Fault attacks against the Miller algorithm in Edwards Coordinates

Nadia El Mrabet

Fault attacks against the Miller algorithm in Edwards Coordinates

Nadia El Mrabet

GREYC - LMNO, University of Caen, France
nadia.el_mrabet@info.unicaen.fr

Abstract. Initially, the use of pairings did not involve any secret entry. However in an Identity Based Cryptographic protocol, one of the two entries of the pairing is secret, so fault attack can be applied to Pairing Based Cryptography to find it. In [18], the author shows that Pairing Based Cryptography in Weierstrass coordinates is vulnerable to a fault attack. The addition law in Edwards coordinates is such that the exponentiation in Edwards coordinates is naturally protected to Side Channel attacks. We study here if this property protects Pairing Based cryptography in Edwards coordinates against fault attacks.
Key words: Pairing Based Cryptography, Edwards coordinates, fault attack.

1 Introduction

Originally, pairings were used in a destructive way. Pairings convert the discrete logarithm problem from an elliptic curve subgroup to the discrete logarithm problem in a finite field. This property was used in the MOV [29] and Frey Ruck attack [19] . This pairing property permits the construction of new protocols. The first constructive use of pairings was the tripartite key exchange of A. Joux [22]. It was followed by original protocols like Identity Based Cryptography (IBC), which was introduced by D. Boneh and M. Franklin in 2001 [10], or short signature schemes [20].

The use of pairings in IBC involves a secret entry during the pairing calculation. Several pairing implementations exist, for example [32] and [5]. Side Channel Attacks (SCA) against pairing based cryptography were first developed three years ago ([30], [33] and [24]).

In [30], D. Page and F. Vercauteren introduce a fault attack against the particular case of the Duursma and Lee algorithm. The fault attack consists in modifying the number of iterations of the algorithm. This idea was complete in [18] in application to the Miller algorithm in Weierstrass coordinates. In [33] the authors conclude that if the secret is used as the first argument of the pairing computation, then it can not be found. This countermeasure is not one, as concluded in [18]. This three articles consider the case of Weierstrass coordinates. Recently, Edwards coordinates were introduced for computing pairings [6, 8, 23, 3].

Edwards curves became interesting for elliptic curve cryptography when it was proved by Bernstein and Lange in [7] that they provide addition and doubling formulas faster than all addition formulas known at that time. The advantage of Edwards coordinates is that the addition law can be complete and thus the exponentiation in Edwards coordinates is naturally protected against Side Channel Attacks.

Our contribution is to find out if Pairing Based Cryptography in Edwards coordinates is protected against fault attack. We show that a fault attack against the Miller algorithm in Edwards coordinates can be done through the resolution of a non linear system.

The outline of this article is as follow.

First we will give a short introduction to pairings in SectionMiller. After that we recall the background of Edwards coordinates in Section and to pairing in Edwards coordinates in Section 3 and to pairing based Cryptography in Section 4. Section 5 presents our fault attack against Pairing Based Cryptography in Edwards coordinates, finally, we conclude in Section 6.

2 Pairings and the Miller algorithm

First, we recall the definition and property of pairings, before introducing the property of Edwards curves and Pairing Based Cryptography over Edwards curves.

2.1 A short introduction to pairings

In this section we give a brief overview of the definitions of pairings on elliptic curves and of Miller's algorithm [28] used in pairing computation. Let q be a prime power not divisible by 2, E an elliptic curve over $\mathbb{F}_q$ and r a prime factor of $\#(E(\mathbb{F}_q))$. Suppose r^2 does not divide $\#(E(\mathbb{F}_q))$ and k be the *embedding degree* with respect to r, i.e. the smallest integer such that r divides $q^k - 1$. We denote $\mathcal{O}$ the point at infinity of the elliptic curve.

Definition 1. *A pairing is a bilinear and non degenerate function:*

$$\begin{aligned} e : G_1 \times G_2 &\to G_3 \\ (P, Q) &\to e(P, Q) \end{aligned}$$

where $\mathbb{G}_1$ and $\mathbb{G}_2$ are subgroups of order r on the elliptic curve and $\mathbb{G}_3$ is generally μ_r, the subgroup of the r-th roots of unity in $\mathbb{F}_{q^k}$. In general we take $\mathbb{G}_1 = E(\mathbb{F}_q)[r]$ and $\mathbb{G}_2 \subset E(\mathbb{F}_{q^k})[r]$, where we denote by $E(K)[r]$ the subgroup of K-rational points of order r of the elliptic curve E. We also denote $E[r]$ the subgroup of points of order r defined over the algebraic closure of $\mathbb{F}_q$.

Let $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$. The goal of Miller's algorithm is to construct a rational function $f_{s,P}$ associated to the point P and to some integer s and to evaluate

this function at the point Q (in fact at a divisor associated to this point). The function $f_{s,P}$ is such that the divisor associated to it is:

$$\mathrm{div}(f_{s,P}) = s(P) - (sP) - (s-1)(\mathcal{O}).$$

Suppose we want to compute the sum of iP and jP. Take h_1 the line going through iP and jP and h_2 the vertical line through $(i+j)P$. Miller's idea was to make use of the following relation

$$f_{i+j,P} = f_{i,P} f_{j,P} \frac{h_1}{h_2}, \tag{1}$$

in order to compute $f_{s,P}$ iteratively. Moreover, Miller's algorithm uses the double-and-add method to compute $f_{s,P}$ in $\log_2(s)$ operations [28].

The reduced Tate pairing

The reduced Tate pairing, denoted $\widehat{e}_{Tate}$, is defined by:

$$\begin{aligned} &\mathbb{G}_1 \times \mathbb{G}_2 \mapsto \mathbb{G}_3 \\ &(P,Q) \mapsto \widehat{e}_{Tate}(P,Q) = f_{r,P}(Q)^{\frac{p^k-1}{r}}. \end{aligned}$$

3 Background on Edwards curves

Definition and properties Edwards showed in [16] that every elliptic curve E defined over an algebraic number field is birationally equivalent over some extension of that field to a curve given by the equation:

$$x^2 + y^2 = c^2(1 + x^2y^2). \tag{2}$$

In this paper, we use the notion of Twisted Edwards curves denoted $E_{a,d}$ and defined over a field $\mathbb{F}_q$, where q is a power of prime different from 2 :

$$E_{a,d} := \{(x,y) \in \mathbb{F}_q^2 \text{ such that } ax^2 + y^2 = 1 + dx^2y^2\}$$

They were introduced by Bernstein et al in [8] as a generalization of Edwards curves.

On a twisted Edwards curve, we consider the following addition law:

$$(x_1,y_1),(x_2,y_2) \to \left(\frac{x_1y_2 + y_1x_2}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2} \right). \tag{3}$$

The neutral element of this addition law is $O = (0,1)$. For every point $P = (x,y)$ the opposite element is $-P = (-x,y)$.

In [7], it was shown that this addition law is *complete* when d is not a square. This means it is defined for all pairs of input points on the Edwards curve with no exceptions for doubling, neutral element etc.

In the following sections we use projective coordinates. A projective point (X,Y,Z) satisfying $(aX^2 + Y^2)Z^2 = Z^4 + dX^2Y^2$ and $Z \neq 0$ corresponds to the affine point $(X/Z, Y/Z)$ on the curve $ax^2 + y^2 = 1 + dx^2y^2$. The Edwards curve has two points at infinity $(0:1:0)$ and $(1:0:0)$. The fastest formulas for computing pairings over Edwards curves are given in [3].

3.1 Pairings over Edwards curves

For efficiency reasons, we restrict the domain of the Tate pairing to a product of cyclic subgroups of order r on the elliptic curve. In general, the point P can be chosen such that $\langle P \rangle$ is the unique subgroup of order r in $E(\mathbb{F}_q)$. In order to get a non-degenerate pairing, we take Q a point of order r, $Q \in E(\mathbb{F}_{q^k}) \backslash E(\mathbb{F}_q)$. Moreover, if the embedding degree is even, it was shown that the subgroup $\langle Q \rangle \subset E(\mathbb{F}_{q^k})$ can be taken so that the x-coordinates of all its points lie in $\mathbb{F}_{q^{k/2}}$ and the y-coordinates are products of elements of $\mathbb{F}_{q^{k/2}}$ with $\sqrt{\alpha}$, where α is a non square in $\mathbb{F}_{q^{k/2}}$ and $\sqrt{\alpha}$ is in $\mathbb{F}_{q^k}$ (see [25, 3] for details).

The same kind of considerations apply to Edwards curves and Twisted Edwards curves [3]. Using the trick of [25] the point $Q \in E(\mathbb{F}_{q^k})$ is written $(X_Q\sqrt{\alpha}; Y_Q; Z_Q)$ using a twist of degree 2. The element X_Q, Y_Q, Z_Q and α are in $\mathbb{F}_{q^{k/2}}$ and $\sqrt{\alpha} \in \mathbb{F}_{q^k}$. The point P is written (X, Y, Z) with X, Y and $Z \in \mathbb{F}_q$. In the following algorithm we used the denominator elimination trick [25].

Algorithm 1: Miller (P, Q, s)

Data: $s = (s_n \dots s_0)$(binary decomposition), $P \in \mathbb{G}_1$ $Q \in \mathbb{G}_2$;
Result: $f_{s,P}(Q) \in \mathbb{G}_3$;
$T \leftarrow P$, $f \leftarrow 1$,
for $i = n - 1$ **to** 0 **do**
  $T \leftarrow [2]T$ and $f \longleftarrow f^2 \times g_d(Q)$
  if $s_i = 1$ **then**
    $T \leftarrow T + P$ and $f \longleftarrow f \times g_a(Q)$
  end
end
return $f = f_{s,P}(Q) \in \mathbb{F}^*_{q^k}$

Fig. 1. Miller's algorithm

The equation of the function g_d and g_a are described in the following Sections.

Doubling step We now take a look into the details of the computation of a Miller iteration. The doubling step is done for each iteration of the Miller's algorithm. We note $T = (X_1, Y_1, Z_1)$. Following [3] the doubling formulas for $2T = (X_3, Y_3, Z_3)$ are:

$$\begin{aligned} X_3 &= (2X_1Y_1)(2Z_1^2 - aX_1^2 - Y_1^2), \\ Y_3 &= Y_1^4 - a^2X_1^4, \\ Z_3 &= (aA_1^2 + Y_1^2)(2Z_1^2 - aA_1^2 - Y_1^2). \end{aligned}$$

The function g_d has the following equations:

$$g_d(Q) = c_{Z^2}\eta'\sqrt{\alpha} + c_{XY}y_0 + c_{XZ}$$

where

$$\eta' = \frac{Z_Q + Y_Q}{X_Q} \text{ and } y_0 = \frac{Y_Q}{Z_Q},$$
$$c_{Z^2} = X_1(2Y_1^2 - 2Y_1Z_1),$$
$$c_{XY} = 2Z_1(Z_1^2 - aX_1^2 - Y_1Z_1),$$
$$c_{XZ} = Y_1(2aX_1^2 - 2Y_1Z_1).$$

Addition step This step is done only when the current bit of s is equal to 1. We note $T = (X_1, Y_1, Z_1)$ and $P = (X_P, Y_P, Z_P)$. Following [3] the addition formulas for $T + P = (X_3, Y_3, Z_3)$ in extended Edwards form are:

$$T_1 = \frac{X_1Y_1}{Z_1} \text{ and } T_P = \frac{X_PY_P}{Z_P},$$
$$X_3 = (T_1Z_P + T_PZ_1)(X_1Y_P - X_PY_1),$$
$$Y_3 = (T_1Z_P + T_PZ_1)(Y_1Y_P + aX_1X_P),$$
$$Z_3 = (X_1Y_P - X_PY_1)(Y_1Y_P + aX_1X_P).$$

The function g_a has the following equations:

$$g_a(Q) = c_{Z^2}\eta'\sqrt{\alpha} + c_{XY}y_0 + c_{XZ}$$

where

$$\eta' = \frac{Z_Q + Y_Q}{X_Q} \text{ and } y_0 = \frac{Y_Q}{Z_Q},$$
$$c_{Z^2} = X_1X_P(Y_1Z_P - Y_PZ_1),$$
$$c_{XY} = Z_1Z_P(X_1Z_P - Z_1X_P + X_1Y_P - Y_1X_P)$$
$$c_{XZ} = X_PY_PZ_1^2 - X_1Y_1Z_P^2 + Y_1Y_P(X_PZ_1 - X_1Z_P).$$

4 Identity based cryptography

The aim of identity based encryption is that the users public key are their identity, and a trusted authority sends them their private key. This trusted authority creates all the private keys related to an identity based protocol. The general scheme of identity based encryption is described in [10].

The most useful property in pairing based cryptography is bilinearity:

$$\forall (n, m) \in \mathbb{Z}^2,\ e([n]P, [m]Q) = e(P, Q)^{nm}.$$

Pairings permit several protocol simplifications and original scheme creation, for example Identity Based Cryptography (IBC) protocols. A nice survey of protocols using pairings is done in [15]. A recent book [17] is dedicated to IBC.

The general scheme of an identity based encryption is described in [10], we briefly recall it. We describe an exchange between Alice and Bob using the Boneh and Franklin scheme [10].

The public data are an elliptic curve E defined over $\mathbb{F}_q$, for q a power of a prime p, $\mathbb{G}_1$ a sub-group of E and $\mathbb{G}_3$ a sub-group of $\mathbb{F}_{q^k}$, where k is the embedding degree of E relatively to $r = \#\mathbb{G}_1$, a pairing $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_3$, and P_{pub} a generator of $\mathbb{G}_1$. Let $H_1 : \{0,1\}^\star \to \mathbb{G}_1,\ H_2 : \mathbb{G}_1 \to \{0,1\}^n$, be two hash functions, with n the bitlenght of the message.

The public key of Bob is $Q_B = H_1(Id_B) \in \mathbb{G}_1$, where Id_B is the identity of Bob. His private key is constructed by a trusted authority denoted T_A. T_A chooses an integer s kept secret and computes $K_{pub} = [s]P_{pub} \in \mathbb{G}_1$ the public key of the protocol, and the private key of Bob by $P_B = [s]Q_B \in \mathbb{G}_1$.

With the public data Alice can compute $Q_B = H_1(Id_B)$, and the pairing $g_B = e(Q_B, K_{pub})$.

She chooses an integer m and sends to Bob $C = \langle [m]P_{pub}, M \oplus H_2(g_B^m) \rangle$, which we denote $C = \langle U, V \rangle$.

To decipher the message C, Bob recover his private key and compute $V \oplus H_2(e(P_B, U))$.

Considering the property of bilinearity :

$$\underline{e(P_B, U)} = e([s]Q_B, [m]P_{pub}) = e(Q_B, [s]P)^m = e(Q_B, K_{pub})^m = \underline{g_B^m}.$$

Consequently, Bob can read the message by computing $V \oplus H_2(g_B^m)$.

The important point is that to decipher a message using an Identity Based Protocol, a computation of a pairing involving the private key and the message is done. Side Channel Attacks can be therefore applied to find this secret. The particularity of Identity Based Protocol is that an attacker can know the algorithm used, the number of iterations and the exponent. The secret is only one of the arguments of the pairing. The secret key influences neither the execution time nor the number of iterations of the algorithm.

5 Fault Attack against Pairing Based Cryptography

The goal of a fault injection attack is to provoke mistakes during the calculation of an algorithm, for example by modifying the internal memory, in order to reveal sensitive date. This attack needs very precise timing, position and expensive apparatus to be performed. Nevertheless, new technologies could allow this attack [21].

5.1 Description of the fault attack

The goal of a fault injection attack is to provoke mistakes during the calculation of an algorithm, for example by modifying the internal memory, in order to reveal sensitive data. This attack needs a very precise positioning and an expensive apparatus to be performed. Nevertheless, new technologies could allow for this attack [21].

We follow the scheme of attack described in [30] and completed in [18]. We assume that the pairing is used during an Identity Based Protocol, the secret point is introduced in a smart card or an electronic device and is a parameter of the pairing. In order to find the secret, we modify the number of iterations in the Miller's algorithm by the following way.

First of all, we have to find the flip-flops belonging to the counter of the number of iterations (i.e. $\log_2(s)$) in the Miller's algorithm. This step can be done by using reverse engineering procedures. Once we found it, we provoke disturbances in order to modify it and consequently the number of iterations of the Miller's algorithm. For example the disturbance can be induced by a laser [2]. Lasers are nowadays thin enough to make this attack realistic [21]. Counting the clock cycles, we are able to know how many iterations the Miller loop has done. Each time, we record the value of the Miller loop and the number of iterations we made. The aim is to obtain a couple $(\tau, \tau+1)$ of two consecutive values, corresponding to τ and $\tau+1$ iterations during the Miller's algorithm.

We denote the two results by $F_{\tau,P}(Q)$ and $F_{\tau+1,P}(Q)$. To conclude the attack, we consider the ratio $\frac{F_{\tau+1,P}(Q)}{F_{\tau,P}(Q)^2}$. By identification in the basis of $\mathbb{F}_{q^k}$, we are lead to a system which can reveal the secret point, which is described in Section 5.4.

The probability for obtaining two consecutive numbers is sufficiently large to make the attack possible [18]. In fact, for an 8-bits architecture only 15 tests are needed to obtain a probability larger than one half, $P(15, 2^8) = 0.56$, and only 28 for a probability larger than 0.9.

5.2 The τ^{th} step

We execute the Miller algorithm several times. For each execution we provoke disturbance in order to modify the value of $\log_2(s)$, until we find the result of the algorithm execution for two consecutive iterations, the τ^{th} and $(\tau+1)^{th}$ iterations of algorithm 1. We denote the two results by $F_{\tau,P}(Q)$ and $F_{\tau+1,P}(Q)$. After τ iterations, the algorithm 1 will have calculated $[j]P$. During the $(\tau+1)^{th}$ iteration, it calculates $[2j]P$ and considering the value of the $(\tau+1)^{th}$ bit of $\log_2(s)$, it either stops at this moment, or it calculates $[2j+1]P$. In order to simplify the equations, we consider $k = 4$, but the method described can be generalised for $k \geq 4$. We denote $B = \{1, \gamma, \sqrt{\alpha}, \gamma\sqrt{\alpha}\}$ the basis used for written the elements of $\mathbb{F}_{q^k}$, this basis is constructed by a tower extensions [4].

5.3 Finding j

We know $\log_2(s)$, the order of the point Q,(as P and Q have the same order). By counting the number of clock cycles during the pairing calculation, we can find

the number τ of iterations. Then reading the binary decomposition of $\log_2(s)$ gives us directly j. We consider that at the beginning $j = 1$, if $s_{n-1} = 0$ then $j \leftarrow 2j$, else $j \leftarrow 2j+1$, and we go on, until we arrive to the $(n-1-\tau)^{th}$ bit of s. For example, let $s = 1000010000101$ in basis 2, and $\tau = 5$, at the first iteration we compute $[2]P$, at the second, as $s_{n-1} = 0$ we only make the doubling, so we calculate $[4]P$, it is the same thing for the second, third and fourth step so we have $[32]P$ in T.

At the fifth iteration, $s_{n-6} = 1$, then we make the doubling and the addition, so $j = 2 \times 32 + 1$, i.e. $j = 65$.

5.4 Curve and equations

In [30, 34, 18], only the affine coordinates case is treated. In [30, 34], a simple identification of the element in the basis of $\mathbb{F}_{q^k}$ gives the result. Here, the difference between these cases and Edwards coordinates is that we solve a nonlinear system.

Using the equation of the pairing calculation proposed in Section 3.1, we find a nonlinear system of k equations using the equality $g(Q) = R$, where $g(Q)$ defines the equation of update of f during the Miller's algorithm. This system is solvable with the resultant method. To solve the system in Edwards coordinates we need k to be greater than 2.

The embedding degree. In order to simplify the equations, we consider case $k = 4$. As the important point of the method is the identification of the decomposition in the basis of $\mathbb{F}_{q^k}$, it is easily applicable when k is larger than 2.

We denote $B = \{1, \gamma, \sqrt{\alpha}, \gamma\sqrt{\alpha}\}$ the basis of $\mathbb{F}_{q^k}$, constructed by a tower extensions. The point $P \in E(\mathbb{F}_q)$ is given in Jacobian coordinates, $P = (X_P, Y_P, Z_P)$ and the point $Q \in E(\mathbb{F}_{q^k})$ also. As k is even, we can use a classical optimisation in pairing based cryptography which consists in using the twisted elliptic curve to write $Q = (X_Q\sqrt{\alpha}; Y_Q; Z_Q)$, with X_Q, Y_Q, Z_Q and $\alpha \in \mathbb{F}_{q^2}$ and $\sqrt{\alpha} \in \mathbb{F}_{q^4}$, as described in Section 3.1.

Case 1: $s_{\tau+1} = 0$. We know the results of the τ^{th} and $(\tau+1)^{th}$ iterations of the Miller's algorithm, $F_{\tau,P}(Q)$ and $F_{\tau+1,P}(Q)$. We examine what happens during the $(\tau+1)^{th}$ iteration.

The doubling step gives:

$$F_{\tau+1,P}(Q) = (F_{\tau,P}(Q))^2 \times g_d(Q)$$

As we suppose that $s_{\tau+1} = 0$, the additional step is not done. The return result of the Miller's algorithm is $F_{\tau+1,P}(Q) = (F_{\tau,P}(Q))^2 g_d(Q)$. We dispose of $F_{\tau,P}(Q)$, $F_{\tau+1,P}(Q)$ and the point $Q = (X_Q\sqrt{\alpha}; Y_Q; Z_Q)$, with X_Q, Y_Q and $Z_Q \in \mathbb{F}_{q^2}$. Recall that the coordinates of Q can be freely chosen and that we describe the

attack for $k = 4$, this can easily be generalised for $k > 4$.
We can calculate the value $R \in \mathbb{F}_{q^k}^*$ of the ratio $\frac{F_{\tau+1,P}(Q)}{(F_{\tau,P}(Q))^2}$,

$$R = R_3\gamma\sqrt{\alpha} + R_2\sqrt{\alpha} + R_1\gamma + R_0,$$

where $R_3,\ R_2,\ R_1,\ R_0 \in \mathbb{F}_q$.

Moreover, we know the theoretical form of R in the basis $B = \{1, \gamma, \sqrt{\alpha}, \gamma\sqrt{\alpha}\}$ which depends of coordinates of jP and Q:

$$R = g_d(Q) = c_{Z^2}\eta'\sqrt{\alpha} + c_{XY}y_0 + c_{XZ},$$

where the c_{Z^2}, C_{XY}, c_{XZ} are in $\mathbb{F}_q$ and $\eta',\ y_0 \in \mathbb{F}_{q^2}$.

When the secret is the first argument
This position was presented as a counter measure to SCA in [33]. We know the point Q, thus the value of η' and $y_0 \in \mathbb{F}_{q^2}$ and their decomposition in $\mathbb{F}_{q^2}$, $\eta' = \eta'_0 + \eta'_1\gamma$, $y_0 = y_{00} + y_{01}\gamma$, where $(1, \gamma)$ defines the basis of $\mathbb{F}_{q^2}$. The elements c_{Z^2}, c_{XY} and c_{XZ} are in $\mathbb{F}_q$. Using the equality :

$$c_{Z^2}(\eta'_0 + \eta'_1\gamma)\sqrt{\alpha} + c_{XY}(y_{00} + y_{01}\gamma) + c_{XZ} = R_0 + R_1\gamma + R_2\sqrt{\alpha} + R_3\gamma\sqrt{\alpha}$$

by identification in the basis of $\mathbb{F}_{q^k}$, we obtain, after simplification, the following system of equations in $\mathbb{F}_q$:

$$\begin{cases} c_{XZ} = \lambda_2 \\ c_{XY} = \lambda_1 \\ c_{Z^2} = \lambda_0 \end{cases}$$

The value λ_0, λ_1 and λ_2 are known. With the resultant method we recover the coordinates of the secret point P. An example is given in the appendix.
When the secret is the second argument

We know the point P, thus the value of c_{Z^2}, c_{XY} and $c_{XZ} \in \mathbb{F}_q$. Using the equality :

$$c_{Z^2}(\eta'_0 + \eta'_1\gamma)\sqrt{\alpha} + c_{XY}(y_{00} + y_{01}\gamma) + c_{XZ} = R_0 + R_1\gamma + R_2\sqrt{\alpha} + R_3\gamma\sqrt{\alpha}$$

By identification in the basis of $\mathbb{F}_{q^k}$, we can recover the value η' and y_0, and thus the coordinate of the point Q.

$$\begin{cases} \eta'_0 = \frac{R_2}{c_{Z^2}} \text{ and } \eta'_1 = \frac{R_3}{c_{Z^2}}, \\ y_{00} = \frac{R_0 - c_{XZ}}{c_{XY}} \text{ and } y_{01} = \frac{R_1}{c_{XY}} \end{cases}$$

Indeed, once we have $y_0 \left(= \frac{Y_Q}{Z_Q}\right)$, using the elliptic curve we can find the value of $x_0 \left(= \frac{X_Q}{Z_Q}\right)$, and the coordinates of point Q.

Case 2: $s_{\tau+1} = 1$. In this case, the $(\tau+1)^{th}$ iteration involves the addition in the Miller's algorithm.

Thus, at the $(\tau+1)^{th}$ iteration, Miller's algorithm compute $F_{\tau+1,P}(Q) = (F_{\tau,P}(Q))^2 g_d(Q) g_a(Q)$. We could repeat the scheme of the previous case, and thanks the resolution of a non linear system, we can recover the secret point, whatever its position is. TO obtain the system, we juste have to develop the product $g_d(Q)g_a(Q)$. Using the polynomial reduction for the base of $\mathbb{F}_{p^{k/2}}$ and $\mathbb{F}_{p^k}$, we find the system by identification in this basis.

6 Conclusion

We have study if the Miller algorithm in Edwards coordinates is vulnerable to a fault attack. We demonstrate that it is the case, whatever is the position of the secret. Consequently, the property of Edwards curves does not protect Pairing Based Cryptography in Edwards coordinates toward fault attack. A discussion about weakness to this fault attack of pairings based on this algorithm was done in [18]. The authors shows that the Weil pairing is directly sensitive to the fault attack described, and presents some methods to override the final exponentiation are given; and then, for a motivated attacker, the final exponentiation will no longer be a natural counter measure for the Tate and Ate pairings [12]. Thus implementation of Pairing Based Cryptography in Edwards coordinates must be protected. A possible protection could be to use an asynchrone clock to confuse the attack and physical shield to protect the counter.

References

1. Abraham D.G., Dolan G.M., Double G.P. and Stevens J.V.: *Transaction Security System* IBM Systems Journal, vol 30, p 206 – 229, 1991.
2. Anderson R. and Kuhn M.: *Tamper Resistance – a Cautionary Note* The Second USENIX Workshop on Electronic Commerce Proceedings, p 1–11, Okland, California 1996.
3. C. Arène, T. Lange, M. Naehrig and C. Ritzenhaler: *Faster Pairing Computation of the Tate pairing*, Cryptology ePrint Archive, Report 2009/155, `http://eprint.iacr.org/2009/155`,2009.
4. J.C.Bajard and N.El Mrabet: *Pairing in cryptography: an arithmetic point de view*, Advanced Signal Processing Algorithms, Architectures, and Implementations XVI, part of SPIE, August 2007.
5. G. M. Bertoni, L. Chen, P. Fragneto, K. A. Harrison and G. Pelosi.: *Computing Tate pairing on smartcards*, Proceedings of Ches'05, Workshop on Cryptographic Hardware and Embedded Systems 2005 (CHES 2005) Edinburgh, Scotland.
6. D. J. Bernstein and T. Lange: *Performance evaluation of a new side channel resistant coordinate system for elliptic curves*, `cr.yp.to/antiforgery/newelliptic-20070410.pdf`, 2007.
7. D. J. Bernstein and T. Lange: *Faster additions and doubling on elliptic curves*, ASIACRYPT 2007, vol 4833, p 29–50, Springer-Verlag, 2007.
8. D. J. Bernstein, P. Birkner, M. Joye, T. Lange and C. Peters: *Twisted Edwards curves*, Afrycacrypt 2008, p 389–405,Springer-Verlag.

9. S. Ionica and A. Joux: *Another Approach to Pairing Computation in Edwards Coordinates*, INDOCRYPT '08: Proceedings of the 9th International Conference on Cryptology in India, Springer-Verlag
10. Boneh D., Franklin M.: *Identity-based encryption from the Weil pairing.* Extended abstract in Crypto 2001, LNCS 2139, pp. 213-229, 2001
11. Brier E., Joye M.: *Point multiplication on elliptic curves through isogenies*, AAECC 2003, LNCS., vol. 2643, 2003, 43?50.
12. Boneh D., DeMillo R. and Lipton R.: *On the importance of checking cryptographic protocols faults*, Advances in Cryptology Eurocrypt 1997, Lecture Notes in Comput. Sci., vol. 1233, Springer-Verlag, Berlin, 1997, 37–51.
13. Cohen, H., Frey, G. (editors): *Handbook of elliptic and hyperelliptic curve cryptography.* Discrete Math. Appl., Chapman & Hall/CRC (2006)
14. Yang B., Wu K. and Karri R.: *Scan Based Side Channel Attack on Dedicated Hardware Implementation of Data Encryption Standard* Test Conference 2004, proceedings ITC 2004, p 339 – 344.
15. Ratna Dutta and Rana Barua and Palash Sarkar: **Pairing-Based Cryptographic Protocols: A Survey**, In Cryptology ePrint Archive, Report 2004/064, 2004.
16. Edwards H. *A normal Form for Elliptic Curve* Bulletin of the American Mathematical Society Vol. 44, Number 3, July 2007.
17. M. Joye and G. Neven: **Identity-Based Cryptography**, vol. 2 of Cryptology and Information Security Series, IOS Press.
18. N. El Mrabet, *What about Vulnerability to a Fault Attack of the Miller's Algorithm During an Identity Based Protocol?*, Advances in Information Security and Assurance 2009, LNCS 5576, p 122–134, Springer-Verlag.
19. Frey G., Müller M., Rück H.G.: The Tate Pairing and the Discrete Logarithm Applied to Elliptic Curve Cryptosystems, IEEE Transactions Inf. Theory, 45, 1717-1719;1999
20. Galbraith S.: K.G.: Pairings, Chapter IX, *Advances in Elliptic Curve Cryptography*, F. Blake and G. Seroussi and N. Smart editors, Series: London Mathematical Society Lecture Note Series (No. 317),Cambridge University Press,2005
21. Habing D.H: *The Use of Lasers to Simulate Radiation-Induced Transients in Semiconductor Devices and Circuits* IEEE Transactions On Nuclear Science, vol.39, pp. 1647-1653, 1992.
22. Joux A.: *one round protocol for tripartite Diffie-Hellman*, Algorithmic Number Theory: Fourth International Symposium, Lecture Notes in Computer Science, 1838 (2000), 385-393. Full version: Journal of Cryptology, 17 (2004), 263-276
23. Ionica S. and Joux A.: *Faster Pairing Computation on Edwards Curves* pre-print presented at the C2 conference : http://c2-2008.inria.fr/C2/
24. Tae Hyun Kim, Tsuyoshi Takagi, Dong-Guk Han, Ho Won Kim and Jongin Lim: *Side Channel Attacks and Countermesures on Pairing based Cryptosystems over Binary Fields*, The 5th International Conference on Cryptology and Network Security (CANS 2006), LNCS 4301, Springer-Verlag 2006, 168-181.
25. Koblitz N., Menezes A. J.: *Pairing-based cryptography at high security levels*, Proceedings of the Tenth IMA International Conference on Cryptography and Coding, Springer-Verlag, LNCS 3796, 2005, 13-36;
26. Macwilliams F.J. and Sloane N.J.A.: *The Theory of Error-Correcting Codes II* North-Holland Mathematical Library, vol. 16, North-Holland, Amsterdam, 1998.
27. Menezes A.: *An introduction to pairing-based cryptography* Notes from lectures given in Santander, Spain, 2005
http://www.cacr.math.uwaterloo.ca/ ajmeneze/publications/pairings.pdf

28. Miller V.: *The Weil pairing and its efficient calculation*, J. Cryptology, 17 (2004), 235-261.
29. MenezesA., Okamoto T. and Vanstone S.A.: *Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field*, IEEE Trans. Inf . Theory 39, numéro 5, pages 1639-1646, 1993.
30. Page Dan and Vercauteren Frederik: *Fault and Side Channel Attacks on Pairing based Cryptography*, IEEE Transactions on Computers, vol. 55, no. 9, pp. 1075-1080, Sept., 2006.
31. PARI/GP, version `2.1.7`, Bordeaux, 2005, http://pari.math.u-bordeaux.fr/
32. Scott M.: *Computing the Tate Pairing*, Lecture Notes in Computer Science 3376, Springer-Verlag, 2005. ed.Cryptography track - RSA-2005, - , San Francisco, USA, 293 - 304.
33. Whelan C. and Scott M.: *Side Channel Analysis of Practical Pairing Implementation: Which Path is More Secure ?*, Lecture Notes in Computer Science, Volume 4341 Springer-Verlag ed.VietCrypt 2006, 25-SEP-06 - 28-SEP-06, Hanoi, Vietnam, 99 - 114.
34. Whelan C. and Scott M.: *The Importance of the Final exponentiation in Pairings when considering Fault Attacks* Lecture Notes in Computer Science, Volume 4575/2007 Springer-Verlag ed. Pairing07, Tokyo.

A The probability for the fault attack.

The important point of this fault attack is that we can obtain two consecutive couples of iterations, after a realistic number of tests. The number of picks with two consecutive number is the complementary of the number of picks with no consecutive numbers. The number $B(n, N)$ of possible picks of n numbers among N integers with no consecutive number is given by the following recurrence formula:

$$\begin{cases} N \leq 0, n > 0, B(n, N) = 0, \\ \quad \forall N, n = 0 B(n, N) = 1 \\ \qquad B(n, N) = \sum_{j=1}^{N} \sum_{k=1}^{n} B(n-k, j-2). \end{cases}$$

With this formula, we can compute the probability to obtain two consecutive numbers after n picks among N integers. This probability $P(n, N)$ is

$$P(n, N) = 1 - \frac{B(n, N)}{C_{n+N}^{n}}$$

B Example of resolution of a system

We consider the Edwards elliptic curves given in [6]: $E_{1,-1}$ over $\mathbb{F}_q$
with $q = 2^5 20 + 2^3 63 - 2^3 60 - 1$.

We consider that after a differential attack, we obtain the following values for c_{Z^2} and c_{XY}:

$$\begin{cases} c_{Z^2} = 340483761541219253591134293755215103931312112021481471447934253402934279329298538846116722969540525733078205154818523398590977903233845501192089410893868180 7, \\ c_{XY} = 175208068457016790878745084332426428593619960800642137258585409170745219031354476823850136133478591743769409441759263897379859991288038826511945916794250369 8 \end{cases}$$

To solve the system

$$\begin{cases} X(2Y^2 - 2YZ) = c_{Z^2}, \\ 2Z(Z^2 - aX^2 - YZ) = c_{XZ}, \\ (aX^2 + Y^2)Z^2 = Z^4 + dX^2Y^2, \end{cases}$$

we use the following Pari-GP [31] code:

```
q = 2^520 + 2^363 - 2^360 - 1;

a=Mod(1,q);

d=Mod(-1,q);

cZZ = 340483761541219253591134293755215103931312112021481471447934253\
      402934279329298538846116722969540525733078205154818523398590\
      779032338455011920894108938681807;

cXY = 175208068457016790878745084332426428593619960800642137258585409\
      170745219031354476823850136133478591743769409441759263897379859\
      599912880388265119459167942503698;
```

We construct the polynomial corresponding to each line of the system:

```
x = X*(2*Y^2-2*Y*Z) - cZZ;

y = - cXY + 2*Z*(Z^2-a*X^2-Y*Z);

z = (a*X^2+Y^2)*Z^2 - Z^4 - d*X^2*Y^2;
```

We apply the resultant method to obtain one equation in one unknown value:

```
 Z1 = polresultant(x,y,X);

 Z2 = polresultant(x,z,X);
```

```
Z3 = polresultant(Z1,Z2,Y);
```

$Z3$ is the final equation in Z, it is an equation of degree 16. We can find the solution of this equation:

```
polrootsmod(Z3,p)
```

We find 4 solutions in Z.

We are looking for points on the elliptic curve, thus Z must be different from 0. So we have 3 possible values.

```
[Mod(0, q),
Mod(901525105405827680078932099881135208347014760557116161414786496\
689808758208101433139075821047553434266076497551527897572311775271 6\
434381663459747649106 1913,q),
Mod(943029634660650213489325189263739902235878374981732742513043934\
826893737618341124368185574033314859408984546420993323406357349949 8\
869096417100823041016331 32,q),
Mod(239921668486407187599373200028884318556413779905064814412212597\
769212334243720211757081644203501531374800699632031977903077472562 7\
052961436635715286188378362, q)]~
```

To each of the 3 non zero value, using equation $Z2$ we find one value for Y:

```
Z = Mod(901525105405827680078932099881135208347014760557116161414786496\
689808758208101433139075821047553434266076497551527897572311775271 6\
434381663459747649106 1913,q)

Y = [Mod(261664723692376771412519800619210191801678649210732534557505099892\
174925093897971705778547452808016256451379938199954909912375542916450574\
500329476293105973,q)]~


Z = Mod(943029634660650213489325189263739902235878374981732742513043934\
826893737618341124368185574033314859408984546420993323406357349949 8\
869096417100823041016331 32,q)
Y = [Mod(56777,q)]~


Z = Mod(239921668486407187599373200028884318556413779905064814412212597\
769212334243720211757081644203501531374800699632031977903077472562 7\
052961436635715286188378362, q)]
```

```
Y = Mod(81575159314153714336575239334859469061793115798076715720159756229 58\
230307824554135471310507202775003271478125582659372844508775611877643204\
80065590487910657,q)
```

Using these 3 couples of values, we find 6 triplets and an exhaustive research gives us the correct secret point.