Misusing Misuse-Resistance in APE

Dhiman Saha$^1$, Sukhendu Kuila$^2$, Dipanwita Roy Chowdhury$^1$

$^1$Dept. Of Computer Science & Engineering, IIT Kharagpur, INDIA
$^2$Dept. Of Mathematics, Vidyasagar University, INDIA

DIAC 2014, Santa Barbara, USA
Nonce-based Encryption

- Formalized by Rogaway

- Primary Condition
  - *Uniqueness* of the nonce in every instantiation of the cipher

- Interesting Consequence
  - Automatic protection from Differential Fault Analysis (DFA)

- DFA assumption
  - Ability to induce faults in the intermediate state of the cipher while replaying the encryption with the same plaintext.

  - No longer holds due to introduction of nonce
Misuse-Resistance

- A desirable property for authenticated ciphers.
- Avoids maintaining a nonce-generator
- Suited for resource-constrained environments
- Addressed in CAESAR selection portfolio

However, there is some collateral damage.
  - Nonce assumption no longer holds
  - Opens up the ciphers for DFA

This work explores this idea to mount efficient DFA on misuse-resistant AE scheme APE
Authenticated Permutation-based Encryption – APE
- Introduced first in FSE 2014
- First misuse-resistant permutation-based AE scheme
- Inspired from SPONGE
- Targeted for lightweight environments
- Basically a mode of operation
- Can be instantiated with permutations of hashes like SPONGENT/QUARK/PHOTON

Reintroduced in CAESAR
- Along with HANUMAN & GIBBON
- Part of PRIMATEs family of authenticated ciphers
- Now with new indigenous permutation called PRIMATE
The PRIMATE Permutation

- Internal permutation for APE/HANUMAN/GIBBON
  - Inspired from FIDES authenticated cipher
  - Structurally follows AES round function

- Has two variants
  - PRIMATE-80/120
    - Internal state realized as \((5 \times 8) / (7 \times 8)\) five-bit elements

- Component Transformations
  - SubBytes
  - ShiftRows
  - MixColumns
  - Round constant addition
PRIMATE-APE

- N[·] – Nonce block
- A[·] – Associated data block
- M[·] – message block
- K – Key (160 bit for APE-80)
- The IVs are predefined and vary according to the nature of the length of message and associated data.
- This work uses APE-80 (can be extended to APE-120)
Misusing Misuse-Resistance

- Concept of faulty collisions:
  - Not a real collision
  - Attacker induces a fault in the state of the cipher so that two different plaintexts produce the same tag.

- Idea: To find faulty collisions
  - Feasible due to misuse-resistance
    - Observation: APE is misuse-resistant up to a common prefix.

- Common prefix implication:
  - Plaintexts can be of the following form:
    - $M_1 = x_0 \ || \ x_1 \ || \ x_2 \ || \ ... \ || \ x_i \ || \ ... \ || \ x_w$
    - $M_2 = x_0 \ || \ x_1 \ || \ x_2 \ || \ ... \ || \ x'_i \ || \ ... \ || \ x_w$
A Faulty Collision

- Exploits: Misuse-resistance + Online nature
  - Induce random word fault in \((i-1)\)th ciphertext output
  - Observe faulty \((i-1)\)th output & manipulate \(i\)th message input

\[
\begin{align*}
\text{Plaintext1} & = M[1] \mid | M[2] \mid | \ldots \mid | M[i] \mid | M[i+1] \mid | \ldots \mid | M[w] \\
\text{Ciphertext1} & = C[1] \mid | C[2] \mid | \ldots \mid | C[i] \mid | C[i+1] \mid | \ldots \mid | C[w] \\
\text{Tag} & = T
\end{align*}
\]
Implications of a Faulty Collision

- Ability to replay the encryption

- Recall
  - This is one of the fundamental requirements to mount differential fault analysis attacks

- Next, we explore the prospect of DFA in the presence of faulty collisions

- Fault model assumed is random word fault
  - Recall: word in case of APE is a 5-bit vector
Fault Induction

- Fault induced at the input of 10\textsuperscript{th} round of the final iteration of APE
- Next study the fault diffusion in the differential state in the remaining rounds
Fault Diffusion

- Observe: Exactly 3 specific unaffected columns at the start of $r^{th}$ round due to diagonal word fault at the start of $(r-2)^{th}$ round.
- Helps to identify fault source diagonal by observing differential state
- Exploits the non-square nature of state matrix
Diagonal Fault Analysis

- Advanced differential fault attack
  - Introduced in 2009, specially suited for AES-like constructions
  - Has been highlighted in the book Fault Analysis in Cryptography as one of the most efficient DFA on AES
    - Exploits equivalence of fault induced in the same diagonal of the state matrix

- Can be applied on APE
  - But not directly
  - Last round MixColumn inclusion - major deviation from AES
  - Makes classical diagonal attack inefficient
  - Need some adaptation
    - Focus on recovering the state instead of the key
The Fault Invariant

- The diagonal principle:
  - *Equivalence of faults limited to a diagonal*
- The relation matrix is governed by MixColumns
Diagonal Fault Analysis of APE

- Inbound phase
  - Invert the differential state (computed from correct and faulty output) to reach up to state after last round SubBytes.
  - Use unaffected columns to identify source fault diagonal and load appropriate relation matrix
  - Solve equations involving fault invariant to generate hyper-state
  - Hyper-State is a special structure where every element is a set of candidates computed after equation solving
    - Helps capture the notion of candidate states for the correct state
EscApe (contd.)

- The Outbound phase
  - Apply ShiftRows to Hyper-state
  - Compute Kernel (Refer paper for details)
  - Apply MixColumns to Kernel

- Reduce message space by verifying candidates against last ciphertext block
  - Exploits the availability of last ciphertext block
  - Simulations confirm large-scale reduction due to this

- Reduced message space directly corresponds to reduced key space.
**EscApe: The Final Picture**

### Compute Hyper-State By Verifying Fault Invariants After Inverse SubBytes

<table>
<thead>
<tr>
<th>18°F2</th>
<th>F1</th>
<th>18°F5</th>
<th>2°F4</th>
<th>2°F3</th>
</tr>
</thead>
<tbody>
<tr>
<td>8°F2</td>
<td>18°F1</td>
<td>11°F5</td>
<td>3°F4</td>
<td>19°F3</td>
</tr>
<tr>
<td>5°F2</td>
<td>11°F1</td>
<td>20°F5</td>
<td>5°F4</td>
<td>30°F3</td>
</tr>
<tr>
<td>F2</td>
<td>20°F1</td>
<td>15°F5</td>
<td>19°F4</td>
<td>8°F3</td>
</tr>
<tr>
<td>F2</td>
<td>15°F1</td>
<td>6°F5</td>
<td>22°F4</td>
<td>31°F3</td>
</tr>
</tbody>
</table>

### Inbound Phase
(Repeat for each faulty cipher-text)

- **After Inverse ShiftRows**
- **After Inverse MixColumns**
- **Differential State**

### Outbound Phase

- **After Hyper-State ShiftRows**
- **After Kernel MixColumns**
- **Final Reduction in Message (Key) Space**
In the presence of faulty collision:

<table>
<thead>
<tr>
<th>Fault Count</th>
<th>Fault Type</th>
<th>Avg. Final Key Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Random word fault at the start of 10\textsuperscript{th} round in the last iteration of APE</td>
<td>$2^{80}$</td>
</tr>
<tr>
<td>2</td>
<td>225</td>
<td>$2^{25}$</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>$2^{5}$</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Epilogue

- Shown how the desirable property of misuse-resistance becomes the gateway for DFA
- First fault analysis of SPONGE when used in the context of authenticated encryption
- EscApe: efficient diagonal attack on APE
  - 2 faults lead to a practical attack, 4 give the unique key
- Removal of final truncation of FIDES in APE makes EscApe highly efficient
- Finally, it’s evident that
  - Misuse-resistance,
  - Design of underlying permutation and
  - Choice of mode of operation
can all contribute to the susceptibility of authenticated ciphers to fault attacks
Thank You

- Please forward any queries to crypto@dhimans.in

- Full version of the paper: http://de.ci.phe.red
  or, CAESAR mailing list