A Generic Countermeasure Against Fault Injection Attacks on Asymmetric Cryptography

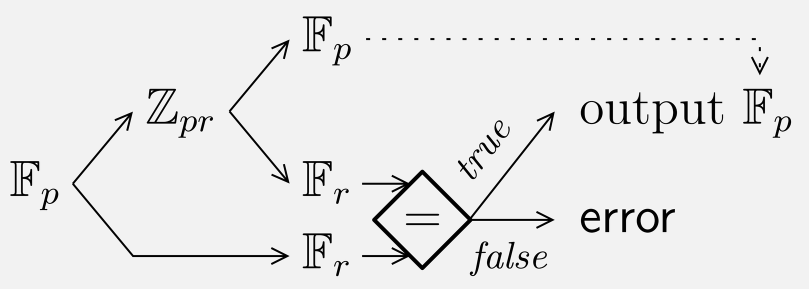
Using Modular Extension to Provably Protect Elliptic Curve Cryptography in Theory and Practice

Pablo Rauzy, Martin Moreau, Sylvain Guilley, and Zakaria Najm

{*firstname.lastname*}@telecom-paristech.fr

Abstract. We propose a new *modular extension* based countermeasure for elliptic curve scalar multiplication (ECSM) that we prove correct and secure. The fault non-detection probability of our proposed countermeasure is inversely proportional to the security parameter. We implement an ECSM protected with our countermeasure on an ARM Cortex-M4 microcontroller: a systematic fault injection campaign for several values of the security parameter confirms our theoretical prediction and the security of the obtained implementation and provides figures for practical performance.

1. Modular extension protection scheme



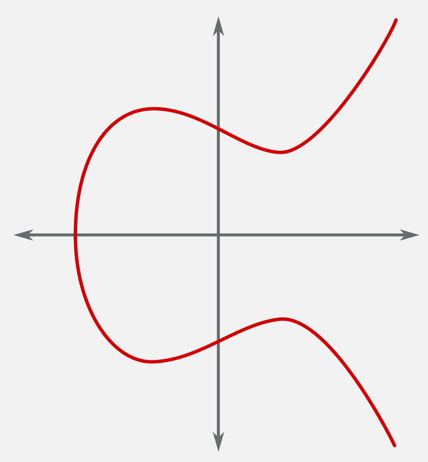
Computation integrity.
- Verifying signature is costly.
- Repeating the whole computation too.

Cost-effective redundancy.
- Compute in direct product $\mathbb{Z}_{pr}$ and in $\mathbb{F}_r$.
- Invariant: $\mathbb{Z}_{pr} \bmod r = \mathbb{F}_r$.

2. Elliptic curves and the projective plane

Elliptic Curve $E(\mathbb{F}_p)$.
- A "point at infinity" denoted by $\mathcal{O}$.
- A set of points (x, y) satisfying an equation of the form: $y^2 = x^3 + ax + b$.

Projective coordinates.
- To avoid divisions, a third coordinate Z is added.
- Projective $(X : Y : Z) \iff$ affine $(X/Z, Y/Z)$.
- New equation: $Y^2Z = X^3 + aXZ^2 + bZ^3$.
- By convention, $\mathcal{O}$ is represented by $(X : Y : 0)$.



3. Test-free elliptic curve scalar multiplication (ECSM)

Algorithm: TF-ECDBL(Q, n).
Input : $Q = (X_1 : Y_1 : Z_1) \in E(\mathbb{Z}_n)$
Output: $(X : Y : Z) = 2Q \in E(\mathbb{Z}_n)$

~~**if** Q *is* $\mathcal{O}$ **then return** Q~~

$A = 3(X_1^2 + 2aZ_1(X_1 + Z_1))$

$X = 2Y_1Z_1(A^2 - 8X_1Z_1Y_1^2)$
$Y = A(12X_1Z_1Y_1^2 - A^2) - 8Z_1^2Y_1^4$
$Z = 8Z_1^3Y_1^3$

return $(X : Y : Z)$

TF-good scalar.
Let $P \in E(\mathbb{Z}_n)$, $k > 0$,
k is TF-good w.r.t. P if and only if $\forall i > 1 \in \mathbb{N}$:
- $ord(P) \nmid \lfloor k/2^i \rfloor$,
- $ord(P) \nmid \lfloor k/2^i \rfloor - 1$, when $k_i = 1$,
- $ord(P) \nmid \lfloor k/2^i \rfloor - 2$, when $k_i = 1$.

→ TF-ECSM$_{\mathbf{L2R}}$ is partially correct [3].
Let $P \in E(\mathbb{Z}_n)$, and $k > 0$,
if k is TF-good wrt P and $E(\mathbb{Z}_n)$ then:
- TF-ECSM$_{\mathrm{L2R}}(P, k, n)$ = ECSM$_{\mathrm{L2R}}(P, k, n)$,

else:
- TF-ECSM$_{\mathrm{L2R}}(P, k, n) = \mathcal{O}$.

Algorithm: TF-ECADD(Q, P, n).
Input : $Q = (X_1 : Y_1 : Z_1)$,
$Q = (X_2 : Y_2 : Z_2) \in E(\mathbb{Z}_n)$
Output: $(X : Y : Z) = Q + P \in E(\mathbb{Z}_n)$

~~**if** Q *is* $\mathcal{O}$ **then return** P~~
~~**if** P *is* $\mathcal{O}$ **then return** Q~~
~~**if** $Q = -P$ **then return** $\mathcal{O}$~~
~~**if** $Q = P$ **then return** $2P$~~

$A = Y_2Z_1 - Y_1Z_2$
$B = X_2Z_1 - X_1Z_2$
$C = Z_1Z_2A^2 - (X_1Z_2 + X_2Z_1)B^2$

$X = BC$
$Y = A(X_1Z_2B^2 - C) - Y_1Z_2B^3$
$Z = Z_1Z_2B^3$

return $(X : Y : Z)$

Algorithm: TF-ECSM$_{\mathrm{L2R}}(P, k, n)$.
Input : $P \in E(\mathbb{Z}_n)$, $k > 0$
Output: $Q = [k]P \in E(\mathbb{Z}_n)$

$Q = \mathcal{O}$
for $i = \lceil \log_2 k \rceil - 1, \ldots, 0$ **do**
  Q = TF-ECDBL(Q, n)
  if k_i **then** Q = TF-ECADD(Q, P, n)
return Q

4. State of the art: BOS [1] and BV [2]

Algorithm: ECSM protected with BOS countermeasure BOS(P, k, p).
Input : $P \in E(\mathbb{F}_p)$, $k > 0$
Output: $Q = [k]P \in E(\mathbb{F}_p)$

Choose a small prime r, a curve $E(\mathbb{F}_r)$, and a point P_r on that curve.
Determine the combined curve $E(\mathbb{Z}_{pr})$ and point P_{pr} using the CRT.

$(X_{pr} : Y_{pr} : Z_{pr})$ = ECSM(P_{pr}, k, pr)
$(X_r : Y_r : Z_r)$ = ECSM(P_r, k, r)

if $(X_{pr} \bmod r : Y_{pr} \bmod r : Z_{pr} \bmod r) = (X_r : Y_r : Z_r)$ **then**
  return $(X_{pr} \bmod p : Y_{pr} \bmod p : Z_{pr} \bmod p)$
else
  return error

Algorithm: ECSM protected with BV countermeasure BV(P, k, p).
Input : $P \in E(\mathbb{F}_p)$, $k > 0$
Output: $Q = [k]P \in E(\mathbb{F}_p)$

Choose a small random integer r.
Compute the combined curve $E'(\mathbb{Z}_{pr})$.

$(X_{pr} : Y_{pr} : Z_{pr})$ = ECSM(P, k, pr)

if $(X_{pr} \bmod r : Y_{pr} \bmod r : Z_{pr} \bmod r) \in E'(\mathbb{Z}_{pr}) \bmod r$ **then**
  return $(X_{pr} \bmod p : Y_{pr} \bmod p : Z_{pr} \bmod p)$
else
  return error

→ BOS is incorrect.
When k is TF-bad (not TF-good) wrt P_r and $E(\mathbb{F}_r)$, BOS returns error even without faults.

→ BV is incorrect.
When k is TF-bad wrt P and $E'(\mathbb{F}_r)$, BV returns error even without faults in very specific cases.

6. RMGN [4]

Algorithm: TF-ECSM with modular extension protection RMGN(P, k, p).
Input : $P \in E(\mathbb{F}_p)$, $k > 0$
Output: $Q = [k]P \in E(\mathbb{F}_p)$

Choose a small prime r.

$(X_{pr} : Y_{pr} : Z_{pr})$ = TF-ECSM(P, k, pr)
$(X_r : Y_r : Z_r)$ = TF-ECSM$(P \bmod r, k, r)$

if $(X_{pr} \bmod r : Y_{pr} \bmod r : Z_{pr} \bmod r) = (X_r : Y_r : Z_r)$ **then**
  return $(X_{pr} \bmod p : Y_{pr} \bmod p : Z_{pr} \bmod p)$
else
  return error

→ RMGN is correct.
RMGN always returns the correct result.
However, its resistance to fault attack is weakened in the case of TF-bad scalar.

→ TF-bad scalar probability is low is practice.
The probability of a scalar k to be TF-bad wrt a point $P \in E(\mathbb{F}_r)$ is

$$\mathbb{P}_{\text{TF-bad}_P}(k) \approx 1 - \left(1 - \frac{1}{ord(P)}\right)^{\lceil \log_2 k \rceil - \lceil \log_2 ord(P) \rceil} = O\left(\frac{1}{ord(P)}\right).$$

In practice when r is on 32 bits, $\mathbb{P}_{\text{TF-bad}_P}(k) \approx 10^{-8}$.

7. Formal security analysis of RMGN

→ Inversion in $\mathbb{Z}_{pr}$ is possible in the modular extension context.
To invert z in $\mathbb{F}_p$ while computing in $\mathbb{Z}_{pr}$, one has:
- $z = 0 \bmod r \implies (z^{p-2} \bmod pr) \equiv z^{-1} \mod p$,
- otherwise $(z^{-1} \bmod pr) \equiv z^{-1} \mod p$.

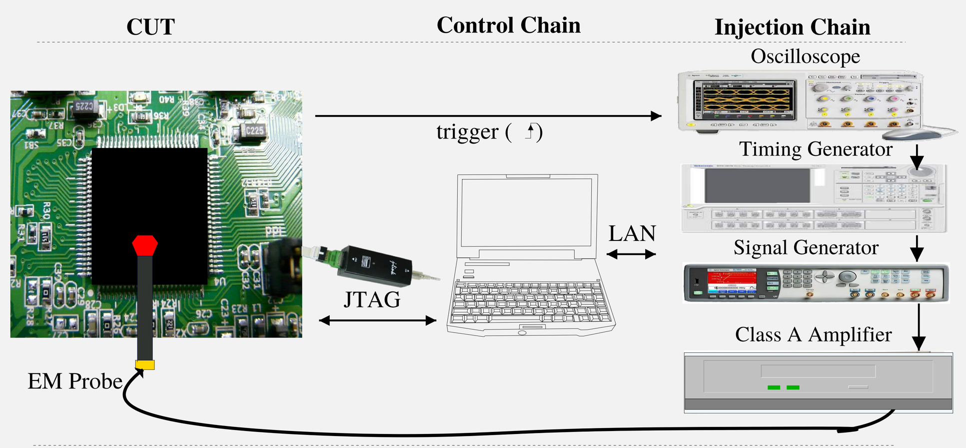
Fault model.
Each injected fault can be:
- randomizing or zeroing any intermediate variable;
- skipping any number of consecutive instructions.

Secure algorithm.
An algorithm is secure if:
- it returns the good result when there is no faults; and
- it return either the good result or error otherwise, with an overwhelming probability.

→ RMGN is secure.
The probability of non-detection $\mathbb{P}_{\text{n.d.}} = O(\frac{1}{r})$.

Security parameter.
Thus, r is the security parameter. It should be prime, private, and dynamically chosen.

8. Practical case study with RMGN



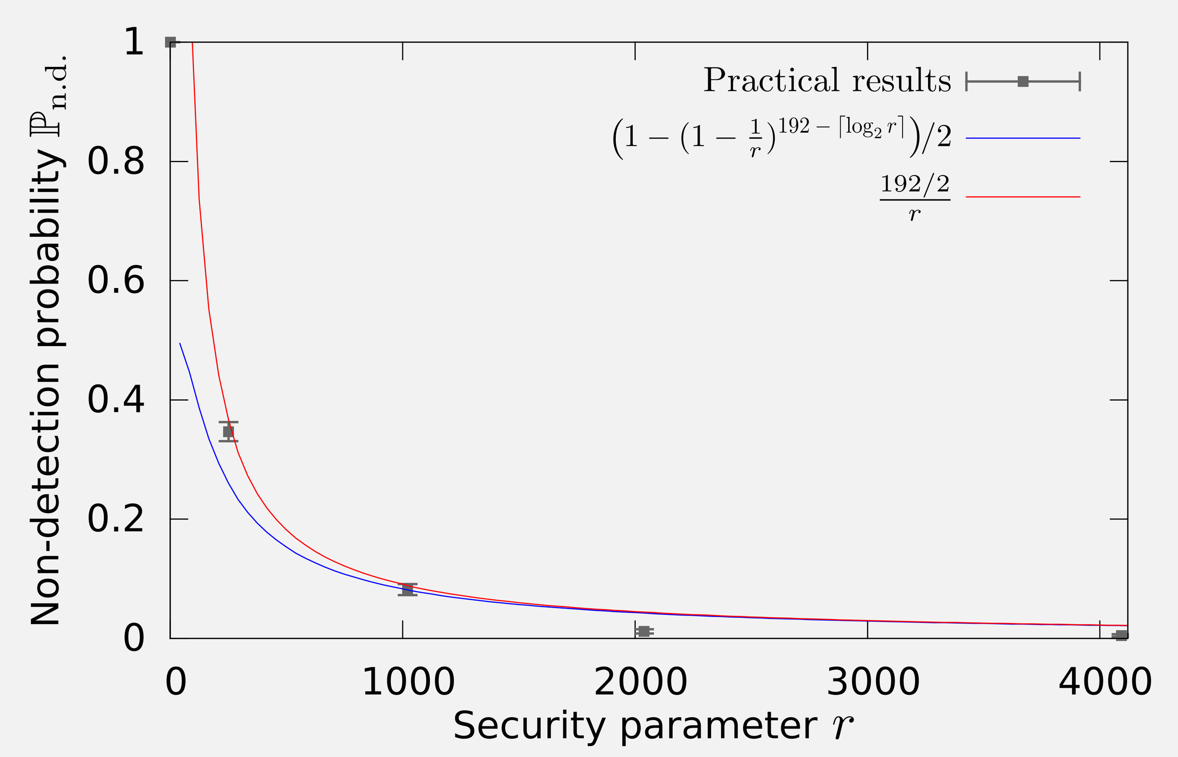
Setup.
- ARM Cortex-M4 STM32.
- NIST curve P-192.
- C and `mini-gmp`.
- Compiled with `gcc -O0`.

Security results.

r value	r size (bit)	positive % true	positive % false	negative % true	negative % false
1	1	0.00	0.00	2.74	97.3
251	8	63.7	0.00	2.56	33.8
65521	16	97.8	0.00	2.21	0.00
4294967291	32	97.2	0.00	2.81	0.00
18446744073709551557	64	99.8	0.00	0.21	0.00

Performance results.

r value	r size (bit)	time (ms) $\mathbb{Z}_{pr}$	time (ms) $\mathbb{F}_r$	time (ms) test	over head
1	1	683	24	$\ll 1$	$\times 1.04$
251	8	883	91	$\ll 1$	$\times 1.43$
65521	16	883	189	$\ll 1$	$\times 1.56$
4294967291	32	832	172	$\ll 1$	$\times 1.47$
18446744073709551557	64	996	246	$\ll 1$	$\times 1.82$



[1] J. Blömer, M. Otto, and J.-P. Seifert. Sign Change Fault Attacks on Elliptic Curve Cryptosystems. In L. Breveglieri, I. Koren, D. Naccache, and J.-P. Seifert, editors, *Fault Diagnosis and Tolerance in Cryptography*, 2006.
[2] Y.-J. Baek and I. Vasyltsov. How to Prevent DPA and Fault Attack in a Unified Way for ECC Scalar Multiplication - Ring Extension Method. In E. Dawson and D. Wong, editors, *Information Security Practice and Experience*, 2007.
[3] J. Fan, B. Gierlichs, and F. Vercauteren. To Infinity and Beyond: Combined Attack on ECC Using Points of Low Order. In B. Preneel and T. Takagi, editors, *CHES*, 2011.
[4] P. Rauzy, M. Moreau, S. Guilley, and Z. Najm. A Generic Countermeasure Against Fault Injection Attacks on Asymmetric Cryptography. IACR ePrint, 2015. `http://eprint.iacr.org/2015/882`