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 PESSAC CEDEX, FRANCE
Email: *{n.floissac;y.lhyver}@serma.com*

28th September 2011

SERMA TECHNOLOGIES

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

SERMA TECHNOLOGIES

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

SERMA TECHNOLOGIES

Aim

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 $\mathbf{K_N}$

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 :
  Let $\mathbf{C} = \mathbf{S_{SR,N-1}} \oplus \mathbf{I\_MC}(\mathbf{K_{N-1}})$
- Exploit the faulty result at end of penultimate round
- Retrieve penultimate round key $\mathbf{K_{N-1}}$

AES variant differences

Case AES-192

- RotWord and SubWord are not applied on last column $\mathbf{K}_{10}$
- 2 first columns of $\mathbf{K}_{11}$ depend on 2 last columns of $\mathbf{K}_{10}$
- 2 last columns of $\mathbf{K}_{11}$ do not impact 2 last columns of $\mathbf{K}_{12}$

Case AES-256

- Only SubWord is applied on last column of $\mathbf{K}_{12}$
- All columns of $\mathbf{K}_{14}$ depend on 4 columns of $\mathbf{K}_{12}$
- Columns of $\mathbf{K}_{13}$ do not impact columns of $\mathbf{K}_{14}$, except the last one : RotWord and SubWord transformations

Goal

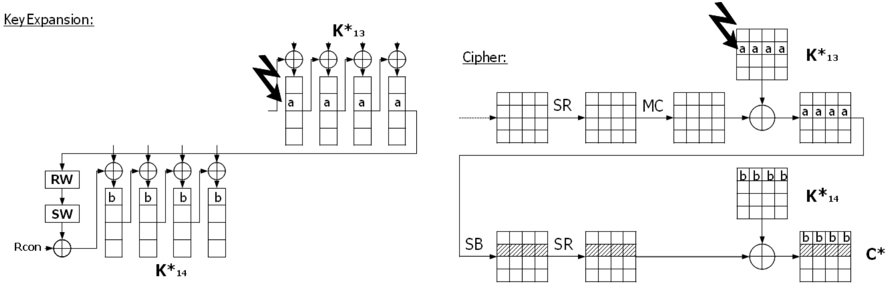
Original attack

DFA on KeyExpansion of AES-128 : C. H. Kim and J.-J. Quisquater, 2008

Attack on AES-192 and AES-256

Apply technics used on original attack with the previous methodology

DFA on KeyExpansion AES-256 : Extension



Extension : analysis I

Differences with original attack

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

SERMA TECHNOLOGIES

Extension : analysis II

Exploitation

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

$\mathbf{K_{N-1}}$

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

SERMA TECHNOLOGIES

Extension : conclusions

AES-192

- $\mathbf{K}_{12}$ is found
- 4 bytes of $\mathbf{K}_{11}$ missing :
  - Exhaustive search
  - Reproduction of DFA on KeyExpansion

Extension : conclusions

AES-192

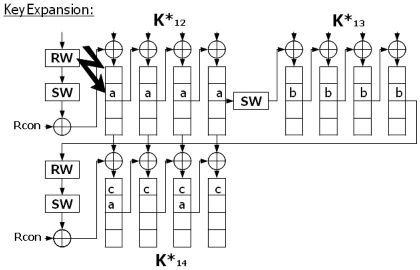
- $\mathbf{K}_{12}$ is found
- 4 bytes of $\mathbf{K}_{11}$ missing :
  - Exhaustive search
  - Reproduction of DFA on KeyExpansion

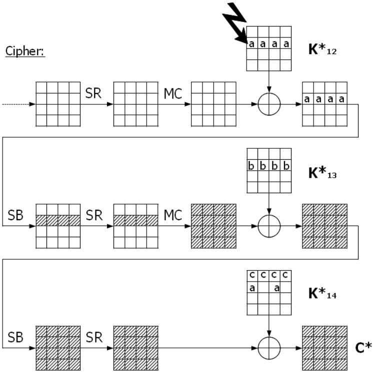
AES-256

- $\mathbf{K}_{14}$ is found
- 12 bytes of $\mathbf{K}_{13}$ missing : reproduction of DFA on KeyExpansion

SERMA TECHNOLOGIES

Reproduction : Fault diffusion on AES-256





DFA on AES-256 : analysis

Reproduction : Find $\mathbf{K}^*_{14}$

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

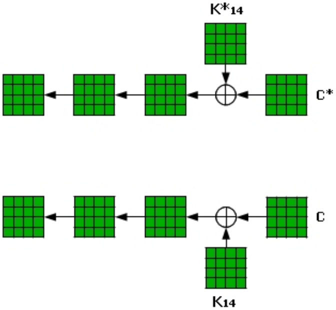
SERMA TECHNOLOGIES

DFA on AES-256 : exploitation

Exploitation

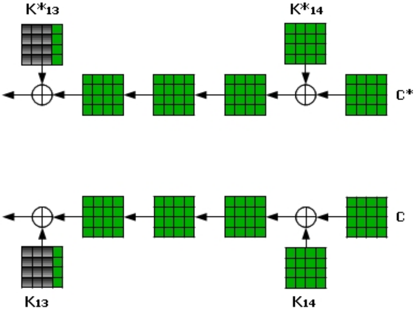
- Search on **a** and **i** gives hypotheses on $\mathbf{K}^*_{14}$
- Correct and faulty output known : Use Inverse MixColumn trick with $\mathbf{K}^*_{14}$ and $\mathbf{K}_{14}$ to obtain $\mathbf{S}_{ARK,13}$
- Find good hypothesis on $\mathbf{K}^*_{14}$

DFA on AES-256 : exploitation I

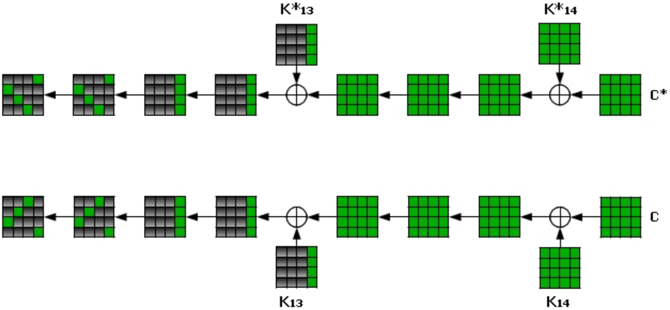


SERMA TECHNOLOGIES

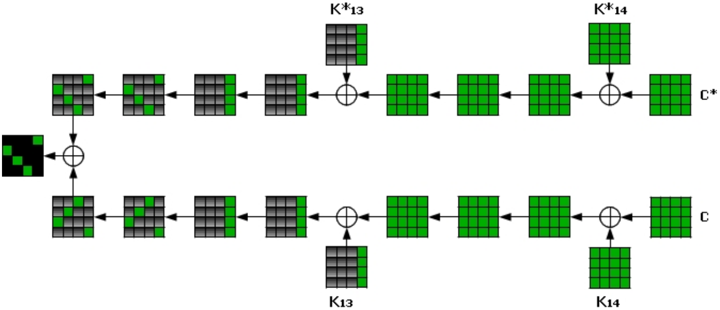
DFA on AES-256 : exploitation I



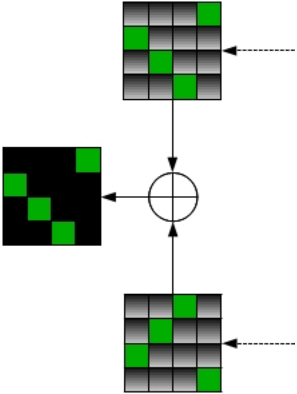
DFA on AES-256 : exploitation I



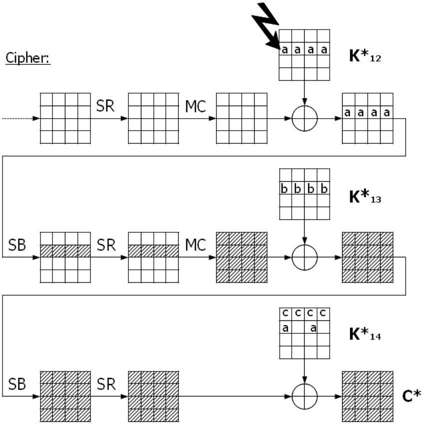
DFA on AES-256 : exploitation I



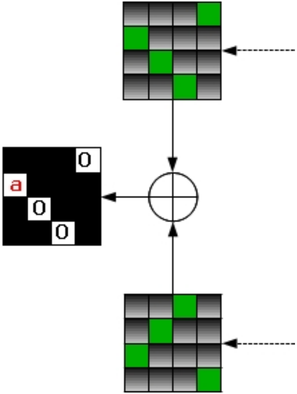
DFA on AES-256 : exploitation I



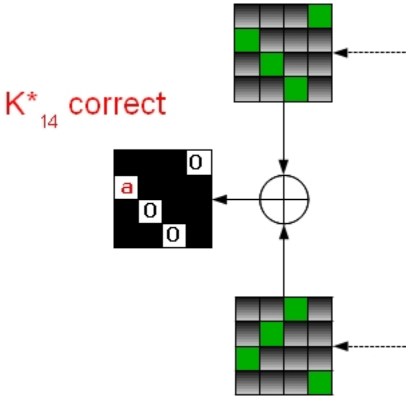
DFA on AES-256 : exploitation I



DFA on AES-256 : exploitation I



DFA on AES-256 : exploitation I



DFA on AES-256 : exploitation II

Second step of reproduction

- Known Data :
  - $\mathbf{K_{14}}$ and $\mathbf{K^*_{14}}$
  - $\mathbf{i}$, $\mathbf{a}$ and $\mathbf{b}$
  - $\mathbf{C}'$ $(= S_{SR,13} \oplus I\_MC(K_{13}))$ and $\mathbf{C}'^*$ $(= S^*_{SR,13} \oplus I\_MC(K^*_{13}))$
- Let $\mathbf{K}' = \mathbf{I\_MC}(\mathbf{K_{13}})$ and $\mathbf{K}'^* = \mathbf{I\_MC}(\mathbf{K^*_{13}})$
- Solve equation : for a given byte $\{\mathbf{i}, (\mathbf{j} - \mathbf{i})[4]\}$, where $\mathbf{j}$ in $[0..3]$
  $$\mathbf{a} = \mathbf{I\_Sb}(\mathbf{C}' \oplus \mathbf{K}') \oplus \mathbf{I\_Sb}(\mathbf{C}'^* \oplus \mathbf{K}' \oplus \mathbf{b})$$
- Exhaustive search on $\mathbf{K}'\{\mathbf{i}, (\mathbf{j} - \mathbf{i})[4]\}$

SERMA TECHNOLOGIES

DFA on AES-256 : exploitation III

End of adaptation

- 2 couples $(\mathbf{C}'_1, \mathbf{C}'^*_1)$ and $(\mathbf{C}'_2, \mathbf{C}'^*_2)$ give 4 bytes of $\mathbf{K}'$
- Reiteration of attack for each line gives $\mathbf{K}'$
- Retrieve $\mathbf{K}_{13}$ 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

SERMA TECHNOLOGIES

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

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

SERMA TECHNOLOGIES

Thank you for your attention.

Any Questions???