J-DFA

A Novel Approach for Robust Differential Fault Analysis

Luca Magri, Silvia Mella
University of Milan

Pasqualina Fragneto, Beatrice Rossi, Filippo Melzani
STMicroelectronics

Attack Scenario

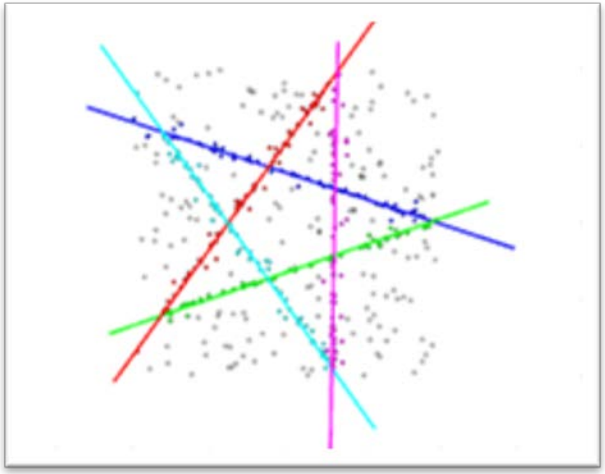
Most Differential Fault Analysis require some kind of knowledge by the attacker on the effect of the faults

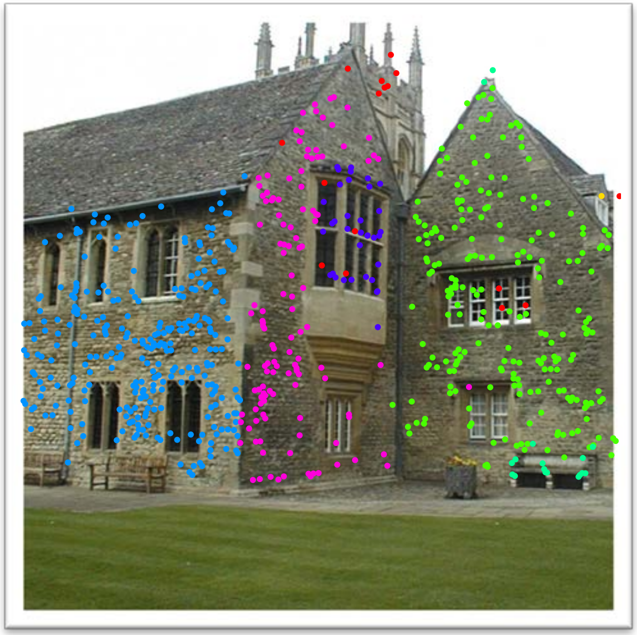
- Every fault provides information about the secret key, based on the model assumed a-priori by the attacker
- Discrepancies between model and experiments can lead to wrong solution (or no solution) for the key

This work: Application of a specific clustering technique with the purpose of softening the a-priori knowledge on the injection technique

J-Linkage

- Clustering technique that tackles the problem of fitting multiple models to data corrupted by noise and outliers
- Originally proposed for geometric model fitting in Computer Vision
  - homography estimation, plane fitting, motion segmentation

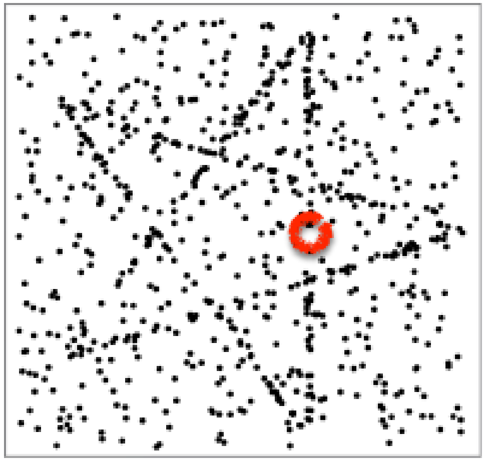




Properties of J-Linkage

- Based on conceptual data representation: each point is represented with the characteristic function of the set of models that fit the point
- A tailored agglomerative clustering is used to group points belonging to the same model
- Does not require prior specification of the number of models, nor it necessitates parameters tuning
- Robust to outliers

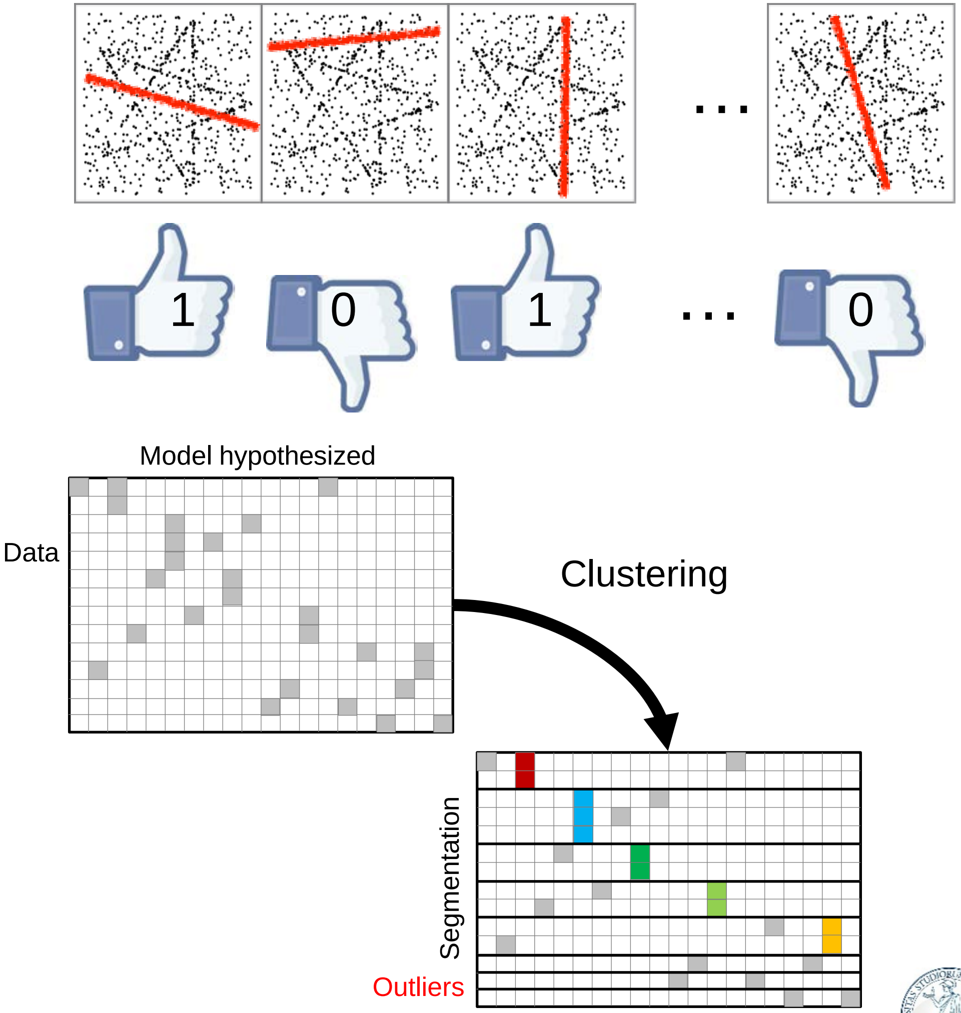
J-Linkage: geometric example



Model Hypothesis

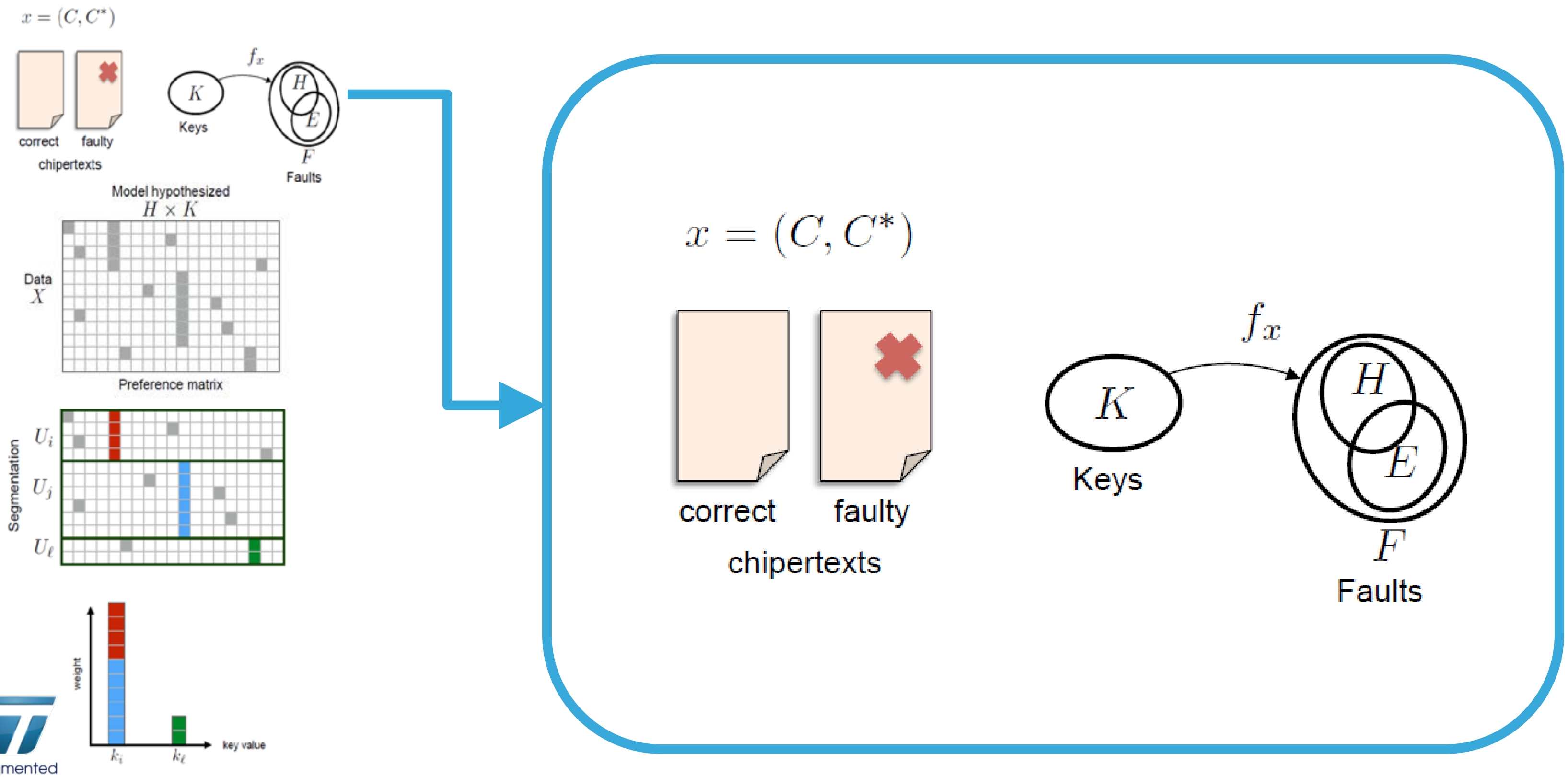
Points’ Preferences

Preference Matrix



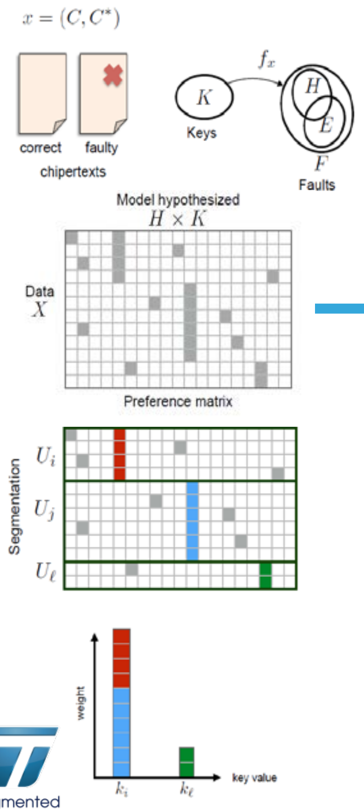
J-DFA: Data Mapping

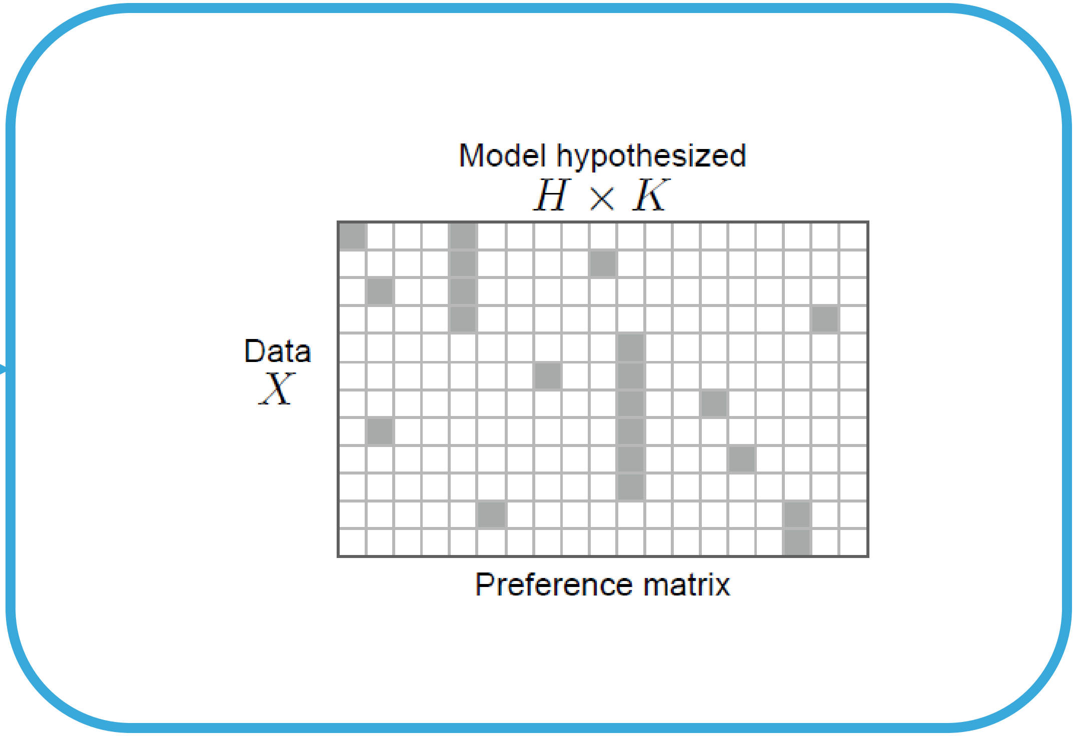
An experiment defines a map between possible key values and the set of possible faults.



J-DFA: Conceptual Representation

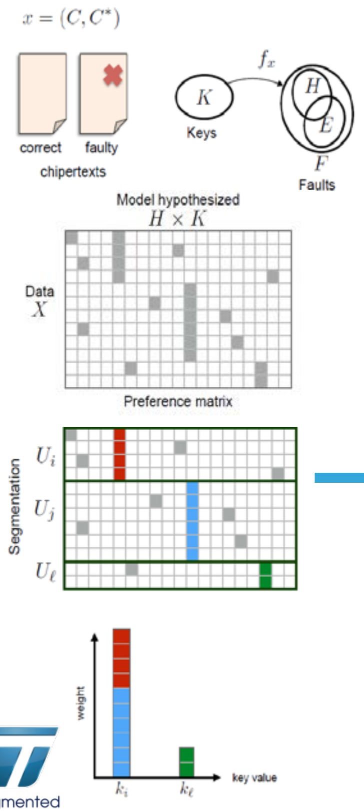
The preference matrix is built, representing every datum by the votes it grants to the set of putative models.

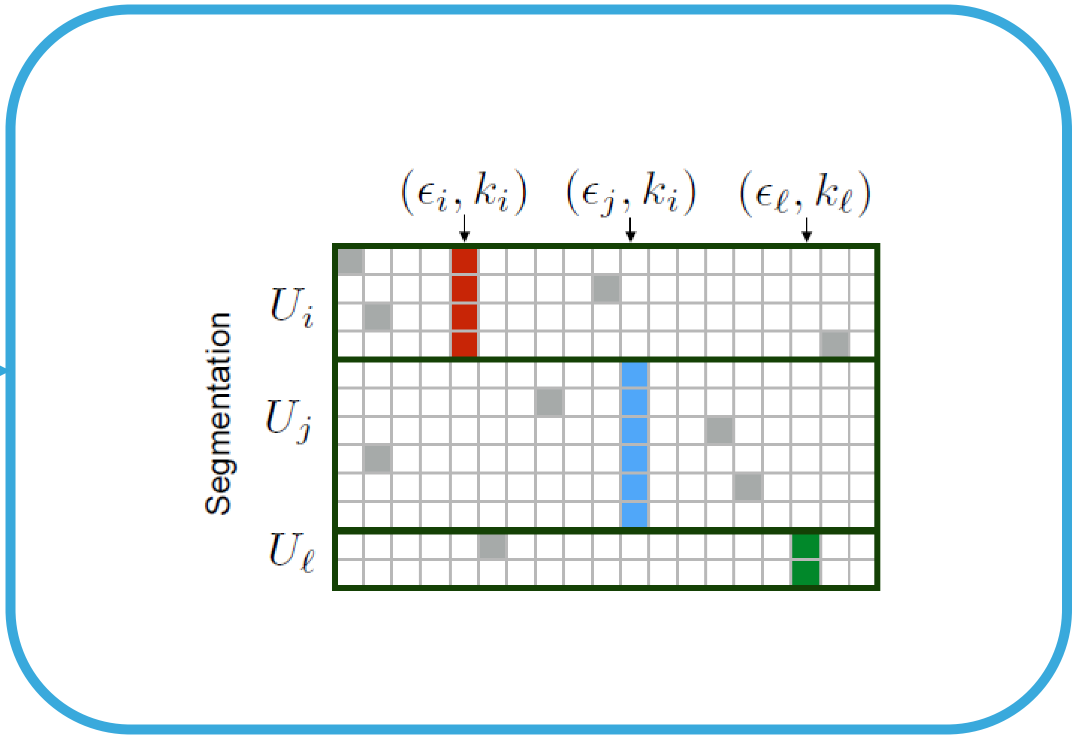




J-DFA: Clustering

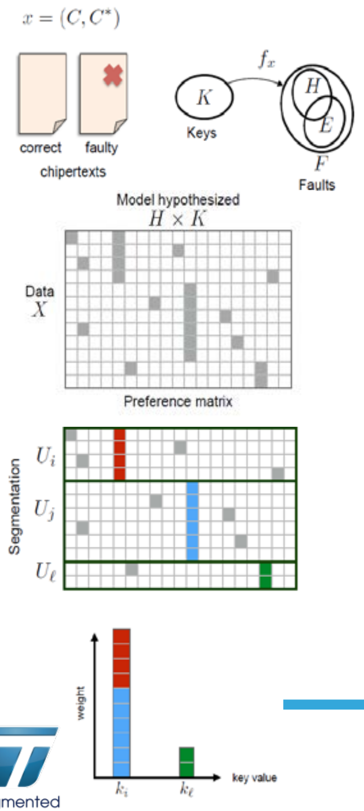
J-Linkage segments the preference matrix in clusters.
Most preferred models per cluster are extracted.
The same key may appear as preferred by several clusters.

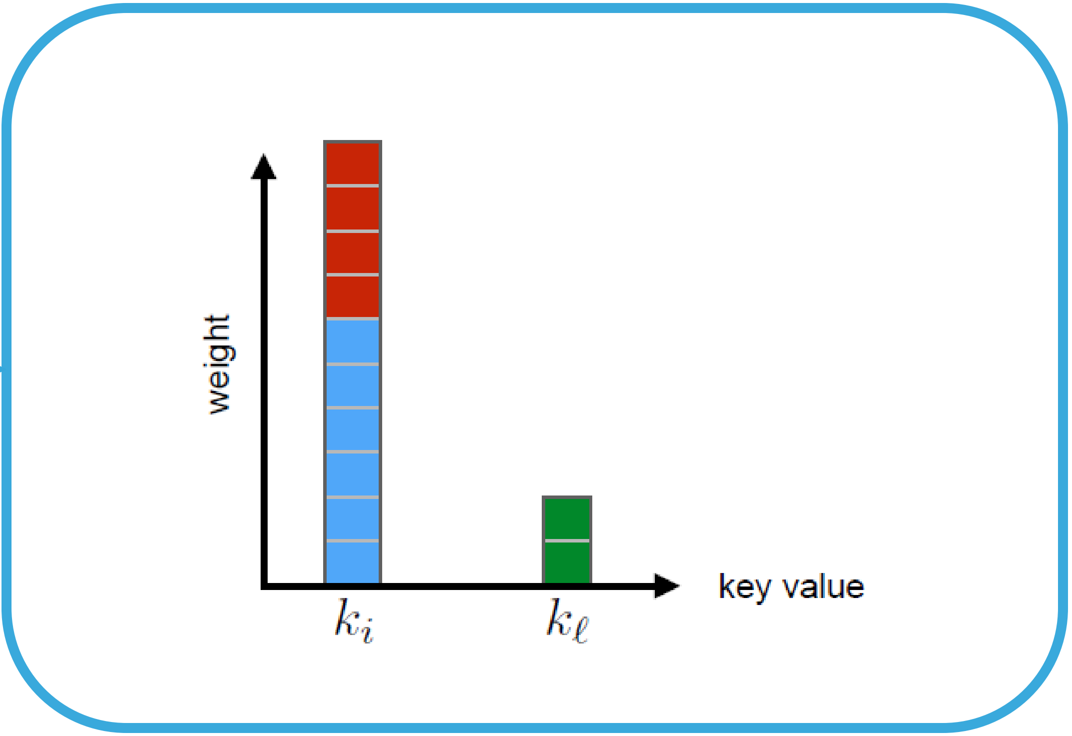




J-DFA: Ranking of the Keys

Votes are aggregated with respect to keys and the most preferred one is retained.





J-DFA: example

- DFA described in [Giraud] as a reference
  - Fault: one bit at the beginning of the last round of AES
    $E \in \{0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80\}$
  - Experiment: a couple of correct and faulty ciphertexts
  - Data mapping: based on
    $\text{SubBytes}^{-1}(\mathrm{c} \oplus \mathrm{k}) \oplus \text{SubBytes}^{-1}(c^{*} \oplus \mathrm{k}) \in E$
- Among all the possible faults some can be filtered a-priori
  - When correct and faulty ciphertexts differ for more than a byte
- Experiments related to faults not included in the model are managed as outliers
  - They cannot be identified a-priori
  - They severely compromise the success of a classical DFA

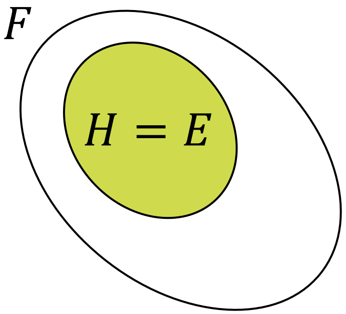
[Giraud] C. Giraud. *DFA on AES*. IACR Cryptology ePrint Archive, 2003.

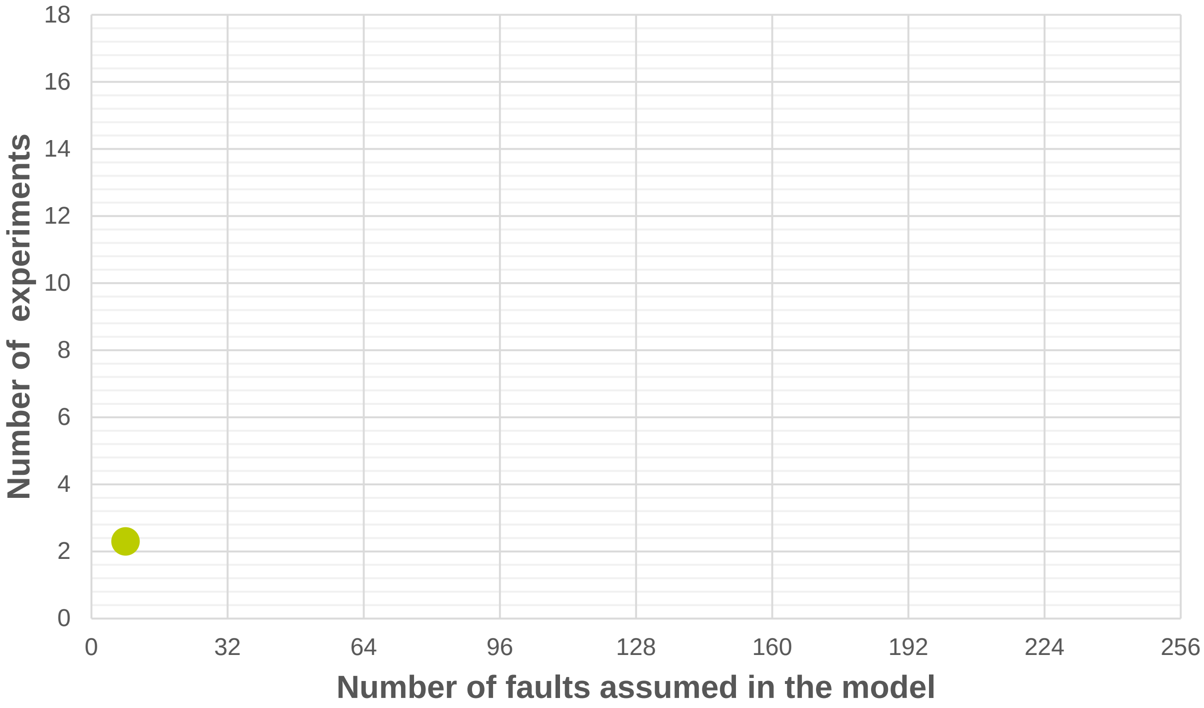
J-DFA with profiling

- Faults are generated through SW simulation
  - The set of possible fault effects E are defined at the beginning
- The assumed fault model H is exactly equal to E
  - The case when the attacker through profiling completely characterize the injection technique on the target device

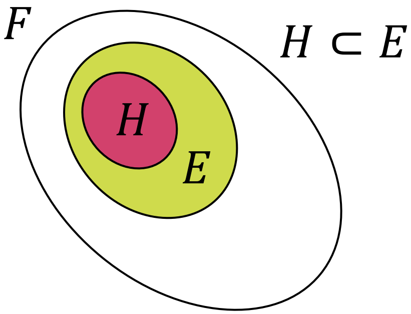
Possible Faults	**Average number of experiments to identify the correct byte of the key**
[Giraud]: $\|E\| = 8$	2.1
Only the least significant bit: $\|E\| = 1$	1.9
All but the most significant bit: $\|E\| = 127$	210,3

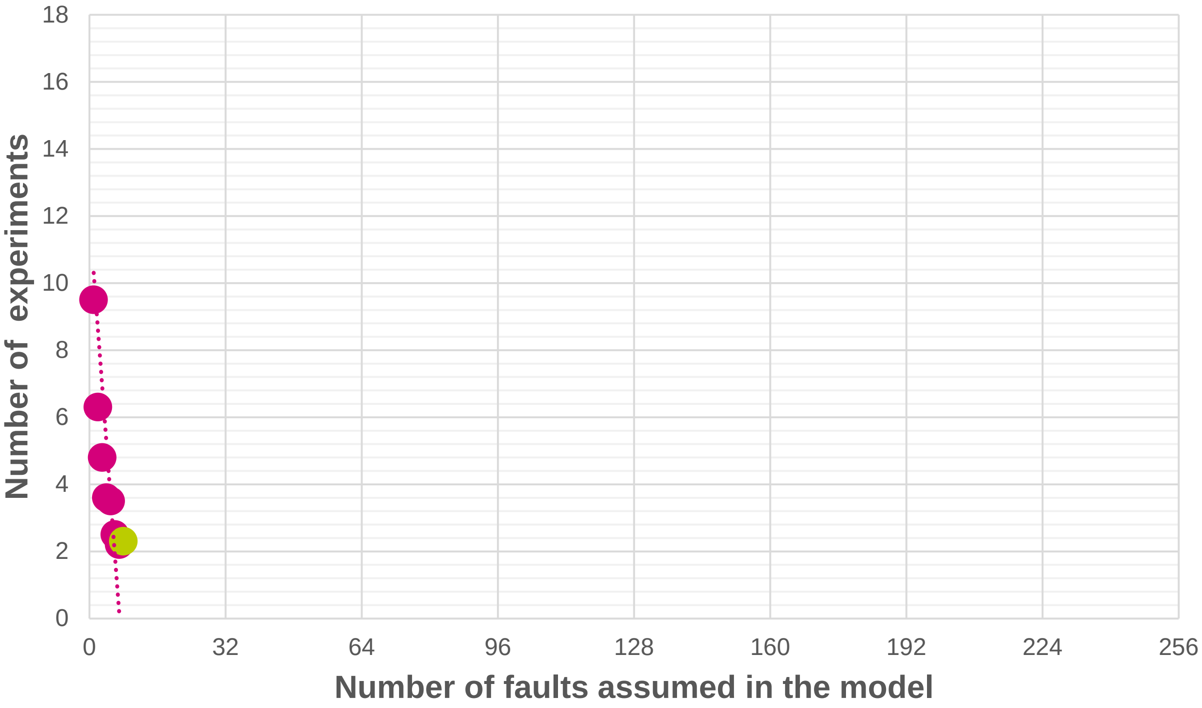
J-DFA with profiling



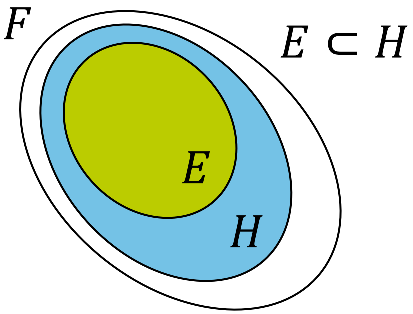


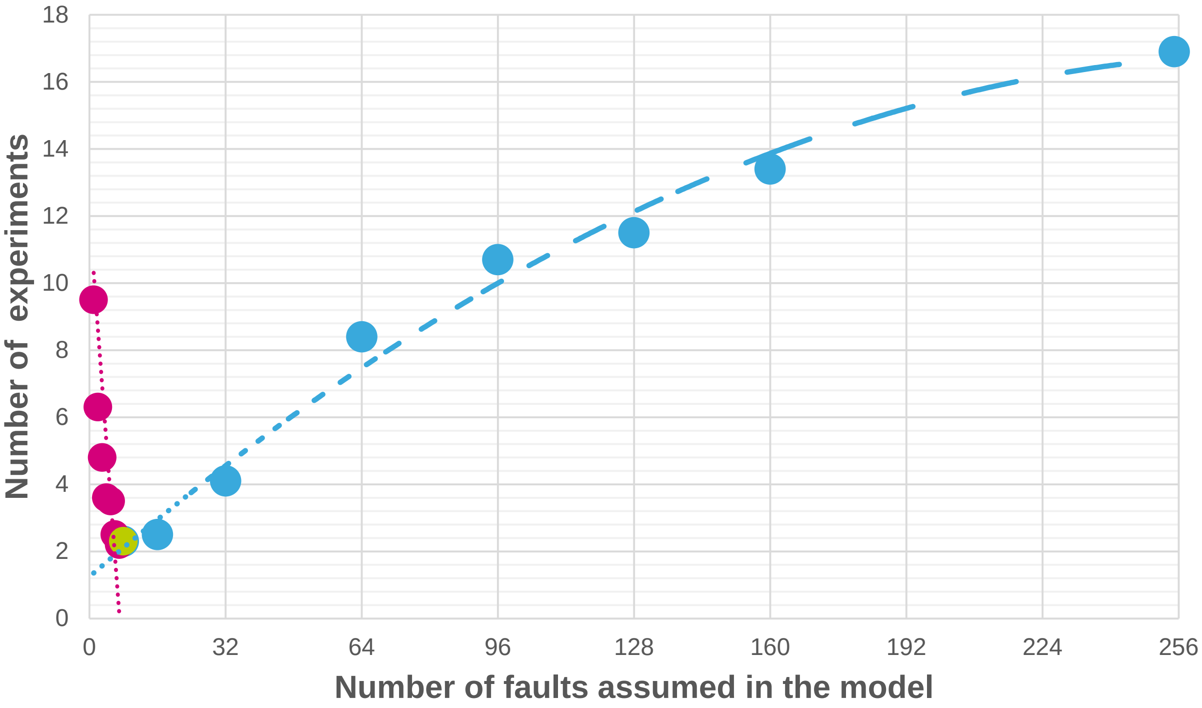
J-DFA without profiling





J-DFA without profiling





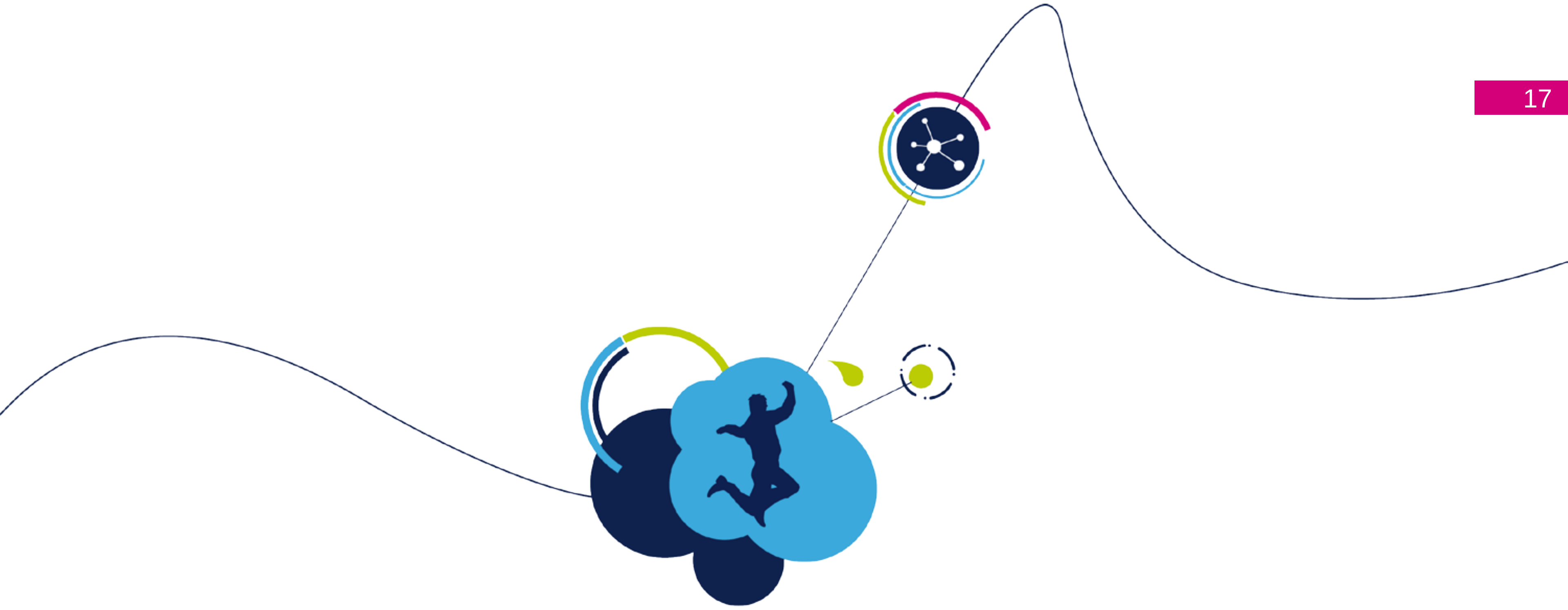
Working Conditions

- J-DFA works in case…
  - All the experiments fit in the model (which is limited to a subset)
    - It is assumed by many papers introducing new DFA attacks
  - Some of the experiments fit in the model
    - The others are managed as outliers
  - The model includes all the possible faults in a class
    - Differently from classical DFA

- J-DFA does not work in case…
  - None of the experiments fit in the model
    - Like any other approach that uses a wrong model

Conclusions

- J-DFA works!
  - Convenient tool to replicate classical DFA attacks
- J-DFA works even in case the experiments do not perfectly fit into the assumed model
  - Outliers are managed by J-linkage
  - The fault model can be extended up to an entire class of effects
- In principle J-DFA can be applied to any known DFA, by just adapting the "Data Mapping" stage
  - Still, the computational effort needs to be evaluated



Thank you