jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2019, VOL. 65, NO. 2, PP. 203–209
Manuscript received December 23, 2018; revised April, 2019. DOI: 10.24425/ijet.2019.126302

Combined small subgroups and side-channel attack on elliptic curves with cofactor divisible by 2^m

Michał Wroński

***Abstract*—Nowadays, alternative models of elliptic curves like Montgomery, Edwards, twisted Edwards, Hessian, twisted Hessian, Huff's curves and many others are very popular and many people use them in cryptosystems which are based on elliptic curve cryptography. Most of these models allow to use fast and complete arithmetic which is especially convenient in fast implementations that are side-channel attacks resistant. Montgomery, Edwards and twisted Edwards curves have always order of group of rational points divisible by 4. Huff's curves have always order of rational points divisible by 8. Moreover, sometimes to get fast and efficient implementations one can choose elliptic curve with even bigger cofactor, for example 16. Of course the bigger cofactor is, the smaller is the security of cryptosystem which uses such elliptic curve. In this article will be checked what influence on the security has form of cofactor of elliptic curve and will be showed that in some situations elliptic curves with cofactor divisible by 2^m are vulnerable for combined small subgroups and side-channel attacks.**

***Keywords*—Small subgroups attack, side-channel attack, alternative models of elliptic curves**

I. Introduction

It is well known that during implementations of protocols using elliptic curve cryptography it should be checked if:

- the point which multiplicity k one wants to compute lies on elliptic curve,
- computed point is not the point at infinity or it should be ensured that scalar k will be smaller than prime order r of the given point,
- side-channel resistant method of point scalar multiplication like ladder or complete arithmetic is implemented.

Because very frequently base field used for ECC is prime field $\mathbb{F}_p$, where p and $r = \#E(\mathbb{F}_p)$ are large coprime primes, it is easy to forget about checking if given point which lies on elliptic curve is the one of the correct order. Alternative models of elliptic curves like Montgomery, Edwards, twisted Edwards or Huff's curves have always order of form hr, where r in cryptographic applications is large prime and h is cofactor divisible by 4, so checking if given point has correct order should be always performed. If one wants to fast find if order of given point $P \neq O$ is equal to prime r, then the easiest way is to compute $[r]P$ and check if it is equal to O. If yes, then P has order r. Otherwise, P has order different from r. Sometimes checking of the order of a given point is not performed because it requires computing of point scalar multiplication by r, so it is very inefficient and small subgroup attacks which may be in this case performed seems not to be too dangerous. If elliptic curve is cyclic, then one can find point P' of order hr and then perform attack faster than generic Pollard's Rho for point P of order r for about $\alpha\sqrt{h}$ times, where α denotes the ratio of expected time of computing discrete logarithm on elliptic curve using Pollard's rho to Gaudry-Schost algorithm over the same field and the same size of interval. It is worth to note that for small h the gain is very small. Unfortunately, if order of the point is not checked and cofactor is divisible by 2^m, then adversary who performed side-channel attack has knowledge about m the least significant bits of private key k. Combined attack using points of low order and side-channel attack is described in [1], where it is assumed that adversary has unlimited access to the device and he is able to make fault injections. Genkin et al. in [2] described the flush+reload attack on Curve25519 using side-channel leakage during interactions of the point at infinity, order-2, and order-4 elements in the Montgomery ladder. In this article will be only assumed that order of the given point is not checked and then the scalar multiplication of the point lying on the curve but having not prime order r may be done.

II. Alternative models of elliptic curves

In this section are described basic information about alternative models of elliptic curves having order of rational points always divisible at least by 4. For these alternative models of elliptic curve combined small subgroups and side-channel attack may be performed if order of given point is not checked.

A. Montgomery curves

Montgomery curves were developed by Montgomery and described in [3]. Montgomery curve over field $\mathbb{K}$ is given by the equation $E_{M,a,b}/\mathbb{K} : by^2 = x^3 + ax^2 + x$, where $a, b \in \mathbb{K}$ and $b(a^2 - 4) \neq 0$. The neutral element of group addition law is point at infinity O and opposite point to the point $P = (x, y)$ is point $-P = (x, -y)$. For every Montgomery curve exists some isomorphic elliptic curve in Weierstrass form. The order of rational points of Montgomery curve is always divisible by 4. Moreover, every Montgomery curve is birationally equivalent with some twisted Edwards curve. On Montgomery curve it is possible to use Montgomery ladder using XZ coordinates.

B. Edwards and twisted Edwards curves

Edwards and tiwsted Edwards curves are well described in [4], [5] and [6].

Twisted Edwards curve over field $\mathbb{K}$ with characteristic different from 2 is given by the equation $E_{TE,a,d}/\mathbb{K} : ax^2 + y^2 = 1 + dx^2y^2$, where $a, d \in \mathbb{K}$ and $a(d-1) \neq 0$. Moreover, a is square in $\mathbb{K}$ and d is not square or a is not square in $\mathbb{K}$ and then d is a square. For every twisted Edwards curve exists some isomorphic elliptic curve in short Weierstrass form. If

M. Wroński is with Institute of Mathematics and Cryptology, Faculty of Cybernetics, Military University of Technology, Warsaw, Poland (e-mail: michal.wronski@wat.edu.pl).

$a = 1$, then curve is an Edwards curve. The point $(0, 1)$ is neutral element of group addition law and for point $P = (x, y)$ and negation of point P is $-P = (-x_1, y_1)$. Because it is possible to use the same formula for points addition and point doubling and no special cases like addition of opposite point or point at infinity need to be considered, on twisted Edwards curve it is possible to use complete arithmetic.

C. Huff's curves

Huff's curves were firstly described in [7]. Let $\mathbb{K}$ denote a field of characteristic different from 2. Huff's curve is given by equation

$$E_{Hu,a,b}/\mathbb{K} : ay(x^2 - 1) = by(x^2 - 1),$$

where $a, b \in \mathbb{K}^*$ and $a^2 \neq b^2$. The neutral point of group addition law is $O = (0, 0)$ and opposite point for $P = (x, y)$ is point $-P = (-x, -y)$. Every elliptic curve over field of odd characteristic which contains a copy of $\mathbb{Z}/4\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ is birationally equivalent with some Huff's curve. On Huff's curves it is possible to use complete arithmetic for point scalar multiplication.

III. Methods of point scalar multiplication on elliptic curves

A. Binary (double-and-add) method

Double-and-add method is one of the simplest method of point scalar multiplication on elliptic curves. Its origin comes from fast modular powering method. In double-and-add method the private key k needs to be written in binary form. Computations start from the most significant bit and ends after computations for the least significant bit. If the given bit is equal to 0, then it is necessary to make doubling of the actual point and if bit is equal to 1, then additionally it is required to make addition of the generator to the given point. This method is presented in the algorithm 1 and its resistance against side-channel attacks is described in subsection VII-A.

Algorithm 1 Binary (double-and-add) method

```
Input: $P \in E$,$k \in Z$, $k = (k_{d-1}, ..., k_0)_2$
Output: $Q = [k]P$
$Q = O$;
for $j = l - 1$ to $0$ by $-1$ do
    $Q = doubling(Q)$;
    if $k_j = 1$ then
        $Q = addition(Q, P)$;
    end
end
return $Q$;
```

If points addition and doubling is computed using the same formulas and no special cases need to be considered, then it is possible to use double-and-add method with complete arithmetic. In this case doubling is computed as addition of two the same points. The algorithm of point scalar multiplication using double-and-add method and complete arithmetic is presented in the algorithm 2. Resistance of this method against side-channel attacks is described in subsection VII-B.

Algorithm 2 Binary (double-and-add) method using complete arithmetic

```
Input: $P \in E$,$k \in Z$, $k = (k_{d-1}, ..., k_0)_2$
Output: $Q = [k]P$
$Q = O$;
for $j = l - 1$ to $0$ by $-1$ do
    $Q = addition(Q, Q)$;
    if $k_j = 1$ then
        $Q = addition(Q, P)$;
    end
end
return $Q$;
```

B. Ladders

One of the most popular method of point scalar multiplication which is simple side-channel attacks resistant is an analogue of powering ladder. The best example of such ladder is Montgomery ladder given by Montgomery in [3], which allows one to perform point scalar multiplication on Montgomery curve using XZ coordinates in constant time (if implementation is proper) for each bit. The single step of this method is presented in algorithm 3.

It is also possible to use Montgomery ladder for every model of elliptic curve using algorithm 4.

In ladder algorithms, for every bit of private key k from the pair of points $([m]P, [m + 1]P)$ one gets always $([2m]P, [2m + 1]P)$ if given bit is equal to 0 or $([2m + 1], [2m + 2]P)$, if given bit is equal to 1. The resistance of this method against side-channel attacks is described in section VII-C.

Algorithm 3 The single step of the Montgomery ladder using XZ coordinates.

```
Input: $P_1, P_2 \in E_{M,a,b}(\mathbb{K})$, where $P_1 = (X_1, Z_1)$, $P_2 = (X_2, Z_2)$, $4a_{24} = a + 2$ and $P_3 = (X_3, Z_3)$,
       $P_1, P_2, P_3 \in E_{M,a,b}/\mathbb{K} : by^2 = x^3 + ax^2 + x$
Output: $P_4 = (X_4, Z_4)$, $P_5 = (X_5, Z_5)$, $P_4, P_5 \in E_M : by^2 = x^3 + ax^2 + x$
    $A = X_2 + Z_2$;
    $AA = A^2$;
    $B = X_2 - Z_2$;
    $BB = B^2$;
    $E = AA - BB$;
    $C = X_3 + Z_3$;
    $D = X_3 - Z_3$;
    $DA = D \cdot A$;
    $CB = C \cdot B$;
    $X_5 = Z_1 \cdot (DA + CB)^2$;
    $Z_5 = X_1 \cdot (DA - CB)^2$;
    $X_4 = AA \cdot BB$;
    $Z_4 = E \cdot (BB + a_{24} \cdot E)$;
return $P_4 = (X_4, Z_4)$, $P_5 = (X_5, Z_5)$;
```

Algorithm 4 Montgomery ladder.

```
Input: Point $P$ and positive integer $k = (k_m, k_{m-1}, ..., k_0)$
Output: Point $Q = [k]P$
$R_0 = O$;
 $R_1 = P$;
 for $i$ from $m$ downto $0$ do
    if $d_i = 0$ then
        $R_1 = addition(R_0, R_1)$;
         $R_0 = doubling(R_0)$;
    end
    else
        $R_0 = addition(R_0, R_1)$;
         $R_1 = doubling(R_1)$;
    end
end
return $R_0$;
```

IV. Pohlig-Hellman and fault attacks on ECC

A. Pohlig-Hellman attack

Pohlig-Hellman attack was developed by Pohlig and Hellman in 1976 and described in [8]. It uses fact that if elliptic curve has order of group of rational points divisible by many prime factors, then it is easy to find the private key k. Let's assume that point P from curve E has order which is product of coprime factors $p_1, \dots, p_n$. Numbers $p_1, \dots, p_n$ need not to be prime but may be the power of prime number. Let $Q = [k]P$, where k is the private key and Q is the public key and $\pi = \prod_{j=1}^n p_i$. One can now compute $Q_1 = \left[\prod_{j=2}^n p_j\right] Q = [k]\left(\left[\prod_{j=2}^n p_j\right] P\right), Q_2 = \left[\prod_{j=1,j\neq 2}^n p_j\right] Q = [k]\left(\left[\prod_{j=1,j\neq 2}^n p_j\right] P\right), \dots, Q_n = \left[\prod_{j=1,j\neq n}^n p_j\right] Q = [k]\left(\left[\prod_{j=1,j\neq i}^n p_j\right] P\right)$. It is obvious that points $\left[\prod_{j=2}^n p_j\right] P$, $\left[\prod_{j=1,j\neq 2}^n p_j\right] P, \dots, \left[\prod_{j=1,j\neq n}^n p_j\right] P$ has orders $p_1, \dots, p_n$ respectively. Because numbers $p_1, \dots, p_n$ are small, then using Chinese remainder theorem for system of linear congruences

$$\begin{cases} k \equiv l_1(mod\ p_1), \\ k \equiv l_2(mod\ p_2), \\ \dots \\ k \equiv l_n(mod\ p_n), \end{cases}$$

it is easy to find k and the attack is finished.

B. Fault attacks

The description of fault attacks may be found for example in [9]–[11]. These attacks may be performed for implementations if one does not check if point which multiplicity should be computed lies on elliptic curve. There are many kinds of these attacks but two the most popular are:

1) attack on twisted curve,
2) attack on arithmetic which does not use the all parameters of elliptic curve.

The first of these attacks uses fact that the twist of elliptic curve may have order having small (comparing to the size of the base field) cofactors. In this case the adversary, for example, may find the point P^t which does not lie on elliptic curve over prime field $E/\mathbb{F}_q : y^2 = x^3 + ax + b$ but it lies on its quadratic twist $E^t/\mathbb{F}_q : cy^2 = x^3 + ax + b$, where c is not square in the field $\mathbb{F}_q$. In the next step the point P^t is sent to Alice. Even if implementations of points addition and point doubling use all parameters of elliptic curve, then the operations will be also proper for twisted curve. If $\#E^t$ has small factors, then adversary may find Alice's private key k using Pohlig-Hellman attack.

The second attack is similar but uses fact that implementations of elliptic curve operations does not require all parameters of elliptic curve. In this case adversary may use every elliptic curve which has the same parameters, used for elliptic curve operations, as curve E. It is possible to manipulate parameters which arithmetic on elliptic curve does not use. For example, for short Weierstrass curve over prime field $E/\mathbb{F}_q : y^2 = x^3 + ax + b$ one can use points addition and point doubling formulas which does not use parameter b. So adversary can find some elliptic curve of the form $E'/\mathbb{F}_q : y^2 = x^3 + ax + b'$, which order of group of rational points has small divisors. As same as in the first attack, also in this case the Pohlig-Hellman attack may be used to find k. Presented ideas may be used also for binary curves.

V. Small subgroups attack

Small subgroups attack was firstly described in the case of discrete logarithm computation over prime fields in [12]. Zuccherato in [13] showed how small subgroups attacks can be performed in the case of elliptic curves. Sometimes if implementation is not made properly, then the order r of the point is not checked. In this case, if elliptic curve has the point of order $\#E(\mathbb{F}_q)$ which has small divisors, one can make the attack described below:

1) one should find a point of order $\#E(\mathbb{F}_q)$ (if group of rational points of elliptic curve is not cyclic, then it is possible to find several points of different orders and the first steps of the attack must be then made several times, once for each point),
2) for all small coprime divisors $p_1, p_2, \dots, p_n$ of curve order $\#E$, it is necessary to compute the products $\pi_i = \prod_{j=1,j\neq i}^n p_j$,
3) in the next step one sends to Alice point P of order $\#E$, and then receives from Alice point $Q = [k]P$,
4) now it is necessary to compute $Q_1 = [\pi_1]Q = [k]([\pi_1]P), Q_2 = [\pi_2]Q = [k]([\pi_2]P), \dots, Q_n = [\pi_n]Q = [k]([\pi_n]P)$, respectively,
5) now let's see that every of points $Q_1, \dots, Q_k$ has order being divisor of π_i and because of that it is easy to find for each $i = \overline{1,n}$ the result of operation $l_i = k\ mod\ p_i$,
6) now one has to solve the system of linear congruences
$$\begin{cases} L \equiv l_1(mod\ p_1), \\ L \equiv l_2(mod\ p_2), \\ \dots \\ L \equiv l_n(mod\ p_n), \end{cases}$$
which by Chinese remainder theorem has exactly one solution in the set $\{0, \dots, \prod_{j=1}^n p_j - 1\}$,
7) now it is necessary to compute $L = k\ mod \prod_{j=1}^n p_j$ and it is easy to see that $\left[\prod_{j=1}^n p_j\right] Q = [k]\left(\left[\prod_{j=1}^n p_j\right] P\right)$,
8) making substitutions $P' = \left(\left[\prod_{j=1}^n p_j\right] P\right), Q' = \left[\prod_{j=1}^n p_j\right] Q$ one receives $Q' = [k]P'$,

9) using $k = m \cdot \prod_{j=1}^{n} p_j + L$, one gets $Q' = \left[m \cdot \prod_{j=1}^{n} p_j + L\right] P' = \left[m \cdot \prod_{j=1}^{n} p_j\right] P' + [L]P'$, which is equivalent to $Q' - [L]P' = \left[m \cdot \prod_{j=1}^{n} p_j\right] P'$ and finally to $\left[\left(\prod_{j=1}^{n} p_j\right)^{-1}\right](Q' - [L]P') = [m]P'$,

10) because $k = m \cdot \prod_{j=1}^{n} p_j + L < r$, then $m < \frac{r-L}{\prod_{j=1}^{n} p_j}$ and finally the sought value is in much smaller interval so one can use Pollard's lambda or Gaudry-Schost algorithm to find m.

Now it is easy to see that expected time for find m will be about $\alpha\sqrt{\prod_{j=1}^{n} p_i}$ shorter than expected time of finding k if no other informations are given. If cofactor is equal to 2^m then small subgroups attack will be faster for about $\alpha \cdot 2^{\frac{m}{2}}$. Of course adversary also knows the $k \ mod \ \prod_{j=1}^{n} p_i$, which gives some knowledge about k and may help to perform successful attack, what will be showed later.

VI. Working example

Given is an elliptic curve in twisted Edwards form $E_{TE,a,d}/\mathbb{F}_p : ax^2 + y^2 = 1 + dx^2y^2$ over prime field, where $p = 2^{255} - 19, a = 25, d = 2$. The order $\#E_{TE,a,d}(\mathbb{F}_p)$ of this curve is equal to $2^3 \cdot r$, where r is large prime equal to

$$r = 7237005577332262213973186563042994240\backslash$$

$$810436479626757246112137716528066531403.$$

This elliptic curve may be found in [6].

Now let's see that the order of this curve is divisible by 8. We will use this fact to perform combined attack using small subgroups and side-channel attack.

Because group of rational points of curve $E_{TE,a,d}/\mathbb{F}_p$ is cyclic, firstly is found the point of order $\#E_{TE,a,d}(\mathbb{F}_p)$. In the next step this point is sent to Alice. Alice will use her private key k for computing $Q = [k]P$, where in this example

$$P = (2561695780460675429183073143611335391\backslash$$

$$2445229295974976997496753296672030409897,$$

$$415879197266491886855167603920446349017\backslash$$

$$47251551319301365710869190965495637268)$$

and

$$k = 3246952806144862811508043550870821340\backslash$$

$$8051927704950993777692826804070954214002.$$

Then the adversary will compute $Q_1 = [r]Q$ and $P_1 = [r]P$. Because order of the point Q_1 is one of the divisors of 8, it is very easy to find out that $Q_1 = [2]P_1$. So now it is known that $k \equiv 2(mod \ 8)$. Moreover, also three the least significant bits of the private key k are known and this knowledge may be used by adversary to perform side-channel attack with bigger probability of success. It is worth to note that in [14] are presented methods of protection against this kind of attacks. The simplest way of protection against these attacks is to always use private keys k divisible by 8. But even then simple side-channel attacks need not to be much harder. One may find out that the last operations which are performed for bits equal to 0 has similar time of computing. If there are operations which requires much more/less time then there is a big probability that these operations are performed for bits of private key k equal to 1. Below will be presented some implementations of twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ arithmetic in inverted coordinates using different methods.

VII. Side-channel attacks using small subgroups attack

One should note that if cofactor of elliptic curve is divisible by 2^m, then small subgroups attacks gives knowledge about m least significant bits of private key k. This knowledge may be very dangerous, because adversary may:

1) get know what type of point scalar multiplication was used in implementation,
2) having this knowledge and characteristics from side-channel attack he can use some methods to find differences in characteristics when bit of private key is equal to 0 and when the bit of private key is equal to 1 and therefore in some cases he will be able to find the whole private key.

Of course the more bits of private key are known the easier is to perform side-channel attack and then find the whole private key. It is worth to note that even knowledge about a few bits may be very useful.

A. Double-and-add method

Classical double-and-add method for point scalar multiplication is one of the most vulnerable for side-channel attacks. Even without knowledge of any bit it should be easy to find the all bits of private key after performing this attack. On the figure 1 are presented numbers of clock cycles required for any operation of points addition or doubling and therefore because addition is longer than doubling it is easy to find the private key k. This and other implementations were implemented in Magma and use built field arithmetic.

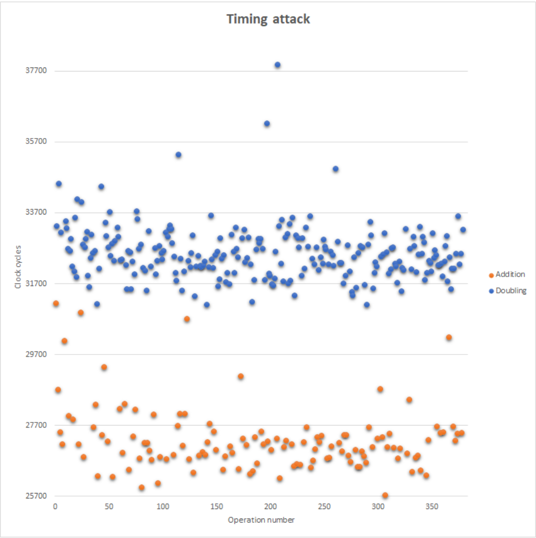

Fig. 1. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with binary method for point scalar multiplication.

B. Complete arithmetic

Complete arithmetic is much more side-channel attacks resistant than classic binary (double-and-add) method. Unfortunately, if bad implemented, this method may be also vulnerable for side-channel attacks. For example, using built field arithmetic, no matter if complete arithmetic is used, some operations during point doubling will be squares and in point addition they must be computed as normal multiplication. Moreover, for simplicity and efficiency reasons, the generator which is used in points addition may have Z coordinate equal to 1 and because of that multiplication by 1 will be much faster in generic situations than multiplication by some random number. So it is easy to see that points addition and doubling may not take exactly the same time and in such case if adversary has knowledge about even few bits, he can use this knowledge to find some more bits of private key k. This can be done by comparing the value of known bits with given characteristics from side-channel attacks. To prevent simple side-channel attacks, complete arithmetic should be implemented to be constant time. On the figure 2 is presented timing attack on badly implemented point scalar multiplication using complete arithmetic. The biggest mistake is to set in the generator P its Z-coordinate to 1 and therefore during point addition in the first step is computed $Z_2 \cdot 1$, where Z_2 is Z-coordinate of the second point. In point doubling in this step will be computed $Z_2 \cdot Z_2$, so if multiplication is not implemented as constant time, then points addition should be faster than point doubling.

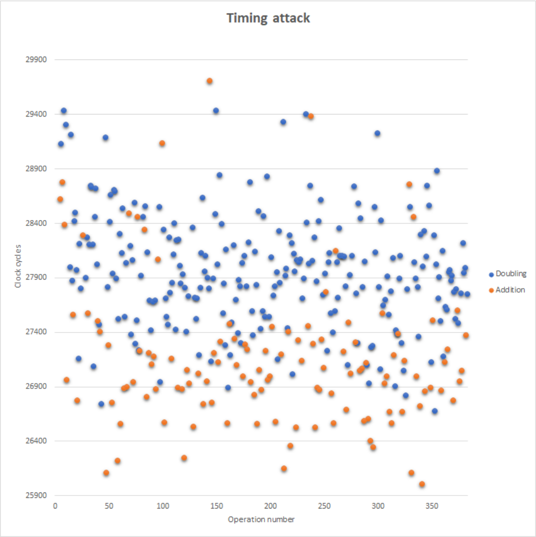

Fig. 2. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with complete arithmetic binary method of point scalar multiplication and Z-coordinate of generator equal to 1.

It is easy to see that knowledge about three the least significant bits is helpful, because adversary can see that point addition is faster than point doubling and therefore there is a big probability that it is not an accident. So for the others bits of private key there will be a big probability that points addition will be faster than doubling. This knowledge may be crucial in the process of searching for private key k. For example, operations from 380 to 383 presented in the figure 2 took:

- operation number 380 (doubling, bit of private key k is equal to 0): 27980 clock cycles,
- operation number 381 (doubling, bit of private key k is equal to 1, the first operation for this bit): 27832 clock cycles,
- operation number 382 (addition, bit of private key k is equal to 1, the second operation for this bit): 27564 clock cycles,
- operation number 383 (doubling, bit of private key k is equal to 0): 28028 clock cycles.

This situation is presented in figure 3.

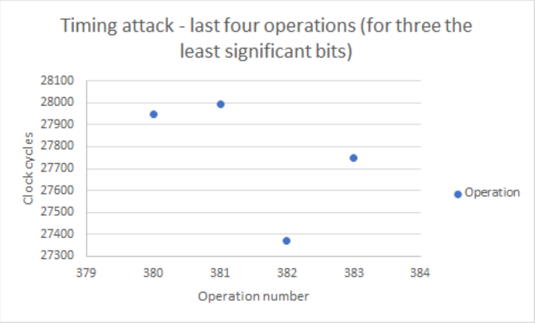

Fig. 3. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with binary method of point scalar multiplication and Z-coordinate of generator equal to 1 - three the least significant bits.

Because adversary knows that point scalar multiplication by private key k requires 383 operations and that three the least significant bits are equal to $0, 1, 0$, he may deduce that implementation uses complete arithmetic which has some implementational mistakes. Let's see that points addition took smaller amount of time (operation 382) than point doubling (operations 380, 381 and 383). It is also easy to see in this case that arithmetic does not use simple binary method, because results presented in the figure 3 are not similar to the results presented in the figure 1, where binary method with different formulas for addition and doubling was used. Therefore, adversary may guess that the average time of points addition is smaller than the average time of point doubling and he may easier find the private key k. If adversary knows three the least significant bits, then the best situation for him would be if not all of these bits are zeros, because in this case he may find some differences between their characteristics and it should be easier for him to perform side-channel attack. The probability that in this case not all bits are equal to zero is equal to $\frac{7}{8}$. It is worth to note that even if all the least significant bits are equal to zero, it is possible to get some information about used arithmetic and adversary may guess which operations are points additions and which are point doublings.

In the figure 4 is presented situation when Z-coordinate of the generator is set to some large random value. In this case it is much harder to guess which bit is equal to 0 and which is equal to 1. Even knowledge about three the least significant bits is not as helpful as previous.

C. Ladders

Using ladders like Montgomery ladder or others seems to be invulnerable for side-channel attacks even if there are some

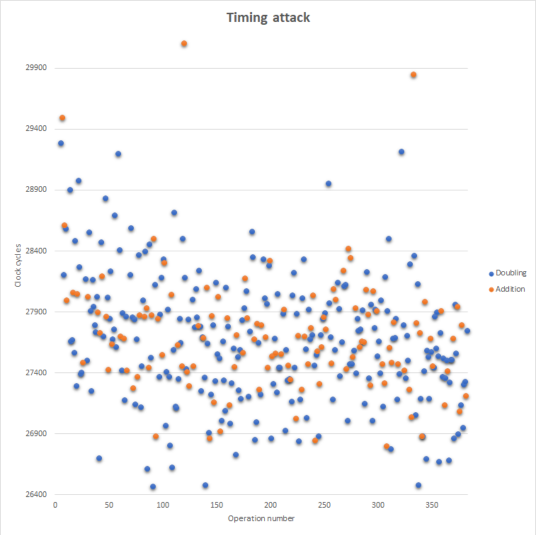

Fig. 4. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with Z-coordinate of generator being large random number.

implementational mistakes. One of such mistakes is using for example *if-else if* for checking if the given bit is equal to 0 or to 1 instead of using *case* or even *if-else* clauses. Such mistakes may result in that for one of the values (for example for value 1) of bits of the private key there will be always performed some additional cycles during checking of the first condition. Instead of that, even then it is hard to get some knowledge about any others bits of private key and knowledge of a few the least significant bits seems not to be too much helpful. Figures 5 and 6 present timing attack for ladder implementation of point scalar multiplication for Edwards curve in inverted coordinates.

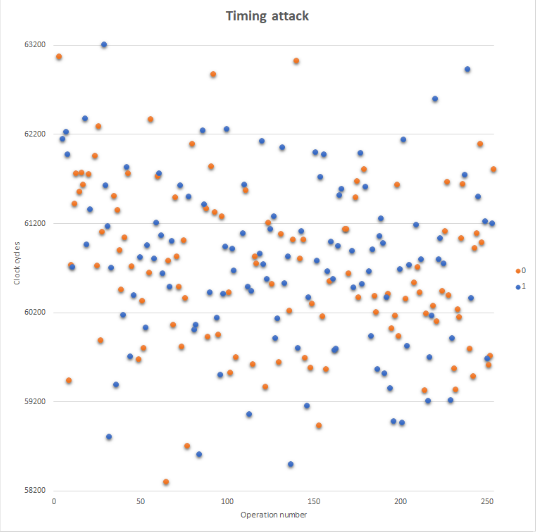

Fig. 5. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with ladder method for point scalar multiplication - number of clock cycles for all operations for given bit.

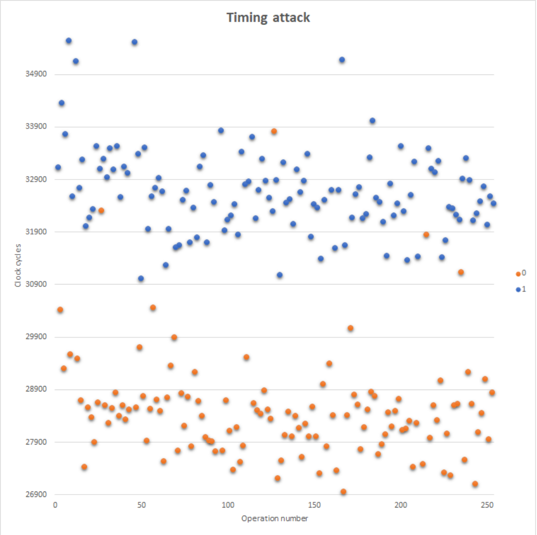

Fig. 6. Timing attack for twisted Edwards curve $E_{TE,25,2}/\mathbb{F}_{2^{255}-19}$ with ladder method for point scalar multiplication - number of clock cycles for eah operation of points addition and point doubling.

VIII. Conclusion

Using alternative models of elliptic curves with cofactor bigger than 1 requires checking during computations if order of given point is correct (large prime) or using private keys which are divisible by cofactor of elliptic curve. If the order of point is not checked, then small subgroups attack may be performed but in most cases this attack will not be dangerous because the gain should be small. Unfortunately, if cofactor is equal to or divisible by 2^m, then combined attack using small subgroups attack and side-channel attack may give knowledge about m least significant bits of private key k and therefore if there are some other mistakes in implementation of point scalar multiplication, then characteristics taken from side-channel attacks may be very dangerous and adversary may be able even to guess the whole private key k.

References

[1] J. Fan, B. Gierlichs, and F. Vercauteren, “To infinity and beyond: Combined attack on ecc using points of low order,” in *Cryptographic Hardware and Embedded Systems – CHES 2011*, B. Preneel and T. Takagi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 143–159.

[2] D. Genkin, L. Valenta, and Y. Yarom, “May the fourth be with you: A microarchitectural side channel attack on several real-world applications of curve25519,” in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS ’17. New York, NY, USA: ACM, 2017, pp. 845–858. [Online]. Available: http://doi.acm.org/10.1145/3133956.3134029

[3] M. Peter, “Speeding the pollard and elliptic curve methods of factorization,” *Mathematics of Computation*, vol. 48, pp. 243–264, 1987.

[4] E. Harold, “A normal form for elliptic curves,” *Bulletin of the American Mathematical Society*, vol. 44, no. 3, pp. 393–422, April 2007.

[5] D. J. Bernstein and T. Lange, “Faster addition and doubling on elliptic curves,” in *Advances in Cryptology – ASIACRYPT 2007*, K. Kurosawa, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 29–50.

[6] D. J. Bernstein, P. Birkner, M. Joye, T. Lange, and C. Peters, “Twisted edwards curves,” in *Proceedings of the Cryptology in Africa 1st International Conference on Progress in Cryptology*, ser. AFRICACRYPT’08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 389–405. [Online]. Available: http://dl.acm.org/citation.cfm?id=1788634.1788672

[7] M. Joye, M. Tibouchi, and D. Vergnaud, “Huff’s model for elliptic curves,” in *Algorithmic Number Theory*, G. Hanrot, F. Morain, and E. Thomé, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 234–250.

[8] S. Pohlig and M. Hellman, “An improved algorithm for computing logarithms over and its cryptographic significance,” *IEEE Trans. Inf. Theor.*, vol. 24, no. 1, pp. 106–110, September 1978. [Online]. Available: https://doi.org/10.1109/TIT.1978.1055817

[9] I. Biehl, B. Meyer, and V. Müller, “Differential fault attacks on elliptic curve cryptosystems,” in *Advances in Cryptology — CRYPTO 2000*, M. Bellare, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 131–146.

[10] M. Ciet and M. Joye, “Elliptic curve cryptosystems in the presence of permanent and transient faults,” *Designs, Codes and Cryptography*, vol. 36, no. 1, pp. 33–43, Jul 2005. [Online]. Available: https://doi.org/10.1007/s10623-003-1160-8

[11] S. Neves and M. Tibouchi, “Degenerate curve attacks: extending invalid curve attacks to edwards curves and other models,” *IET Information Security*, vol. 12, no. 3, pp. 217–225, 2018. [Online]. Available: https://doi.org/10.1049/iet-ifs.2017.0075

[12] C. H. Lim and P. J. Lee, “A key recovery attack on discrete log-based schemes using a prime order subgroup.” Springer-Verlag, 1997, pp. 249–263.

[13] Z. R., “Methods for avoiding the small-subgroup attacks on the diffie-hellman key agreement method for s/mime,” *RFC 2785*, March 2000.

[14] D. J. Bernstein, “Curve25519: New diffie-hellman speed records,” in *Public Key Cryptography - PKC 2006*, M. Yung, Y. Dodis, A. Kiayias, and T. Malkin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 207–228.