



2023

Hélène Le Boudier, Ronan Lashermes et Fabien Autrel

Sécurité matérielle : les attaques physiques

Leçon 3 : *Attaques par injections de fautes*



IMT Atlantique

Bretagne-Pays de la Loire
École Mines-Télécom

Objectifs Pédagogiques :

- Comprendre ce qu'est une faute matérielle.
- Connaître les différents moyens d'injection de faute.
- Exploiter une faute matérielle, le "personal identification number (PIN)" reste notre cas d'usage.
- Savoir se protéger des fautes matérielles.

Table des matières

1	Introduction	3
2	Techniques d'injection de faute	4
2.1	Injections par rayonnement lumineux	4
2.2	Injections de fautes par rayonnement électromagnétique	5
2.3	Injections de fautes par perturbation d'horloge	5
3	Effet d'une injection de faute	6
3.1	Faute sur les données	6
3.2	Faute sur les instructions	6
4	Exploitation des fautes	7
4.1	Differential fault analysis (DFA)	7
4.2	Exemples d'exploitation de fautes contre une vérification code PIN	7
5	Contre-mesures	8
5.1	Redondance	8
5.2	Bouclier	8
6	Conclusion	9

1 Introduction

Cette leçon est entièrement dédiée aux attaques physiques par injections de fautes [1].

L'idée d'injecter des fautes physiques vient de la conquête spatiale. Les rayons cosmiques génèrent des erreurs dans les circuits des ordinateurs envoyés dans l'espace. Les environnements radiatifs peuvent également générer des fautes. À l'origine les premières fautes physiques étaient donc accidentelles et non malveillantes.

Dans ce cours quand on parle **d'attaques par injections de fautes** FIA, c'est l'idée de perturber le bon fonctionnement du circuit, dans le but d'obtenir des informations ou de le détourner.

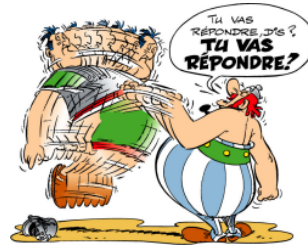


illustration Albert Uderzo

De nombreux moyens différents peuvent être utilisés pour injecter une faute physique. Les FIA sont très puissantes. Cependant, ce sont généralement des attaques invasives ou semi-invasives, elles comportent un risque non négligeable qui est l'endommagement voir la destruction du circuit. En effet, certains outils d'injection peuvent détruire une partie du circuit. C'est pourquoi, lors d'une attaque par injection de fautes, l'attaquant cherche généralement à minimiser le nombre de fautes à injecter.

Les méthodes pour les analyser sont multiples. Ce cours vous présente un rapide aperçu des techniques les plus répandues. Les valeurs notées d'une étoile * signifient qu'elles sont fautes.

2 Techniques d'injection de faute

2.1 Injections par rayonnement lumineux

Un moyen d'injection de fautes est le laser [2, 3, 4, 5, 6]. Le rayonnement lumineux émis par un laser ou une source lumineuse focalisée permet d'injecter des fautes dans un circuit. De tels dispositifs d'injections de fautes nécessitent une ouverture du boîtier, il s'agit donc d'attaques invasives. Ce type d'attaque nécessite une ouverture chimique ou mécanique dans un environnement dit en salle blanche. Aussi ce type d'attaque est très onéreux. D'une part par sa préparation et par le coût du laser lui-même. Les premiers bancs laser utilisés étaient des lasers de découpe, maintenant il existe des bancs laser spécialisés dans l'injection de fautes.

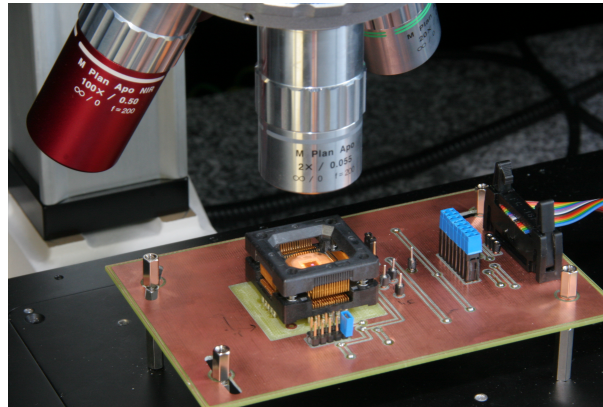




FIGURE 1 – Banc laser de la plateforme Micropacs, du centre de Microélectronique de Provence, de L'École des Mines de Saint-Étienne

La FIGURE 1 est l'un des bancs d'injection de faute par laser de la plateforme Micropacs , du centre de Microélectronique de Provence, de L'École des Mines de Saint-Étienne .

L'énergie du rayonnement lumineux utilisé est absorbée par le silicium du circuit. Lorsque l'énergie transmise est supérieure au seuil permettant à des électrons de passer dans la bande de conduction du silicium, des paires électron-trou sont alors créées le long du faisceau lumineux. Ce phénomène est illustré dans la FIGURE 2 reprise de la thèse de Roscian [7].

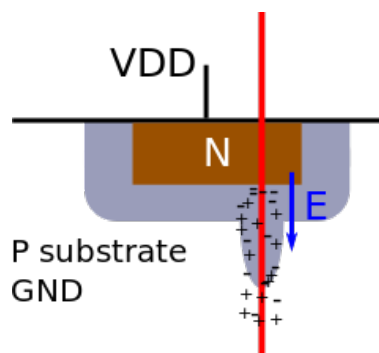


FIGURE 2 – Effet d'un tir laser, VDD = potentiel d'alimentation, GND (Ground) = Masse = potentiel 0, P et N type du semi-conducteur.

Ces paires électron-trou peuvent aboutir à l'apparition d'un courant photoélectrique au niveau d'un transistor. Ce courant entraîne alors l'apparition d'un pic de tension qui peut se propager dans des blocs de logique combinatoire. Ce phénomène est appelé single event transient (SET). Le pic de tension induit peut entraîner l'apparition d'une faute s'il est échantillonné par un élément mémoire comme un registre.

2.2 Injections de fautes par rayonnement électromagnétique

Une autre méthode de plus en plus utilisée est l'injection de fautes par rayonnement électromagnétique[8, 9, 10, 11]. Le circuit attaqué est placé sous une antenne en champ proche. L'attaquant fait ensuite circuler un courant très fort durant un temps très court (les paramètres exacts dépendent de la puce ciblée). Par couplage électromagnétique (similaire aux plaques à induction en cuisine), une partie de cette énergie va être transmise dans les lignes métalliques à la surface de la puce (pistes d'alimentation, d'horloge, bus de données, ...). Le courant généré dans la puce peut en conséquence générer des erreurs.

La Figure 3 représente une plateforme permettant d'injecter des fautes par rayonnement électromagnétique sur un circuit. Cette plateforme est la propriété de l'*Inria* de Rennes.



FIGURE 3 – Faustine, plateforme d'injections de fautes du LHS INRIA Rennes. ©Inria / Photo C. Morel

2.3 Injections de fautes par perturbation d'horloge

Un circuit synchrone est constitué d'un ou plusieurs blocs logiques encadrés par des registres cadencés par une horloge interne commune, notée clk (clock). Les blocs logiques correspondent aux fonctions utilisées par l'algorithme théorique. Les registres sont des éléments mémoire du circuit, correspondant aux blocs de l'algorithme de chiffrement par blocs. L'horloge interne donne à intervalles réguliers, une impulsion électrique, simultanément à tous les composants du processeur. À chaque front montant de l'horloge, les données sont mises à jour à la sortie des registres. La FIGURE 4 représente la logique synchrone des circuits, les droites verticales rouges représentent les fronts montants de l'horloge.

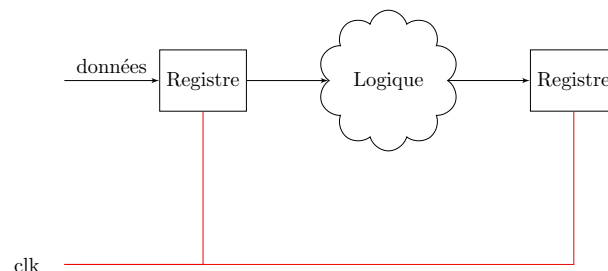


FIGURE 4 – Circuit cadencé par une horloge interne.

Il apparaît évident que la période d'horloge doit être suffisamment longue pour assurer un bon déroulement des calculs (voir FIGURE 5). Plus précisément, nous avons les notations suivantes :

- t_{clk} : la période d'horloge.
- t_{Max} : le temps auquel a lieu le dernier changement de valeur des données en entrée des registres avant d'être stable. C'est le temps du bit qui a le chemin le plus long pour aller d'un registre à un autre.
- t_s : le temps pendant lequel la donnée doit rester stable en entrée d'un registre pour être échantillonnée correctement.

- L'équation suivante doit toujours être vérifiée :

$$t_{\text{clk}} > t_{\text{Max}} + t_s \quad (1)$$

Si un attaquant vient modifier le signal de l'horloge interne du circuit, cette équation (1) peut ne plus être vérifiée. On parle d'injections de fautes par glitch d'horloge [12, 13, 14].

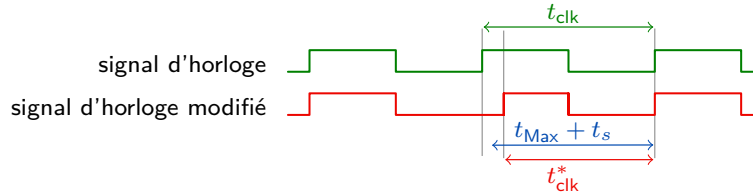


FIGURE 5 – Clock glitch

3 Effet d'une injection de faute

Les conséquences d'une faute physique sont multiples. En cas d'échec, le circuit peut être détruit, c'est l'effet indésirable d'un mauvais réglage du laser ou du générateur de rayonnement électromagnétique. Une faute réussie va perturber le flot de données ou le flot de contrôle.

3.1 Faute sur les données

Un modèle de faute par perturbation du flot de données représente les effets d'une attaque modifiant les données manipulées par le programme. Les modèles de fautes classiques sont les suivant.

- **Collage** : au niveau binaire ou assembleur, il s'agit de la mise à 0 ou à 1 d'un ou plusieurs bits d'une cellule mémoire ou d'un registre. On parle de la corruption de la mémoire ou d'un registre.
- **Bit-flip** : ce modèle représente une modification de la valeur d'affectation d'une variable ou d'une valeur intermédiaire.

3.2 Faute sur les instructions

Un modèle de fautes par perturbation du flot de données représente les effets d'une attaque modifiant les instructions du programme. Les modèles de fautes possibles sont les suivant.

- **Saut d'instruction** : l'instruction n'est pas exécutée. Par exemple on peut sauter la vérification d'un code PIN.
- **Remplacement d'instruction** : l'instruction est remplacée par une autre. On peut l'utiliser par exemple pour modifier un test conditionnel avant un branchement par exemple.

4 Exploitation des fautes

4.1 DFA

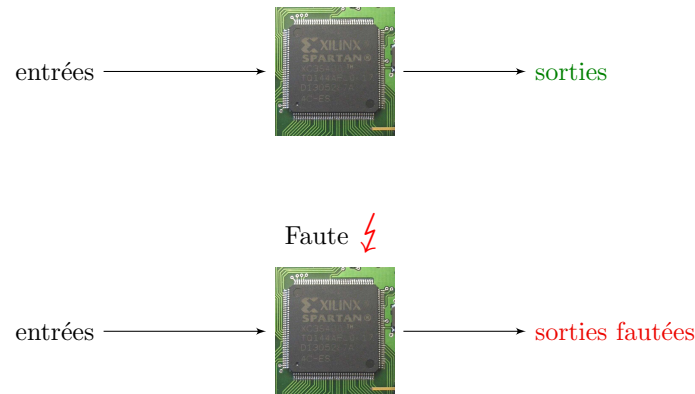


FIGURE 6 – Principe de la DFA

Les attaques de type **differential fault analysis (DFA)** [15, 16, 17, 18], que l'on peut traduire en français par analyse des différences liées aux fautes, exploitent des résultats erronés obtenus par injections de fautes (voir FIGURE 6). L'idée est d'analyser la différence avec les résultats obtenus sans injections de faute. Ce sont des attaques très fréquentes notamment en rétro-conception d'algorithme ou dans un contexte de cryptanalyse.

4.2 Exemples d'exploitation de fautes contre une vérification code PIN

Dans le cas d'une vérification de code PIN, une faute sur les données peut permettre d'augmenter la valeur du compteur d'essais. Avec une injection de faute par laser un attaquant, peut également forcer la valeur de sortie à **vrai**. Ainsi l'algorithme de vérification retourne vrai quelque soit le PIN candidat proposé.

Une faute sur les instructions peut sauter et donc ne pas effectuer l'instruction qui décrémente le compteur d'essais. Pire on peut imaginer une faute qui saute carrément la vérification de code PIN.

En pratique, une bonne implémentation de vérification de code PIN est associée de drapeaux. Ces derniers doivent vérifier que toutes les instructions ont bien été réalisées [19].

5 Contre-mesures

5.1 Redondance

Dans le cas des attaques par injections de fautes, la plupart des contre-mesures visent principalement à ajouter de la **redondance** dans le code à exécuter. Cette redondance peut être :

- **spatiale** : plusieurs éléments distincts du circuit réalisent la même opération en parallèle,
- **temporelle** : une même opération est répétée plusieurs fois.

La redondance a tout de même ses limites notamment dû au fait qu'un attaquant peut injecter plusieurs fautes durant l'exécution d'un même programme [14, 11].

Une autre forme de redondance est l'utilisation de codes détecteurs d'erreurs ou mieux des codes correcteurs d'erreurs (pour se protéger du déni-de-service). Il s'agit dans ce cas de modifier l'algorithme pour intégrer ces codes. Il est par exemple possible de modifier les sous-fonctions d'un programme pour calculer la modification des valeurs intermédiaires en parallèle d'un bit de parité sur ces valeurs intermédiaires.

5.2 Bouclier

Pour protéger une puce, il est possible d'utiliser un **bouclier** (shield en anglais). Il s'agit de pistes métalliques positionnées au dessus et en dessous de la puce. Ces pistes fournissent d'une part une protection électromagnétique passive pour les pistes en dessous. De plus, pour éviter que l'attaquant ne troue le bouclier (en utilisant un Focused Ion Beam par exemple), il est possible de faire transiter des données pseudo-aléatoires dans les pistes du bouclier. On vérifie à l'autre bout que les données soient toujours présentes, correctes et à des latences nominales pour s'assurer de l'intégrité du bouclier.

6 Conclusion

Les attaques par injections de fautes sont redoutables. Il existe différentes techniques pour injecter des fautes, laser, modification d'horloge et rayonnement électromagnétique. Dans le cas du code PIN, permettent de contourner la sécurité mise en place. Il existe d'autres types d'attaques dites DFA qui analysent la différence entre une sortie fautée et une sortie normale. Il existe des contremesures logicielles pour rendre ces attaques plus difficiles à réaliser. Néanmoins, seules les contremesures matérielles sont réellement efficaces.

Acronymes

PIN personal identification number. 2, 6, 7, 9

DFA differential fault analysis. 2, 7, 9

FIA fault injection attack. 3

LHS Laboratoire Haute Sécurité. 5

SET single event transient. 4

Références

- [1] Ronan Lashermes. Attaques par faute.
- [2] Cyril Roscian, Jean-Max Dutertre, and Assia Tria. Frontside laser fault injection on cryptosystems-application to the aes'last round. In *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 119–124. IEEE, 2013.
- [3] Marc Lacruche, Nicolas Borrel, Clement Champeix, Cyril Roscian, Alexandre Sarafianos, Jean-Baptiste Rigaud, Jean-Max Dutertre, and Edith Kussener. *Laser fault injection into sram cells : Picosecond versus nanosecond pulses*. PhD thesis, 2015.
- [4] Michel Agoyan, Jean-Max Dutertre, Amir-Pasha Mirbaha, David Naccache, Anne-Lise Ribotta, and Assia Tria. Single-bit dfa using multiple-byte laser fault injection. In *2010 IEEE International Conference on Technologies for Homeland Security (HST)*, pages 113–119. IEEE, 2010.
- [5] Mathieu Dumont, Pierre-Alain Moëllic, Raphael Viera, Jean-Max Dutertre, and Rémi Bernhard. An overview of laser injection against embedded neural network models. In *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, pages 616–621. IEEE, 2021.
- [6] Raphael Viera, Jean-Max Dutertre, Mathieu Dumont, and Pierre-Alain Moëllic. Permanent laser fault injection into the flash memory of a microcontroller. In *2021 19th IEEE International New Circuits and Systems Conference (NEWCAS)*, pages 1–4. IEEE, 2021.
- [7] Cyril Roscian. *Cryptanalyse physique de circuits cryptographiques à l'aide de sources LASER*. PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne, 2013.
- [8] Nicolas Moro. *Security of assembly programs against fault attacks on embedded processors*. PhD thesis, Université Pierre et Marie Curie - Paris VI, 2014. <https://tel.archives-ouvertes.fr/tel-01147122>.
- [9] Pierre Bayon, Lilian Bossuet, Alain Aubert, Viktor Fischer, François Poucheret, Bruno Robisson, and Philippe Maurine. Contactless electromagnetic active attack on ring oscillator based true random number generator. In *International Workshop on Constructive Side-Channel Analysis and Secure Design*, pages 151–166. Springer, 2012.
- [10] Amine Dehbaoui, Jean-Max Dutertre, Bruno Robisson, and Assia Tria. Electromagnetic transient faults injection on a hardware and a software implementations of aes. In *2012 Workshop on Fault Diagnosis and Tolerance in Cryptography*, pages 7–15. IEEE, 2012.
- [11] Vanthanh Khuat, Oualid Trabelsi, Laurent Sauvage, and Jean-Luc Danger. Multiple and reproducible fault models on micro-controller using electromagnetic fault injection. In *2021 IEEE International Joint EMC/SI/PI and EMC Europe Symposium*, pages 667–672. IEEE, 2021.
- [12] Loic Zussa, Jean-Max Dutertre, Jessy Clediere, and Assia Tria. Power supply glitch induced faults on fpga : An in-depth analysis of the injection mechanism. In *2013 IEEE 19th International On-Line Testing Symposium (IOLTS)*, pages 110–115. IEEE, 2013.
- [13] Josep Balasch, Benedikt Gierlichs, and Ingrid Verbauwhede. An in-depth and black-box characterization of the effects of clock glitches on 8-bit mcus. In *2011 Workshop on Fault Diagnosis and Tolerance in Cryptography*, pages 105–114. IEEE, 2011.
- [14] Ludovic Claudepierre, Pierre-Yves Péneau, Damien Hardy, and Erven Rohou. Traitor : a low-cost evaluation platform for multifault injection. In *Proceedings of the 2021 International Symposium on Advanced Security on Software and Systems*, pages 51–56, 2021.
- [15] Christophe Giraud. DFA on AES. In H. Dobbertin and V. Rijmen and A. Sowa, editor, *Advanced Encryption Standard - AES*, volume 3373 of *Lecture Notes in Computer Science*. Springer, 2005.
- [16] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In *CRYPTO*, 1997.
- [17] Ronan Lashermes, Guillaume Reymond, Jean-Max Dutertre, Jacques Fournier, Bruno Robisson and Assia Tria. A DFA on AES Based on the Entropy of Error Distributions. In *FDTC*, 2012.
- [18] Gilles Piret and Jean-Jacques Quisquater. A differential fault attack technique against spn structures, with application to the aes and khazad. In *International workshop on cryptographic hardware and embedded systems*, pages 77–88. Springer, 2003.
- [19] Lionel Riviere. *Sécurité des implémentations logicielles face aux attaques par injection de faute sur systemes embarqués*. PhD thesis, PhD thesis, Telecom Paris Tech, 2015.



IMT Atlantique Bretagne - Pays de la Loire - www.imt-atlantique.fr

Campus de Brest
Technopôle Brest-Iroise
CS 83818
29238 Brest Cedex 03
T +33 (0)2 29 00 11 11
F +33 (0)2 29 00 10 00

Campus de Nantes
4, rue Alfred Kastler - La Chantrerie
CS 20722
44307 Nantes Cedex 03
T +33 (0)2 51 85 81 00
F +33 (0)2 51 85 81 99

Campus de Rennes
2, rue de la Châtaigneraie
CS 17607
35576 Cesson Sévigné Cedex
T +33 (0)2 99 12 70 00
F +33 (0)2 99 12 70 08