*International Journal of Research in Advent Technology, Vol.3, No.4, April 2015*
*E-ISSN: 2321-9637*

# Implementation of Fault Attacks on Elliptic Curve Cryptosystems

Anubhav Saxena[1], Varun Prakash Saxena[2], Sandip Mal[3]
*MTech (CSMC), Central University of Jharkhand, INDIA[1]*
*Assistant Professor (CS), GWECA, Ajmer, INDIA[2]*
*Assistant Professor (CSMC), Central University of Jharkhand, INDIA[3]*
*Email: anubhav.saxena@cuj.ac.in[1], varunsaxena82@gmail.com[2]* , sandip.mal1987@gmail.com[3]

**Abstract-** The main motivation behind Elliptic Curve Cryptography is to find a Public Key Family which provides the same level of security as Discrete Log Systems or RSA but with shorter operands. Through Fault Attacks, the adversary disturbs the computation of Cryptographic device to obtain information about Secret Key. This paper uses Elliptic Curve Point Multiplication Algorithm based on a binary sequence to build a Cryptographic algorithm and provides a way to retrieve the key by application of Fault Attacks on the Algorithm.

**Index Terms**- Elliptic Curve Cryptography, Elliptic Curve Point Multiplication, Fault Attacks, Euclidean Addition Chain.

## 1. INTRODUCTION

Elliptic Curve Cryptosystems were first introduced by Miller in 1986 and Koblitz in 1987 [1], [2].The basic idea behind finding such systems is to find another cyclic group in which the Discrete Logarithmic problem is difficult, Ideally more difficult than in $Z_p^*$ . Since Elliptic Curves are based on Discrete Log problem, so all the Discrete Log protocols such as Diffie-Hellman etc. can be realized through Elliptic Curves. Elliptic Curve Point Multiplication computes dP (=P+P+P----+P, d times) for a given point P on the curve and an arbitrary scalar d. Certain techniques are available in the literature to speed up the processing and efficiency of this protocol such as Montgomery Ladder Algorithm, Double and End Algorithm etc.In this paper we are using a technique proposed by Knuth to compute the point multiplication which uses"addition chain" [3], [4].
Cryptographic Systems build using these technique are vulnerable to fault attacks [6]. Fault Attacks were first introduced by Boneh, DeMillo and Lipton in 1996 which aims to recover the private key by inserting faults into the RSA Cryptosystem [5].In this paper we do two things:

- Firstly we mention a Cryptographic algorithm based on point Multiplication and addition chain technique (Section 3).
- We provide a way to recover the secret key by insertion of faults in the computation of the system (Section 4).

Through this paper our target is to demonstrate the application of fault attack on Cryptosystem build using Elliptic Curves.

## 2. ECC BACKGROUND

The Elliptic Curve over $Z_p$ ($p > 3$) is the set of all pairs $(x, y) \in Z_p$ which fulfills

$$y^2 \equiv x^3 + ax + b \bmod p \qquad (1)$$

together with an imaginary point at infinity $\Theta$, where a, b belongs to $Z_p$ and the condition

$$4a^3 + 27b^2 \neq 0 \bmod p \qquad (2)$$

The definition of Elliptic curve requires that the curve is non singular which means that the plot has no self-intersection or vertices [7]. Elliptic Curves had certain properties which are not required for cryptographic purpose, so curves which does not satisfies Eq. (2) are not chosen for cryptography. According to the Group properties there must exists a neutral element in the group which satisfies the following Group property.
$P + \Theta = P$ where P belongs to E; There is no such point on the curve which satisfies the group condition. So we define an abstract point at infinity as the neutral element.
$P+\Theta = P$ for every P belongs to E, According to the group def. we can also define the inverse -P of any group element P as $P + (-P) = \Theta$ for every P belongs to E.
Given two points on the curve $P=(x_1, y_1)$ and $Q= (x_2, y_2)$ the addition of these two points is a point R on the curve whose coordinates $x_3$ and $y_3$ can be calculated by following equations[7]:

$$x_3 = s^2 - (x_1 + x_2) \text{ , } y_3 = s(x_1 - x_3) - y_1 \qquad (3)$$

Where

$$s=\begin{cases}(y_2-y_1)/(x_2-x_1) \text{ if } P\neq Q\\ (3x_1^2+a)/(2y_1) \text{ if } P=Q\end{cases}$$

In case of point addition ($P \neq Q$), s is the slope of the line through P and Q. In case of point doubling ($P = Q$), s is the slope of the tangent through P. These equations are Analytical expressions used by electronic devices to implement group operations in them. The point Multiplication Equation is defined as:

$$P + P + P+P \text{ (d times)} = dP = T \qquad (4)$$

d is the private key which is an integer, while the public key T is a point on the curve[7].

## 3. RELATED WORK

In this section first we present the idea of Euclidean Addition chains [3] and later how they can be used to speed up the multiplication process [4].

An EAC solves the following problem: Let K be an integer ($K > 0$) given as input, starting from the integer 1 and at each step computing the sum of two previous results, What is the minimum no. of steps required to reack K ?

Example1. Let K=37, Choose a integer g (17) which is co prime to K according to the following rule: g should be close to K/Φ where Φ= (1 + √5=2) termed as golden steps [3]. take K and g and apply the subtractive form of Euclidean's Algorithm as shown below[3].

37 - 17 = 20
20- 17 = 3
17 - 3 = 14
14 - 3 = 11
11 - 3 = 8
8- 3 = 5
5- 3 = 2
3- 2 = 1
2 - 1 = 1
1 - 1 = 0

Reading the first integer of each row from bottom to top gives the EAC (1, 2, 3, 5, 8, 11, 14, 17, 20, 37).

Now we will get a binary sequence related to this chain. Consider step 3 which is 3- 2 = 1 can be rewritten as 3 = 1 + 2.If we use biggest of two integers of this step i.e. 3 and 2 in step 4 then step 4 is termed as Big step otherwise it is termed as Small step. For simplification we consider the EAC from step 4 onwards. For a Big step we write the binary value 1 and for the Small step the value is 0.

The binary sequence corresponding to EAC can be written as (1,2,3,5,8,11,14,17,20,37)

(1 1 0 0 0 0 1)

Let l be the length of the binary sequence and $c_i$ denotes the $i^{th}$ binary value. Now we describe the point multiplication algorithm which uses the binary sequence representing the EAC.

**Algorithm 1: Point Multiplication with EAC**

Input: P, 2P and the binary sequence

Output: Q = kP

1. $U_1 = 2P$, $U_2 = P$
2. for i = 4 to l do
3. $U_1 = U_1 + U_2$
4. if $c_i = 1$ then
5. $U_2 \leftarrow U_1$
6. Else
7. $U_2 \leftarrow U_2$
8. end if
9. end for
10. return $Q=U_1+U_2$

This algorithm computes the output with fewer computations as compared to the point multiplication Eq. (4).

## 4. OUR APPROACH

In this section we will describe how the binary sequence related to addition chain can be recovered through the computation of algo. (1). First the adversary executes the algo. (1) and observes the correct output which is shown in Table [1].

**TABLE 1**

| **Steps** | $\mathbf{c_i}$ | $\mathbf{U_1}$ | $\mathbf{U_2}$ |
|---|---|---|---|
| 1 | | 2P | P |
| 2 | 1 | 3P | 2P |
| 3 | 1 | 5P | 3P |
| 4 | 0 | 8P | 3P |
| 5 | 0 | 11P | 3P |
| 6 | 0 | 14P | 3P |
| 7 | 0 | 17P | 3P |
| 8 | 1 | 20P | 17P |
| | | Q=37P | |

Now the adversary keeps the last bit ($c_8$) as it is (corresponding to Table [1]) and flips the bit ($c_7$) which is shown in Table [2].Adversary observes the output.

| TABLE 2 | | | |
|---|---|---|---|
| **Steps** | **$c_i$** | **$U_1$** | **$U_2$** |
| 1 | | 2P | P |
| 2 | 1 | 3P | 2P |
| 3 | 1 | 5P | 3P |
| 4 | 0 | 8P | 3P |
| 5 | 0 | 11P | 3P |
| 6 | 0 | 14P | 3P |
| 7 | 1 | 17P | 14P |
| 8 | 1 | 31P | 17P |
| | | Q=48P | |

The adversary now sets the last bit ($c_8$) to 1(corresponding to Table [1]) and flips the second last bit ($c_7$).Adversary observes this output.

| TABLE 3 | | | |
|---|---|---|---|
| **Steps** | **$c_i$** | **$U_1$** | **$U_2$** |
| 1 | | 2P | P |
| 2 | 1 | 3P | 2P |
| 3 | 1 | 5P | 3P |
| 4 | 0 | 8P | 3P |
| 5 | 0 | 11P | 3P |
| 6 | 0 | 14P | 3P |
| 7 | 1 | 17P | 14P |
| 8 | 1 | 31P | 17P |
| | | Q=48P | |

Now he compares. If the result of Table [2] and Table [3] is same the last bit is 1 otherwise it is 0.
Now we recover the second last bit ($c_7$) in Table [1]. For this calculation we will use step 7 and step 6 of Table [1] computing the original output. Now the adversary keeps the last bit ($c_7$) as it is (corresponding to Table [1]) and flips the bit ($c_6$) which is shown in Table [4].

| TABLE 4 | | | |
|---|---|---|---|
| **Steps** | **$c_i$** | **$U_1$** | **$U_2$** |
| 1 | | 2P | P |
| 2 | 1 | 3P | 2P |
| 3 | 1 | 5P | 3P |
| 4 | 0 | 8P | 3P |
| 5 | 0 | 11P | 3P |
| 6 | 1 | 14P | 11P |
| 7 | 0 | 25P | 11P |
| 8 | 1 | 36P | 25P |
| | | Q=61P | |

The adversary now sets the bit ($c_7$) to 1(corresponding to Table [1]) and flips the bit ($c_6$).

| TABLE 5 | | | |
|---|---|---|---|
| **Steps** | **$c_i$** | **$U_1$** | **$U_2$** |
| 1 | | 2P | P |
| 2 | 1 | 3P | 2P |
| 3 | 1 | 5P | 3P |
| 4 | 0 | 8P | 3P |
| 5 | 0 | 11P | 3P |
| 6 | 1 | 14P | 11P |
| 7 | 1 | 25P | 14P |
| 8 | 1 | 39P | 25P |
| | | Q=64P | |

If the result of Table [4] and Table [5] is same then second last bit ($c_7$) is 1 otherwise it is 0. So it is clear that to recover any $c_i$ bit we have to use rows corresponding to $c_i$ and $c_{i-1}$ bit. In one table we flip the $c_{i-1}$ bit and observe the output, into another table we set the $c_i$ bit as 1 and flip the $c_{i-1}$ bit. Then we compare the results of both the tables, if the results are equal the bit is 1 and if not it is zero. This comparison logic is derived from the fact that we have only four cases if we compare two rows of table 1(i.e. 00,01,10,11) and we can find a relation between them and the correct output of Table [1],if we apply sufficient no. of test cases. This approach can be applied recursively to recover all the bits except the first bit. There are two possibilities for first bit either 1 or 0, the adversary with these two possibilities and the known cipher text can easily recover the remaining bit. For this attack to be successful the adversary must be able to inject faults into the execution of algo. (1) in a reverse order.

## 5. CONCLUSION

In this paper we had used Point Multiplication algorithm with EAC to compute private key efficiently. This type of approach is used by small cryptographic devices to secure data. Later we had provided a way to recover the scalar k and private key which will break the entire Cryptosystem.For effective implementation of point multiplication algorithm, techniques are required which must be safe against such type of Fault Attacks. Overall this paper is about Implementation of Fault Attacks on Cryptosystems.

## ACKNOWLEDGEMENT

First of all I would like to acknowledge to my External Guide Varun Prakash Saxena for helping me in deciding the topic, giving freedom to explore the literature and providing motivation and support throughout the learning process of this work. Secondly I would acknowledge to my Internal Guide Sandip

Mal for providing me all the resources needed for this research work and providing valuable inputs time to time. Without the help and efforts of these two people the work won't be a successful endeavor.

**REFERENCES**

[1] N.Koblitz,"Elliptic curve cryptosystems", Mathematics of Computation, vol. 48, no.177, pp.203-209, 1987.
[2] V.S.Miller,"Use of elliptic curves in cryptography", CRYPTO 1985, pp, 417-428, 1986.
[3] Andrew Byrne, Francis Crowe, William Marnane, Nicolas Meloni, Arnaud Tisserand, et al.. SPA Resistant Elliptic Curve Cryptosystem Using Addition Chains. International Journal of High Performance Systems Architecture, 2007, 1 (2), pp.133-142. , 10.1504/IJH- PSA.2007.015399. lirmm-00176433.
[4] D.E. Knuth, The art of computer programming. Vol. 2: Seminumerical algorithms, Addison-Wesley, Reading, Massachusetts, second edition, 1981.
[5] D.Boneh, R.A. DeMillo and R.J.Lipton,"On the importance of checking cryptographic protocols for faults", Lecture Notes in computer Sci-ence, vol. 1233, pp.37-51, 1997.
[6] S. Pontarelli, G.C. Cardarilli, M. Re, and A. Salsano. "Error detection in addition chain based ecc point multiplication", IEEE International On-Line Testing Symposium, pp. 192–194, 2009.
[7] C. Paar, J. Pelzl, Understanding Cryptography, 239 DOI 10.1007/978-3-642-04101-3 9, c Springer-Verlag Berlin Heidelberg 2010.
[8] Annual Workshop on Elliptic Curve Cryptography ECC. http://cacr.math. uwaterloo.ca/conferences.
[9] Cryptool — Educational Tool for Cryptography and Cryptanalysis. https://www.cryptool.org/.