TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 4
ISSN: 2054 - 7420

Binary Division Attack for Elliptic Curve Discrete Logarithm Problem

Boris S. Verkhovsky, Yuriy S. Polyakov
Department of Computer Science, New Jersey Institute of Technology, USA;
verb73@gmail.com, polyakov@njit.edu

ABSTRACT

Elliptic curve cryptography (ECC) is an approach to public key cryptography (PKC) that is based on algebraic operations with elliptic curves defined over finite fields. Security of elliptic curve cryptography is based on the hardness of the elliptic curve discrete logarithm problem (ECDLP). Although there is no theoretical proof that ECDLP is intractable, no general-purpose sub-exponential running time algorithm has been found for solving the ECDLP if the elliptic curve parameters are chosen properly. In this study, we develop a new security attack based on the binary division of elliptic curve points over prime fields that may be used to solve the ECDLP when the order q of elliptic curve satisfies the congruence $q = 2 \pmod 4$. To perform the binary division, we devise a novel algorithm of point halving on elliptic curves defined over prime fields that applies to the cases when $q = 1 \pmod 2$ and $q = 2 \pmod 4$. The binary division attack has exponential worst-case asymptotic time complexity but in certain practical cases can be used to solve the ECDLP in a relatively efficient way. We therefore make a recommendation to avoid the case of $q = 2 \pmod 4$ in elliptic curve cryptosystems.

Keywords: Elliptic Curve Cryptography, Discrete Logarithm Problem, Security Attack, Point Halving, Cryptoanalysis, Public-Key Cryptography.

1 Introduction

Elliptic curve cryptography (ECC) is an approach to public key cryptography (PKC) that is based on the algebra of elliptic curves defined over finite fields. ECC is more efficient than RSA and discrete logarithm (DL) systems: smaller keys in ECC can be used to achieve the same security level as in RSA and DL systems [1]. The ECC algorithms substantially outperform both RSA and DL systems when carrying out private-key operations, such as digital signature generation and decryption. The benefits of ECC are most pronounced when processing power, storage, bandwidth, or power consumption is constrained.

An elliptic curve E over a field K is defined by a Weierstrass equation [1]

DOI: 10.14738/tnc.24.293
Publication Date: 4th August 2014
URL: http://dx.doi.org/10.14738/tnc.24.293

$$E: y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \ , \tag{1}$$

where constants $a_1, a_2, a_3, a_4, a_6 \in K$ and discriminant $\Delta \neq 0$.

If the characteristic of *K* is not equal to 2 or 3, then Equation (1) can be simplified to

$$E: y^2 = x^3 + ax + b \ . \tag{2}$$

where coefficients $a, b \in K$. Here, the transformed coordinates *x* and *y* are obtained using the admissible change of variables

$$(\mathrm{x}, y) \to \left(\frac{x - 3a_1^2 - 12a_2}{36}, \frac{y - 3a_1x}{216} - \frac{a_1^3 + 4a_1a_2 - 12a_3}{24} \right). \tag{3}$$

The discriminant of this curve is evaluated as $\Delta = -16\left(4a^3 + 27b^2\right)$.

The security of all ECC schemes is based on the hardness of the elliptic curve discrete logarithm problem (ECDLP) formulated as follows [1]: given an elliptic curve *E* defined over a finite field $\mathbb{F}_u$, a point $P \in E\left(\mathbb{F}_u\right)$ of order *n*, and a point $Q \in \langle P \rangle$, where $\langle P \rangle$ is the subgroup of *E* generated by *P*, find the integer $t \in [0, n-1]$ such that

$$Q = tP \ . \tag{4}$$

The integer *t* is called the discrete logarithm *Q* to the base of *P*, denoted $t = \log_P Q$.

The main attacks on the ECDLP include Pohlig-Hellman algorithm [1], Pollard's rho attacks and its modifications [1-5], and several isomorphism attacks that attempt to efficiently reduce ECDLP to the discrete logarithm problem (DLP) known to have sub-exponential algorithms [1]. The most efficient general-purpose attack on the ECDLP is a combination of the Pohlig-Hellman algorithm and Pollard's rho algorithm (or its modifications), which has a fully-exponential running time of $O\left(\sqrt{p}\right)$, where *p* is the largest prime divisor of *n*. Sub-exponential algorithms devised using isomorphism attacks are available only for special cases [1].

The special-purpose attacks, particularly those associated with polynomial-time and subexponential-time running times, are examined to devise countermeasures for verifying that a given elliptic curve is immune to these attacks. Currently, the following cryptographically weak special cases and corresponding countermeasures are known. (1) The Pohlig-Hellman algorithm reduces the computation of $t = \log_P Q$ to the computation of discrete logarithms in the prime order subgroups of $\langle P \rangle$ [1]. This implies the elliptic curve parameters should be selected to yield the order *n* of *P* that is divisible by a large prime. (2) For prime-anomalous elliptic curves ($\#E\left(\mathbb{F}_u\right) = p$) , ECDLP can be transformed to an equivalent DLP as part of the Araki-Satoh-Semaev-Smart attack [1]. It is simple to circumvent this attack by verifying that

$\#E(\mathbb{F}_u) \neq p$. (3) In the case when gcd(*n*,*u*) = 1, the Weil and Tate pairing attacks find an isomorphism between $\langle P \rangle$ and a subgroup of order *n* of the multiplicative group $\mathbb{F}_{u^k}^*$ of some extension field $\mathbb{F}_{u^k}$ [1]. To ensure that an elliptic curve *E* defined over $\mathbb{F}_u$ is not susceptible to the Weil and Tate attacks, it is sufficient to check that *n*, the order of the base point $P \in E(\mathbb{F}_u)$, does not divide $u^K - 1$ for all small *k* for which the DLP in $\mathbb{F}_{u^k}^*$ is tractable (for $n > 2^{160}$, the verification interval of *k* is [1,20]). (4) To protect against the Weil descent attack specifically designed for binary fields [1], it is suggested to avoid the use of elliptic curves $\mathbb{F}_{2^m}$, where *m* is composite.

The two kinds of elliptic curves recommended by the National Institute of Standards and Technology (NIST) for cryptographic protocols are elliptic curves over binary fields $\mathbb{F}_{2^m}$ (with the characteristic of 2) and elliptic curves over prime fields $\mathbb{F}_p$ (with the characteristic of *p*) [1, 6]. Ten specific elliptic curves over $\mathbb{F}_{2^m}$ and five elliptic curves $\mathbb{F}_p$ are recommended in the FIPS 186-2 standard for U.S. federal government use [6].

In this paper, we devise a "binary division" attack for elliptic curves defined over prime fields $\mathbb{F}_p$ that is based on the binary division of the integer $t = \log_P Q$. The binary division approach was previously examined for binary fields [7]. It was shown that this method has exponential complexity due to the fact that every point halving for the elliptic curves over binary fields yields two distinct points, which requires the consideration of two branches at each step where division is carried out.

2 Binary Division Algorithm

Consider an elliptic curve *E* defined over prime field $\mathbb{F}_p$ by Eq. (2). Let *P* be a generator point such that there is no point $A \in E(\mathbb{F}_p)$ that satisfies $P = 2A$. In other words, the point *P* is not divisible by 2. In this case, the sought integer *t* in Eq. (4) can be found using the following binary division algorithm:

Algorithm 1. Binary Division Algorithm for ECDLP

INPUT: $Q \in E(\mathbb{F}_p)$, $P \in E(\mathbb{F}_p)$, *P* is not divisible by 2

OUTPUT: $t_n t_{n-1} .. t_1 t_0$ (*t* in binary format)

1. Set $R \leftarrow Q$, $i \leftarrow 0$.
2. While ($R \neq O$ and $R \neq P$)
   - 2.1 If *R* is divisible by 2, then $t_i \leftarrow 0$.
   - 2.2 Else $t_i \leftarrow 1$, $R \leftarrow R - P$.
   - 2.3 $R \leftarrow R/2$.
   - 2.4 $i \leftarrow i + 1$.
3. If $R = P$ then $t_i \leftarrow 1$.

4. Else $t_i \leftarrow 0$.

Here, $t_i = 0$ denotes the case when point Q_i is divisible by 2, and $t_i = 1$ corresponds to the case when there is no such point $A \in E\left(\mathbb{F}_p\right)$ that satisfies $Q_i = 2A$.

To find the integer *t*, one needs to have both an efficient point divisibility criterion and an efficient point halving algorithm for elliptic curves over prime fields $\mathbb{F}_p$. There is an efficient point halving algorithm for binary fields [1, 5, 8, 9, 10] that is used to perform efficient scalar multiplication for elliptic curves over binary fields. However, there is no known point halving algorithm for elliptic curves defined over prime fields [11].

3 Point Halving Algorithm over Prime Fields

3.1 Formulation of the problem

The group law for elliptic curve *E* given by Eq. (2) over prime field $\mathbb{F}_p$ has the following rule for point doubling [1]:

Let $P = \left(x_1, y_1\right) \in E\left(\mathbb{F}_p\right)$, where $P \neq -P$. Then $2P = \left(x_2, y_2\right)$, where

$$x_2 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1,\ y_2 = \left(\frac{3x_1^2 + a}{2y_1}\right)\left(x_1 - x_2\right) - y_1. \tag{5}$$

To solve the inverse problem of finding a point $A \in E\left(\mathbb{F}_p\right)$ such that $P = 2A$, one needs to solve the system of nonlinear equations for x_1 and y_1 given the values of x_2 and y_2. To the best of our knowledge, there is no efficient general-purpose algorithm for solving system (5). The naïve approach of substituting every point $A \in \langle P \rangle$ into $\left(x_1, y_1\right)$ until a match is found (or no match if the system has no solution) requires the worst-case number of operations equal to the order of point *P*, which ultimately results in the exponential asymptotic complexity of same or higher order as the exhaustive search algorithm for ECDLP [1].

At the same time, efficient algorithms for some special cases can be devised. We separate the further discussion into the cases of odd and even elliptic curve orders.

3.2 Elliptic curve of odd order

Theorem 1: If the number *q* of points on elliptic curve $E\left(\mathbb{F}_p\right)$ given by Eq. (2) is odd, i.e., $\#E\left(\mathbb{F}_p\right) \bmod 2 = 1$, then for every point $P \in E\left(\mathbb{F}_p\right)$ there exists such a point $A \in E\left(\mathbb{F}_p\right)$ that

$$P = 2A \tag{6}$$

and

$$A=\left(\frac{q+1}{2}\right)P. \tag{7}$$

Proof: For every point $P\in E\left(\mathbb{F}_p\right)$

$$qP=O \tag{8}$$

because *q* is the order of elliptic curve *E*. Here, *O* is the point at infinity. We need to show that expression (7) implies (6). Indeed,

$$2A=\left(q+1\right)P=qP+P=O+P=P. \tag{9}$$

Corollary 1: Every point of $E\left(\mathbb{F}_p\right)$ with odd order *q* is divisible by two.

This suggests that the Binary Division Algorithm cannot solve ECDLP when the order of elliptic curve $E\left(\mathbb{F}_p\right)$ is odd (no generator point *P* that is indivisible by two can be selected), and thus any odd-order elliptic curve $E\left(\mathbb{F}_p\right)$ is immune to the Binary Division Attack developed in this study.

3.3 Elliptic curve of even order

3.3.1 Theorems and challenges

The order of $E\left(\mathbb{F}_p\right)$ can be even only if the curve contains at least one point with the *y*-coordinate of zero. Generally, for each *x*-coordinate such that $P=\left(x,y\right)\in E\left(\mathbb{F}_p\right)$, there is another point $Q=\left(x,p-y\right)\in E\left(\mathbb{F}_p\right)$, which follows from the square root operation performed when finding the value of *y*-coordinate for a given value of *x*-coordinate in Eq. (2). These two points coalesce into a single point when *y* = 0. Since the equation

$$x^3+ax+b\equiv 0\ \left(\mathrm{mod}\,p\right) \tag{10}$$

can practically have only 0, 1, or 3 roots (2 roots may occur only if the discriminant $\Delta=-16\left(4a^3+27b^2\right)=0$, which is not acceptable for ECC), the order of $E\left(\mathbb{F}_p\right)$ with at least one *y*-coordinate of 0 is the sum of 1 (for the O point) + the number of points with non-zero *y*-coordinate multiplied by 2 + 1 or 3 (depending on the number of roots to Eq. (10)). It is evident that this sum is always even.

Theorem 2: Let the number *q* of points on elliptic curve $E\left(\mathbb{F}_p\right)$ given by Eq. (2) be even, i.e., $\#E\left(\mathbb{F}_p\right)\mathrm{mod}2=0$. Let the points $P,A\in E\left(\mathbb{F}_p\right)$ such that $P=2A$ exist. If $q=2^m r$, where *r* is odd, $m\geq 0$, and

$$rP = O, \tag{11}$$

then

$$A = \left(\frac{r+1}{2}\right)P \ . \tag{12}$$

Proof: Indeed, Equations (11) and (12) imply that

$$2A = (r+1)P = rP + P = O + P = P \ . \tag{13}$$

It should be noted that Theorem 1 is a special case of Theorem 2 when $m = 0$.

Let us denote the points with the *y*-coordinate of 0 as Z_i, where integer $i \in \{1,3\}$. Let $\#(Z)$ denote the number of such points on a specific elliptic curve. The definition of point doubling given by Eq. (5) implies

$$2Z_i = O \ . \tag{14}$$

This suggests that expression (12) is not unique and the following values of A are also possible:

$$A = \left(\frac{r+1}{2}\right)P + V_i \ . \tag{15}$$

In order to apply the Binary Division Algorithm to the elliptic curve of even order, one also needs to find the divisibility criterion and determine if expressions (12) and (15) can be used to find the coordinates of point $A = P/2$ for all divisible points on even-order elliptic curves. This analysis is performed using numerical experiments.

3.3.2 Numerical experiments

Consider elliptic curve (2) defined over the prime field $\mathbb{F}_{23}$. According to Hasse's theorem [1],

$$p + 1 - 2\sqrt{p} \le \#E(\mathbb{F}_p) \le p + 1 + 2\sqrt{p} \ , \tag{16}$$

which implies that $15 \le \#E(\mathbb{F}_{23}) \le 33$. Our goal is to consider all even-order cases and both scenarios $\#(Z) = 1$ and $\#(Z) = 3$. A representative sample is listed in Table 1.

Table 1: Even-order cases of $E(F_{23})$

Curve #	*a*	*b*	#(*E*)	#(*Z*)
1	5	15	16	3
2	7	15	18	1
3	16	1	20	1
4	12	10	20	3
5	14	1	22	1
6	8	1	24	3
7	13	1	26	1
8	6	1	28	3
9	19	1	30	1
10	7	1	32	3

For each elliptic curve in Table 1, the following procedure was executed:

1. Count the order $\#E\left(\mathbb{F}_{23}\right)$ for each elliptic curve (including the at infinity point *O*).
2. Compute 2*A* for each point $A \in E\left(\mathbb{F}_{23}\right)$.
3. Compute the order of each point *A*, which is denoted as #(*A*).
4. Count $\#\left(Z\right)$.

To simplify the analysis, the following definitions are introduced:

- *Odd point*: An elliptic curve point that is not divisible by two;
- *Even point*: An elliptic curve point that is divisible by two.

The results of numerical experiments are listed in Tables 2 through 11. The even points are underlined. Only the points with $y<p/2$ are listed because points (x,y) and $(x, p-y)$ have the same order and divisibility property.

Table 2: Results for *E*: $y^2=x^3+5x+15$, #(*E*) = 16, #(*Z*) = 3

A	(5,2)	(6,10)	(7,5)	(12,3)	(13,0)	<u>(14,0)</u>	(18,7)	(19,0)	<u>(22,3)</u>
2A	(22,3)	(14,0)	(22,20)	(22,20)	O	<u>O</u>	(22,3)	O	(14,0)
#(*A*)	8	4	8	8	2	2	8	2	4

Table 3: Results for *E*: $y^2=x^3+7x+15$, #(*E*) = 18, #(*Z*) = 1

A	(1,0)	<u>(7,4)</u>	(8,10)	<u>(9,5)</u>	<u>(10,2)</u>	<u>(13,7)</u>	(18,4)	(20,6)	(21,4)
2A	<u>O</u>	(13,7)	(9,18)	(9,18)	(7,4)	(10,21)	(13,7)	(7,4)	(10,2)
#(*A*)	2	9	6	3	9	9	18	18	18

Table 4: Results for E: $y^2=x^3+16x+1$, #(E) = 20, #(Z) = 1

A	(0,1)	(1,8)	(2,8)	(9,0)	(11,6)	(12,9)	(14,5)	(16,11)	(18,7)	(20,8)
$2A$	(18,16)	(11,6)	(12,9)	O	(2,8)	(2,15)	(11,6)	(18,16)	(12,9)	(9,0)
#(A)	20	20	5	2	10	5	20	20	10	4

Table 5: Results for E: $y^2=x^3+12x+10$, #(E) = 20, #(Z) = 3

A	(1,0)	(3,2)	(7,0)	(10,7)	(11,1)	(14,1)	(15,0)	(18,3)	(19,6)	(20,4)	(21,1)
$2A$	O	(10,16)	O	(19,6)	(19,17)	(19,17)	O	(19,17)	(10,16)	(10,16)	(10,16)
#(A)	2	10	2	5	10	10	2	10	5	10	10

Table 6: Results for E: $y^2=x^3+14x+1$, #(E) = 22, #(Z) = 1

A	(0,1)	(1,4)	(3,1)	(4,11)	(5,9)	(6,5)	(8,2)	(17,0)	(18,6)	(20,1)	(22,3)
$2A$	(3,1)	(0,1)	(6,18)	(18,6)	(6,18)	(20,22)	(20,1)	O	(0,1)	(18,17)	(3,1)
#(A)	11	22	11	22	22	11	22	2	11	11	22

Table 7: Results for E: $y^2=x^3+8x+1$, #(E) = 24, #(Z) = 3

A	(0,1)	(2,5)	(3,11)	(6,9)	(7,3)	(8,5)	(10,0)	(12,10)	(13,5)	(15,0)	(16,4)	(17,6)	(21,0)
$2A$	(16,4)	(0,22)	(0,22)	(0,22)	(10,0)	(16,19)	O	(0,1)	(10,0)	O	(16,19)	(16,19)	O
#(A)	6	12	12	12	4	6	2	12	4	2	3	6	2

Table 8: Results for E: $y^2=x^3+13x+1$, #(E) = 26, #(Z) = 1

A	(0,1)	(2,9)	(4,5)	(10,2)	(11,7)	(14,11)	(15,11)	(16,2)	(17,11)	(18,8)	(19,0)	(20,2)	(21,6)
$2A$	(2,9)	(21,17)	(0,1)	(21,17)	(14,11)	(4,5)	(20,2)	(4,18)	(2,14)	(0,22)	O	(14,12)	(20,21)
#(A)	13	13	13	26	26	13	26	26	26	26	2	13	13

Table 9: Results for E: $y^2=x^3+6x+1$, #(E) = 28, #(Z) = 3

A	(0,1)	(1,10)	(3,0)	(5,8)	(6,0)	(7,8)	(8,3)	(9,5)	(10,7)	(11,8)	(14,0)	(15,4)	(17,5)	(20,5)	(21,2)
$2A$	(9,18)	(7,8)	O	(15,19)	O	(15,19)	(15,19)	(7,8)	(9,5)	(9,18)	O	(9,18)	(15,4)	(7,8)	(7,15)
#(A)	14	14	2	14	2	7	14	7	14	14	2	7	14	14	14

Table 10: Results for E: $y^2=x^3+19x+1$, #(E) = 30, #(Z) = 1

A	(0,1)	(2,1)	(3,4)	(4,7)	(6,3)	(9,2)	(10,8)	(11,0)	(12,5)	(15,2)	(16,10)	(17,4)	(20,3)	(21,1)	(22,2)
$2A$	(4,7)	(12,5)	(17,19)	(0,22)	(15,2)	(0,1)	(6,20)	O	(12,18)	(17,19)	(4,16)	(20,3)	(6,20)	(20,3)	(15,2)
#(A)	5	6	30	5	15	10	30	2	3	15	10	15	15	30	30

Table 11: Results for E: $y^2=x^3+7x+1$, #(E) = 32, #(Z) = 3

A	(0,1)	(1,3)	(2,0)	(3,7)	(4,1)	(5,0)	(6,11)	(7,5)	(10,6)	(11,11)	(13,9)	(15,10)	(16,0)	(18,5)	(19,1)	(21,5)	(22,4)
$2A$	(18,5)	(11,11)	O	(6,12)	(18,5)	O	(11,12)	(18,5)	(6,12)	(5,0)	(6,12)	(18,5)	O	(11,11)	(11,12)	(6,12)	(5,0)
#(A)	16	8	2	16	16	2	8	16	16	4	16	16	2	8	8	16	4

3.3.3 Observations

Tables 2-11 suggest that all even-order elliptic curves contain both odd and even points. This implies that the Binary Division Algorithm presented in Section 2 can generally be applied to any even-order elliptic curve.

Let $q = \#E(\mathbb{F}_{23})$. For all cases when $q \equiv 2 \pmod 4$, Equation (11) holds, which implies that expression (12) can be used to find the coordinates of point $A = P/2$ when $P = 2A$ exists. On the other hand, the tables corresponding to $q \equiv 0 \pmod 4$ contain a number of points with the order that is even and not divisible by the odd number *r*, which does not allow one to use Theorem 2 in this case.

It should be noted that $q \equiv 2 \pmod 4$ is equivalent to $q = 2r$, where *r* is odd, suggesting that expression (12) can be transformed to

$$A = \left(\frac{q+2}{4}\right) P \ . \tag{17}$$

Next we need to determine when a certain point $P \in E(\mathbb{F}_p)$ is divisible by two. Tables 3, 6, 8, and 10, corresponding to $q \equiv 2 \pmod 4$, show that for even points the order is *r* or a divisor of *r*. On the other hand, all odd points have even orders. This implies that the divisibility criterion for the case of $q \equiv 2 \pmod 4$ can be expressed as

If $q \equiv 2 \pmod 4$ and $(q/2)P = O$, then *P* is divisible by *two*. (18)

Our further analysis of the results of experimental data for this case suggests that when $(q/2)P = O$ does not hold, expression $(q/2)P = Z$ is valid, where *Z* is a point with the *y*-coordinate of zero, which can be restated as

If $q \equiv 2 \pmod 4$ and $(q/2)P = Z$, then *P* is not divisible by *two*. (19)

Expressions (18)-(19) can be considered as the Euler criterion for the elliptic curves over prime fields that correspond to the case of $q \equiv 2 \pmod 4$.

When the number of points with the *y*-coordinate of 0 is one or higher, point halving no longer has a unique solution, as shown by Eqs. (14) and (15). Table 1 suggests that in the case of $q \equiv 2 \pmod 4$, there is only one such point, denoted for simplicity as *Z* (the index *i* is dropped). This means that every point halving operation yields exactly two points in this scenario.

As an example, consider point (12,5) in Table 10. The first point found by Eq. (17) is (12,18). The second point found with Eq. (15) is (2,1). When each of this points is doubled, the result is the same: (12,5).

Combining expressions (15), (17)-(19), we can formulate the following conjecture:

Conjecture 1: Let the number q of points on elliptic curve $E\left(\mathbb{F}_p\right)$ given by Eq. (2) satisfy the congruence $q \equiv 2 \pmod 4$. Let a point $P \in E\left(\mathbb{F}_p\right)$ be given. If $(q/2)P = O$, then P is divisible by two and the two points $A \in E\left(\mathbb{F}_p\right)$ satisfying $P = 2A$ can be computed as

$$A = \left(\frac{q+2}{4}\right)P \text{ and } A = \left(\frac{q+2}{4}\right)P + Z, \tag{20}$$

where Z is the point with the y-coordinate of zero. If $(q/2)P = Z$, then P is not divisible by two.

This conjecture is valid for all of the experiments we ran, but it needs to be either formally proven or numerically (for a large number of elliptic curves with various prime characteristics) verified.

Tables 2, 5, 7, 9, 11 suggest that for the case of $q \equiv 0 \pmod 4$ and $\#(Z) = 3$, the divisibility criterion can be formulated as follows:

Conjecture 2: Let $q \equiv 0 \pmod 4$ and $\#(Z) = 3$. If $(q/4)X = O$, then X is divisible by two; if $(q/4)X = Z_i$, where Z_i is a point with the y-coordinate of zero, then X is not divisible by two.

In this case, Equation (12) cannot be used because its necessary condition (11) is rarely satisfied.

4 Counting Points on Elliptic Curves

Section 3 implies that the binary division attack can be applied only to even-order elliptic curves. Moreover, a non-brute-force point halving algorithm is available only for the case when the even order q of elliptic curve satisfies the congruence $q \equiv 2 \pmod 4$. In view of the above, the binary attack has to incorporate an algorithm for counting the number of points on elliptic curves (2) defined over prime fields.

Main practical algorithms for counting the order of elliptic curves over prime fields include Baby Step Giant Step (BSGS), Mestre's algorithm (improved BSGS), Schoof's algorithm, and Schoof-Elikes-Atkin (SEA) method [12]. They are implemented in standard number-theory software packages, such as Pari-GP and Sage.

The BSGS and Mestre's algorithms have the asymptotic computational complexity of $O\left(\sqrt[4]{p}\right)$; the most efficient variant has the space complexity of $O\left(n^2\right)$, where $n = \log p$. Schoof's algorithm has the computational complexity of $O\left(n^5\right)$ and space complexity of $O\left(n^3\right)$ - it was the first deterministic polynomial-time algorithm for counting points on elliptic curves. The SEA algorithm is probabilistic and has the asymptotic running time of $O\left(n^4\right)$ and space complexity between $O\left(n^3 \log n\right)$ and $O\left(n^4\right)$.

5 Examples

Section 3 suggests that the Binary Division Algorithm may practically be used only in the case of $q \equiv 2 \pmod 4$. When $q \equiv 1 \pmod 2$, every point is divisible by two, and thus the necessary condition for a generator point does not hold. When $q \equiv 0 \pmod 4$, no non-brute force point halving algorithm is known.

According to Eq. (20), each step in the loop of the Binary Division Algorithm for $q \equiv 2 \pmod 4$ should be run for two different values of R. This branching for each bit may theoretically run for all n bits of integer t, resulting in the exponential complexity of 2^n. Let us consider several examples to determine if certain branches can be truncated at early stages.

Example 1: Consider the problem of finding t in $(16,10) = t(21,1)$ for elliptic curve $E(\mathbb{F}_{23})$: $y^2=x^3+19x+1$ (the answer is 21). The recursive application of Algorithm 1 is illustrated in Fig. 1 as a binary tree. The root node has (16,10) as the initial value for R. As R is not divisible by 2, the bit t_0 is set to 1, and the new value of R is set to $R-(12,1)=(12,18)$. Then the breadth-first traversal (BFT) algorithm is used to visit both child nodes of the root node: (12,5) and (2,22). Figure 1 shows that $A=\left(\frac{q+2}{4}\right)P$ (left node) always yields an even point (bit: 0) and $A=\left(\frac{q+2}{4}\right)P+Z$ (right node) always gives an odd point. This follows from the fact that $q=2r$, where r is odd. It should be noted that some points are repeated, for example, point (12,18). This observation can be used to ignore certain "irrelevant" branches: If point R has previously been visited (traversed), then the current point R should not be traversed. The comparison of the current point with any points already visited can be implemented using a dynamic hash table.

The solution to Example 1 is retrieved as a bit sequence in the reverse order. In this case, it is "10101", which is 21. The number of binary divisions (scalar multiplications given by Eq. (17)) needed to find the solution is 7.

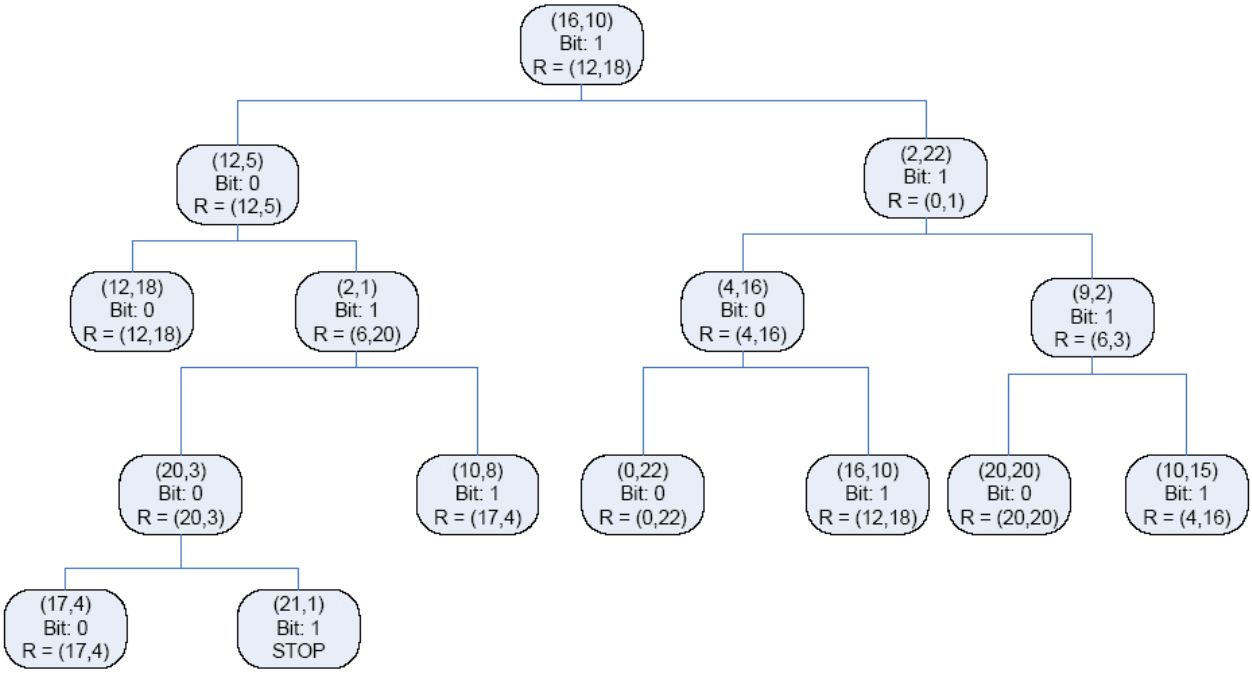

Figure 1: Tree Representation of the Solution for Example 1

Example 2: Consider the problem of finding t in $(10,15)=t(3,4)$ for elliptic curve $E(\mathbb{F}_{23})$: $y^2=x^3+19x+1$ (the answer is 19). In this case, the solution is "11001", which corresponds to 19. The number of binary divisions is 9.

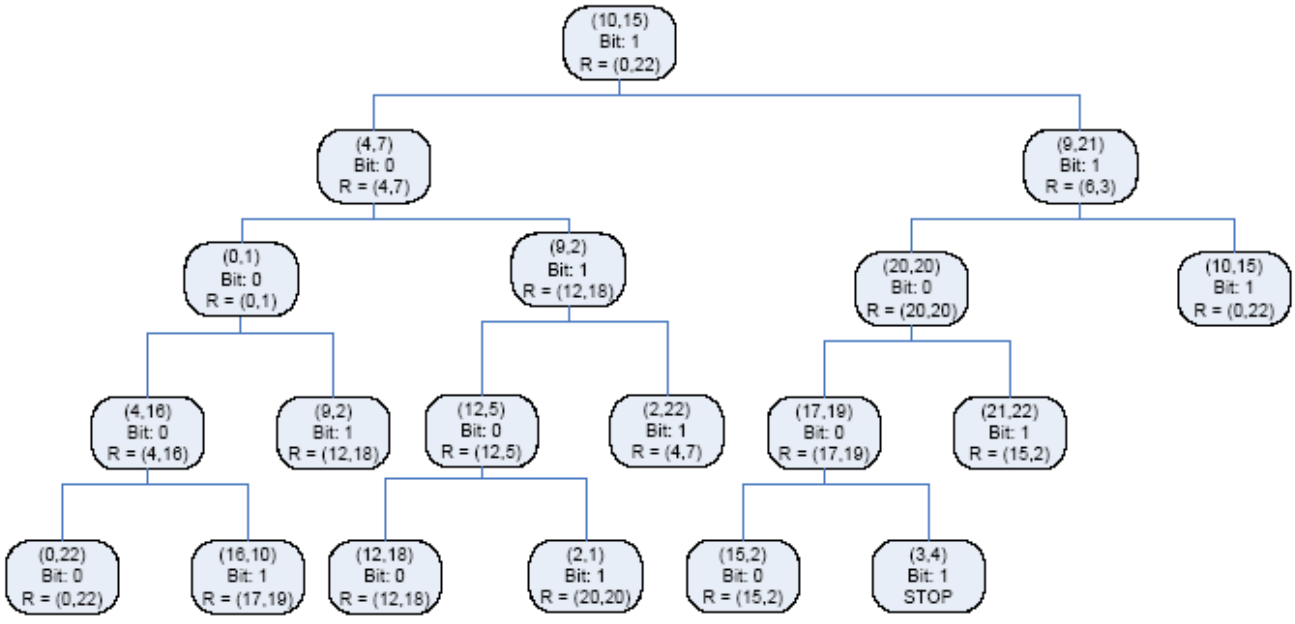

Figure 2: Tree Representation of the Solution for Example 2

Example 3: Consider the problem of finding t in $(11,16)=t(10,2)$ for elliptic curve $E(\mathbb{F}_{23})$: $y^2=x^3+13x+1$ (the answer is 17). In this case, the solution is "10001", which corresponds to 17. The number of binary divisions is 7.

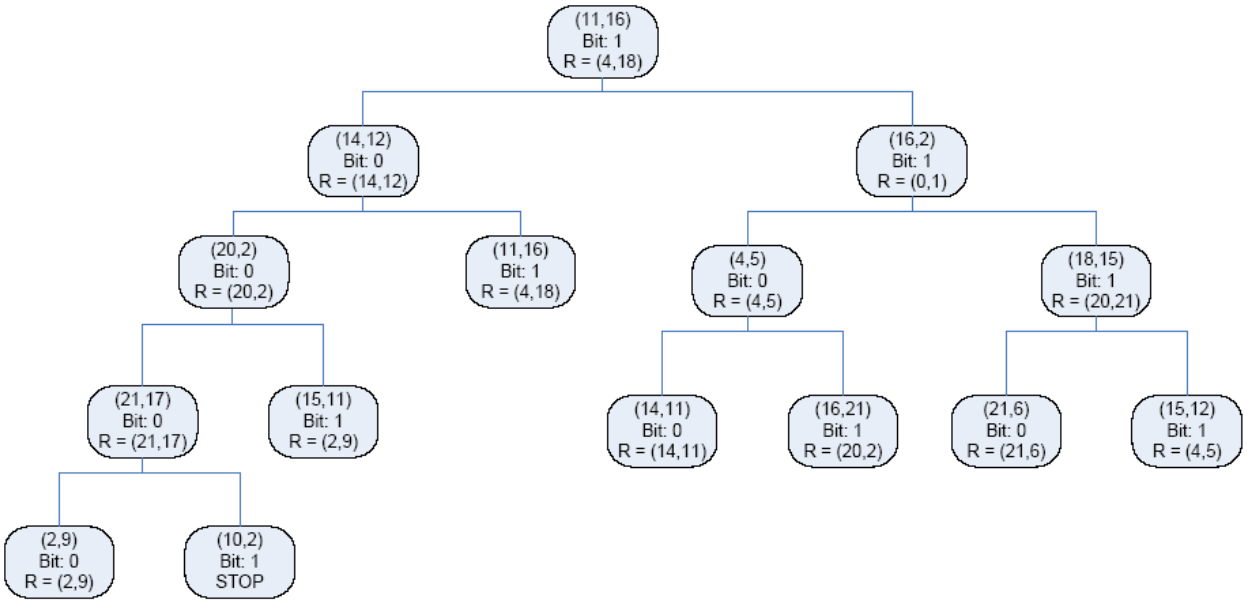

Figure 3: Tree Representation of the Solution for Example 3

6 Analysis

The most time-consuming operations in the binary tree implementation of Algorithm 1 for the case where $q \equiv 2 \pmod 4$ include (1) the counting of the number of points on elliptic curve, (2) the scalar multiplication involved in each point halving, and (3) the branching at each point halving leading to a binary tree traversal.

The first operation has to be executed only once and has a polynomial time complexity of $O(n^4)$, where $n=\log p$ (see Section 4 for details).

Equation (17) can be efficiently computed using one of the double-and-add methods, such as windowed, sliding-window, wNAF, or Montgomery ladder algorithm [1]. These algorithms generally require $O(k)$ iterations of point doubling and addition, where $k=\log q$.

The branching at each point results in at most $q/2$ binary divisions. Certain branches, as illustrated in the Examples, may be truncated at early stages. Still, the worst-case number of binary divisions is $O(q)=O(2^k)$, leading to the overall complexity of $O(k2^k)$. This implies that the binary division attack developed in this paper has exponential time complexity due to non-uniqueness of the point halving operation for elliptic curves defined over prime fields, which was also observed for the binary field case [7].

7 Conclusion

The binary division attack developed in this study can be used to solve the ECDLP when the order q of elliptic curve $E\left(\mathbb{F}_p\right)$ given by Eq. (2) satisfies the congruence $q \equiv 2 \pmod 4$. Although in the worst-case scenario the algorithm has an exponential asymptotic time complexity, in certain cases the number of visited branches in the binary tree representation of the algorithm (for example, see Fig. 1) may be relatively small making the solution of ECDLP practically feasible. Therefore, our recommendation is to avoid the case of $q \equiv 2 \pmod 4$ in practical ECC systems.

REFERENCES

[1]. D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*, New York: Springer, 2004.

[2]. R. Gallant, R. Lambert, and S. Vanstone, "Improving the parallelized Pollard lambda search on binary anomalous curves", *Math. Comput.*, vol. 69, pp. 1699–1705, 1999.

[3]. P. van Oorschot and M. Wiener, "Parallel collision search with cryptanalytic applications", *J. Cryptol.*, vol. 12, pp. 1–28, 1999.

[4]. M. Wiener and R. Zuccherato, "Faster attacks on elliptic curve cryptosystems", in *Selected Areas in Cryptography'98*, Berlin: Springer-Verlag, LNCS 1556, 1998, pp. 190–200.

[5]. F. Zhang and P. Wang, "Speeding up elliptic curve discrete logarithm computations with point halving", *Des. Codes Cryptogr.*, vol. 67, pp. 197–208, 2013.

[6]. NIST, *Digital Signature Standard*, FIPS Publication 186-2, February 2000.

[7]. A. V. Bessalov, "A method of solution of the problem of taking the discrete logarithm on an elliptic curve by division of points by two", *Cybern. Syst. Anal.*, vol. 37, no. 6, pp. 820–823, 2001.

[8]. E. Knudsen, "Elliptic scalar multiplication using point halving", in *Advances in Cryptology-ASIACRYPT'99*, Lecture Notes in Computer Science 1716, 1999, pp. 135–149.

[9]. R. Schroeppel, "Elliptic curve point halving wins big", in *2nd Midwest Arithmetical Geometry in Cryptography Workshop*, Urbana, 2000.

[10]. D. Hankerson, K. Karabina, and A. Menezes, "Analyzing the Galbraith-Lin-Scott point multiplication method for elliptic curves over binary fields", *IEEE Trans. Comput.*, vol. 58, no. 10, pp. 1411-1420, 2009.

[11]. K. Wong et al., "Fast elliptic scalar multiplication using new double-base chain and point halving", *Appl. Math. Comput,*, vol. 183, pp. 1000–1007, 2006.

[12]. R. Schoof, "Counting points on elliptic curves over finite fields", *Journal de Theorie des Nombres de Bordeaux*, vol. 7, pp. 219-254, 1995.