Key Committing Security of AEZ and More

Yu Long Chen, Antonio Flórez-Gutiérrez, Akiko Inoue, Ryoma Ito, Tetsu Iwata, Kazuhiko Minematsu, Nicky Mouha, Yusuke Naito, Ferdinand Sibleyras, Yosuke Todo
Data Confidentiality and Authenticity

- **Data confidentiality**
  - No outsider can learn anything about data
- **Data authenticity**
  - No outsider can manipulate data
Authenticated Encryption

\[ \text{Dec outputs } M = \text{Dec}(N, A, C, T) \in \{0, 1\}^{|C|} \text{ if } T \text{ is correct and } \bot \text{ otherwise} \]

We require that \( \text{Dec}_K(N, A, \text{Enc}_K(A, M)) = M \)
Key Committing Security

• Example: Security as proposed by Bellare and Hoang @Euro22

• Probability that an attacker can find two inputs of AE that have the same ciphertext (including tags)
  • CMT-1: different keys
  • CMT-3: different (K, N, A) pairs
  • CMT-4: different (K, N, A, M) pairs
  • CMT-3 = CMT-4 has been proven by BH22
Encode-then-Encipher via Wide-Block Cipher

• First encode the message (for example append with zeros), then apply WBC for enciphering

• Analyzing key committing security against EtE
  • WBC itself is not an AE and we need to specify where to insert $0^\tau$

• In this work, we focus on
  • AEZ: appending is specified
  • Adiantum: prepend and append with zeros
  • HCTR2: prepend and append with zeros
AEZ

- EtE using 128-bit TBC
  - Zero string concatenation at the end of plain text
  - Length of zero string is an arbitrary byte, not considered to exceed 128 bits
  - Input length = plaintext length + $\tau$ = ciphertext length
- Input length 256 bits or more: AEZ-core (**this work!**)
- Input length less than 256 bits: AEZ-tiny
  - Feistel with a minimum of 8 rounds
  - Number of steps varies depending on input length
Key Committing Security of AEZ

• $O(1)$ CMT-4 attack against general AEZ
• CMT-1 attacks
  • $\tau = n$: birthday complexity $O(2^{n/2})$
  • $\tau < n$: attack based different algorithms
  • Tightness of attack against general AEZ -> Provable security result for $\tau = n$, assuming the primitives are ideal
Collision-Finding for CMT-1 Attacks Against AEZ-Core
CMT-1 Attack Complexities Against gAEZ

- attack based on 4-tree algorithm, repeated 4-tree algorithm, and birthday attack
Differential Propagation in CMT-1 Attack Against Full-Spec AEZ

Underlying TBC follows the full specification of AEZ-core (full-spec AEZ)
- Choose distinct keys \((K, K')\) -> the difference in certain intermediate states becomes 0
- CMT-1 attack against full-spec AEZ with complexity \(O(2^{27})\)
- A numerical example of CMT-1 attack
EtE-Adiantum

- WBC designed by Crowley and Biggers [CB18]
- Widely deployed in practice as a disk sector encryption scheme on Android devices
- NH [BHK+99] and Poly1305 [Ber05], AES-256, and XChaCha12
- Results:
  - $O(1)$ CMT-4 attack against both prepending and appending cases
  - CMT-1 attack with birthday complexity
    - $O(2^{n/2})$ for appending case
    - $O(2^{\tau/2})$ for prepending case
  - Tightness of attack against prepending case -> provable security result assuming cryptographic permutation inside XChaCha12 is ideal
  - Using s-way collision probability of permutation-based Davies-Meyer
Collision-Finding for CMT-1 Attacks Against Adiantum
XChacha’s Block Function

- Input: key $K$ (256 bits), nonce = $(n_1, n_2)$ (128,64 bits)
- Output: $Y$ (512 bits)
- $Init = (\text{const (128)} || K (256) || n_1 (128))$
- $P =$ Chacha permutation (20 rounds)
- $\text{HChacha}(Init) = \text{tr}_256(P(Init) + Init)$
  - “+” is 32-bit word-wise modular addition (16 additions)
  - “tr_256” concatenates the first and the last 128 bits
- $Init' = (\text{const (128)} || L (256) || 0^{32} || 0^{32} || n_2 (64))$

EtE-HCTR2

- WBC designed by Crowley, Huckleberry, and Biggers [CHB21]
- Based on HCTR [WFW05]
- Polynomial hash function, AES, and XCTR mode of stream encryption
- Results:
  - $O(1)$ CMT-4 attack against both prepending and appending cases
  - CMT-1 attack with birthday complexity
    - $O(2^\tau / 2)$ for appending case
    - $O(2^n / 2)$ for prepending case
Collision-Finding for CMT-1 Attacks Against HCTR2
## Summarization

<table>
<thead>
<tr>
<th>Scheme</th>
<th>CMT-1 A</th>
<th>CMT-1 P</th>
<th>CMT-4 (A &amp; P)</th>
<th>Proof</th>
</tr>
</thead>
<tbody>
<tr>
<td>general AEZ</td>
<td>$O(2^{n/2})$</td>
<td>(not specified)</td>
<td>$O(1)$</td>
<td>$n/2$ (Sect. 7.1)</td>
</tr>
<tr>
<td>full-spec AEZ</td>
<td>$O(2^{27})$</td>
<td>(not specified)</td>
<td>$O(1)$</td>
<td>—</td>
</tr>
<tr>
<td>EtE-Adiantum</td>
<td>$O(2^{n/2})$</td>
<td>$O(2^{n/2})$</td>
<td>$O(1)$</td>
<td>$n/2$ (Sect. 7.2)</td>
</tr>
<tr>
<td>EtE-HCTR2</td>
<td>$O(2^{n/2})$</td>
<td>$O(2^{n/2})$</td>
<td>$O(1)$</td>
<td>—</td>
</tr>
</tbody>
</table>
Authenticated Enciphering

• Tim Beyne, Yu Long Chen, and Wonseok Choi
• An alternative definition for authenticated encryption
• Follows the work of Mihir Bellare, Phillip Rogaway: *Encode-Then-Encipher Encryption: How to Exploit Nonces or Redundancy in Plaintexts for Efficient Cryptography*. (AC2000)
• Key observation: nonce/tag pair has the same relation as the message/ciphertext pair