From AES-128 to AES-192 and AES-256, How to Adapt Differential Fault Analysis Attacks on KeyExpansion

FDTC 2011, Nara, Japan

Noémie Floissac and Yann L’Hyver

SERMA TECHNOLOGIES ITSEF
30, Avenue Gustave Eiffel, 33608 PEassa CEDEX, FRANCE
Email: \{n.floissac;y.lhyver\}@serma.com

28th September 2011
Overview

Adaptation of DFA to AES-192 and AES-256
DFA on KeyExpansion of AES-192 and AES-256
Results and conclusion

Background

AES

- Symmetric algorithm based on iterations of SubBytes, ShiftRows, MixColumn and AddRoundKey
- Each round key is provided by KeyExpansion algorithm
- 3 variants: AES-128, AES-192 and AES-256

DFA on AES-128

- General concept: fault injection on last rounds, differential analysis of correct and faulty results, obtain (last round) key
- Attack performed on State and KeyExpansion
### Overview

**DFA on AES-192 and AES-256**

#### Fault on State
- From 2010: several papers present DFA on these variants
- Based on DFA on AES-128: A. Barenghi and al

#### Fault on KeyExpansion
- Nothing presented concerning full AES key recovery
Adapt DFA on KeyExpansion from AES-128 to AES-192 and AES-256
Methodology used on AES-192 and AES-256

**Let $N$ last AES round**

**Extension**
- Inject fault on the last rounds like for DFA on AES-128
- Retrieve last round key $K_N$

**Reproduction**

**Aim**: Retrieve respectively the 8 and 16 bytes of missing key
- Inject fault like for extension but on the previous round
- Reduce AES help to inverse MixColumn trick:
  
  $C = S_{SR,N-1} \oplus I_{MC}(K_{N-1})$

  - Exploit the faulty result at end of penultimate round
  - Retrieve penultimate round key $K_{N-1}$
AES variant differences

Case AES-192
- RotWord and SubWord are not applied on last column $K_{10}$
- 2 first columns of $K_{11}$ depend on 2 last columns of $K_{10}$
- 2 last columns of $K_{11}$ do not impact 2 last columns of $K_{12}$

Case AES-256
- Only SubWord is applied on last column of $K_{12}$
- All columns of $K_{14}$ depend on 4 columns of $K_{12}$
- Columns of $K_{13}$ do not impact columns of $K_{14}$, except the last one: RotWord and SubWord transformations
Overview
Adaptation of DFA to AES-192 and AES-256
DFA on KeyExpansion of AES-192 and AES-256
Results and conclusion

Introduction
Extension on AES-192 and AES-256
Reproduction on AES-256

Goal

Original attack

Attack on AES-192 and AES-256
Apply technics used on original attack with the previous methodology
DFA on KeyExpansion AES-256: Extension

KeyExpansion:

Cipher:

N. Floissac and Y. L’Hyver, SERMA TECHNOLOGIES ITSEF

From AES-128 to AES-192 and AES-256, How to Adapt ...
Overview

Adaptation of DFA to AES-192 and AES-256

DFA on KeyExpansion of AES-192 and AES-256

Results and conclusion

Introduction

Extension on AES-192 and AES-256

Reproduction on AES-256

Extension: analysis I

Differences with original attack

- Fault injected on line \( i \)
- AES-192:
  - \( K^*_{12}\{i,j\} = K_{12}\{i,j\} \), whenever \( j \) equals to 0, 1 or 3
  - \( K^*_{12}\{i,j\} = K_{12}\{i,j\} \oplus a \), whenever \( j \) equal to 2
- AES-256:
  - \( K^*_{14}\{i,j\} = K_{14}\{i,j\} \), for all \( j \)
- Original equation is still true: for a given byte \( \{i, (j - i)[4]\} \), where \( j \) in \([0..3]\)
  - \( a = \text{I}_\text{Sb}(C \oplus K_N) \oplus \text{I}_\text{Sb}(C^* \oplus K^*_N) \)
- Exhaustive search on each byte of \( K_N \) and check on \( a \)
Extension: analysis II

Exploitation

- 2 couples \((C_1, C_1^\ast)\) and \((C_2, C_2^\ast)\) for each line targeted
- Inject a fault on each line of first column of \(K_{N-1}\)
- Retrieve \(K_N\)

\[K_{N-1}\]

- Diffusion gives: \(b = Sb(K_{N-1}\{i, 3\} \oplus a) \oplus Sb(K_{N-1}\{i, 3\})\).
- 2 couples \((a, b)\) known for each line
- Exhaustive search on each byte of \(K_{N-1}\{., 3\}\)
Overview

A daptation of DFA to AES-192 and AES-256

DF A on KeyExpansion of AES-192 and AES-256

Results and conclusion

Introduction

Extension on AES-192 and AES-256

Reproduction on AES-256

Extension : conclusions

AES-192

- $K_{12}$ is found
- 4 bytes of $K_{11}$ missing:
  - Exhaustive search
  - Reproduction of DFA on KeyExpansion

N. Floissac and Y. L’Hyver, SERMA TECHNOLOGIES ITSEF
AES-192

- $K_{12}$ is found
- 4 bytes of $K_{11}$ missing:
  - Exhaustive search
  - Reproduction of DFA on KeyExpansion

AES-256

- $K_{14}$ is found
- 12 bytes of $K_{13}$ missing: reproduction of DFA on KeyExpansion
Adaptation of DFA to AES-192 and AES-256

DF A on KeyExpansion of AES-192 and AES-256

Results and conclusion

Introduction

Extension on AES-192 and AES-256

Reproduction on AES-256

Reproduction : Fault diffusion on AES-256

N. Floissac and Y. L'Hyver, SERMA TECHNOLOGIES ITSEF

From AES-128 to AES-192 and AES-256, How to Adapt...

Overview
DFA on AES-256: analysis

Reproduction: Find $K_{14}^*$

- Retrieve $a$ and $c$
- Line $i$ of injection unknown
- Diffusion gives for a given $i$:
  - $a = K_{12}\{i, j\} \oplus K_{12}^*\{i, j\}$, where $j$ in [0..3]
  - $b = Sb(K_{12}\{i, 3\} \oplus a) \oplus Sb(K_{12}\{i, 3\})$
  - $c = Sb(K_{13}\{i, 3\} \oplus b) \oplus Sb(K_{13}\{i, 3\})$
  - We have:
    
    \[
    c = Sb(K_{13}\{i, 3\}) \oplus Sb(K_{12}\{i, 3\} \oplus a) \oplus Sb(K_{12}\{i, 3\})) \\
    \oplus Sb(K_{13}\{i, 3\})
    \]
- Columns 2 and 3 of $K_{14}$ known: $K_{12}\{i, 3\}$ is known
- Extension: $K_{13}\{i, 3\}$ is known
DFA on AES-256: exploitation

Exploitation

- Search on `a` and `i` gives hypotheses on $K_{14}^*$
- Correct and faulty output known: Use Inverse MixColumn trick with $K_{14}^*$ and $K_{14}$ to obtain $S_{ARK,13}$
- Find good hypothesis on $K_{14}^*$
DFA on AES-256: exploitation I

\[
K^{*14} \rightarrow C^* \\
K_{14} \rightarrow C
\]
Overview
Adaptation of DFA to AES-192 and AES-256
DFA on KeyExpansion of AES-192 and AES-256
Results and conclusion

Introduction
Extension on AES-192 and AES-256
Reproduction on AES-256

DFA on AES-256: exploitation I

N. Floissac and Y. L’Hyver, SERMA TECHNOLOGIES ITSEF
From AES-128 to AES-192 and AES-256, How to Adapt ...
DFA on AES-256: exploitation I

\[ K_{13} \rightarrow K_{14} \rightarrow C^* \]

\[ K_{13} \rightarrow K_{14} \rightarrow C \]
DFA on AES-256: exploitation I

\[ K^{*13} \rightarrow K^{*14} \rightarrow C^* \]

\[ K_{13} \rightarrow K_{14} \rightarrow C \]
DFA on AES-256: exploitation I
DFA on AES-256: exploitation I

N. Floissac and Y. L’Hyver, SERMA TECHNOLOGIES ITSEF

From AES-128 to AES-192 and AES-256, How to Adapt...
DFA on AES-256: exploitation I
DFA on AES-256: exploitation I
Overview

Adaptation of DFA to AES-192 and AES-256

DFA on KeyExpansion of AES-192 and AES-256

Results and conclusion

Introduction

Extension on AES-192 and AES-256

Reproduction on AES-256

DFA on AES-256 : exploitation II

Second step of reproduction

- **Known Data:**
  - $K_{14}$ and $K^*_{14}$
  - $i$, $a$ and $b$
  - $C' (= S_{SR,13} \oplus I\_MC(K_{13}))$ and $C'^* (= S^*_{SR,13} \oplus I\_MC(K^*_{13}))$

- Let $K' = I\_MC(K_{13})$ and $K'^* = I\_MC(K^*_{13})$

- Solve equation : for a given byte $\{i, (j - i)[4]\}$, where $j$ in $[0..3]$
  - $a = I\_Sb(C' \oplus K') \oplus I\_Sb(C'^* \oplus K' \oplus b)$

- Exhaustive search on $K'\{i, (j - i)[4]\}$
DFA on AES-256: exploitation III

End of adaptation

- 2 couples \((C'_1, C'_1^*)\) and \((C'_2, C'_2^*)\) give 4 bytes of \(K'\)
- Reiteration of attack for each line gives \(K'\)
- Retrieve \(K_{13}\) and so initial AES key
Summary

First DFA on KeyExpansion of AES-192 and AES-256 variants

- Adaptation of existing attack
- Twice the number of faults of the original attack: a total of 16
### First DFA on KeyExpansion of AES-192 and AES-256 variants

- Adaptation of existing attack
- Twice the number of faults of the original attack: a total of 16

### Conclusion

- DFA on KeyExpansion can be adapted
- DFA on KeyExpansion of AES-192 and AES-256 is more complex than original attack on AES-128
- Subject is still open
Thank you for your attention.

Any Questions???