An Overview of CAESAR

Mridul Nandi
Indian Statistical Institute, Kolkata

SEPTEMBER 2016
1 Introduction

2 CAESAR Competition

3 TriviA: A Streamcipher Based AE Scheme

4 Hardware Implementation of TriviA

5 ELmD: A Blockcipher Based AE Scheme
A Brief Overview

A proper integration of Encryption and Authentication

First Formalized by Bellare and Namprempre [Asiacrypt 00]
- Proposed $EtM$ (used in IPSec), $MtE$ (used in SSL/TLS) and $E&M$ (used in SSH).
- Proposed formal security model for Privacy and Authenticity
- $EtM$ strongest in this security model

Other Important Works
- AE proposed by Jutla, Gligor et al. (XCBC and XECB), Rogaway et al. (OCB)
- Later CCM, EAX (improved CCM), EAX’ (Update over EAX) and GCM. GCM was recommended by NIST (SP 800-38D)
1 Introduction

2 CAESAR Competition
   - Classification of CAESAR Candidates by Structure

3 TriviA: A Streamcipher Based AE Scheme

4 Hardware Implementation of TriviA

5 ELmD: A Blockcipher Based AE Scheme
CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness

Announced in 2013
- Offer advantages over AES-GCM
- Suitable for widespread adoption

Functional requirements
- The algorithm receives $PMN, SMN$ (optional), $AD$ and $M$
- The algorithm outputs $C$ and $T$
- Privacy for $M$ and $SMN$, Authenticity for $PMN, SMN, AD$ and $M$
First Round of CAESAR

57 Submissions (March, 2014)

Classification by Primitives of Important Submissions

- BC-Based - CLOC, SILC, ELMd, OTR, COPA, Joltik, OCB
- SC-Based - TriviA, Acorn, AEGIS
- Sponge Based - Ascon, PRIMATEs
Filter of First Round Candidates

Elemination of Several Candidates

- 28 candidates are eliminated (some are withdrawn)
- Some were broken. Some were inefficient

Some Important Cryptanalysis

- Forgery of COBRA, POET, PAES, LAC
- Cryptanalysis of XLS constructions
- Forgery and Key recovery of Marble
- Forgery of iFEED in both standard model and INT-RUP model
- Forgery and state recovery of PANDA
- INT-RUP Forgery of AES-CPFB
Second Round of CAESAR

- 29 Submissions (July, 2015)

Several Attack After Second Round Announcements

- Key Recovery of 2.5 Round Pi-Cipher
- Forgery of ICEPOLE
- INT-RUP Attack on Rate-1 BC based AE (OCB, iFEED)
- Fault Attack on PAEQ, PRIMATEs, Minalpher, CLOC-SILC
Introduction

CAESAR Competition

TriviA : A Streamcipher Based AE Scheme

Hardware Implementation of TriviA

ELmD : A Blockcipher Based AE Scheme

Classification of CAESAR Candidates by Structure

Third Round of CAESAR

- 15 Submissions (August, 2016)

Structural Classification

- OTP Mode
  - Counter Mode - N/A
  - Streamcipher Mode - N/A
  - Sequential Feedback Mode without Counter
    - Sponge Mode - Ascon, Ketje, Keyak, NORX, Tiaoxin
    - NON-Sponge Mode - ACORN, AEGIS, AES-JAMBU, CLOC-SILC, MORUS

- OCB Mode - Deoxys, OCB, OTR

- Encrypt-Mix-Encrypt Mode - COLM

- Hash-Counter Mode - AEZ
Structural Classification of all CAESAR Candidates

- OTP Mode
  - Counter Mode
  - Streamcipher Mode
  - Sequential Feedback Mode without Counter
    - Sponge Mode
    - NON-Sponge Mode
- OCB Mode
- Encrypt-Mix-Encrypt Mode
- Hash-Counter Mode
Uses counter value for encryption of each block
- Encryption of different blocks can be parallel or the $M/C$ block can be sequentially fed
- May or may not be online
- iFEED, AES-CPFB, PAEQ, Pi-Cipher, OMD

Figure: Counter Mode
OTP Mode: Streamcipher Mode

- Uses expander function (such as streamcipher)
- Takes the state, updates state and generate random value
- This random value XORed with $M$ to generate $C$
- TriviA, Wheesht, Sablier, Raviyoyla

Figure: Streamcipher Mode
OTP Mode

3. Sequential Feedback Mode without Counter

- Similar to Streamcipher mode
- Except, the state also contains the previously processed $M$ or previously generated $C$
- Two types
  - Sponge Mode
  - Non-Sponge Mode
- Ascon, ICEPOLE, PRIMATEs are Sponge Modes
- ACorrn, CLOC-SILC, MORUS are Non-Sponge Modes
- $d$-block delay online security
Sponge and NON-Sponge Constructions

**Figure:** Sponge Constructions

**Figure:** Non-Sponge Constructions
OCB or Tweakable Blockcipher Mode

- ECB like structure
- Nonce can not be misused
- AES-OCB, AES-OTR

Figure: OCB
Encrypt-Mix-Encrypt Mode

- Encryption module between two collision resistant online hash functions
- 0-Block delay Online
- ElmD, COPA, Marble, KIASU

![Diagram of Encrypt-Mix-Encrypt Mode]

**Figure: ElmD**
Hash-Counter Mode (2-pass construction)

- Whole $M$ is Hashed generate the tag and an IV
- IV is used in counter mode to generate $C$
- Not Online
- SIV, BTM, AEZ

**Figure:** Hash-Counter Mode
1 Introduction

2 CAESAR Competition

3 TriviA : A Streamcipher Based AE Scheme

4 Hardware Implementation of TriviA

5 ELmD : A Blockcipher Based AE Scheme
TriviA Encryption Mode

- Joint work with Avik Chakraborti
- CAESAR candidate, Accepted at CHES 2015 and JCEN 2016

- **TriviA-SC** - Updated version of Trivium.
- **EHC-Hash** - Universal Hash follows EHC technique.
- **TriviA-SC** generates encryption and authentication key stream.
Circuit of TriviA-SC
TriviA-SC Informations

- **384-bit state** - A (132-bit), B (105-bit) and C (147-bit)
- Loaded with **128-bit key** and **128-bit nonce**.
- **1152-round initialization**.
- **64-bit parallelism**
- **Nonlinearity** in the output
- **KeyExt64** - From output, **StExt64** - From state
Circuit of EHC Hash

- $D/M$ and $D'/M'$
- 32-bit Multiplier
- MUX
- $K$ and $K'$
- 32-bit Multiplier
- VHorner$_{32/5}$
- VHorner$_{64/4}$
- 256-bit
- 64-bit
- 16-bit
- 160-bit

Authenticated Encryption
Introduction

CAESAR Competition

TriviA : A Streamcipher Based AE Scheme
Hardware Implementation of TriviA
ELmD : A Blockcipher Based AE Scheme

EHC-Hash Informations

- Used underlying Fields - $F_{2^{32}}(\alpha)$ and $F_{2^{64}}(\beta)$

- **Expand / Encode-Hash-Combine**
  - **Expand** by ECCodex $d$ (VHorner$_{64/d}$)
  - **Blockwise Hash** by PDP-Hash (32-bit Multiplier)
  - **Combine** by VMult$_{\alpha,d}$ (VHorner$_{32/d+1}$)

- One 32-bit Multiplication for 64-bit block.

- EHC$^{(d,l)}$ is $2^{-128}$ Universal hash (Not $\Delta$-U)
- EHC$^{(d,l)} \oplus K$, $K$ uniform is $2^{-31d}$-pairwise independent.
TriviA: A Streamcipher Based AE Scheme

Hardware Implementation of TriviA

ELmD: A Blockcipher Based AE Scheme

Authenticated Encryption
Informations on TriviA

- **Arbitrary** length $M$ (padded with $10^*$) divided into Blocks

- Block Size $w$ - 64-bit

- Intermediate tag (if any) - Computed after each $ck$ blocks.
  - $ck = 0$ for this Paper (no intermediate tag).
  - $ck \in \{0, 128\}$ for CAESAR submission.

- $|C| = |M|$

- Size of each of the tags - 128-bit
Security Level for TriviA

<table>
<thead>
<tr>
<th>Version</th>
<th>Confidentiality</th>
<th>Authenticity</th>
</tr>
</thead>
<tbody>
<tr>
<td>TriviA-0</td>
<td>128</td>
<td>124</td>
</tr>
<tr>
<td>TriviA-128</td>
<td>128</td>
<td>124</td>
</tr>
</tbody>
</table>
Important Properties of TriviA

- Options for Intermediate Tag.

- **TriviA-SC** - Updated design of a well studied and efficient (both in hardware and software) stream cipher Trivium.

- High security level- 128-bits for confidentiality and 124-bits for Authenticity of plaintext.

- High speed hardware.
1. Introduction

2. CAESAR Competition

3. TriviA: A Streamcipher Based AE Scheme

4. Hardware Implementation of TriviA

5. ELmD: A Blockcipher Based AE Scheme
TriviA-Base Architecture

TriviA-Base

Authenticated Encryption
TriviA-Base Architecture Properties

- No *pipelined* register
- *Parallel* processing of data
- 64-bits/ cycle
- **Long Critical path**: $(2 \times 1) 64$-bit MUX $\rightarrow$ 64-bit XOR $\rightarrow$ 32-bit Mult $\rightarrow$ Tag Updation $\rightarrow$ $(3 \times 1) 160$-bit MUX
- Reduced Speed, Throughput
- Controller *FSM* has 3-bit Register
TriviA-Pipelined Architecture
TriviA-Pipelined Architecture Properties

- 2 operations in series
  - 32-bit multiplication
  - Tag updation

- 3 stage pipeline
  - Increased throughput
  - Increased frequency
  - Extra cycles required

- Controller FSM has 3-bit Register
TriviA ASIC Implementation

- Verilog HDL, Synopsys Design Compiler J-2014.09
- Technology node: UMC 65nm logic SP/RVT Low-K process

Base Implementation
- Area: 23.6 KGE
- Frequency: 1150 MHZ, Throughput: 73.9 Gbps

Pipelined Implementation
- Area: 24.4 KGE
- Frequency: 1425 MHZ, Throughput: 91.2 Gbps
## TriviA ASIC Results Comparison

<table>
<thead>
<tr>
<th>Schemes</th>
<th>KGE</th>
<th>Gbps</th>
<th>Mbps/GE</th>
<th>cpb</th>
</tr>
</thead>
<tbody>
<tr>
<td>TriviA-Base</td>
<td>23.6</td>
<td>73.9</td>
<td>3.13</td>
<td>0.12</td>
</tr>
<tr>
<td>TriviA-Pipelined</td>
<td>24.4</td>
<td>91.2</td>
<td>3.73</td>
<td>0.12</td>
</tr>
<tr>
<td>Scream, iScream</td>
<td>17.29</td>
<td>5.19</td>
<td>0.30</td>
<td>-</td>
</tr>
<tr>
<td>NORX</td>
<td>62</td>
<td>28.2</td>
<td>0.45</td>
<td>-</td>
</tr>
<tr>
<td>Ascon</td>
<td>7.95</td>
<td>7.77</td>
<td>0.98</td>
<td>0.75</td>
</tr>
<tr>
<td>AEGIS-AO1</td>
<td>20.55</td>
<td>1.35</td>
<td>0.07</td>
<td>6.67</td>
</tr>
<tr>
<td>AEGIS-AO2</td>
<td>60.88</td>
<td>37.44</td>
<td>0.61</td>
<td>0.33</td>
</tr>
<tr>
<td>AEGIS-TO1</td>
<td>88.91</td>
<td>53.55</td>
<td>0.60</td>
<td>0.20</td>
</tr>
<tr>
<td>AEGIS-TO2</td>
<td>172.72</td>
<td>121.07</td>
<td>0.70</td>
<td>0.07</td>
</tr>
</tbody>
</table>
TriviA FPGA Results

- Xilinx ISE 14.7
- Default settings, no optimizations
- Pre-layout synthesis
- 5.4x better than AES – CCM
ELmD Mode to encrypt a message of length $l$

Message Padding

$M_{l+1} = M_l = \sum_{i=1}^{l-1} M_i + (M_i^* \| 10^*)$.
Description of $\rho$ function

$W' = X + 2W$

$Y = X + 3W$

$Y_j = X_j + 3X_{j-1} + \ldots + 2^{j-2}.3X_1 + 2^{j-1}.3.\text{IV}$
Generation of IV

\[ D[0] = N \]
\[ 3.2^d. L \]
\[ Z[0] \]
\[ 0 \]
\[ \rho \]
\[ 3.2. L \]
\[ Z[1] \]
\[ \rho \]
\[ 3^d. L \]
\[ Z[d] \]
\[ \rho \]
\[ \cdots \]
\[ IV \]

AD Processing

- Identical to message processing.
- Parallel and Fully Pipeline Implementable.
Security of ELmD

Online Privacy

\[ \text{Adv}_{\text{ELmE}}^{\text{opriv}}(q, \sigma_{\text{priv}}, t) \leq \text{Adv}_{E}^{\text{prp}}(\sigma_{\text{priv}}, \sigma_{\text{priv}}, t') + \frac{5\sigma_{\text{priv}}^2}{2^n} \]

Authenticity

\[ \text{Adv}_{\text{ELmE}}^{\text{forge}}(q, \sigma_{\text{auth}}, t) \leq \text{Adv}_{E}^{\text{prp}}(\sigma_{\text{auth}}, \sigma_{\text{auth}}, t') + \frac{9\sigma_{\text{auth}}^2}{2^n} + \frac{s}{2^n} \]

\( \sigma_{\text{priv}} \) := total no. of blocks of \( q \) forward queries.
\( \sigma_{\text{auth}} \) := total no. of blocks of \( q \) forward and \( s \) forging queries.
Claim 1

$X_j$ is fresh unless $M_j = M_j''$. 
Online Privacy of ELmD

Claim 2

\[ Y_j^i = X_j^i + 3.X_{j-1}^i + \ldots + 2^{i-2}.3.X_1^i + 2^{i-1}.3.IV^i \] is fresh unless

\[ M_{1..j}^i = M_{1..j}''^i. \]
Online Privacy of ELmD

Claim 3
\[ \forall i, \ Y_{i+1}^i \text{ is fresh.} \]
Main Claim

For any forged ciphertext $C^f_{1..l}$, $Y^f_{l+1}$ is fresh.
Low End Device Compatibility

**Online**

$i^{th}$ ciphertext block depends on first $i$ plaintext blocks.

**Nonce Misuse Resistant**

Provides Online PRP Security even if nonce is repeated.

**Problem of Limited buffer**

- Use intermediate tags to stop releasing unverified plaintext.
- Generate intermediate tags after each $k < 128$ blocks:

  $$IT_i = E_K^{-1}(W_{k.i}) + \Delta$$
Design Rationale

**EME type Structure**

EME with linear mixing obtain online PRP Security and fully pipeline implementable.

**Decryption in Lower Level**

Identical Enc-Dec structure: combined hardware implementation area is minimized.

**ρ mixing**

- Easy Intermediate tag generation.
- Plain xor mixing require intermediate tag generation through checksum and more buffer required.
Identical Enc-Dec structure for ELmD.

Hardware Implementation

Combined hardware implementation area is minimized.
ELmD: Robustness

Use as Online Encryption/Decryption only Scheme

- Set associated data as empty and \( IV = 1 \).
- Return \( C_1 \ldots \).

Use as MAC only

- Set Associated data empty and \( IV = 1 \).
- Return \( (M, T = C_{l+1}) \).

Use to check integrity of associated data only

- Set message as empty and \( M_1 = 0 \).
- Return \( (D, T = C_1) \).
Thank you