A Differential Fault Attack Technique Against SPN Structures and the AES

G. Piret*, J.-J. Quisquater

CHES 2003 Workshop
Cologne, Germany
Outline of the Talk

1. General Context.
   → Introduction
   → Cipher Structure
   → Framework of the Attack

2. The Attack.
   → Sketch of an Attack
   → A Practical Attack
   → Dealing with Wrong-Located Faults

3. Application to the AES.
   → About the Linear Transform of the AES
   → The Basic Attack
   → An Improved Attack
   → Implementation on a PC

4. Conclusion.
Introduction: Fault attacks

- First suggestion in 1997: Boneh, DeMillo, Lipton. Fault Attack on RSA-CRT.

- Application to block ciphers, especially DES: Biham, Shamir 1997.

- Several papers about DFA on the AES: BS02, DLV03, G03, …
Fault Attacks: Principle

- Induce faults during cryptographic computation.
  - By changing power supply voltage.
  - By increasing frequency of the external clock.
  - By applying radiations.
- Outputs faulty results.
- Use them to recover the secret key stored in the card.

Key=1010110...
Framework of our Attack

- Faults occurring on **bytes**.
- A faulty ciphertext results from one unique fault.
- Cipher Structure: Substitution-Permutation Network.

- Countermeasure: Double encryption.
Substitution-Permutation Network (SPN)

A round with structure $\sigma[K^r]\circ\theta\circ\gamma$ is iterated several times:

- $\sigma[K^r] =$ Key addition $\quad \sigma[k](a)=b \iff b=a\oplus k$
- $\theta =$ Linear diffusion layer
- $\gamma =$ Non-linear layer
Fault Location

\[ \theta_{R-2} \]

\[ \theta_{R-1} \]
Observation

• The difference before $\theta_{R-1}$ caused by a random fault between $\theta_{R-2}$ and $\theta_{R-1}$ is of the form:

$$(0, \ldots, 0, \alpha, 0, \ldots, 0)$$

The number of such differences is $255n$.

• There are $255n$ corresponding differences before the last S-box layer. They are of the form:

$$(\alpha_1, \ldots, \alpha_n)$$
**Sketch of an Attack**

1. Compute a list $\mathcal{D}$ of the $255^n$ possible differences after $\theta_{R^{-1}}$.

2. Consider a plaintext $P$, the corresponding ciphertext $C$, and the faulty ciphertext $C^*$.

3. For each possible $K^R$, compute the difference:
   \[
   \gamma_R^{-1} \circ \sigma[K^R](C) \oplus \gamma_R^{-1} \circ \sigma[K^R](C^*)
   \]
   If it is in $\mathcal{D}$, add $K^R$ to the list $\mathcal{L}$ of possible candidates.

4. Consider a new plaintext $P$, with corresponding ciphertexts $C$ and $C^*$. Apply step 3 to all candidates of $\mathcal{L}$. 
Some Comments

• 2 pairs \((C, C^*)\) are enough to retrieve \(K^R\), provided the linear layer \(\theta\) is optimal.

• If \(K^R\) is not enough to retrieve the master key \(K\), last round can be peeled off, and the attack repeated to retrieve \(K^{R-1}\).

• Not practical: Time complexity \(2^{8n}\).
A Practical Attack

1. Compute the list $\mathcal{D}$ of possible differences before $\theta_{R-1}$
2. Consider two pairs $(C,C^*)$ and $(D,D^*)$.
3. Consider the 2 left-most bytes of $K^R$. For each of the $2^{16}$ candidates, compute:
   \[ \gamma^{-1}_R \circ \sigma[\langle K_1^R, K_2^R \rangle](\langle C_1, C_2 \rangle) \oplus \gamma^{-1}_R \circ \sigma[\langle K_1^R, K_2^R \rangle](\langle C_1^*, C_2^* \rangle) \]
   \[ \gamma^{-1}_R \circ \sigma[\langle K_1^R, K_2^R \rangle](\langle D_1, D_2 \rangle) \oplus \gamma^{-1}_R \circ \sigma[\langle K_1^R, K_2^R \rangle](\langle D_1^*, D_2^* \rangle) \]
4. Compare the results with the 2 left-most bytes of the differences in $\mathcal{D}$. The $\langle K_1^R, K_2^R \rangle$ for which a match is found for both ciphertext pairs are stored in a list $\mathcal{L}$. 
5. For each candidate of $\mathcal{L}$, try to extend it by one byte (computing both differences to check).
6. Keep extending candidates in $\mathcal{L}$ until they are $n$-bytes long. At this stage, only the right key is remaining.
Faults Occurring at a Wrong Location

• Usually the attacker has no control on the fault location.
• Problem: To distinguish pairs \((C, C^*)\) resulting from a fault occurring between \(\theta_{R-2}\) and \(\theta_{R-1}\) [right pairs] from other pairs [wrong pairs].

• If the diffusion layer \(\theta_{R-1}\) is not optimal: Trivial.
• If \(\theta_{R-1}\) is optimal, it is not possible to decide whether a single pair \((C, C^*)\) is a right pair or not.
Faults Occurring at a Wrong Location

• However if:
  – \((C, C^*)\) is a right pair.
  – \((D, D^*)\) is a wrong pair.

Then applying the attack to these pairs
\(\rightarrow\) no solution for \(K^R\).

• Thus wrong pairs can be distinguished, by considering pairs of pairs \((C, C^*)\).

• Suppose 1 pair \((C, C^*)\) out of 50 is right.
  \(\rightarrow\) \(~10000\) \((100*100)\) pairs \(((C, C^*);(D, D^*))\) need to be examined in order to find \(K^R\). \(\rightarrow\) Feasible!
The AES-128

- 128-bit block, 128-bit key variant. 10 rounds SP Network.
- Knowledge of $K^R$ is enough to retrieve the master key.
- Non-optimal linear diffusion layer: Composition of 2 transformations, \texttt{ShiftRow} and \texttt{MixColumn}.
Basic Attack

\( \theta_{R-1} \)

\( K_1^{R-1} \quad K_2^{R-1} \quad K_3^{R-1} \quad K_n^{R-1} \)

\( K_1^R \quad K_2^R \quad K_3^R \quad K_n^R \)

(C,C*)  (D,D*)

K^R
Basic Attack

- If the fault location can be chosen very precisely: 8 = 4 * 2 pairs \((C, C^*)\) are needed to retrieve \(K^R\). (but in fact, 6 pairs are enough)
- If we cannot choose the byte where the fault occurs: \(~15\) pairs are needed.

An Improved Attack

- It is possible to do better if we deal with faults occurring between \(\theta_{R-3}\) and \(\theta_{R-2}\) (instead of between \(\theta_{R-2}\) and \(\theta_{R-1}\)).
A DFA Technique Against SPN Structures and the AES
Implementation on a PC

• Using 2 right pairs \((C,C^*)\), with fault occurring between \(\theta_{R-3}\) and \(\theta_{R-2}\):
  \(\Rightarrow\) Takes a few seconds.
  \(\Rightarrow\) Unique candidate retrieved in 77% of the cases.
  \(\Rightarrow\) Number of candidates never exceeds 16.

• Applying the attack to 2 pairs one of which is wrong (i.e. corresponds to a fault occurring before \(\theta_{R-3}\)), the obtained set of solutions was always empty.
  \(\Rightarrow\) We can indeed reject wrong pairs !!
Conclusion

• Attack exploits faults on bytes.
• If fault location can be chosen:
  → Requires only 2 faulty ciphertexts.
  → Takes a few seconds.
• If fault location cannot be chosen:
  → Requires ~100 faulty ciphertexts
  → Completes in a few hours.
• Applicable to other ciphers: Khazad, Noekeon, Serpent,…
• The simple and elegant structure of SPNs makes such an efficient attack possible.