1, \ldots, C_n, C'_{n+1}, T'.$$

Since the tag $T' = E_{K'}(M_n)$ is the same for all messages with equal last block the equation, $T^* = T'$ holds. The block C'_{n+1} is of no use and can be discarded. The remaining blocks are our valid ciphertext-tag pair.

Now, for the indicator variant of PFX-INC, assume that the pre-shared indicator I is known to the adversary. Again, we want to forge the encryption $C_1, \ldots, C_n, T^*$ of the message $M = M_1, \ldots, M_n$ with nonce N^* in the indicator variant. We take advantage of the fact that the indicator is encrypted in the same way as the other blocks. Hence, we request the extended message

$$\text{PFX-INC}_K^{N^*}(M_1, \ldots, M_n, I) = C_1, \ldots, C_n, C'_{n+1}, T'.$$

In this scenario the tag T' is of no use for us. But the block C'_{n+1} fulfills the demands for the tag because $C'_{n+1} = E(I \oplus \Delta_{n+1}) \oplus \Delta_{n+1} \oplus M_n$. Hence, $T^* = C'_{n+1}$ and we get the desired ciphertext with its valid tag.

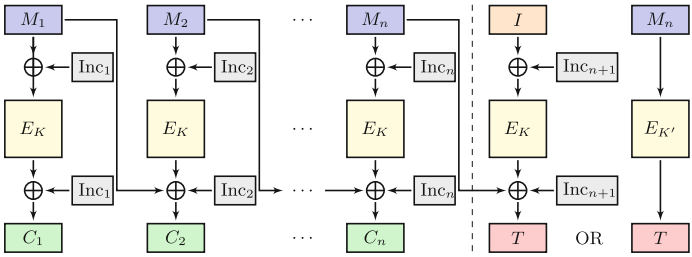


Fig. 4. The PFX-INC authenticated encryption algorithm.

5.6 Attack on PFC-CTR

We demostrate a universal forgery attack on PFC-CTR. Assume we want to forge the encryption $C_1, \ldots, C_n, T^*$ of the message $M = M_1, \ldots, M_n$ with initial value IV^*. Again, we create a gadget $G^r_{\text{PFC-CTR}}$ simulating $\mathrm{MSB}_r(E_K(X))$. To obtain the result of $\mathrm{MSB}_r(E(X))$, $G^r_{\text{PFC-CTR}}$ asks for

$$\text{PFC-CTR}_K^{\overline{\mathrm{IV}}}(M'_1, \overline{M}_2) = C'_1, C'_2, T'$$

with an arbitrary block $\overline{M}_2$, an unused $\overline{\mathrm{IV}} \neq \mathrm{IV}^*$ and $M'_1 = X \oplus (\overline{\mathrm{IV}} + 1)$. The idea of this gadget is to get the result from the second ciphertext block because of the freedom provided by the choice of $\overline{\mathrm{IV}}$ and M'_1 in the equation $(\overline{\mathrm{IV}} + 1) \oplus M'_1 = X$. Hence, the gadget returns $G^r_{\text{PFC-CTR}}(X) := C'_2 \oplus \overline{M}_2$.

The tag now needs to be computed in a different way than the ciphertext blocks. We will take the advantage of the fact that the tag depends only on the length of the message NOB, the last block and the initial value. This means that the tag remains the same as long as the last block, the number of blocks and the initial value are the same. We can then forge our message by the following procedure:

1. Use $G^r_{\text{PFC-CTR}}$ to obtain $A_1, \ldots, A_n = \mathrm{MSB}_r(E_K(\mathrm{Ctr}_1))$, $\mathrm{MSB}_r(E_K(\mathrm{Ctr}_2 \oplus M_1))$, $\ldots$, $\mathrm{MSB}_r(E_K(\mathrm{Ctr}_n \oplus M_{n-1}))$ with $\mathrm{Ctr}_i = \mathrm{IV}^* + i - 1$.
2. Compute $C_i = A_i \oplus M_i$.
3. To get the tag request $\text{PFC-CTR}^{\mathrm{IV}^*}(\overline{M}_1, \ldots, \overline{M}_{n-1}, M_n) = C'_1, \ldots, C'_n, T$ with arbitrary $\overline{M}_1, \ldots, \overline{M}_{n-1}$.

Note that the ciphertext blocks $C_1, \ldots, C_n$ can be also obtained by only one request via an insertion variant of the above algorithm. In this case, we ask for $\text{PFC-CTR}_K^{\mathrm{IV}^*-1}(\overline{M}_0, M_1, \ldots, M_n) = C'_0, C_1, \ldots, C_n, T'$ with an arbitrary block $\overline{M}_0$. This gives us all the necessary blocks for the forgery.

5.7 Attack on IAR-CTR

Assume we want to forge the encryption $C_1, \ldots, C_n, T_1, \ldots, T_t$ of the message $M = M_1, \ldots, M_n$ with initial value IV^*. To achieve this we create a gadget $G^r_{\text{IAR-CTR}}$ to simulate $\mathrm{MSB}_r(E_K(X))$ and get the tags by a special request.

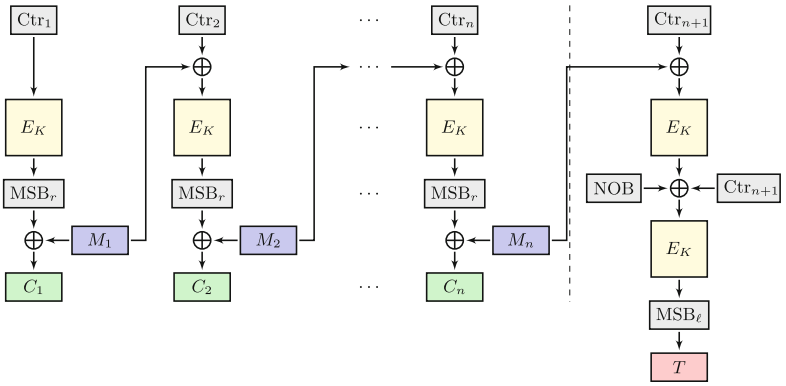


Fig. 5. The PFC-CTR authenticated encryption algorithm.

Algorithm 3: Encryption PFC-CTR$_K^{\mathrm{IV}}(M_1, \ldots, M_n)$

$\mathrm{Ctr}_1 \leftarrow \mathrm{IV}$
$O_1 \leftarrow E_K(\mathrm{Ctr}_1)$
$C_1 \leftarrow M_1 \oplus \mathrm{MSB}_r(O_1)$
for $i = 2$ *to* n **do**
  $\mathrm{Ctr}_i \leftarrow \mathrm{Ctr}_{i-1} + 1$
  $O_i \leftarrow E_K(M_{i-1} \oplus \mathrm{Ctr}_i)$
  $C_i \leftarrow M_i \oplus \mathrm{MSB}_r(O_1)$
$\mathrm{Ctr}_{n+1} \leftarrow \mathrm{Ctr}_n + 1$
$\tau \leftarrow E_K(M_n \oplus \mathrm{Ctr}_{n+1})$
$O_{n+1} \leftarrow E_K(\mathrm{NOB} \oplus \tau \oplus \mathrm{Ctr}_{n+1})$
$T \leftarrow \mathrm{MSB}_\ell(O_{n+1})$
return $C_1, \ldots, C_n, T$

The simplest option for such a gadget would be to set the initial vector such that the first output block is used for the simulation. However, this does not work whenever one block M_i happens to equal $\mathrm{IV}^* + 1$. To avoid this scenario we prepare a message such that the j-th ciphertext output block will be used with $j > t$. To obtain $\mathrm{MSB}_r(E_K(X))$ the gadget picks a (preferably small) $j > t$ and chooses some M'_{j-t} and an unused $\mathrm{IV}' \neq \mathrm{IV}^*$ subject to the constraint

$$X = (M'_{j-t}||0..0) \oplus (\mathrm{IV}' + j).$$

Let $M' = \overline{M}_1, \ldots, M'_{j-t}, \ldots, \overline{M}_j, \ldots, \overline{M}_q$ a new q-block message with M'_{j-t} as the $(j-t)$-th block. The other blocks can be set to arbitrary values. The gadget $G^r_{\text{IAR-CTR}}$ then asks for

$$\text{IAR-CTR}_K^{\mathrm{IV}'}(M') = C'_1, \ldots, C'_j, \ldots, C'_q, T'_1, \ldots, T'_t.$$

At last, $G^r_{\text{IAR-CTR}}$ returns $G^r_{\text{IAR-CTR}}(X) := C'_j \oplus \overline{M}_j$.

Observation 4. *For a given X the gadget $G^r_{IAR\text{-}CTR}$ simulates $MSB_r(E_K(X))$.*

To obtain the tags we take advantage of the fact that the tags depend only on the counter, the number of blocks and the last t message blocks. Let $M_{\mathrm{TAG}} =$

$\overline{M}_1, \ldots, \overline{M}_{n-t-1}, M_{n-t+1}, \ldots, M_n$ be the message with the same t last blocks as M and some arbitrary ones for the other blocks. Hence we get the tags by the request

$$\text{IAR-CTR}_K^{\text{IV}^*}(M_{\text{TAG}}) = C'_1, \ldots, C'_n, T_1, \ldots, T_t. \tag{1}$$

A problem occurs when $n = t$ because then $M = M_{\text{TAG}}$ and a request of the message M is not allowed by attack model. To handle this issue, we create another gadget $G^{\ell}_{\text{IAR-CTR}}$ to obtain a desired tag. Consider the following calculations. Let $\text{Ctr}_i = \text{IV}^* + i$. One tag T_i is constructed by the formula

$$T_i = \text{MSB}_{\ell}(E_K(n \oplus \text{Ctr}_{n+i} \oplus E_K(\text{Ctr}_{n+1} \oplus M_{n-t+i}||0..0)).$$

The task is to find another initial value IV', number of blocks n' and some message blocks such that we get the same T_i. Let $\text{Ctr}'_i = \text{IV}' + i$. By

$$T_i = \text{MSB}_{\ell}(E_K(n' \oplus \text{Ctr}'_{n'+i} \oplus E_K(\text{Ctr}'_{n'+1} \oplus M'_{n'-t+i}||0..0))$$

we get the following constraints:

$$\text{Ctr}_{t+i} \oplus M_{n-t+i}||0..0 = \text{Ctr}'_{n'+i} \oplus M'_{n'-t+i}||0..0,$$
$$n \oplus \text{Ctr}_{n+i} = n' \oplus \text{Ctr}'_{n'+1}$$

By transforming these equations we get

$$\begin{aligned} &\text{Ctr}_{t+i} \oplus \text{Ctr}'_{n'+i} = M_{n-t+i}||0..0 \oplus M'_{n'-t+i}||0..0, \\ &\text{Ctr}_{n+i} \oplus \text{Ctr}'_{n'+i} = n \oplus n', \\ \Rightarrow &d := \text{Ctr}_{t+i} \oplus \text{Ctr}'_{n'+i} = M_{n-t+i}||0..0 \oplus M'_{n'-t+i}||0..0 = n \oplus n'. \end{aligned}$$

It follows that difference d has to be greater or equal to 2^{p-r} because of $d = M_{n-t+i}||0..0 \oplus M'_{n'-t+i}||0..0$. By choosing d one gets the necessary variables n', $M'_{n'-t+i}$, and IV'. Let M_{T_i} be the message

$$\overline{M}_1, \ldots, \overline{M}_{n'-t+i-1}, M_{n-t+i} \oplus \text{MSB}_r(d), \overline{M}_{n'-t+i+1}, \ldots, \overline{M}_{n'}$$

with the desired block at the $(n' - t + i)$-th position and the remaining blocks set to arbitrary values. Hence we can get the tag T_i by the following request

$$\text{IAR-CTR}_K^{\text{IV}^* \oplus d}(M_{T_i}) = C'_1, \ldots, C'_{n'}, T'_1, \ldots, T_i, \ldots, T'_t$$

Observation 5. *With given p-bit integers n, c and r-bit block Z_0 the gadget $G^{\ell}_{IAR\text{-}CTR}$ returns $MSB_{\ell}(E_K(n \oplus c \oplus E_K(c \oplus Z_0||0..0)))$. This can be used to simulate a tag for IAR-CTR.*

We can now forge the desired message by the following procedure:

1. Use $G^{r}_{\text{IAR-CTR}}$ to obtain $A_1, \ldots A_t = \text{MSB}_r(E_K(\text{IV}^* + 1)), \ldots, \text{MSB}_r(E_K(\text{IV}^* + t))$.
2. Use $G^{r}_{\text{IAR-CTR}}$ to obtain $A_{t+1}, \ldots, A_n = \text{MSB}_r(E_K((\text{IV}^* + t + 1) \oplus M_1)), \ldots, \text{MSB}_r(E_K((\text{IV}^* + n) \oplus (M_{n-t})))$.
3. Compute $C_i = A_i \oplus M_i$.
4. Get the tags $T_1, \ldots, T_t$ by the above-mentioned request (1) if $n \neq t$. Otherwise use the gadget $G^{\ell}_{\text{IAR-CTR}}$ to obtain the tags.

Algorithm 4: Encryptions of IAR-CTR and IAR-CFB

IAR-CTR$_K^{\mathrm{IV}}(M_1, \ldots, M_n)$:
$\mathrm{Ctr}_0 \leftarrow \mathrm{IV}$
for $i = 1$ *to* t **do**
  $\mathrm{Ctr}_i \leftarrow \mathrm{Ctr}_{i-1} + 1$
  $O_i \leftarrow E_K(\mathrm{Ctr}_i)$
  $C_i \leftarrow M_i \oplus \mathrm{MSB}_r(O_i)$
for $i = t + 1$ *to* n **do**
  $\mathrm{Ctr}_i \leftarrow \mathrm{Ctr}_{i-1} + 1$
  $O_i \leftarrow E_K((M_{i-t}||0..0) \oplus \mathrm{Ctr}_i)$
  $C_i \leftarrow M_i \oplus \mathrm{MSB}_r(O_i)$
for $i = n + 1$ *to* $n + t$ **do**
  $\mathrm{Ctr}_i \leftarrow \mathrm{Ctr}_{i-1} + 1$
  $\tau_i \leftarrow E_K((M_{i-t}||0..0) \oplus \mathrm{Ctr}_i)$
  $O_i \leftarrow E_K(\mathrm{NOB} \oplus \tau_i \oplus \mathrm{Ctr}_{n+1})$
  $T_{i-n} \leftarrow \mathrm{MSB}_\ell(O_i)$
return $C_1, \ldots, C_n, T_1, \ldots, T_t$

IAR-CFB$_K^{\mathrm{IV}}(M_1, \ldots, M_n)$:
$\mathrm{Ctr}_0 \leftarrow \mathrm{IV}$
for $i = 1$ *to* t **do**
  $\mathrm{Ctr}_i \leftarrow \mathrm{Ctr}_{i-1} + 1$
  $O_i \leftarrow E_K(\mathrm{Ctr}_i)$
  $C_i \leftarrow M_i \oplus \mathrm{MSB}_r(O_i)$
$Y_i = \mathrm{Ctr}_i$
for $i = t + 1$ *to* n **do**
  $Y_i \leftarrow \mathrm{LSB}_{p-r}(Y_{i-1})||C_{i-t}$
  $O_i \leftarrow E_K((M_{i-t}||0..0) \oplus Y_i)$
  $C_i \leftarrow M_i \oplus \mathrm{MSB}_r(O_i)$
for $i = n + 1$ *to* $n + t$ **do**
  $Y_i \leftarrow \mathrm{LSB}_{p-r}(Y_{i-1})||C_{i-t}$
  $O_i \leftarrow E_K((M_{i-t}||0..0) \oplus Y_i)$
  $T_{i-n} \leftarrow \mathrm{MSB}_\ell(O_i)$
return $C_1, \ldots, C_n, T_1, \ldots, T_t$

5.8 Attack on IAR-CFB

In this section, we present a universal forgery attack on IAR-CFB for the case where $t \geq \lceil \frac{p}{r} \rceil$. This scenario is not only within the specified requirements for these parameters but also entirely practical (see e.g. the experiments in [12]), considering that if r is relatively small compared to p, the number of block cipher calls per message increases, reducing the efficiency of the scheme.

Assume we want to forge the encryption $C_1, \ldots, C_n, T_1, \ldots, T_t$ of the message $M = M_1, \ldots, M_n$ with initial value IV^*. Note that the input for the block cipher is not known immediately since the delayed ciphertext is used as part of the input. We create two gadgets $G^r_{\text{IAR-CFB}}$ and $G^\ell_{\text{IAR-CFB}}$ to simulate $\mathrm{MSB}_r(E(X))$ and $\mathrm{MSB}_\ell(E(X))$, respectively. The first gadget $G^r_{\text{IAR-CFB}}$ makes use of one of the first t encryptions. Let $1 \leq j \leq t$ be the desired index position. To obtain $\mathrm{MSB}_r(E_K(X))$ on the j-th position the gadget $G^r_{\text{IAR-CFB}}$ asks for

$$\text{IAR-CFB}_K^{X-j}(\overline{M}_1, \ldots, \overline{M}_t) = C'_1, \ldots, C'_t, T'_1, \ldots, T'_t$$

with an arbitrary $\overline{M}_i$. Since $C'_i = \mathrm{MSB}_r(E_K(X)) \oplus \overline{M}_i$ this gadget returns $G^r_{\text{IAR-CFB}}(X) := C'_j \oplus \overline{M}_j$. We do not fix one position j due to the freedom of initial vectors we can choose. This comes in quite handy for the next gadget.

Observation 6. *The gadget $G^r_{IAR\text{-}CFB}$ returns $MSB_r(E_K(X))$ for a given X.*

In the following the idea behind the second gadget $G^\ell_{\text{IAR-CFB}}$ is explained. We take advantage of the fact that the chained value Y_i (see Algorithm 4) which is used as the input for the ciphertext is updated by the delayed ciphertext blocks. Let $g = \lceil \frac{p}{r} \rceil$. After g blocks we have full control of this chained value.

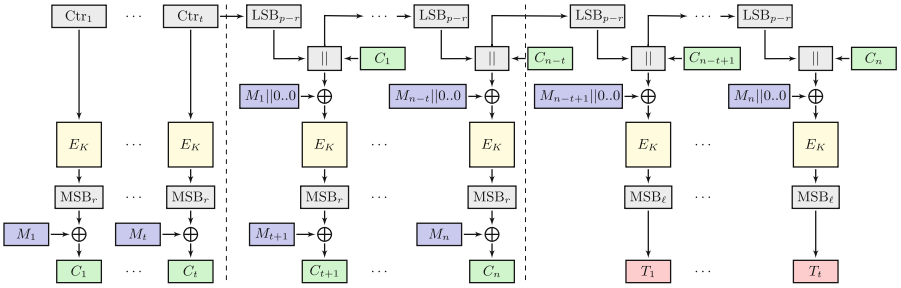


Fig. 6. The IAR-CFB authenticated encryption algorithm.

For the sake of simplicity we assume that $r|p$. The adjustment for the case $r \nmid p$ is described in Appendix A.3. Hence $g = \frac{p}{r}$. Furthermore, let X be the desired message. We split X into g equal sized parts $X_1, \ldots, X_g$. The gadget $G^{\ell}_{\text{IAR-CFB}}$ uses the g-th tag as the result of a t-block message. Before X is used as the input of E_K it will be xored with $M_g||0..0$. Thus, we have to set the ciphertexts as $X_1 = C_1 \oplus M_g$ and $X_i = C_i$ for $2 \leq i \leq g$. Due to $C_i = M_i \oplus \text{MSB}_r(E_K(\text{Ctr}_i))$ we know that

$$M_1 = X_1 \oplus M_g \oplus \text{MSB}_r(E_K(\text{Ctr}_1)), \tag{2}$$

$$M_i = X_i \oplus \text{MSB}_r(E_K(\text{Ctr}_i)). \tag{3}$$

By the following we obtain a procedure for computing the auxiliary message which has to be requested to obtain $\text{MSB}_\ell(E_K(X))$:

1. Find unused IV' such that all of $\{\text{Ctr}_1, \ldots, \text{Ctr}_g\}$ and $\text{IV}' - j$ for $1 \leq j < t$ are unused as the initial vector with $\text{Ctr}_i = \text{IV}' + i$.
2. Obtain $A_i = \text{MSB}_r(E_K(\text{Ctr}_i))$ by $G^r_{\text{IAR-CFB}}$ such that IV' is still unused for a request.
3. For $2 \leq i \leq g$ set $M'_i = X_i \oplus A_i$ (because of (3)).
4. Set $M'_1 = X_1 \oplus A_1 \oplus M'_g$ (because of (2)).

Now, to obtain $\text{MSB}_\ell(E_K(X))$ we ask for

$$\text{IAR-CFB}^{\text{IV}'}_K(M'_1, \ldots, M'_g, \overline{M}_{g+1}, \ldots, \overline{M}_t) = C'_1, \ldots, C'_t, T'_1, \ldots, T'_n$$

where $M'_1, \ldots, M'_g$ and IV' are obtained by the above procedure and the other blocks are arbitrary. Finally return $G^{\ell}_{\text{IAR-CFB}}(X) := T'_g = \text{MSB}_\ell(E_K(X))$.

Observation 7. *With $G^{\ell}_{IAR\text{-}CFB}$ one can get $MSB_\ell(E_K(X))$ without knowledge of the key by g many calls of $G^r_{IAR\text{-}CFB}$ and one oracle request.*

We now have all the required tools for our universal forgery attack:

1. Obtain $O_i = \text{MSB}_r(\text{IV}^* + i)$ by $G^r_{\text{IAR-CFB}}$ for $1 \leq i \leq t$.
2. Compute $C_i = M_i \oplus O_i$.

3. Let $Y_t = \mathrm{Ctr}_t$, for $t+1 \leq i \leq n$
   (a) Compute $Y_i = \mathrm{LSB}_{p-r}(Y_{i-1})||C_{i-t}$
   (b) Obtain $O_i = \mathrm{MSB}_r(M_{i-t}||0..0 \oplus Y_i)$ by $G^r_{\text{IAR-CFB}}$.
   (c) $C_i = M_i \oplus O_i$.
4. For $n+1 \leq i \leq n+t$
   (a) Compute $Y_i = \mathrm{LSB}_{p-r}(Y_{i-1})||C_{i-t}$
   (b) Obtain $T_{i-n} = \mathrm{MSB}_\ell(M_{i-t}||0..0 \oplus Y_i)$ by $G^\ell_{\text{IAR-CFB}}$.

The resulting ciphertext blocks and tags yield the desired forgery.

6 Conclusion

In this paper, we have analyzed the security of three AE algorithm families, namely PFX, PFC and IAR, which were designed as improvements to general-purpose well-established AE schemes such as CCM, GCM or OCB which are widely used in security protocols for wireless networks based on IEEE 802.11 (Wi-Fi), IEEE 802.15.4 (such as Zigbee), as well as LTE and 5G mobile networks. The design objective of PFX, PFC and IAR was to guarantee confidentiality and authenticity in a single-pass process while reducing the number of block cipher calls and avoiding expensive operations like finite field multiplications. As such, they appeared to be well-suited alternatives to standard modes such as CCM or GCM for wireless systems, lightweight wireless sensor networks, and real-time wireless applications.

Our analysis however indicates that these AE schemes cannot provide their claimed security guarantees. We described universal forgery attacks on all three algorithm families, allowing an adversary to compute valid ciphertexts and authentication tags for any message of their choice without knowledge of the secret key. All of our attacks only have linear complexity in the length of the target message and as such are entirely practical. Overall, our analysis implies that the affected schemes should not be used in practice, despite their attractive performance characteristics in the context of wireless and real-time networks.

It remains an interesting open problem to adapt existing well-established and secure cryptographic primitives for authenticated encryption more to the specific requirements of wireless network applications, especially in the context of lightweight wireless sensor nodes or real-time constraints. The forgery attacks on PFX, PFC and IAR illustrate the need for thorough and long-term security analysis of new cryptographic algorithms before considering their deployment, emphasizing the importance of adhering to well-established standardized cryptographic algorithms. For applications where standard solutions such as CCM or GCM are not ideal, a promising line of research would be to comparatively evaluate the NIST lightweight cryptography standard Ascon [7–9] as well as algorithms from the CAESAR final portfolio, which besides Ascon include ACORN [28] with a lighweight focus, AEGIS [29] and OCB for high-throughput networks, and Deoxys [18] and COLM [1,2] for scenarios where defense in depth against e.g. nonce misuse is required. These algorithms have already received extensive cryptanalytic scrutiny over a couple of years and could potentially be included in future versions of standards for wireless encryption.

Acknowledgements. We would like to thank the anonymous reviewers for their insightful comments.

A Appendix

A.1 Notation

In the following we briefly define some notation used in this article. Let M and C respectively denote the bit strings containing the plaintext message and its encryption in some mode by a block cipher E_K with the secret key K. Subscripts for M or C, like M_i, denote the i-th block of M or C respectively. The size of each block depends on the used block cipher (typically 64 or 128 bits). The operator $A \oplus B$ is the bitwise xor operation on two bit strings A and B. The output of a MAC is called tag and denoted by the variable T. The selection of the b most or b least significant bits of a bit string is written as $\mathrm{MSB}_b(\cdot)$ and $\mathrm{LSB}_b(\cdot)$, respectively. The total number of blocks of a message is referred to as NOB. The operator $||$ denotes concatenation of two bit strings. We use 0..0 to abbreviate the repetition of zeros up to a number (such as the block size) which is clear from the context.

A.2 PFC-OCB Scheme

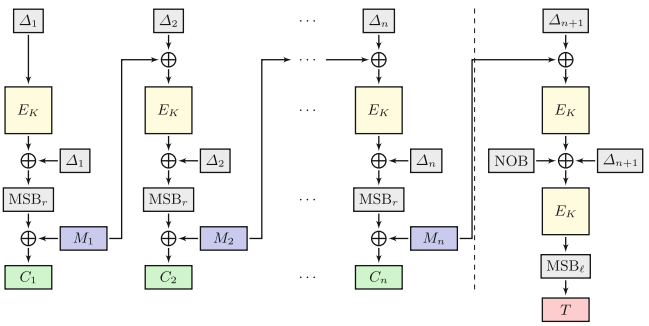


Fig. 7. The PFC-OCB authenticated encryption algorithm.

A.3 Adjustment for $G^{\ell}_{\text{IAR-CFB}}$ for the Case $r \nmid p$

In the following we explain how to adjust the gadget $G^{\ell}_{\text{IAR-CFB}}$ for the case $r \nmid p$. Let $g = \lceil \frac{p}{r} \rceil$ and let $q = p \mod r$. First, we consider the case for $g > 2$. Then, the case $g = 2$ will be discussed. We split the input X in the parts $X_1, \ldots, X_g$ where only X_1 consists of q bits and all other blocks of r bits. In this case, the

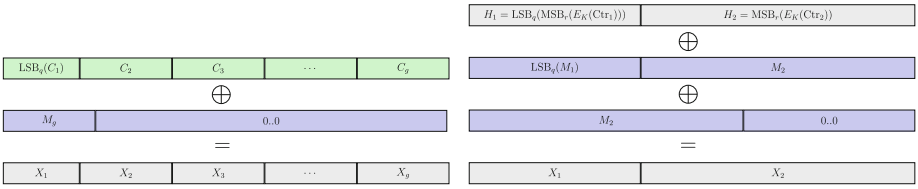
message block M_g influences X_1 and X_2, see Fig. 8a for a visualization. This yields the equations

$$X_1 = \mathrm{LSB}_q(C_1) \oplus \mathrm{MSB}_q(M_g), \tag{4}$$
$$X_2 = C_2 \oplus \mathrm{LSB}_{r-q}(M_g)||0..0, \tag{5}$$
$$X_i = C_i. \tag{6}$$

By combining these equations with $C_i = M_i \oplus \mathrm{MSB}_r(E_K(\mathrm{Ctr}_i))$ we obtain the following relations:

$$\mathrm{LSB}_q(M_1) = X_1 \oplus \mathrm{MSB}_q(M_g) \oplus \mathrm{LSB}_q(\mathrm{MSB}_r(E_K(\mathrm{Ctr}_1))), \tag{7}$$
$$M_2 = X_2 \oplus \mathrm{LSB}_{r-q}(M_g)||0..0 \oplus \mathrm{MSB}_r(E_K(\mathrm{Ctr}_2)), \tag{8}$$
$$M_i = X_i \oplus \mathrm{MSB}_r(E_K(\mathrm{Ctr}_i)), \tag{9}$$

from which we can replace the last two steps (3 and 4) of the message creation step of $G^{\ell}_{\text{IAR-CFB}}$ with three ones above.



(a) Case $g > 2$ (b) Case $g = 2$

Fig. 8. Illustration of the computation of X in the message creation part for $G_{\text{IAR-CFB}}$ in the case $r \nmid p$. Note that M_g is r bits long while X_1 and $\mathrm{LSB}_q(C_1)$ consist of $q = p \bmod r$ bits.

Now we consider the case $g = 2$. Similar to the above case we replace the formulas for the message creation part. Let $s := r - q$. See Fig. 8b for the dependencies in this case. Note that the ciphertexts C_1 and $\mathrm{LSB}_q(C_1)$ are replaced with $\mathrm{LSB}_q(M_1) \oplus \mathrm{LSB}_q(\mathrm{MSB}_r(E_K(\mathrm{Ctr}_1)))$ and $M_2 \oplus \mathrm{MSB}_r(E_K(\mathrm{Ctr}_2))$ respectively because of $C_i = M_i \oplus \mathrm{MSB}_r(E_K(\mathrm{Ctr}_i))$. For ease of presentation, let $H_1 := \mathrm{LSB}_q(\mathrm{MSB}_r(E_K(\mathrm{Ctr}_1)))$ and $H_2 := \mathrm{MSB}_r(E_K(\mathrm{Ctr}_2))$. For the s most significant bits of X_2 there is an "overlap" of two different parts of M_2 as seen in the equations:

$$\mathrm{LSB}_q(X_2) = \mathrm{LSB}_q(M_2) \oplus \mathrm{LSB}_q(H_2), \tag{10}$$
$$\mathrm{MSB}_s(X_2) = \mathrm{LSB}_s(M_2) \oplus \mathrm{MSB}_s(M_2) \oplus \mathrm{MSB}_s(H_2), \tag{11}$$
$$X_1 = \mathrm{MSB}_q(M_2) \oplus \mathrm{LSB}_q(M_1) \oplus H_1. \tag{12}$$

For this overlap we will define $\mathrm{MSB}_s(M_2)$ bitwise. Let $P[i]$ denote the i-th bit of P. For $q \le i < q + s$ we compute

$$M_2[i] := X_2[i] + M_2[i - q] + H_2[i]. \tag{13}$$

The q least significant bits of M_2 can be computed directly by (10). After the computation of M_2 the desired bits of M_1 are obtainable by (12). The adjustment is done by replacing the last 2 formulas in the auxiliary message creation part of $G^{\ell}_{\text{IAR-CFB}}$ by these ones for M_1 and M_2. This concludes the universal forgery for this case.

References

1. Andreeva, E., et al.: AES-COPA v2. Submission to the CAESAR competition (2016)
2. Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Tischhauser, E., Yasuda, K.: Parallelizable and authenticated online ciphers. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013. LNCS, vol. 8269, pp. 424–443. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-42033-7_22
3. Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Yu., Sim, S.M., Todo, Y.: GIFT: a small present. In: Fischer, W., Homma, N. (eds.) CHES 2017. LNCS, vol. 10529, pp. 321–345. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66787-4_16
4. Bellare, M., Namprempre, C.: Authenticated encryption: relations among notions and analysis of the generic composition paradigm. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 531–545. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44448-3_41
5. CAESAR Committee: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness (2019). http://competitions.cr.yp.to/caesar.html
6. Diffie, W., Hellman, M.E.: Privacy and authentication: an introduction to cryptography. Proc. IEEE **67**(3), 397–427 (1979)
7. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2: lightweight authenticated encryption and hashing. J. Cryptol. **34**(3), 33 (2021)
8. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to the CAESAR competition (2016)
9. Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to the NIST Lightweight Cryptography competition (2019)
10. Dworkin, M.J.: SP 800-38D. Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC. Technical report, Gaithersburg, MD, United States (2007)
11. 3rd Generation Partnership Project (3GPP): Security architecture and procedures for 5G System. 3GPP TS 33.501 (2018)
12. Hwang, T., Gope, P.: IAR-CTR and IAR-CFB: integrity aware real-time based counter and cipher feedback modes. Secur. Commun. Netw. **8**(18), 3939–3952 (2015)
13. Hwang, T., Gope, P.: PFX: an essence of authencryption for block-cipher security. Secur. Commun. Netw. **9**(10), 1186–1197 (2016)
14. Hwang, T., Gope, P.: Robust stream-cipher mode of authenticated encryption for secure communication in wireless sensor network. Secur. Commun. Netw. **9**(7), 667–679 (2016)
15. IEEE: IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks–Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC)

and Physical Layer (PHY) Specifications. IEEE Std 802.11-2020/Cor 1-2022 (Corrigendum to IEEE Std 802.11-2020 as amended by IEEE Std 802.11ax-2021, IEEE Std 802.11ay-2021, and IEEE Std 802.11ba-2021), pp. 1–18 (2022)
16. IEEE: IEEE Standard for Low-Rate Wireless Networks. IEEE Std 802.15.4-2020/Cor 1-2022 (Corrigendum to IEEE Std 802.15.4-2020 as amended by IEEE Std 802.15.4z-2020, IEEE Std 802.15.4w-2020, IEEE Std 802.15.4y-2021, and IEEE Std 802.15.4aa-2022), pp. 1–22 (2023)
17. ISO 19772:2009. Information technology – Security techniques – Authenticated encryption (2009)
18. Jean, J., Nikolic, I., Peyrin, T., Seurin, Y.: The deoxys AEAD family. J. Cryptol. **34**(3), 31 (2021)
19. Jutla, C.S.: Encryption modes with almost free message integrity. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 529–544. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44987-6_32
20. Kohno, T., Viega, J., Whiting, D.: CWC: a high-performance conventional authenticated encryption mode. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 408–426. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-25937-4_26
21. Krovetz, T., Rogaway, P.: The software performance of authenticated-encryption modes. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 306–327. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21702-9_18
22. Krovetz, T., Rogaway, P.: The OCB authenticated-encryption algorithm. RFC 7253 (2014). https://www.rfc-editor.org/info/rfc7253
23. McGrew, D.A., Viega, J.: The security and performance of the Galois/Counter Mode (GCM) of operation. In: Canteaut, A., Viswanathan, K. (eds.) INDOCRYPT 2004. LNCS, vol. 3348, pp. 343–355. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30556-9_27
24. Namprempre, C.: Secure channels based on authenticated encryption schemes: a simple characterization. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 515–532. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-36178-2_32
25. NIST: DES modes of operation. FIPS PUB 81 (1980)
26. Rogaway, P.: Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 16–31. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30539-2_2
27. Rogaway, P., Bellare, M., Black, J., Krovetz, T.: OCB: a block-cipher mode of operation for efficient authenticated encryption. In: ACM Conference on Computer and Communications Security, pp. 196–205 (2001)
28. Wu, H.: Acorn: a lightweight authenticated cipher (v3). Submission to the CAESAR competition (2016)
29. Wu, H., Preneel, B.: Aegis: a fast authenticated encryption algorithm (v1.1). Submission to the CAESAR competition (2016)