Fault Attack on a Point Blinding Countermeasure of Pairing Algorithms

Jea Hoon Park, Gyo Yong Sohn, and Sang Jae Moon

Recently, Page and Vercauteren proposed a fault attack on pairing algorithms and two countermeasures against such an attack. The countermeasure uses either a random scalar or a random point to blind the input points. To defeat the countermeasure using a random point, we utilize the point addition formula on an elliptic curve. As a result, we successfully defeat the countermeasure using a random point.

Keywords: Fault attack, pairing algorithm, countermeasure, side-channel attack, elliptic curve.

I. Introduction

Page and Vercauteren first proposed a fault attack on pairing algorithms and two countermeasures against such an attack [1]. After the introduction of the fault attack, although the attack method and countermeasures have since been improved, their countermeasures have been referred to as secure countermeasures [2]-[4].

In [10], Ghosh and others exposed the weaknesses of the countermeasures of Page and Vercauteren. However, due to the fact that Ghosh and others did not consider the bilinearity property of pairing, their attack is infeasible.

This letter shows that the fault attack by Page and Vercauteren can be applied to the countermeasure that blinds the input point using the addition of a random point. The countermeasure does not modify the point addition formula but changes the value of the input point to an arbitrary and unknown value. To apply the fault attack on the countermeasure, we propose a utilization of the addition formula between the input point and a random point instead of the blinded input point. Using the roots of the equations deduced by the utilization, we successfully defeat the countermeasure.

Manuscript received Dec. 17, 2010; revised Feb. 10, 2011; accepted Feb. 24, 2011.

Jea Hoon Park (phone: +82 53 940 8817, email: jenoon65@ee.knu.ac.kr) and Gyo Yong Sohn (corresponding author, email: gysohn74@hanmail.net) are with the School of Computer Science and Engineering, Kyungpook National University, Daegu, Rep. of Korea.

Sang Jae Moon (email: sjmoon@ee.knu.ac.kr) is with the College of IT Engineering, Electronics Engineering, Kyungpook National University, Daegu, Rep. of Korea.

http://dx.doi.org/10.4218/etrij.11.0210.0483

II. Introduction to Tate Pairing

We briefly describe an algorithm of Tate pairing on an elliptic curve. Let E be an elliptic curve over a finite field F_q, and let O be the point at infinity. Let l be a large prime factor of $\#E(F_q)$, and let k be the smallest positive integer such that $l \mid (q^k-1)$. Let $E(F_q)[l]$ be the set of l-torsion points in $E(F_q)$. The Tate pairing then becomes the map

$$e_l : E(F_q)[l] \times E(F_{q^k})[l] \rightarrow F_{q^k}^* / (F_{q^k}^*)^l,$$

given by $e_l(P,Q) = f_{l,P}(D_Q)$. Here, $f_{l,P}$ is a rational function on E whose divisor is equivalent to $l(P)-l(O)$, that is, $div(f_{l,P})=l(P)-([l]P)-(l-1)O$, and D_Q is a divisor of degree 0 equivalent to $(Q)-(O)$. Both $div(f_{l,P})$ and D_Q have disjoint supports. The Tate pairing is well defined and satisfies the bilinearity $e_l(P, tQ)=e_l(tP, Q)=e_l(P, Q)^t$ for any integer $t \neq 0$, and non-degenerate, that is, there exists a $Q \in E(F_{q^k})[l]$ such that $e_l(P,Q) \neq 1 \in (F_{q^k}^*)^l$. The output of the Tate pairing is not unique but is determined up to an element in the quotient group $(F_{q^k}^*)/(F_{q^k}^*)^l$. To produce a unique value, Barreto and others [8] proved that the reduced Tate pairing can be defined as

$$e_l(P,Q) = f_{l,P}(Q)^{\frac{q^k-1}{l}} \in \mu_l \subset F_{q^k}^*,$$

where μ_l is the group of l-th root of unity of F_{q^k}. The

computation of the $(q^k-1)/l$-th power is referred to as a final exponentiation.

In this work, we will consider the pairing algorithm on the supersingular elliptic curve in characteristic 3, that is, $q=3^m$ with gcd(m, 6)=1. Let E be a supersingular elliptic curve over F_{3^m} :

$$E: y^2 = x^3 - x + b, \text{ with } b \in \{-1, 1\}.$$

Let $F_{q^3} = F_q[\rho]/(\rho^3 - \rho - b)$ and $F_{q^6} = F_{q^3}[\sigma]/(\sigma^2 + 1)$. The distortion map $\phi: E(F_q) \to E(F_{q^6})$ is then defined by $\varphi(x, y) = (\rho - x, \sigma y)$, where $\sigma^2 = -1$ for $\sigma, \rho \in F_{q^3}$ and $\rho^3 = \rho + b$ for $\sigma, \rho \in F_{q^6}$. The modified Tate pairing is then defined as $e_m(P, Q) = f_P(\varphi(Q))$ [5]. Several improvements have been proposed [5]-[8]. Algorithm 1 is the Duursma-Lee method for hyperelliptic curves in characteristic 3.

Algorithm 1. Duursma-Lee algorithm.

Input: point $P = (x_P, y_P)$, point $Q = (x_Q, y_Q)$

Output: $e_m(P, Q) = f_P(\varphi(Q)) \in \mu_l \subset F_{q^6}^*$

1 $f \leftarrow 1$
2 for $i = 1$ to m do
3 $x_P \leftarrow x_P^3$, $y_P \leftarrow y_P^3$
4 $\mu \leftarrow x_P + x_Q + b$
5 $\lambda \leftarrow -y_P y_Q \sigma - \mu^2$
6 $g \leftarrow \lambda - \mu\rho - \rho^2$
7 $f \leftarrow f \cdot g$
8 $x_Q \leftarrow x_Q^{1/3}$, $y_Q \leftarrow y_Q^{1/3}$
9 return f^{q^3-1}

III. Fault Attack and Countermeasures by Page and Vercauteren

In 2006, Page and Vercauteren initially proposed a fault attack on pairing algorithms [1]. They assumed a corruption of the loop bound in the 'for' procedure of the pairing algorithms. If an attacker modified the loop bound m to m+1 of algorithm 1, instead of producing a product of polynomials of the form

$$e_m(P, Q) = \prod_{i=1}^{m} (-y_P^{3^i} \cdot y_Q^{3^{m-i+1}} \sigma - \mu_i^2 - \mu_i \rho - \rho^2) \quad (1)$$

with $\mu_i = x_P^{3^i} + x_Q^{3^{m-i+1}} + b$, algorithm 1 produces

$$e_{m+1}(P, Q) = \prod_{i=1}^{m+1} (-y_P^{3^i} \cdot y_Q^{3^{m-i+1}} \sigma - \mu_i^2 - \mu_i \rho - \rho^2) \quad (2)$$

for the loop bound m+1. If the attacker uses g_i to denote the i-th factor of a product produced by algorithm 1, by dividing (2) by (1), he obtains a single factor

$$\frac{e_{m+1}(P, Q)}{e_m(P, Q)} = g_{m+1} = -y_P^{3^{m+1}} \cdot y_Q \sigma - \mu_{m+1}^2 - \mu_{m+1}\rho - \rho^2 \quad (3)$$

after reversing the final powering. Because the attacker knows that $z^{3^m} = z$ for all elements $z \in F_q$, he can extract x_P or y_P, given that he knows x_Q and y_Q, and hence reconstruct the secret point.

As mentioned above, an attack on the pairing algorithms is only successful when the adversary has knowledge of one input point. Therefore, it is natural to utilize point blinding techniques to construct a defense mechanism. Page and Vercauteren also proposed two countermeasures using a point blinding method:

New Point Blinding Techniques.

$$e_m(aP, bQ) = e_m(P, Q)^{ab} = e_m(P, Q),$$

where a and b are random numbers such that $a \cdot b \equiv 1 \pmod{l}$.

Altering Traditional Point Blinding.

$$\begin{aligned} & e_m(P, Q + R) \cdot e_m(P, R)^{-1} \\ & = e_m(P, Q) \cdot e_m(P, R) \cdot e_m(P, R)^{-1} = e_m(P, Q), \end{aligned} \quad (4)$$

where R is a random point.

The two countermeasures above utilize the bilinearity of pairing algorithms and involve random factors. Random factors affect the intermediate computation of pairing algorithms, but these are eliminated at the end of the algorithm.

IV. Weakness of Previous Countermeasure

The countermeasure based on altering the traditional point blinding is vulnerable to the fault attack by Page and Vercauteren. Given the secret point $P=(x_P, y_P)$, the random point $R=(x_R, y_R)$, and the public point $Q=(x_Q, y_Q)$, Q can be chosen by an attacker and R is updated in every execution. We denote $S=e_m(P, R)^{-1}$, $Q+R=T=(x_T, y_T)$. Even if algorithm 1 adopts the countermeasure using T instead of Q as described in (4), the attacker still obtains a single $(m+1)$th factor of a product. Algorithm 1 is executed without a corruption of the loop bound m, resulting in the correct output $e_m(P, Q)$. Subsequently, the attacker executes algorithm 1 after the modification of the loop bound m to m+1. The attacker can deduce a single $(m+1)$th factor of a product by

$$\frac{e_{m+1}(P, T) \cdot S}{e_m(P, Q)} = \frac{e_m(P, T) \cdot S \cdot \hat{g}_{m+1}^{q^3-1}}{e_m(P, Q)} = \hat{g}_{m+1}^{q^3-1}.$$

The final powering is then reversed by the method in [1]:

$$\hat{g}_{m+1} = -y_P^{3^{m+1}} \cdot y_T\sigma - \hat{\mu}_{m+1}^2 - \hat{\mu}_{m+1}\rho - \rho^2, \quad (5)$$

where $\hat{\mu}_{m+1} = x_P^{3^{m+1}} + x_T + b$. Note that the value of $\hat{g}_{m+1}$ can be extracted from the ratio of the pairings without regard to the updating method of the random point R. $\hat{g}_{m+1} \in F_{q^6}$ is denoted by

$$\hat{g}_{m+1} = -D_0\sigma - D_1 - D_2\rho - \rho^2,$$

where D_0, D_1, and D_2 are in F_q. However, the attacker cannot extract x_P or y_P from $\hat{g}_{m+1}$ because x_T and y_T are unknown.

To extract the secret point P, we utilize the point addition formula on an elliptic curve instead of the blinded input point T. In (5), we do not know the coordinates of T, x_T and y_T, but we know the explicit formula for the addition of two points Q and R on a curve defined over F_q. If $Q \neq R$, then $Q+R=T=(x_T, y_T)$, where

$$x_T = \lambda^2 - x_Q - x_R,\ y_T = y_Q + y_R - \lambda^3$$

with $\lambda=(y_Q-y_R)/(x_Q-x_R)$. Thus, after substituting the addition formula for x_T and y_T in (5), we can obtain four equations of the unknown values of x_P, y_P, x_R, and y_R as follows:

$$y_P^3 \cdot (y_Q + y_R - \lambda^3) = D_0, \quad (6)$$

$$x_P^3 + (\lambda^2 - x_Q - x_R) + b = D_2, \quad (7)$$

$$y_P^2 - x_P^3 + x_P - b = 0,$$

$$y_R^2 - x_R^3 + x_R - b = 0.$$

We easily solve the above nonlinear equations to obtain the secret point P. In the case of $Q=R$, we use the doubling formula instead of x_T and y_T in (5).

Our approach to solve the equations is as follows. Eliminating the variables y_P and y_R in (6) and (7) using two elliptic curve equations, we obtain two polynomials in $F_q[x_P, x_R]$. Denote $f(x_P, x_R)$ and $g(x_P, x_R)$, respectively. To find the coordinates of point P or R, we use the resultant method which gives their common roots [9]. At this point, we consider that f and g are defined as polynomials in x_P with the coefficients in $F_q[x_R]$. The resultant of f and g, $Res_{x_P}(f,g)$, gives a polynomial in x_R whose roots are the x_R coordinates of the intersection of f and g. The degree of $Res_{x_P}(f,g)$ is 144. In practice, we can reduce the complexity, thereby using simplified f and g, $\tilde{f}(x_P^3, x_R)$ and $\tilde{g}(x_P^3, x_R)$. In this case, because the degree of the coefficients of f and g are at most 4 and 12, respectively, $Res_{x_P^3}(\tilde{f},\tilde{g})$ is of degree 48. It can be factored into a product of 13 linear factors, a linear factor with a multiplicity of 24, and an irreducible factor over F_q. Given that the leading coefficients of f and g are excepted for candidates of the roots in $Res_{x_P^3}(\tilde{f},\tilde{g})$, we have at most 13 possible solutions for x_R and 13×2 possible points for R (= $(x_R y_R)$). Thus, from (6) and (7), we can extract the secret point P. Similarly, we can also find the secret point P using same method when f and g are defined as polynomials in x_R with coefficients in $F_q[x_P]$.

V. Simulation Result

We simulated our approach in C++ using Shoup's NTL library on a Pentium 2.13-GHz computer with less than 2 GB of memory. The NTL helps perform the arithmetic of finite fields and polynomials using an FFT algorithm.

Let the elliptic curve E: $y^2=x^3-x+1$ over the finite field F_q, where $q=3^m$ and $m=97$. The number of points of E is
#E(F_q)=1908805632340782707542472558694483331020023904 7 (154 bit).
The secret point P is
P=([2DE85A5A2B17D9D41444ED50A80D749BB266B06], [3059A7F8FA2647277429A5006EFEE588ABB2AB8]).
The public point Q is
Q=([28E51C31B47DD808962E6CA4606BDD736E78374], [8E62592D5E1E77D55C770D4DB59D51D32F0531]).
The random point R is
R=([1EF95B1863C99EC17B5686F5A1B5F9AE7979C33], [1FACE5EBA638EC93B2D41123D1112B539CD386C]).
We compute the ratio of pairings by

$$\frac{e_{m+1}(P,Q+R)\cdot S}{e_m(P,Q)} = \hat{g}_{m+1}^{q^3-1}$$
$$= [1212A4AD49B5133B146709F6014C33B93604DCF]$$
$$+ [20DE9C17129CCCFE67275A5B29D3D5949CDF69]\rho$$
$$+ [15AA5BD97EA10974AF4CC0887EA1C6BA62B303]\rho^2$$
$$+ [1A054D26768204C591B098D7F4fCCD9C60445B1]\sigma$$
$$+ [2AF83C1C627F3DABFBDC26418DD1736BEA98B40]\sigma\rho$$
$$+ [2CD8dA6A436C7FBAD71F30DFBA05BB1C3865545]\sigma\rho^2.$$

From the root finding method, we can obtain $\hat{g}_{m+1}$ on the basis of $\{1, \rho, \rho^2, \sigma, \sigma\rho, \sigma\rho^2\}$:

$$\hat{g}_{m+1}$$
$$= -[2D2AAC64217DF287DC09F84E4BA3D7B92BBD36D]$$
$$- [29407E4825A43399A1AD241664BB49EDD653125]\rho - \rho^2$$
$$- [838A295514228CFFFF3F2C0F2A37110D46CB22]\sigma.$$

Thus, after substituting the addition formula for blinded values, we can obtain four equations as follows:

$$\begin{cases} y_P^3 \cdot (y_Q + y_R - \lambda^3) \\ = [838\,A\,295514228\,CFFFF\,3\,F\,2\,C\,0\,F\,2\,A\,37110\,D\,46\,CB\,22], \\ x_P^3 + (\lambda^2 - x_Q - x_R) + b \\ = [29407\,E\,4825\,A\,43399\,A\,1\,AD\,241664\,BB\,49\,EDD\,653125], \\ y_P^2 - x_P^3 + x_P - b = 0, \\ y_R^2 - x_R^3 + x_R - b = 0. \end{cases}$$

Subsequently, as stated above, we can calculate $\tilde{f}(x_P^3, x_R)$ and $\tilde{g}(x_P^3, x_R)$. The result of $Res_{x_P^3}(\tilde{f}, \tilde{g})$ is

$$Res_{x_P^3}(\tilde{f}, \tilde{g}) = (x_R - u_1) \times (x_R - u_2) \times \cdots \times (x_R - u_{13}) \times (x_R^{11} + v_{10} x_R^{10} + \cdots + v_1 x_R + v_0) \times (x_R - w)^{24}.$$

From the above equation, 13 linear factors become the candidates of x_R:

u_1=[3506D9218BB15381D675A0b31F07761E3496FD6],
$u_{2=}$ [9A39EAF6809BE65942AE8E70688A97A7EF69BF],
u_3 =[2290417E9E6718309601C96D2BB96D2E6F67249],
u_4=[1B5085342E90A3F61A1E165F61FE6B2FB49E63B],
u_5= [2F0D1005D82FE558F1C97408C8FB9CF5FF2C43E],
u_6= [2A6C6483F6AA7D0CA6ABA0923DA74CE6FB3A0A2],
u_7=[181DC05DB6280AF124F70FA776132CBB2BED707],
u_8= [48076E348847CD077957F08189A99DDB87B4F],
u_9= [117F65862EB5EBA938422CF16E3BB38B7E42397],
u_{10}= [1408AFEDD7C79945DC7AB7A8A692C49CF76FA7E],
u_{11}= [40173942798BCC002D29762B50C5200EA921E3],
u_{12}= [1EF95B1863C99EC17B5686F5A1B5F9AE7979C33],
u_{13}= [308CB2D8F93488C3B8F26EDCCD25F0513D80FC4].

We can extract x_P or y_P from (5), given that we know the public point Q and the 26 candidates of R, and hence reconstruct the secret point P.

VI. Conclusion

This letter showed that the previous countermeasure is not secure against the fault attack by Page and Vercauteren. We utilized the addition formula on an elliptic curve instead of a blinded point to obtain enough nonlinear equations for finding the secret point. Thus, from the pairing algorithm adopting the countermeasure using a random point, we successfully extracted the secret point. This study provides a warning regarding the usage of the countermeasure with a random point, and we thus recommend the countermeasure with a random scalar among the two previous countermeasures.

References

[1] D. Page and F. Vercauteren, "A Fault Attack on Pairing Based Cryptography," *IEEE Trans. Comput.*, vol. 55, no. 9, Sept. 2006, pp. 1075-1080.

[2] C. Whelan and M. Scott, "The Importance of the Final Exponentiation in Pairings When Considering Fault Attacks," *Proc. Pairing*, *LNCS 4575*, 2007, pp. 225-246.

[3] T. Kim et al., "Power Analysis Attacks and Countermeasures on η_T Pairing over Binary Fields," *ETRI J.*, vol. 30, no. 1, Feb. 2008, pp. 68-80.

[4] M. Shirase, T. Takagi, and E. Okamoto, "An Efficient Countermeasure against Side Channel Attacks for Pairing Computation," *Proc. ISPEC*, *LNCS 4991*, 2008, pp. 290-303.

[5] I. Duursma and H. Lee, "Tate Pairing Implementation for Hyperelliptic Curves $y^2 = x^p - x + d$," *Proc. Asiacrypt*, *LNCS 2894*, 2003, pp.111-123.

[6] S. Kwon, "Efficient Tate Pairing Computation for Elliptic Curves over Binary Fields," *Proc. ACISP*, *LNCS 3574*, 2005, pp. 134-145.

[7] P. Barreto et al., "Efficient Pairing Computation on Supersingular Abelian Varieties," *Design, Codes and Cryptography*, vol. 42, no. 3, Feb. 2007, pp. 239-271.

[8] P. Barreto et al., "Efficient Algorithms for Pairing-Based Cryptosystems," *Proc. CRYPTO*, *LNCS 2442*, 2002, pp. 354-369.

[9] S. Lang, *Algebra*, rev. 3rd ed., vol. 211, *Graduate Texts in Mathematics*, New York, N.Y.: Springer-Verlag, 2002.

[10] S. Ghosh, D. Mukhopadhyay, and D. Chowdhury, "Fault Attack and Countermeasures on Pairing-Based Cryptography," *Int. J. Netw. Security*, vol. 12, no. 1, Jan. 2011, pp. 21-28.