

Performance Evaluation of Protocols Resilient to Physical Attacks



MINISTÈRE DE LA DÉFENSE

Sylvain GUILLEY^{*†}, Laurent SAUVAGE^{*†}, Jean-Luc DANGER^{*†},
Nidhal SELMANE^{*} and Denis RÉAL[‡].

^{*}TELECOM-ParisTech, [†]Secure-IC S.A.S., [‡]French DoD (Information Superiority)

Contact: <sylvain.guilley@TELECOM-ParisTech.fr>

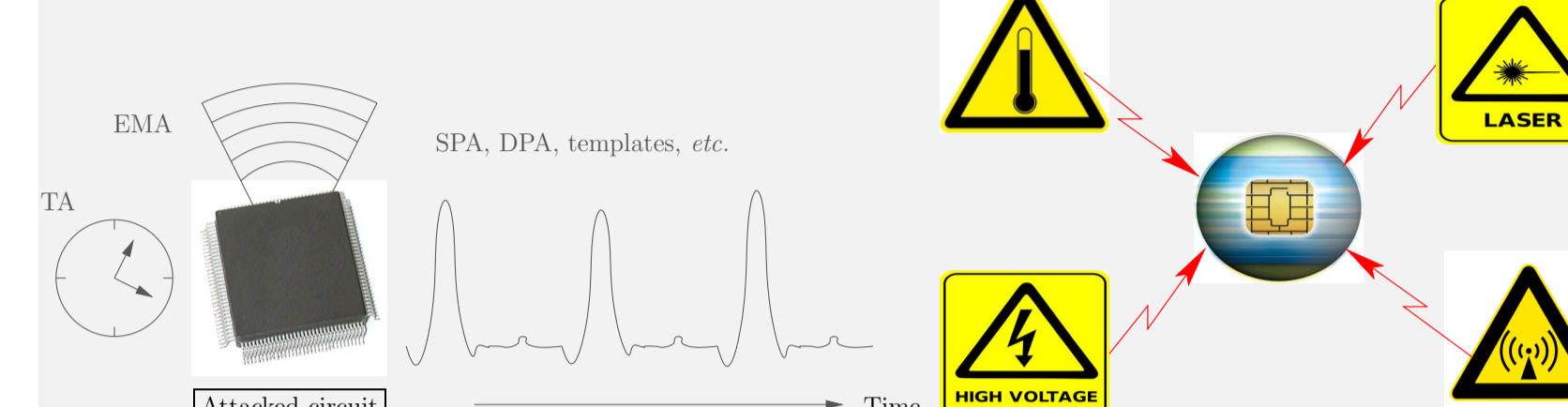


Introduction

Context

- Cryptographic implementations are vulnerable to physical attacks.
- Many countermeasures to resist them have been proposed in the past.
- However, they are often too specific to a given attacker.
- Therefore, a new trend consists in making cryptographic implementations resilient to physical attacks.
- This strategy makes it possible to prove the countermeasure against all possible types of attackers.
- For a given security objective, they all permit to reach the same security level.
- Therefore, they differentiate only according to their efficiency.

Passive and Active Attacks



Two contributions

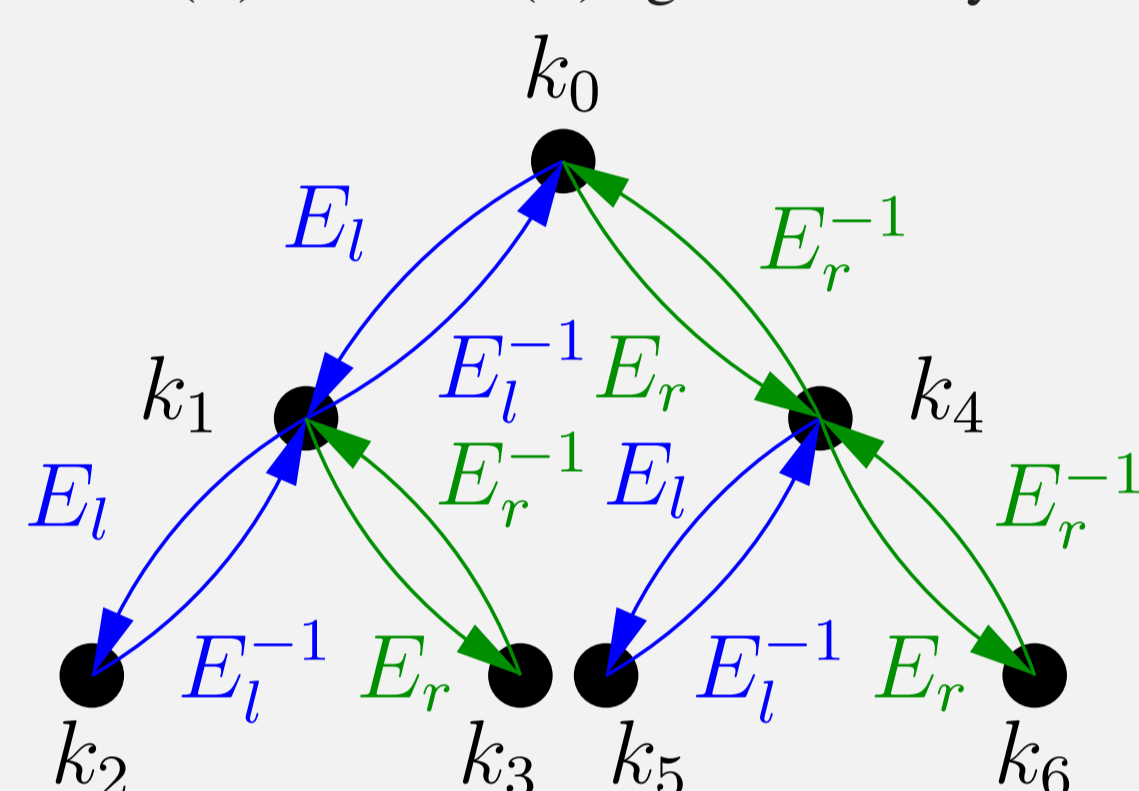
1. Protection against both passive and active attacks.
2. Improvements in terms of:
 - I/O bandwidth and
 - computational performance.

State-of-the-Art

Indexed Key Update (IKU)

IKU is secure against passive attacks.

- Alice (A) and Bob (B) agree on a key thanks to:



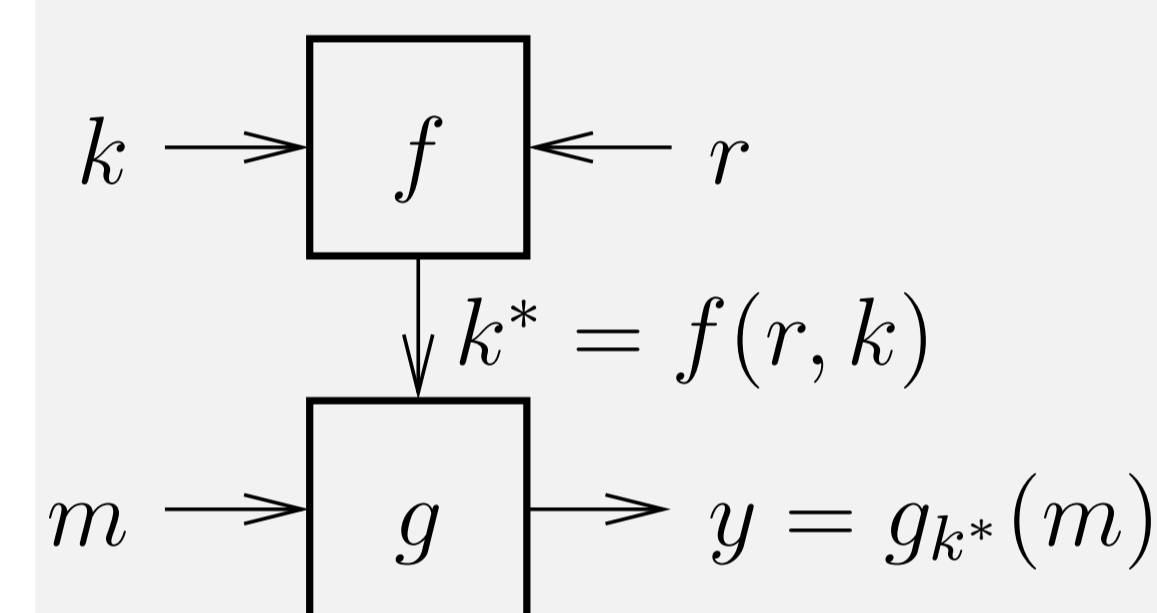
- Then, they use the cryptographic block cipher g with the current indexed key.

P. C. Kocher. "Leak-resistant cryptographic indexed key update", United States Patent 6,539,092, 2003.

Fresh Re-Keying (FRK)

FRK is secure against passive attacks.

- The session key is determined randomly;
- Then, the cryptographic block cipher g is used with the agreed key.



M. Medwed, F.-X. Standaert, J. Großschädl, and F. Regazzoni. "Fresh Re-Keying: Security against Side-Channel and Fault Attacks for Low-Cost Devices". In AFRICACRYPT'2010.

Comparison of IKU & FRK

Step	Single-block IKU
#1	A sends C_A → B receives C_A A receives C_B ← B sends C_B
#2	A computes k_C as $k_C = k_{\max}(C_A, C_B)$ B computes k_C as $k_C = k_{\max}(C_A, C_B)$
#3	A receives m ← B sends m
#4	A computes $y = g_{k_C}(m)$
#5	A sends y → B receives y

Step	Single-block FRK
#1	A sends r → B receives r
#2	A computes k^* as $k^* = f(r, k)$ B computes k^* as $k^* = f(r, k)$
#3	A receives m ← B sends m
#4	A computes $y = g_{k^*}(m)$
#5	A sends y → B receives y

Fault Injection Resilience (FIR)

FIR is secure against active attacks.

Algorithm 1: Probabilistic Encryption Algorithm built on top of block cipher g , non-protected against FIAs.

Input : A plaintext x to be encrypted with the key k , shared between the client and the server.
Output: A ciphertext along with a random number.

- 1 Determine a random number r of the same size as x ; /* This number will whiten x */.
- 2 Return the couple $(y = g_k(x \oplus r), r)$.

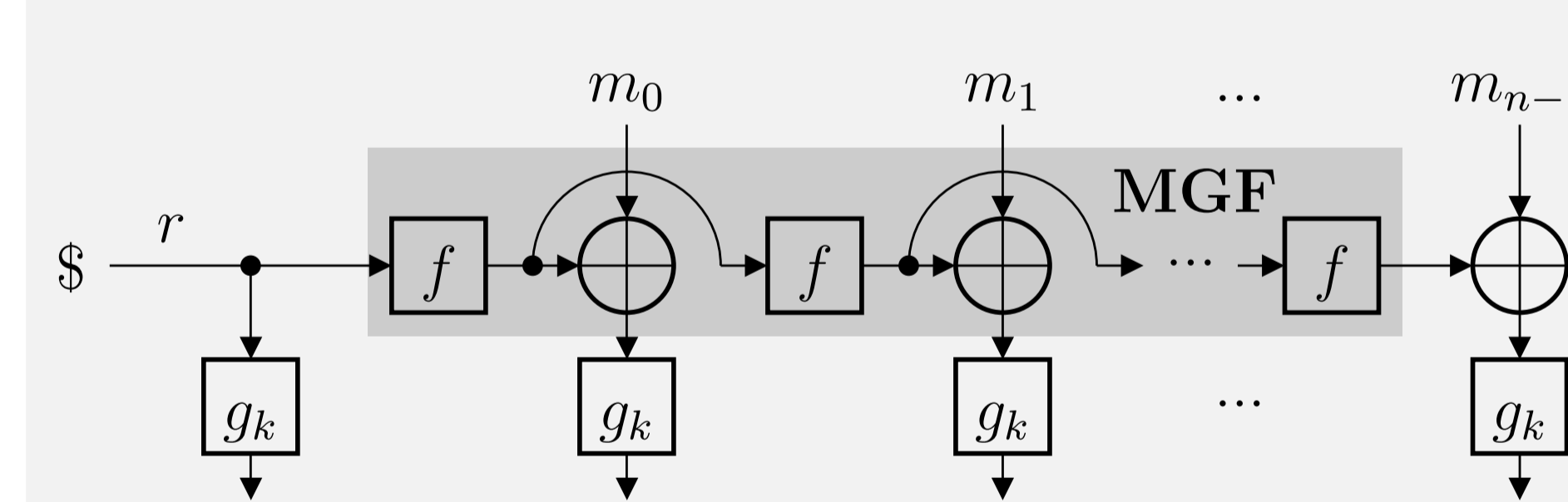
S. Guilley, L. Sauvage, J.-L. Danger, and N. Selmane. "Fault Injection Resilience", in FDTC'2010.

New Resilient Schemes Secure Against Passive and Active Attacks

Open Issue and Proposed Solution

- Problem: asymmetry between passive and active attacks:
 - against passive attacks, a key can be used $\eta > 1$ times,
 - against active attacks, without protections, two encryptions enable an attack (DFA).
- Solution: prevent the attacker from choosing the plaintext.
 - this does not forbid passive attacks, since ciphertext attacks can be done, but
 - against active faults, the attack hypotheses are denied.

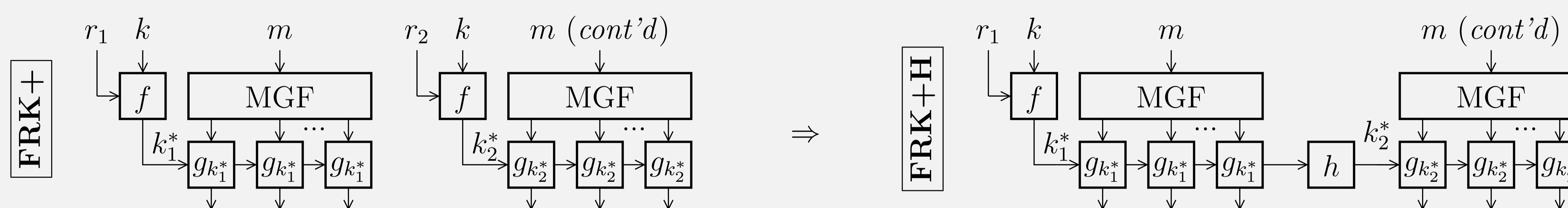
Blinding the Plaintext thanks to a MGF



Resilient MGF, used as partial AONT (All-or-Nothing Transform).

Optimizations

- IKU+ and FRK+**: Because of the MGF (Mask Generation Function) the attacker cannot choose the plaintext.
- Trick**: Replace the strong encryption function in IKU by a lightweight equivalent: IKU* and IKU+*.
- FRK+H**: instead of drawing a new key when necessary, the partners A and B can simply hash it with a lightweight algorithm h (hence the name FRK+H).



Summary

The differences between IKU+* and FRK+H fade away when $n \rightarrow +\infty$.

Notations:

- D is the IKU key tree depth;
- B is the size in bits of the g block cipher;
- n is the number of blocks to encrypt;
- η is the number of queries for a passive attack to be successful;
- $[X]$ is the performance of operation X ;
- E and g are cryptographic-grade operations, whereas
- f is a lightweight operation.

Protocol	I/O [bit]	Performance [computation time]
1-bl. IKU	$2D + 2B$	$(2D - 2)[E] + [g]$
1-bl. FRK	$3B$	$[f] + [g]$
n -bl. IKU	$2D + 2Bn$	$(2D - 3 + n)[E] + n[g]$
n -bl. FRK	$3Bn$	$n \cdot ([f] + [g])$
n -bl. IKU+	$2D + (1 + \frac{\eta}{\eta-1})nB$	$(2D - 3 + \frac{\eta}{\eta-1})[E] + n(\frac{\eta}{\eta-1}[g] + [f])$
n -bl. FRK+	$(\frac{2\eta}{\eta-1})nB$	$\frac{\eta}{\eta-1}[f] + n(\frac{\eta}{\eta-1}[g] + [f])$
n -bl. IKU*	$2D + 2Bn$	$(2D - 3 + n)[f] + n[g]$
n -bl. IKU+*	$2D + (1 + \frac{\eta}{\eta-1})nB$	$(2D - 3 + \frac{\eta}{\eta-1})[f] + n(\frac{\eta}{\eta-1}[g] + [f])$
n -bl. FRK+H	$B + (1 + \frac{\eta}{\eta-1})nB$	$\frac{\eta}{\eta-1}[f] + n(\frac{\eta}{\eta-1}[g] + [f])$

Additional Requirements

- IKU & IKU* require NVM but no TRNG;
- IKU+ & IKU+* require both NVM and TRNG;
- FRK, FRK+ & FRK+H require TRNG but no NVM.

Acknowledgements

- ANR Project ARPEGE ANR-09-SEGI-013: SecReSoC (Secured Reconfigurable System on Chip).
- Housseem MAGHREBI, for the presentation of this poster.