ECC Protections against both Observation and Pertubation Attacks

Audrey LUCAS[1,2] and Arnaud TISSERAND[1,3]
[1]CNRS – [2]IRISA UMR 6074 / [3]Lab-STICC UMR 6285

Among physical attacks, two types are considered as important threats for embedded crypto-processors: *observation attacks* [1] (or side channel attacks) and *perturbation attacks* [2] (or fault injection attacks). The first ones use external measurements of the circuit execution to guess secrets (*e.g.* timings, power consumption, electromagnetic radiation). The second ones disrupt the circuit and exploit its unspecified behavior, directly or not, to deduce secrets. In most of state of the art works, countermeasures only protect the crypto-processor from one type of attacks (*i.e.* side channel or fault attacks). However, in many cases the crypto-system stays or becomes vulnerable to the other type of attacks.

In this work, we focus on protections against both types of attacks simultaneously for scalar multiplication in *elliptic curve cryptography* (ECC). This is the main operation: $k \times P$ where k is a scalar (the secret key in some primitives) and P a public curve point. Scalar multiplication is vulnerable to observation attacks when using weak algorithms where *point addition* and *point doubling* operations have different cost or behavior [3]. One countermeasure consists in using *double and add always* algorithm [4]. Unfortunately, this algorithm is weak to fault attacks (injecting a fault during a dummy point addition does not impact the final result, revealing that the target operation was a dummy one). A common protection against fault attacks on ECC is to verify if the current point is on the curve (by applying its coordinates into the curve equation) [5]. Verification of the current point detects an attack at a specific time during scalar multiplication. For instance, one can choose to only verify the final point of the scalar multiplication for a very low overhead (2 field multiplications, 4 field additions and 1 field multiplication by a scalar). But the secret key may leak before the end. More frequent verification is possible to detect attacks as soon as possible. For instance, one can verify the result after the systematic point doubling in each iteration. This leads to regular scalar multiplication for some coordinate types.

Unlike the current point, the scalar k is not protected by this verification method. To alleviate this vulnerability, we developed a new countermeasure to protect k. Our aim is to count that at the ith iteration, there is a point addition or not. Our protection is efficient against fault attacks named *bit flips*. The protection cost is about $\frac{\log_2(k)}{2}$ integer additions for one scalar multiplication. We are working on ways to perform this verification without leakage to avoid observation attacks. We combined and tested these two countermeasures on different coordinates types for Weierstrass curves. Table 1 reports the computation time overheads in the worst case (*i.e.*, a verification after each point doubling). The cost of this type of regular protection is quite small, but it can be reduced if one choose to only verify the point coordinates less frequently (*e.g.*, w-NAF algorithms).

Although the scalar protection is very cheap, it does not protect against *stuck-at* faults on k digits. We are working on protections against this type of fault without observation leakage.

Algorithms	Coordinates		
	Affine	Jacobian	Projective
Double and add	5.7%	17.2%	9.8%
Montgomery ladder	6.0%	18.0%	8.8%
NAF	6.1%	16.9%	11.2%
w-NAF	7.2%	21.2%	12.5%

Table 1: Computation overheads in the worst case for Weierstrass curves.

Acknowledgments

This work is partly funded by DGA-PEC and the HAH project (Labex CominLab and Lebesgue, Brittany Region, `http://h-a-h.inria.fr/`).

References

[1] S. Mangard, E. Oswald, and T. Popp. *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007.

[2] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan. The sorcerer's apprentice guide to fault attacks. *Proceedings of the IEEE*, 94(2):370–382, February 2006.

[3] Eric Brier and Marc Joye. Weierstraß Elliptic Curves and Side-Channel Attacks. In *Proc. Public Key Cryptography - PKC*, pages 335–345, Paris, France, 2002.

[4] Jean-Sébastien Coron. Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems. In *Proc. Cryptographic Hardware and Embedded Systems- CHES*, pages 292–302, Worcester, MA, USA, 1999.

[5] I. Biehl, B. Meyer, and V. Müller. Differential Fault Attacks on Elliptic Curve Cryptosystems. In *Proc. Advances in Cryptology - CRYPTO*, pages 131–146, Santa Barbara, California, USA, 2000.