Improved Fault Attack Against Eta Pairing

Yunqi Dou, Jiang Weng, and Chuangui Ma
(Corresponding author: Yunqi Dou)

Zhengzhou Information Science and Technology Institute
Zhengzhou, Henan Province, 450002, China
(Email: douyunqi@126.com)
(Received Oct. 16, 2012; revised and accepted Feb. 19, 2013)

Abstract

In recent years, an increasing number of cryptographic protocols based on bilinear pairings have been developed. With the enhancement of implementation efficiency, the algorithms of pairings are usually embedded in identity aware devices such as smartcards. Although many fault attacks and countermeasures for public key and elliptic curve cryptographic systems are known, the security of pairing based cryptography against the fault attacks has not been studied extensively. In this paper, we present an improved fault attack against the Eta pairing and generalize the attack to general loop iteration. We show that whatever the position of the secret point is, it can be recovered through solving the non-linear system obtained after the fault attack.

Keywords: Eta pairing, fault attack, Miller's algorithm, pairing based cryptography

1 Introduction

In 1984, Shamir proposed a challenge for the cryptographer community to design a protocol based on user's identity [18]. This challenge was solved almost twenty years later by Boneh and Franklin in 2001, who proposed the first practical identity based encryption (IBE) scheme based on pairings [3]. Since then, bilinear pairings have become an important tool in cryptography, and pairing based cryptography (PBC) has been developed to be a vital research field. Pairings also have been used as building blocks by numerous schemes, such as attribute based encryption [8], short signatures [4], and anonymous group signatures [5]. Through the past years research, pairings can be implemented efficiently on identity aware and resource constrained devices such as smartcards [17].

Since Kocher gave a number of remarkably simple timing attacks in his seminal work [11] in 1996, an increasingly popular form of attack known as side-channel analysis has been developed. Fault attacks which exploit the leakage of information through the faulty outputs of the cryptographic device have evolved at the same time. The fault attacks against the traditional cryptographic protocols have been extensively studied. However, in the context of pairing based cryptography there are only a few works about the fault attacks [7, 14, 16, 20]. The fault attacks against pairing based cryptography differ fundamentally from the fault attacks known in the elliptic curve cryptography. In the elliptic curve cryptography, the secret is usually the scalar which affects the sequence of operations. Thus, the secret may be easily computed through timing or power analysis. In contrast, the secret in the pairing based cryptography is a point on the elliptic curve used as one of the arguments of the pairing. The secret influences neither the execution time nor the sequence of the pairing algorithm. As mentioned in [16], this may be the main reason why the fault attacks had not been considered against pairing based cryptography for a long time.

Page and Vercauteren [16] presented the first fault attacks against pairing algorithms. They introduced two similar fault attacks against pairing algorithms based on Duursma and Lee's algorithm [6]. The fault attacks consisted in modifying the algorithm iterations number. By inducing extra iterations they were able to isolate a single contribution to the Miller loop. Later this idea was further applied to the Miller's algorithm by Mrabet [14]. The vulnerability of several algorithms for the Weil, Tate and Eta pairings in presence of fault attacks was studied in [20]. Whelan and Scott described the fault attacks against the Weil and Eta pairings by injecting faults into intermediate values in the last loop iteration of the algorithms.

Mrabet [14] promised that for all the coordinate systems (i.e. affine, projective, Jacobian and Edwards coordinates) a fault attack against Miller's algorithm could be done through the resolution of a nonlinear system. She made an assumption that the adversary was able to read the intermediate states of the device before the final exponentiation through some microelectronic methods [1, 21], but it may be unrealistic nowadays. A fault attack against the Tate pairing in Edwards coordinates was presented in [7]. The authors assumed that the adversary was able to inject fault at loop variable, so the Miller's algorithm would execute for only one iteration. Recently, Mrabet *et al.* [15] recalled different types of fault attacks against the pairing algorithms and gave a good overview of countermeasures to foil the attacks.

In this paper, we are especially interested in the fault attack against an algorithm for the Eta pairing. Our contribution is to improve the fault attack against the Eta pairing, not only for the last iteration, but for possible iterations. Whelan and Scott consider if a fault is injected into the coordinates, the non-linear equation obtained may be difficult to solve. According to this, we make an assumption to inject a specified fault into the coordinates and describe precisely the way to realize this fault attack independently of the position of secret point.

The outline of this article is as follows. First we give the definition of the Eta pairing and recall the algorithm of Whelan and Scott to compute the Eta pairing in Section 2. In Section 3 we present our fault attack against the Eta pairing to improve the result of [20] and finally, we conclude in Section 4.

2 The Eta Pairing

Traditionally two types of pairings have been considered in the literature, the Weil pairing and the Tate pairing. In general, the Tate pairing is always regarded as more efficient than the Weil pairing for ordinary elliptic curve at common levels of security [9, 13]. However, other related pairings are available which are more efficient in certain situations, for example the Eta pairing on certain supersingular elliptic curve. In this section, we firstly give the formal definition of the Eta pairing restricted to the case of elliptic curves of characteristic two. Then we introduce the algorithm of [20] to compute the Eta pairing.

Supersingular elliptic curves over finite fields F_{2^m} for some odd m are given by the equation

$$E : y^2 + y = x^3 + x + b,$$

with $b = 0, 1$. The embedding degree of these curves is equal to 4 and the order of $E(F_{2^m})$ is equal to $2^m + 1 \pm 2^{\frac{m+1}{2}}$. So we need the extension field $F_{2^{4m}}$ of F_{2^m}. There exists $s \in F_{2^2}$ with $s^2 = s + 1$ which is a zero of the irreducible polynomial $x^2 + x + 1$ over F_{2^m}. Thus $F_{2^{2m}} \cong F_{2^m}(s) \cong F_{2^m}[x]/(x^2+x+1)$. Further there exists $t \in F_{2^4}$ with $t^2 = t + s$ which is a zero of the irreducible polynomial $x^2 + x + s$ over $F_{2^{2m}}$. Thus $F_{2^{4m}} \cong F_{2^{2m}}(t) \cong F_{2^{2m}}[x]/(x^2 + x + s)$. Hence the elements of $F_{2^{4m}}$ can be also represented in the form

$$c_0 + c_1 s + c_2 t + c_3 st,$$

with $c_i \in F_{2^m}$.

Further for the supersingular elliptic curves over a finite field with characteristic 2, there exists a distortion map:

$$\psi : \begin{cases} E(F_{2^m}) \to E(F_{2^{4m}}) \\ (x, y) \mapsto (x + s + 1, y + sx + t). \end{cases}$$

Definition 1. *Let $n|\#E(F_{2^m})$, $P, Q \in E(F_{2^m})[n]$ and $f_{2^m,P}$ be some function with divisor: $div(f_{2^m,P}) = 2^m(P)-(2^m P)-(2^m-1)(O)$. The Eta pairing η is defined to be*

$$\eta : \begin{cases} E(F_{2^m})[n] \times E(F_{2^m})[n] \to F_{2^{4m}} \\ (P, Q) \mapsto f_{2^m,P}(\psi(Q)). \end{cases}$$

In general, this definition will not give a non-degenerate bilinear map, but for some special cases it is. In the case of characteristic 2 for $N = 2^{2m} + 1$ and $M = 2^{2m} - 1$, Barreto *et al.* [2] deduced that

$$\eta(P, Q)^{M2^{m+1}} = t_N(P, \psi(Q))^M,$$

where t_N is the Tate pairing. Hence, this pairing is a non-degenerate bilinear pairing for the given parameters. Next, we consider the algorithm given in [20] for the Eta pairing, and present the complete description out.

The field $F_{2^{4m}}$ is constructed as extension of F_{2^m} by the irreducible polynomial $x^4 + x + 1$. Let α be a zero of this polynomial. Thus, an element $a \in F_{2^{4m}}$ can be represented as $a = a_0 + a_1\alpha + a_2\alpha^2 + a_3\alpha^3$ with $a_i \in F_{2^m}$. We assume that all coefficients are stored in four different memory cells, and denote the element by $[a_0][a_1][a_2][a_3]$. Furthermore, we give the multiplication in $F_{2^{4m}}$ in above representation form by using the relation $\alpha^4 = \alpha + 1$, $\alpha^5 = \alpha^2 + \alpha$ and $\alpha^6 = \alpha^3 + \alpha^2$. Let $a, b \in F_{2^{4m}}$, then we have the following formulas:

$$\begin{aligned} a \cdot b &= (a_0 + a_1\alpha + a_2\alpha^2 + a_3\alpha^3)(b_0 + b_1\alpha + b_2\alpha^2 + b_3\alpha^3) \\ &= a_0b_0 + \alpha(a_0b_1 + a_1b_0) + \alpha^2(a_0b_2 + a_1b_1 + a_2b_0) \\ &\quad + \alpha^3(a_0b_3 + a_1b_2 + a_2b_1 + a_3b_0) \\ &\quad + \alpha^4(a_1b_3 + a_2b_2 + a_3b_1) + \alpha^5(a_2b_3 + a_3b_2) + a_3b_3\alpha^6 \\ &= [a_0b_0 + a_1b_3 + a_2b_2 + a_3b_1] \\ &\quad [a_0b_1 + a_1b_0 + a_1b_3 + a_2b_2 + a_3b_1 + a_2b_3 + a_3b_2] \\ &\quad [a_0b_2 + a_1b_1 + a_2b_0 + a_2b_3 + a_3b_2 + a_3b_3] \\ &\quad [a_0b_3 + a_1b_2 + a_2b_1 + a_3b_0 + a_3b_3] \end{aligned}$$

Algorithm 1 gives the algorithm of Whelan and Scott to compute the Eta pairing.

Algorithm 1. Algorithm to compute the Eta pairing

Input: $P = (x_P, y_P)$, $Q = (x_Q, y_Q) \in E(F_{2^m})[n]$;
Output: $\eta(P, Q)$;
1: $l \leftarrow [1][0][0][0]$, $v \leftarrow [1][0][0][0]$
2: $T \leftarrow P$
2: for $j = m - 1$ to 0 do
3: $\quad \lambda = x_T^2 + 1$
4: $\quad l_j \leftarrow [y_Q + y_T + \lambda(x_Q + x_T + 1)][\lambda + x_Q + 1][\lambda + x_Q][0]$
5: $\quad l \leftarrow l^2 \cdot l_j$
6: $\quad T \leftarrow 2T$
7: $\quad v_j \leftarrow [x_Q + x_T + 1][1][1][0]$
8: $\quad v \leftarrow v^2 \cdot v_j$
9: end for
10:return l/v

Next, we give a theorem which will be used in Section 3.

Theorem 1. *Let E be an elliptic curve defined over F_2 in the form:*

$$E : y^2 + y = x^3 + x + b,$$

then we have for any $P \in E\backslash\{O\}$:

$$-P = (x_P, y_P + 1)$$

and

$$2^i P = (x_P^{2^{2i}} + i, y_P^{2^{2i}} + i \cdot x_P^{2^{2i}} + \tau(i))$$

with

$$\tau(i) = \begin{cases} 0 & i \equiv 0, 1 \ mod\ 4 \\ 1 & i \equiv 2, 3 \ mod\ 4 \end{cases}$$

Proof : For the proof see Section 3.5.5 in [12].

3 Fault Attack against the Eta Pairing

In this section, we will consider fault attack against the Eta pairing. By corrupting the data the algorithm works on or by interfering with the algorithm execution, the adversary produces corrupted outputs and uses these to recover the secret. We assume that the pairing is implemented on an electronic device. We restrict this study to the case where the secret is used as the first argument of the pairing. If the secret is used as the second argument, a similar attack can be applied. This attack needs a very precise positioning and expensive apparatus to be performed. However, a random value error and specific values (i.e. all 0s or 1s) are realistic to induce [10, 19]. For simplicity, we do not deliberately distinguish l_j and v_j in algorithm 1, and denote them as l and v uniformly.

3.1 Description of the Fault Attack

At first we briefly present the idea of the fault attack against the Eta pairing. The algorithm 1 computes the Eta pairing iteratively as

$$\eta(P, Q) = \prod_{j=0}^{m-1} f_j(P, Q)^{2^{m-1-j}},$$

where f_j is appropriate function.

The idea behind the fault attack is to involve a fault in one of the function values $f_j(P, Q)$. Let $\eta(P, Q)'$ denote a function value in which a fault has been injected. Dividing the faulty pairing value $\eta(P, Q)'$ by a valid pairing value $\eta(P, Q)$, we get the following relationship

$$\frac{\eta(P, Q)'}{\eta(P, Q)} = \left(\frac{f_j(P, Q)'}{f_j(P, Q)}\right)^{2^{m-1-j}}.$$

There are a number of possible locations into which the fault can be injected. A fault can be injected into any cell of l or v, or any of the coordinates x_T, y_T, x_Q, y_Q, or the slope λ. In [20], the authors think if a fault is injected into any cell of l or v in the last iteration, the effect will be local, and they present the detail result in this case. According to the case in [20] that a fault is injected into any cell of l or v in the last iteration, we generalize it to arbitrary loop iteration. For simplicity, we assume that we can inject a fault randomly into l_0 at i loop iteration of algorithm 1. Moreover, we can know the value i through timing or simple power analysis. As $f_i(P, Q), f_i(P, Q)' \in F_{2^{4m}}$, we can get

$$\left(\frac{\eta(P, Q)'}{\eta(P, Q)}\right)^{2^{1+i+3m}} = \frac{f_i(P, Q)'}{f_i(P, Q)} = \frac{[l_0]'[l_1][l_2][l_3]}{[l_0][l_1][l_2][l_3]}.$$

So we can obtain the similar equation as for the fault attack in the last iteration [20], and recover the secret point.

However, if the faults are injected into one of the coordinates or λ, all locations and all subsequent operations in which that coordinate is used may be affected. If the fault affects λ or x_Q, and if the fault is injected in a loop prior to the final loop, the non-linear equation obtained will be more difficult to solve.

In the following section, we will improve the fault attack against the Eta pairing in two directions. On the one hand, in order to reduce the complexity of the equation, we make a stronger assumption. We assume that a specified fault is injected into λ, for example altering λ to 0 in the last iteration. And we show that the secret can be recovered whether P or Q is private. On the other hand, making use of the idea of Page and Vercauteren in [16], we generalize the fault attack to the general loop iteration.

3.2 Fault in the Value λ in the Last Iteration

We consider at first a fault in λ in the last iteration. If the fault affects λ, then the cells l_0, l_1, l_2 will be corrupted. Therefore, the division of faulty and valid pairing will not cancel the function l, leaving a relationship of form

$$\frac{\eta(P, Q)'}{\eta(P, Q)} = \frac{(l'/v)}{(l/v)} = \frac{[l_0]'[l_1]'[l_2]'[l_3]}{[l_0][l_1][l_2][l_3]} = [N_0][N_1][N_2][N_3].$$

Given $\eta(P, Q)'$ and $\eta(P, Q)$, the adversary can compute N_0, N_1, N_2 and N_3. According to the algorithm 1, we get the following equations:

$$\lambda = x_T^2 + 1, \quad (1)$$

$$l_1 = \lambda + x_Q + 1, \quad (2)$$

$$l_0 = y_Q + y_T + \lambda(x_Q + x_T + 1), \quad (3)$$

where T has the form $2^i P$ for some $i \in \{0, ..., m-1\}$.

In addition, due to assuming altering λ to 0, we can obtain the following equation

$$
\begin{aligned}
&[N_0][N_1][N_2][N_3] \\
&= \frac{[y_Q + y_T][x_Q + 1][x_Q][0]}{[y_Q + y_T + \lambda(x_Q + x_T + 1)][\lambda + x_Q + 1][\lambda + x_Q][0]} \\
&= \frac{[l_0 + \lambda(x_Q + x_T + 1)][l_1 + \lambda][l_1 + \lambda + 1][0]}{[l_0][l_1][l_1 + 1][0]}
\end{aligned}
$$

Using the knowledge of the multiplication in $F_{2^{4m}}$ in Section 2, we can get the equation system below

$$
\begin{cases}
(N_0 + 1)l_0 + (N_2 + N_3)l_1 + \lambda(x_Q + x_T + 1) = N_2, \\
N_1 l_0 + (N_0 + N_2 + 1)l_1 + \lambda = N_2 + N_3, \\
N_2 l_0 + (N_0 + N_1 + N_3 + 1)l_1 + \lambda = N_0 + N_3 + 1, \\
N_3 l_0 + (N_1 + N_2)l_1 = N_1.
\end{cases} \quad (4)
$$

Solving the given system of equations above, we can compute l_0, l_1, λ and $\lambda(x_Q + x_T + 1)$.

In the last loop iteration we have

$$T = 2^{m-1}P = (x_P^{2^{2m-2}}, y_P^{2^{2m-2}} + \tau(m-1))$$

according to the Theorem 1 in Section 2.

In order to compute the x-coordinate of P, we use the following formula

$$x_T = x_P^{2^{2m-2}} = (x_P^{2^{m-2}})^{2^m} = x_P^{2^{m-2}} = (x_P^{2^m})^{2^{-2}} = x_P^{1/4}.$$

That is

$$x_P = x_T^4 = (\lambda - 1)^2.$$

In order to compute the y-coordinate of P, we can use the elliptic curve equation

$$E : y^2 + y = x^3 + x + b.$$

So we can get y_P through solving the quadratic equation over finite field F_{2^m}. Alternatively, we can also use the following equations

$$
\begin{cases}
l_0 = y_Q + y_T + \lambda(x_Q + x_T + 1), \\
x_T = x_P^{2^{m-2}}, \\
y_T = y_P^{2^{m-2}} + \tau(m-1).
\end{cases}
$$

So we can get

$$y_P^{2^{m-2}} = l_0 + y_Q + \lambda(x_Q + x_T + 1) + \tau(m-1).$$

That is

$$y_P = (l_0 + y_Q + \lambda(x_Q + x_T + 1) + \tau(m-1))^4.$$

Note: If Q is secret point and knowing P, we can also recover it using the same method, altering λ to 0 in the last loop iteration. Using the Equation (2), we can get $x_Q : x_Q = \lambda + l_1 + 1$. In order to compute y_Q, we have gotten the value $\lambda(x_Q + x_T + 1)$ from Equation (4) and use the following equations

$$
\begin{cases}
l_0 = y_Q + y_T + \lambda(x_Q + x_T + 1), \\
y_T = y_P^{2^{m-2}} + \tau(m-1).
\end{cases}
$$

So we can get

$$y_Q = l_0 + y_P^{2^{m-2}} + \tau(m-1) + \lambda(x_Q + x_T + 1).$$

Besides, one example of our attack is given in appedix.

3.3 Fault in the General Loop Iteration

In this section we generalize the fault attack to the general loop iteration using the idea of Page and Vercauteren in [16]. Assuming we can inject the fault randomly at $\Delta = m-1-j$ $(0 \le j \le m-1)$ loop iteration of algorithm 1. Using the ability, we calculate many erroneous pairing values with the aim of collecting a pair (altering λ to 0 at Δ and $\Delta + 1$ loop iteration respectively)

$$
\begin{aligned}
\eta(P,Q)' &= \prod_{i=0}^{\Delta-1} f_i(P,Q)^{2^{m-i-1}} \cdot \prod_{i=\Delta}^{m-1} (f_i(P,Q)')^{2^{m-i-1}}, \\
\eta(P,Q)'' &= \prod_{i=0}^{\Delta} f_i(P,Q)^{2^{m-i-1}} \cdot \prod_{i=\Delta+1}^{m-1} (f_i(P,Q)')^{2^{m-i-1}}.
\end{aligned}
$$

The attack will naturally require many faulty executions until appropriate values are found. The number of necessary faults will depend on the concrete architecture of the device and the accuracy of the fault. We can know the value of Δ through timing or simple power analysis.

Dividing the faulty pairing value $\eta(P,Q)'$ by $\eta(P,Q)''$, we get the following relationship

$$\frac{\eta(P,Q)'}{\eta(P,Q)''} = \left(\frac{f_\Delta(P,Q)'}{f_\Delta(P,Q)}\right)^{2^{m-1-\Delta}}.$$

Thus we can get

$$
\begin{aligned}
\left(\frac{\eta(P,Q)'}{\eta(P,Q)''}\right)^{2^{\Delta+1+3m}} &= \frac{f_\Delta(P,Q)'}{f_\Delta(P,Q)} \\
&= \frac{[l_0]'[l_1]'[l_2]'[l_3]}{[l_0][l_1][l_2][l_3]} \\
&= [M_0][M_1][M_2][M_3]
\end{aligned} \quad (5)
$$

Given $\eta(P,Q)'$ and $\eta(P,Q)''$, the adversary can compute

$$\left(\frac{\eta(P,Q)'}{\eta(P,Q)''}\right)^{2^{\Delta+1+3m}},$$

and further get M_0, M_1, M_2 and M_3. Expanding the Equation (5), we can obtain a similar system of equations like Equation (4). So we can also compute l_0, l_1, x_T and $\lambda(x_Q + x_T + 1)$. According to the algorithm 1, we also have Equations (2) and (3), but now the point T is $T = 2^\Delta P = (x_P^{2^{2\Delta}} + \Delta, y_P^{2^{2\Delta}} + \Delta \cdot y_P^{2^{2\Delta}} + \tau(\Delta))$.

In order to compute the x_P we use the Equation (2)

$$x_P = (l_1 + \Delta^2 + x_Q)^{2^{m-2\Delta-1}} = (l_1 + \Delta + x_Q)^{2^{m-2\Delta-1}}.$$

Similarly we can derive y_P from the Equation (3)

$$y_P = (l_0 + y_Q + \lambda(x_Q + x_T + 1) + \Delta \cdot x_P^{2^{2\Delta}} + \tau(\Delta))^{2^{m-2\Delta}}.$$

Thus we can completely recover the secret point P. Similarly, if Q is secret point, we can also recover it using the same method.

Comparing to [20], we generalize the fault attack to general loop iteration of Miller loop, which enhances the ability to attack. Moreover, when a fault is injected into λ, the system of equations obtained using our attack method is also easy to solve. Our fault attack can also be used to corrupt the coordinate x_Q, so whether the faults are injected into the cells of l and v, the coordinates or the slope λ, and whether the secret point is P or Q, we can extract the secret.

4 Conclusions

We have presented an improved fault attack against the Eta pairing for any arbitrary loop iteration in this paper. This attack also gives a good solution to the problem that Whelan and Scott consider consequence of corrupting coordinates will not be local and lead to a difficult modular non-linear equation. We assume a specified fault is injected into the coordinates and describe precisely the way to realize this fault attack. Moreover, our idea has important significance for fault attacks against other pairings. As we all know, there are several countermeasures [16, 20] proposed to prevent the fault attacks, for example complex final exponentiation, point blinding and fault detection mechanism. However, it is still an open problem to propose new countermeasures to ensure the efficient and secure implementation of the pairing based cryptography at the same time.

Acknowledgments

This work is supported by the National High Technology Research and Development Program of China (No. 2011AA010803) and National Natural Science Foundation of China (No. 61072047).

References

[1] R. Anderson and M. Kuhn, “Tamper resistance-a cautionary note,” in *Proceedings of the Second USENIX Workshop on Electronic Commerce*, pp. 1–11, Okland, California, 1996.

[2] P. Barreto, S. Galbraith, C. ÓhÉigeartaigh, and M. Scott, “Efficient pairing computation on supersingular abelian varieties,” *Designs, Codes and Cryptography*, vol. 42, no. 3, pp. 239–271, 2007.

[3] D. Boneh and M. Franklin, “Identity-based encryption from the weil pairing,” in *Advances in Cryptology-CRYPTO 2001*, LNCS 2139, Springer-Verlag, pp. 213–229, 2001.

[4] D. Boneh, B. Lynn, and H. Shacham, “Short signatures from the weil pairing,” *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004.

[5] D. Boneh and H. Shacham, “Group signatures with verifier-local revocation,” in *Proceedings of the 11th ACM Conference on Computer and Communications Security*, pp. 168–177, 2004.

[6] I. Duursma and H. Lee, “Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$,” in *Advances in Cryptology-ASIACRYPT 2003*, LNCS 2894, Springer-Verlag, pp. 111–123, 2003.

[7] S. Ghosh, D. Mukhopadhyay, and D. RoyChowdhury, “Fault attack and countermeasures on pairing based cryptography,” *International Journal Network Security*, vol. 12, no. 1, pp. 21–28, 2011.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[9] R. Granger, D. Page, and N. Smart, “High security pairing-based cryptography revisited,” in *Proceedings of Algorithmic Number Theory Symposium - ANTS VII*, LNCS 4096, Springer-Verlag, pp. 480–494, 2006.

[10] C. Kim and J. Quisquater, “Faults, injection methods, and fault attacks,” *IEEE Design & Test of Computers*, vol. 24, no. 6, pp. 544–545, 2007.

[11] P. C. Kocher, “Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems,” in *Advances in Cryptology-CRYPTO96*, LNCS 1109, Springer-Verlag, pp. 104–113, 1996.

[12] G. Liske, *Fault attacks in pairing-based cryptography.* Masters thesis, University of Paderborn, 2011.

[13] A. Menezes and N. Koblitz, “Pairing-based cryptography at high security levels,” in *Proceedings of Cryptography and Coding*, LNCS 3796, Springer-Verlag, pp. 13–36, 2005.

[14] N. El Mrabet, “What about vulnerability to a fault attack of the Millers algorithm during an identity based protocol?,” in *Advances in Information Security and Assurance*, LNCS 5576, Springer-Verlag, pp. 122–134, 2009.

[15] N. El Mrabet, D. Page, and F. Vercauteren, “Fault attacks on pairing-based cryptography,” in *Fault Analysis in Cryptography*, Springer-Verlag, pp. 221–236, 2012.

[16] D. Page and F. Vercauteren, “A fault attack on pairing-based cryptography,” *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1075–1080, 2006.

[17] M. Scott, N. Costigan, and W. Abdulwahab, “Implementing cryptographic pairings on smartcards,” in *Cryptographic Hardware and Embedded Systems-CHES 2006*, LNCS 4249, Springer-Verlag, pp. 134–147, 2006.

[18] A. Shamir, “Identity-based cryptosystems and signature schemes,” in *Advances in Cryptology-CRYPTO 84*, LNCS 196, Springer-Verlag, pp. 47–53, 1984.

[19] S. Skorobogatov and R. Anderson, "Optical fault induction attacks," in *Cryptographic Hardware and Embedded Systems-CHES 2002*, LNCS 2523, Springer-Verlag, pp. 2–12, 2003.
[20] C. Whelan and M. Scott, "The importance of the final exponentiation in pairings when considering fault attacks," in *Pairing-Based Cryptography–Pairing 2007*, LNCS 4575, Springer-Verlag, pp. 225–246, 2007.
[21] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Proceedings of the IEEE International Test Conference*, pp. 339–344, 2004.

Appendix:

We perform our experiment using Magma software package.

Field parameters

Elliptic curve: $y^2 + y = x^3 + x + 1$
Reduction polynomial: $x^{379} + x^{315} + x^{301} + x^{287} + 1$
Input points:
P:
(0x38791B1721C5109810ADDBED960AAD4FE68709EF85A8C67B997A6A5D82D4358F0F2F908601A6299CC31C6BD91D2F216, 0x2B9BCF3F190BC53D5C20B9B5E1D476644866E9122B122409702055CA166DFEC19A5F1ED3591920267D1A65D3953F319)
Q:
(0x3E36A5A789E2D2A19F09F4FCE2A7044AE2695DA6CC3D4FA136E740705BA9993E56AB1AF73F0B5EE0305C4B6BCD7398F, 0x12FB2A298BC843483C5FC6C19BED5CE938B851FFB065453272B6CC6E067A95796C8117BAFE30486FC2287EBB643E22D)

We consider a fault in λ in the last iteration, and we get two outputs from two (valid, faulty) pairing executions.

Output from valid execution
{0x4498523382F27F697585C8BBBCE708BC61A82A939DB9EF7EFE22D02BC199D5C292DA9DE0BA398AC338FD3FF1C6C5B53, 0x99FF313813DEAA7F845333923F50D5B09C97D4D12EE93A656050FB4BC2A7F183E26B5CFC86519C47CE4ACA76EB0F01, 0x71E919046D1EE7B200DE7BD743C4016335D05B18D6D5FFB8701FAE79BE9E71F5E3D697CC2E55F201D6CC916E5585FE6, 0x4BD987B8F89A42F339AB7A4385163AE841D79056B9E26E2E1C804437D5B3210EA438CB13A0864739664A24F08013E1F}
Output from faulty execution
{0x48CD2AA1D9F7528E8E0FABD4F3302503AFD73377377AF91031675147470B7BD3714C2CC6D6479CF3E7F3E3F80D9113D, 0x4BAFAABAD7399A64A464EFD1EE19113359B8539103B4CA571DD3E5E79715DB1B5C7669122A01B3AA4EF8A16AAA20012, 0xE117EF4235D06410F21D4BEE7A0F72B5D4B92A995CCC111E7316BCC94EDA4C738CD1DD9B22FF884F9C2980DF889A4F, 0x60D4FB4199200F26A76B6ADF21A5C67630B669DC32541C3B254C04FC6F99A8483685EAB766735D59F22B192E899C564}
Divide the valid pairing by the faulty pairing
{0x149C10C7D1C377D76273839A712667EE85571A7BBC87E9A5CF051B35B12F023049F3FAF6374DA79A8E1785D40B49C38, 0x6326FAA67F8C60A4DEE0E1F7E8F2F0B8404919683A551CEA4998035F1F265AC9BE3DDF067388B2EFC5BE604A96C08A4, 0x4EB74566CFE16D6852C026D846D4D883712A3EE4E802EDD2BC66AD91AC312354D2E28D4ED94A714E41331D9F173DE89, 0x143618250F0C74A857668CD59907D175A105A59E6163161B74F6A1DFE3B47B9AC24F118180D90DEC20B0650567F3FC2}
Solving the system (4), we get
l_1=0x5807EDADE1A0F31D7CADAE5514C729356782B07D9E8DA9B0714FB78341D419D0DAC7D8424D37672CFD5403319706549;
l_0=0x14EBF98F6F3796237818A9245F42742B0376303A8CBB016B9ED9A4485B28EF99EA1F20644F3D73FBF316D3AB4DF5346;
λ=0x6631480A684221BCE3A45AA9F6602D7F85EBEDDB52B0E61147A8F7F31A7D80EE8C6CC2B5723C39CCCD08485A5A75CC7;
$\lambda(x_Q+x_T+1)$=0x7B49EDD40B89405B17221D0A4C9FBA02962262F442C9161B560C89D5148C4FDD13850E931ED180D40232C9DDF64C295.

When the secret point is Q

$x_Q = \lambda+l_1+1$=0x3E36A5A789E2D2A19F09F4FCE2A7044AE2695DA6CC3D4FA136E740705BA9993E56AB1AF73F0B5EE0305C4B6BCD7398F
Since P is known and we have gotten the value of $\lambda(x_Q + x_T + 1)$, then
$y_T = y_P^{2^{m-2}}+1$=0x7D593E72EF769530536572EF883092C0ADEC03317E175242BA63E1F349DE353D951B394DAFDCBB40330C64CDDF873FE
$y_Q = l_0 + Y_T + \lambda(x_Q + x_T + 1)$=0x12FB2A298BC843483C5FC6C19BED5CE938B851FFB065453272B6CC6E067A95796C8117BAFE30486FC2287EBB643E22D

When the secret point is P

$x_P = x_T^4 = (\lambda-1)^2$=0x38791B1721C5109810ADDBED960AAD4FE68709EF85A8C67B997A6A5D82D4358F0F2F908601A6299CC31C6BD91D2F216
$y_P = (l_0+y_Q+\lambda(x_Q+x_T+1)+\tau(m-1))^4$=0x2B9BCF3F190BC53D5C20B9B5E1D4766448666E9122B122409702055CA166DFEC19A5F1ED3591920267D1A65D3953F319

Yunqi Dou received his B.S. degrees in applied mathematics from the Zhengzhou Information Science and Technology Institute, China, in 2010. He is currently pursuing his M.S degree in department of Information Research, Zhengzhou Information Science and Technology Institute, Zhengzhou, China. His research fields include Elliptic curve cryptography and side-channel attack.

Jiang Weng received his B.S. and M.S. degrees in applied mathematics from the Zhengzhou Information Science and Technology Institute, China, in 2009 and 2012, respectively. He is currently pursuing his Ph.D. degree in department of Information Research, Zhengzhou Information Science and Technology Institute, Zhengzhou, China. His research fields include Elliptic curve cryptography and Pairing-based Cryptography.

Chuangui Ma received his Ph.D. degree in mathematics in 1998 from Zhejiang University. He is currently a professor in the Department of Information Research, Zhengzhou Information Science and Technology Institute, Zhengzhou, China. His research field is information security.