iFeed: the Input-Feed AE Modes

Liting Zhang
Joint work with Wenling Wu, Han Sui and Peng Wang
TCA @ ISCAS
ASK 2013 @ SDUW
2013.08.28
Outline

• Review of AEs
• Basic iFeed Construction
• iFeed AE Modes
• Wrap Up
Basic iFeed Construction

parallel encryption          serial decryption

• \( V \) is an extra value
• Inputs to P should be pairwise distinct & SECRET
Basic iFeed Construction

- In encryption

- Privacy for $X_1$ and $X_2$
- Authenticity for $X_1$ and $X_2$

Closely combine Privacy and Authenticity
Basic iFeed Construction

• In encryption

• For incomplete messages
  – pad the last plaintext block  --- online
  – truncate the last ciphertext block
Basic iFeed Construction

• In decryption

• Offline --- start with the last block

• Authentication at last
iFeed Basic in a Mirror

iFeed Basic

Mirrored iFeed Basic
Mirrored iFeed Basic

parallel encryption  serial decryption

• **Online** decryption --- start with $C_1$
Mirrored iFeed Basic

• In encryption

• For incomplete messages
  – pad the first plaintext block \( \rightarrow \) offline
  – truncate the first ciphertext block
Summary of iFeed Basic

• One-pass
• closely combine Priv and Auth
• inverse-free
  – PRP not SPRP on P
  – We can replace P with compression function CF or tweakable blockcipher TBC
• Parallel encryption, but serial decryption
Summary of iFeed Basic

<table>
<thead>
<tr>
<th></th>
<th>Online encryption</th>
<th>Online decryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>iFeed Basic</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Mirrored iFeed Basic</td>
<td>no</td>
<td>yes</td>
</tr>
</tbody>
</table>

• Online/offline encryption affects little
  – The sender knows the plaintext lengths --- usually has full messages in hand
Summary of iFeed Basic

<table>
<thead>
<tr>
<th></th>
<th>Online encryption</th>
<th>Online decryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>iFeed Basic</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Mirrored iFeed Basic</td>
<td>no</td>
<td>yes</td>
</tr>
</tbody>
</table>

- Online/offline decryption is
  - Important --- decrypting on-the-fly
  - Offline can be solved --- if the sender sends from the last ciphertext block
Outline

• Review of AEs
• Basic iFeed Construction
• iFeed AE Modes
• Wrap Up
Applying iFeed Basic

• Keep the inputs to P, CF and TBC pairwise distinct & SECRET
  – Generating secret masks XORed to the inputs to P
  – Carefully formatting the inputs to CF or TBC

• Process associated data
  – Introducing a MAC
The iFeed AE Mode

- A PMAC-like MAC processing $A = AD || PMN$
- CF is a compression function
The iFeed AE Mode

- Mirrored iFeed Basic to process SMN=S
- S can have any length here
The iFeed AE Mode

- Mirrored iFeed Basic to process Message=M
- M can have any length
The iFeed AE Mode Encryption

- **Input**
  - Key K
  - A=AD || PMN
  - S=SMN
  - M=Message

- **Output**
  - CS, CM, T_c
The iFeed AE Decryption

• Input
  – Key K
  – A=AD||PMN
  – CS, CM, TC

• Output
  – (S, M) or ⊥
Compression Function CF

- $|\text{Sum}| = k\geq n$ bits
- $|\text{num}| = a$ bits
- $|M_i| = n$ bits

For each $K$, CF can process at most $\text{MIN}\{2^a, 2^{n/2}\}$ blocks, including AD, PMN, SMN, and M.
<table>
<thead>
<tr>
<th>category</th>
<th>Hash function</th>
<th>Input length $L_1+L_2$</th>
<th>$L_1$ (message)</th>
<th>$L_2$</th>
<th>Output length $L_H$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO/IEC bc-based</td>
<td>Hash-function 1</td>
<td>2n</td>
<td>$n$</td>
<td>$n$</td>
<td>$\leq n$</td>
</tr>
<tr>
<td>ISO/IEC bc-based</td>
<td>Hash-function 2</td>
<td>3n</td>
<td>$n$</td>
<td>2$n$</td>
<td>$\leq 2n$</td>
</tr>
<tr>
<td>ISO/IEC bc-based</td>
<td>Hash-function 3</td>
<td>12$n$</td>
<td>4$n$</td>
<td>8$n$</td>
<td>2$n$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>Hash-function 4</td>
<td>12$n$</td>
<td>3$n$</td>
<td>9$n$</td>
<td>3$n$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>RIPEMD-160</td>
<td>672</td>
<td>512</td>
<td>160</td>
<td>$\leq 160$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>RIPEMD-128</td>
<td>640</td>
<td>512</td>
<td>128</td>
<td>$\leq 128$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>SHA-1</td>
<td>512</td>
<td>512</td>
<td>160</td>
<td>$\leq 160$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>SHA-256</td>
<td>512</td>
<td>512</td>
<td>256</td>
<td>$\leq 256$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>SHA-224</td>
<td>512</td>
<td>512</td>
<td>256</td>
<td>224</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>SHA-384</td>
<td>1024</td>
<td>512</td>
<td>512</td>
<td>384</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>WHIRLPOOL</td>
<td>512</td>
<td>512</td>
<td>512</td>
<td>$\leq 512$</td>
</tr>
<tr>
<td>ISO/IEC dedicated</td>
<td>SM3</td>
<td>512</td>
<td>256</td>
<td>256</td>
<td></td>
</tr>
</tbody>
</table>
Summary of iFeed[CF]

- Depending on nonce (PMN, SMN)
- Provably secure with $O(L^2q^2/2^n)$
- In the ideal model
- Avoiding generating many masks
- Supporting any-length AD, PMN, and SMN

- Parallel encryption, but serial decryption
Variants

• iFeed[BC] and iFeed[TBC]
  – Secure in the standard model
  – Needing to generate many masks, like OCB[1,2,3]
    • Gray code
    • Finite field multiplication
    • LFSR

• iFeed[CF, BC, TBC ] with Mirrored iFeed Basic
Outline

• Review of AEs
• Basic iFeed Construction
• iFeed AE Modes
• Wrap Up
Wrap Up

• Too many criteria restrict the design of AE
  – **Security** - model, provable, tight bounds,
  – **Efficiency** - key size, rate, parallelizability, memory occupation, HW occupation, SW/HW speed, ...
  – **Usability** - nonce (PMN, SMN), associated data, online, one-pass, inverse-free, patent, ...

• Many AEs have been designed or being under design

• We introduce a new method to combine Privacy and Authenticity --- iFeed
Thanks

Q & A

Special thanks to Lei Wang for his insightful observations.