An Analysis of Parallelizable Authenticated Encryption

Kazuhiko Minematsu
NEC Corporation
Joint work with Akiko Inoue

Asian Symmetric-key Workshop 2018 Nov 14 Kolkata, India

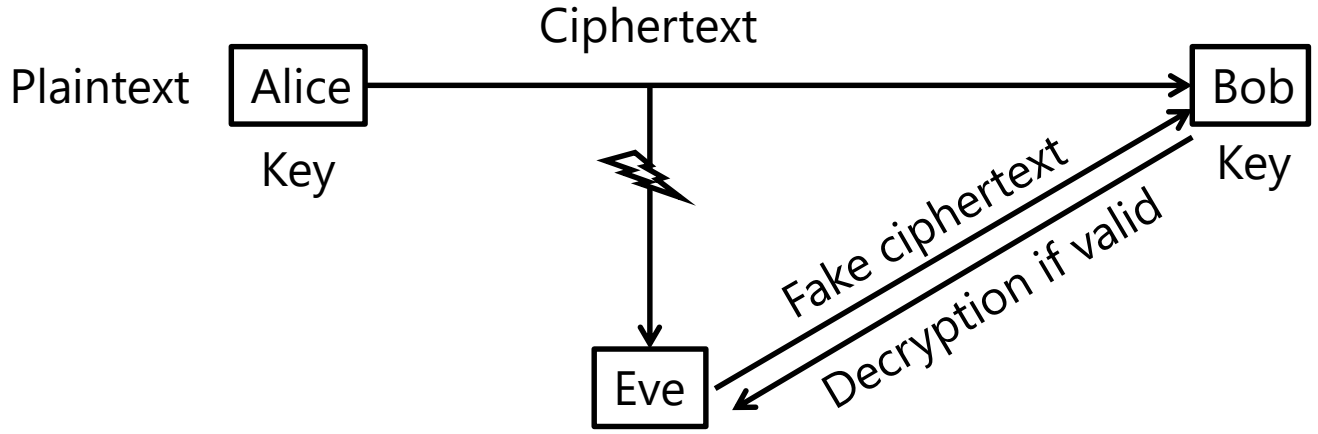
Cryptanalysis of OCB2

(ePrint 2018/1040)

Kazuhiko Minematsu
NEC Corporation
Joint work with Akiko Inoue

Authenticated Encryption (AE)

- Symmetric-key encryption for confidentiality and integrity
- Fundamental research area in symmetric-key cryptography
- Increasing adoption in the real world:
  - Internet (TLS), SSH, Wifi, IoT, ..



Some of Popular Schemes

- NIST recommendations : GCM and CCM
- ISO standards (ISO/IEC 19772) : 6 schemes
- Internet standards (RFC) : GCM, CBC+HMAC, OCB, ChaCha20Poly1305,...
- CAESAR

OCB (Offset CodeBook)

- Celebrated AE scheme
- Blockcipher mode
- Strong features:
  - Rate-1 operation (AE as fast as enc-only mode)
  - Parallelizablility
  - Provable security

OCB Versions

- OCB1 (ACM CCS 2001) by Rogaway et al. [RBBK01]
- OCB2 (Asiacrypt 2004) by Rogaway [Rog04]
- OCB3 (FSE 2011) By Krovetz and Rogaway [KR11]
- Each received significant attentions:
  - OCB1 considered for IEEE 802.11 (Wifi)
  - OCB2 for ISO/IEC 19772
  - OCB3 for RFC 7253
- OCB3 is in CAESAR finalists

[RBBK01] Rogaway, Bellare, Black, Krovetz : OCB: A block-cipher mode of operation for efficient authenticated encryption. ACM CCS 2001
[Rog04] Rogaway : Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. ASIACRYPT 2004
[KR11] Krovetz, Rogaway : The Software Performance of Authenticated-Encryption Modes. FSE 2011

Security of (all versions of) OCB

- Provable security : if AES is a strong pseudorandom permutation (SPRP), OCB-AES is secure
  - (Converse) if OCB-AES is insecure, AES should be broken
- Birthday Bounds : assuming AES=SPRP, $2^{n/2}$ message blocks are needed to break OCB

Security of (all versions of) OCB

- Extensive third-party analyses :
  - Ferguson [Fer02], Sun et al. [SWZ12] : birthday attacks. bounds are tight
  - Andreeva et. al [ABLMMY14], Ashur et. al [ADL17] : attacks under misuse scenarios (e.g. nonce repeated at encryptions)
  - Aoki and Yasuda [AY13] : Relaxing SPRP assumption for a modified OCB
  - Ritam and Nandi [BN17] : Improving the bound for OCB3
- Strong belief on the security of OCB

[Fer02] Ferguson. Collision attacks on OCB. Comments to NIST.
[ABLMMY14] Andreeva, Bogdanov, Luykx, Mennink, Mouha, and Yasuda. How to Securely Release Unverified Plaintext in Authenticated Encryption. Asiacrypt 2014.
[ADL17] Ashur, Dunkelman, Luykx. Boosting Authenticated Encryption Robustness with Minimal Modifications. CRYPTO 2017.
[AY13] Aoki and Yasuda. The Security of the OCB Mode of Operation without the SPRP Assumption. ProvSec 2013.
[BN17] Bhaumik and Nandi. Improved Security for OCB3. Asiacrypt 2017.
[SWZ12] Sun, Wang, Zhang. Collision Attacks on Variant of OCB Mode and Its Series. Inscrypt 2012.

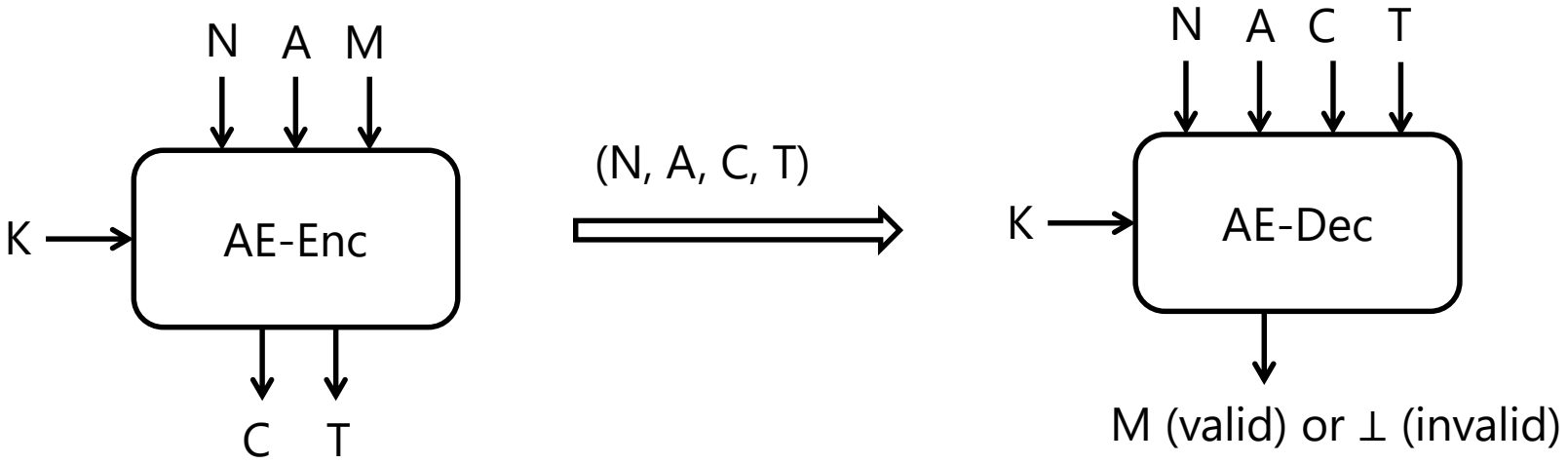
… wait !!

What we found

- Structural forgery attacks against OCB2
  - independent of the underlying blockcipher
- Simple and practical : one encryption query, then forgery
  - (Almost) known plaintext
  - Existential forgery
  - (Almost) universal forgery after the first forgery
    - Reforging attacks
- Verified by the reference code written by Krovetz

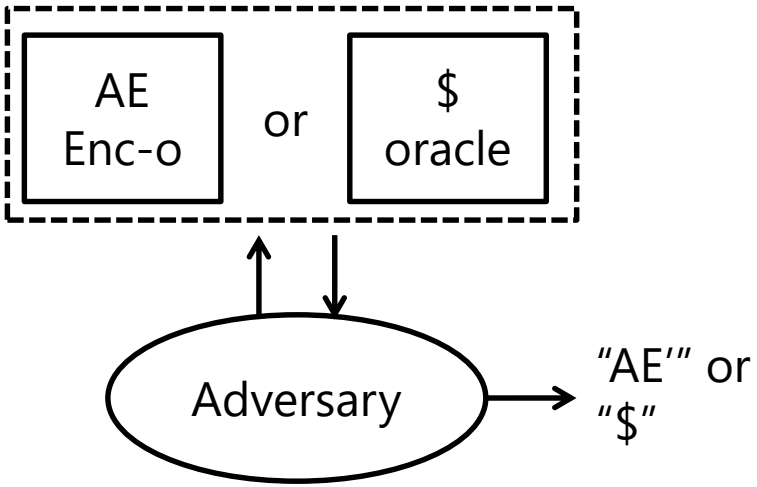
Syntax of AE

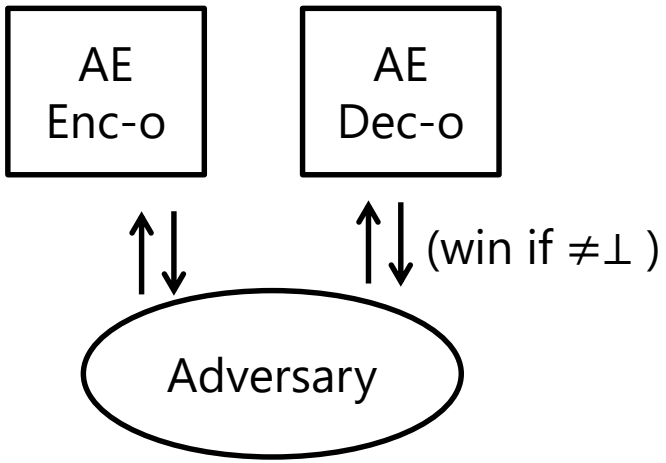
- Six variables: Key (K), Nonce (N), AD (A), Plaintext (M), Ciphertext (C), and Tag (T)
- AE.Enc (N,A,M) -> (C,T)
- AE.Dec (N,A,C,T) -> M if valid, ⊥ if invalid



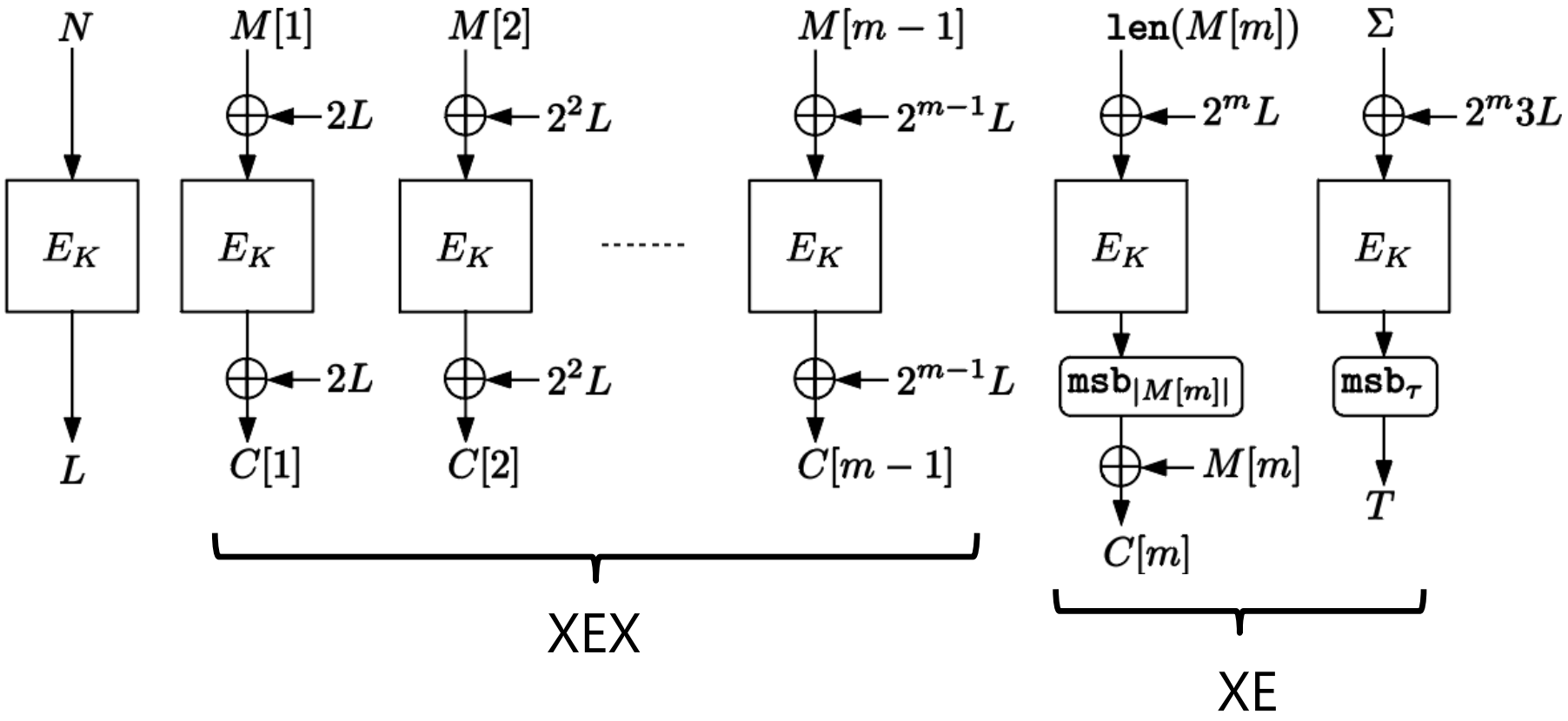
Two security notions

- Privacy (PRIV) : distinguish ciphertexts from random using encryption queries
- Authenticity (AUTH) : create a forgery using encryption and decryption queries
- Nonce-respecting adversary





OCB2



- 2L and 3L : GF(2^n) multiplication by **x** and **x+1**
- When |M[m]| =n , Checksum Σ = M[1] + … +M[m-1] + M[m]
- When AD is present, take PMAC(AD) and XOR it to T

Minimal Attack

- Suppose AD is always empty

1. First, encrypt (N, M) :

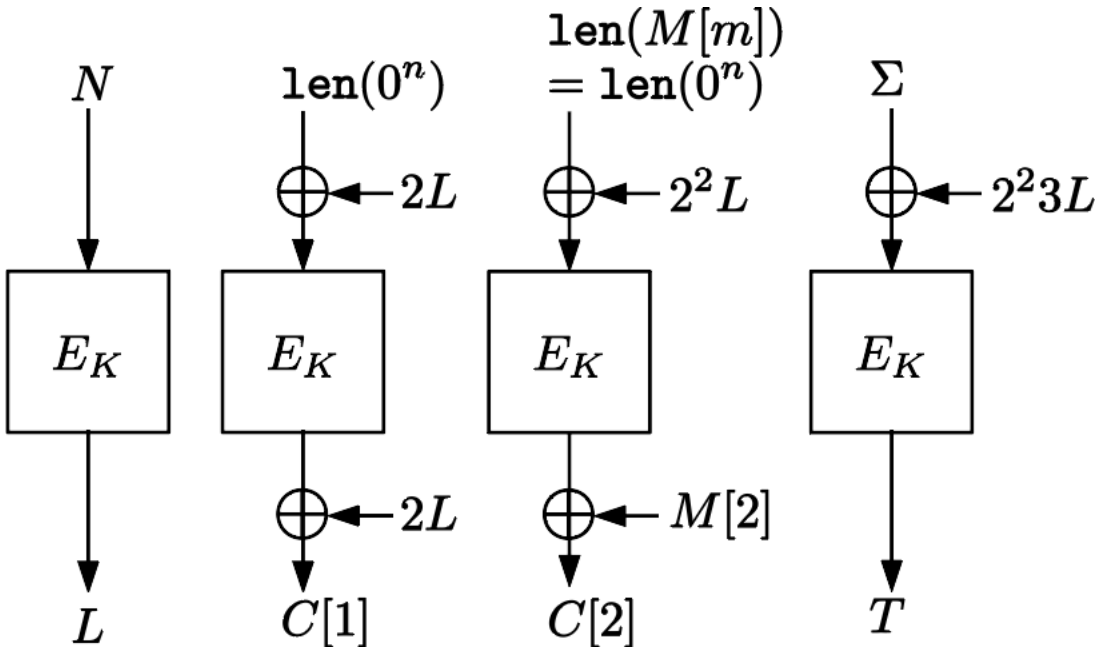
- $M = len(0^n) \| M[2]$ for any n-bit M[2]
- Get (C =C[1]C[2] ,T)

2. Decrypt (N, C' , T') s.t.

- $C' = C[1] + len(0^n)$, $T' = M[2] + C[2]$
- Done !

Why ?

- Encryption query :

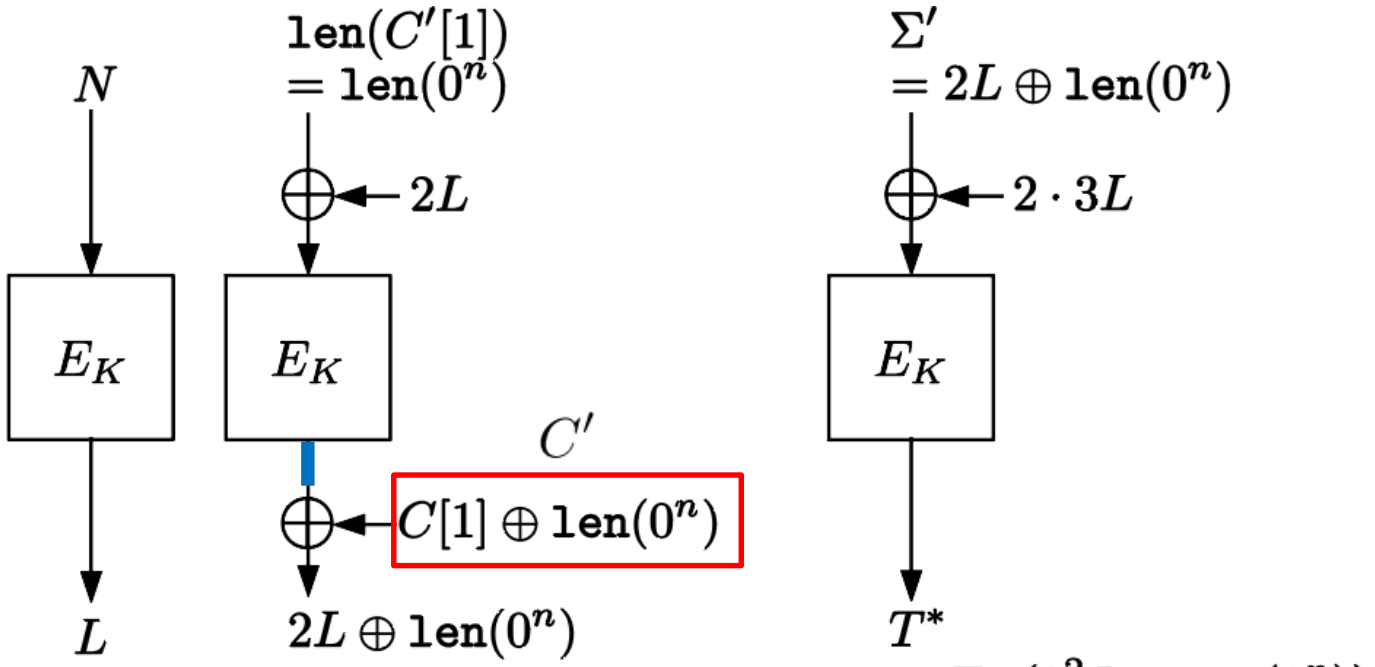


$$C[1] = 2L \oplus E(2L \oplus \mathtt{len}(0^n))$$

$$M[2] \oplus C[2] = E(2^2 L \oplus \mathtt{len}(0^n))$$

Why ?

- Decryption query :
- $3(2L) + 2L = 2^2L + 2L + 2L = 2^2L$



$$C[1] = 2L \oplus E(2L \oplus \mathtt{len}(0^n))$$

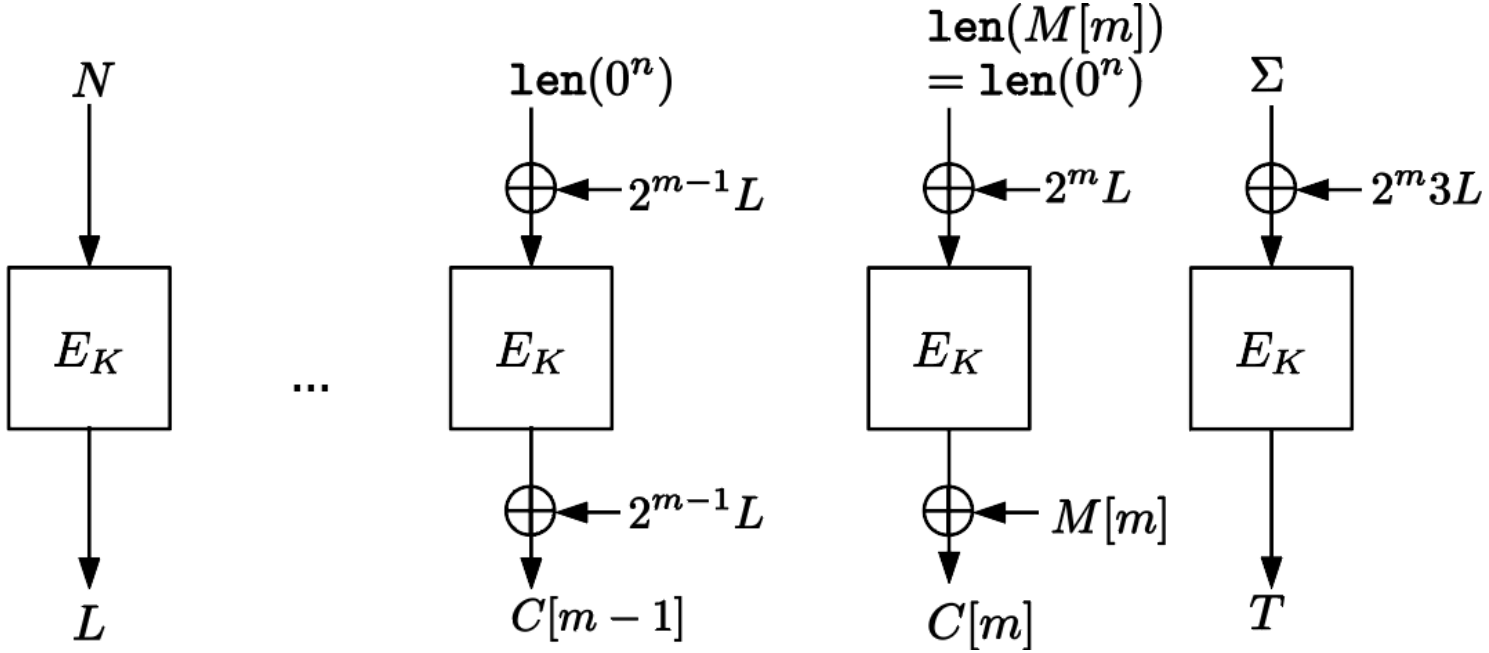
$$M[2] \oplus C[2] = E(2^2L \oplus \mathtt{len}(0^n))$$

Remarks

- AD in the first query may be arbitrary (we do not need the info of T)
- Works independent of the spec of PMAC
- Works independent of the spec of len(*)
- Tag may be truncated (success prob = 1 for any length)

Extension : Longer messages

- Just make the last block and tag follow the minimal attack
- Query m-block M, get m-block C, truncate by one block and modify the last block and tag to form C′



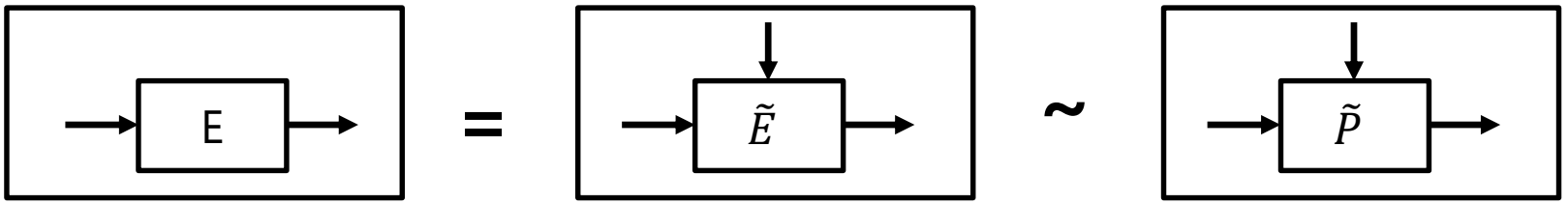
Extension : Universal Forgery

- Dec oracle in the minimal attack reveals L = E(N)
  - $M' = 2L + \text{len}(0^n)$, thus (N,L) is I/O pair of E !
  - Once L is known, other I/O pairs are known as well
- Enables raw access to E
- Can be used to implement universal forgery
  - We choose (N, A, M)
  - We query some (but not querying (N,A,M)), incl. forgeries
  - and build a forgery (N, A, C, T) which results in M
- See ePrint for details

The flaw in the proof of [Rog04]

- How come ? Proof existed ?
- A flaw in the hybrid argument :

1. Observe OCB2 is a mode of XEX*
   - i.e. OCB2[E] = ΘCB2[XEX*[E]]
2. Prove XEX* (combination of XE & XEX) is a secure TBC
3. Prove ΘCB2[$\tilde{P}$] is a secure AE



OCB2[E] ΘCB2[$\tilde{E}$] ΘCB2[$\tilde{P}$]

XEX*

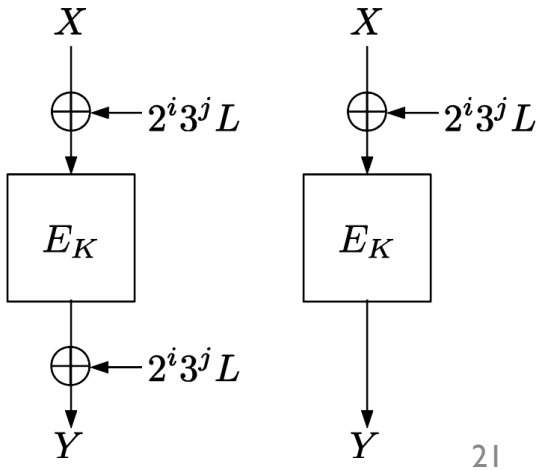
- Combination of XE and XEX
- Not a usual TBC : tagged (additional bit “b” in a tweak)
  - b = 0 means the use of XE
  - b = 1 means the use of XEX

Tweak = (b, N, i, j)

$$\mathrm{XEX}_E^{N,i,j}(X) \stackrel{\mathrm{def}}{=} E(2^i L \oplus X) \oplus 2^i L,$$

$$\mathrm{XE}_E^{N,i,j}(X) \stackrel{\mathrm{def}}{=} E(2^i 3^j L \oplus X),$$

$$\mathrm{XEX}_E^{*,b,N,i,j}(X) = \begin{cases} \mathrm{XEX}_E^{N,i,j}(X) & \text{if } b = 1; \\ \mathrm{XE}_E^{N,i,j}(X) & \text{if } b = 0. \end{cases}$$



Security of Tagged TBC

- Adversary must follow the following access rules :
- 1. No decryption query when b = 0
- 2. Once query (b, T), then never query (1-b, T) for any T = (N, i, j)

- Tag-respecting adversary
- [Rog04] : XEX* is a secure tagged TBC against tag-respecting adversary

Incorrect Hybrid

- Hybrid argument of OCB2 :

$$\mathbf{Adv}^{\mathsf{priv}}_{\mathrm{OCB2_P}}(\mathcal{A}) = \mathbf{Adv}^{\mathsf{priv}}_{\Theta\mathrm{CB2}_{\mathrm{XEX}^*_{\mathsf{P}}}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{tprp}}_{\mathrm{XEX}^*_{\mathsf{P}}}(\mathcal{B}) + \mathbf{Adv}^{\mathsf{priv}}_{\Theta\mathrm{CB2}_{\tilde{\mathsf{P}}}}(\mathcal{A}),$$

$$\mathbf{Adv}^{\mathsf{auth}}_{\mathrm{OCB2_P}}(\mathcal{A}_{\pm}) = \mathbf{Adv}^{\mathsf{auth}}_{\Theta\mathrm{CB2}_{\mathrm{XEX}^*_{\mathsf{P}}}}(\mathcal{A}_{\pm}) \leq \mathbf{Adv}^{\mathsf{tsprp}}_{\mathrm{XEX}^*_{\mathsf{P}}}(\mathcal{B}_{\pm}) + \mathbf{Adv}^{\mathsf{auth}}_{\Theta\mathrm{CB2}_{\tilde{\mathsf{P}}}}(\mathcal{A}_{\pm})$$

- Adv B and B± are tag-respecting and simulating A and A±
- In [Rog04] this part is briefly touched:
  - " It is here that one has formalized the intuition that the first m − 1 tweakable-blockcipher calls to OCB1 need to be CCA-secure but the last two calls need only be CPA-secure. "

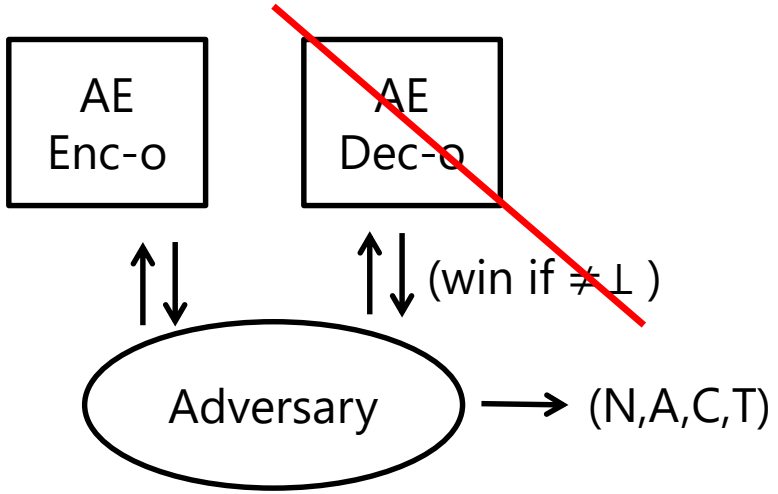
Incorrect Hybrid

- Recall the minimal attack:
- Q1. encrypt (N,M) with |M|=2n, receive (C, T)
- Q2. decrypt (N, C′, T′) with |C′|=n

- Then, the oracle invokes
  - $XEX^{*,1,N,1,0}$, $XEX^{*,0,N,2,0}$, $XEX^{*,0,N,2,1}$ for Q1
  - $XEX^{*,0,N,1,0}$, $XEX^{*,0,N,1,1}$ for Q2
- Violation of the second access rule !
- We cannot simulate w/o violating what XEX* permits

AUTH notion at [Rog04]

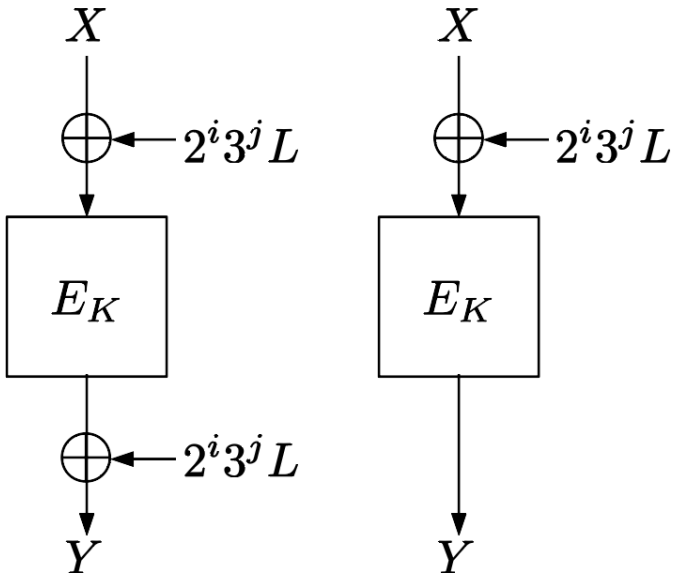
- Authenticity notion in [Rog04] :
  - queries to enc oracle, then submit a decryption query
- Never invokes decryption oracle
  - Win or lose is determined outside the game
  - needs to change to invoke dec oracle inside the game so that hybrid works
    - $\mathrm{Adv}^{\mathrm{auth}}_{F}(A) - \mathrm{Adv}^{\mathrm{auth}}_{G}(A) = \mathrm{Adv}^{\mathrm{IND}}_{F,G}(A)$ when $\mathrm{Adv}^{\mathrm{auth}}_{F}(A) = \Pr[A^{F} => 1]$ for some game of A querying F
- Not capture the multiple decryption queries
  - generic upper bound [BGK94] from single decryption query may result in a weaker bound (e.g. see the case of EAX [MLI13])



[BGM94] Bellare, Goldreich, Mityagin. The Power of Verification Queries in Message Authentication and Authenticated Encryption. ePrint 2004/309.
[MLI13] Minematsu, Luck, Iwata. Improved Authenticity Bound of EAX, and Refinements. ProvSec 2013.

Effect to Related Schemes

- Roughly, attacks are possible when
- 1. the last block encryption uses XE
- 2. The mask of that XE may be used another query in the form of XEX



Effect to Related Schemes

- OCB1 and OCB3 : safe from our attacks
- OCB1 : the last block is XE, but separated
  - Different mask generation from XEX
- OCB3 : the last block is either XEX (when full) or XE (when partial), and XE mask is separated from XEX masks

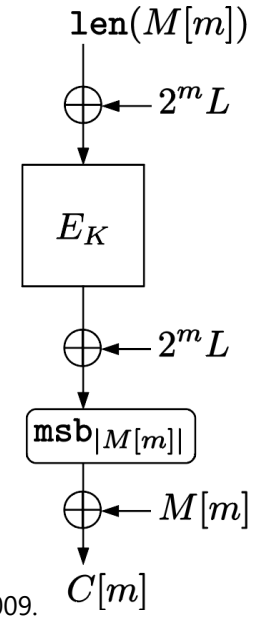
Effect to Related Schemes

- OTR [Min14] : removing inverse from OCB, two-round Feistel with XE-based round function. Totally XE-only
- OPP [GJMN16] : Permutation-based OCB, wider block w/ fast mask generation. Always XEX (or a variant of XPX [Men14])
- Other OCB(2)-related schemes seem to safe from our attacks

[Min14] Minematsu. Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. Eurocrypt 2014.
[GJMN16] Granger, Jovanovic, Mennink, Neves. Improved Masking for Tweakable Blockciphers with Applications to Authenticated Encryption. Eurocrypt 2016.

Fix OCB2

- We just need to use XEX for the last block (which we call OCB2f)
  - hybrid argument then works fine
- Or, use a variant of OCB using a variant of XEX* called XEX+ [MM08]
  - Raw E call instead of XE, input separation
- Or, OCB3 of course !



[MM08] Minematsu and Matsushima. Generalization and Extension of XEX* Mode. IEICE-A 2009.

Follow-up

- We see quick follow-ups on ePrint:
- Poettering (2018/1087) [Pot18] and Iwata (2018/1090) [Iwa18]
  - Presented Privacy attack under IND-CCA and Universal forgery (Poettering)
  - Plaintext recovery (Poettering and Iwata)
- Interesting concurrent work …

[Pot18] Poettering. Breaking the confidentiality of OCB2. ePrint 2018/1087
[Iwa18] Iwata. Plaintext recovery attack of OCB2. ePrint 2018/1090

Summary

- OCB2 is totally broken
- Practical forgery is possible
- A small flaw in the proof lead us to surprisingly powerful attacks
- Not applicable to the general structure of OCB

Lessens learned:

- Proof quality is important
- Active third-party verification is even more important

Thanks !