MORUS
A Fast Authenticated Cipher

Hongjun Wu    Tao Huang

Nanyang Technological University
Different Design Approaches:

- Fast
  - AES-NI (AEGIS)
  - SIMD (MORUS)

- Lightweight
  - Mode (JAMBU)
  - Dedicated (ACORN)
MORUS: Main features

• Fast
  • MORUS-1280 is **0.69 clock cycles/byte** on Haswell
  • The only CAESAR candidate which is not based on AES-NI, but faster than AES-GCM on Haswell
  • Likely the fastest CAESAR candidate on the processors with SIMD (SSE2, AVX2) instructions, but no AES-NI

• Nonce-based
Design software-efficient ciphers

Two general methods for designing software efficient ciphers

• Method 1:
  • reduce the number of computations in the design

• Method 2:
  • use operations that can be computed fast on CPUs
Design software-efficient ciphers

• Method 1:
  • reduce the number of computations in the design

• How to design a cipher using Method 1?
  • Method 1a: Proper security margin (should not be too large)
    (the designers should/can analyze the security of the ciphers)
  • Method 1b: Typically stream cipher requires much less operations than block cipher (anyway, block cipher more robust than stream cipher)
  • Method 1c: Efficient design
    Example: how to achieve high security with less operations ....
Design software-efficient ciphers

• Method 2:
  • use operations that can be computed fast on CPUs

• Some efficient operations on CPUs:
  • SIMD (single instruction multiple data)
    • SSE2: 128-bit registers (available on many CPUs)
    • AVX2: 256-bit registers (available on the latest Intel Haswell CPUs)
  • AES-NI (AES new instruction set)
    • The design of AEGIS
Design software-efficient ciphers

- Design efficient ciphers using SIMD instructions
  - Salsa (stream cipher using 128-bit SSE2 instructions, 2005)
  - Blake, JH (hash functions using 128-bit SSE2 instructions, 2008)
  - MORUS, NORX (authenticated ciphers using 256-bit AVX2 instructions, 2014)
MORUS: Design

• MORUS is a fast software cipher
  • Encryption: stream cipher
  • Authentication: almost for free
  • Use SIMD instructions
MORUS: Design

- MORUS-1280-128: 1280-bit state, 128-bit key
- MORUS-1280-256: 1280-bit state, 256-bit key
- MORUS-640-128: 640-bit state, 128-bit key

- Tag: 128-bit

- MORUS-640 benefits from 128-bit SIMD instructions
- MORUS-1280 benefits from 128/256-bit SIMD instructions
MORUS: Design

• MORUS-1280 (128-bit/256-bit key):
  The cipher state consists of five 256-bit words
  XOR, AND, SHIFT operations are used

• In each step,
  5 rounds are used to update the state, and
  256-bit keystream word is generated
MORUS: Design

• MORUS-640 (128-bit key):
  The cipher state consists of five 128-bit words
• Each step consists of 5 rounds, and
  128-bit keystream word is generated
The state update of MORUS in one step: 5 rounds
MORUS: Design

• Initialization
  • 16 steps
  • key XORed to the state at the end of the initialization

• Finalization
  • 8 steps
  • Part of secret state and length (ad, message) are used to update the state in finalization
  • Generate 128-bit tag from the state
MORUS: Security

• We analyzed differentials involving the low weight input differences
  • The probability of state collision is much less than $2^{-128}$ (it is tremendously difficult to eliminate the difference in the state)
• The high weight input differences likely lead to even lower probability of state collision
MORUS: Performance

Speed on Haswell

<table>
<thead>
<tr>
<th></th>
<th>16B</th>
<th>64B</th>
<th>512B</th>
<th>1024B</th>
<th>4096B</th>
<th>16384B</th>
</tr>
</thead>
<tbody>
<tr>
<td>MORUS-640(EA)</td>
<td>28</td>
<td>7.72</td>
<td>1.95</td>
<td>1.58</td>
<td>1.18</td>
<td>1.11</td>
</tr>
<tr>
<td>MORUS-640(DV)</td>
<td>28</td>
<td>7.99</td>
<td>1.97</td>
<td>1.56</td>
<td>1.23</td>
<td>1.16</td>
</tr>
<tr>
<td>MORUS-1280(EA)</td>
<td>33.9</td>
<td>8.28</td>
<td>1.59</td>
<td>1.12</td>
<td>0.78</td>
<td><strong>0.69</strong></td>
</tr>
<tr>
<td>MORUS-1280(DV)</td>
<td>35.8</td>
<td>8.46</td>
<td>1.63</td>
<td>1.13</td>
<td>0.80</td>
<td><strong>0.69</strong></td>
</tr>
</tbody>
</table>

MORUS-640 is slower since it uses only 128-bit SIMD

AES-128-GCM: 1.03
MORUS: Performance

• MORUS is likely the fastest on the platforms with SIMD but no AES-NI
  • Reason 1: MORUS benefits from SIMD
  • Reason 2: We carefully removed the redundant operations in the cipher
MORUS: Performance

• MORUS is expected to be very fast on hardware
  • Critical path is very short in each step (8XOR, 3AND)
  • 128-bit (256-bit) keystream is generated for MORUS-640 (MORUS-1280)
Conclusion

• MORUS benefits from SIMD
• Likely the fastest candidate on platforms with SIMD but no AES-NI