ISSN 2347 - 3983

Volume 8. No. 7, July 2020

International Journal of Emerging Trends in Engineering Research

Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter 11872020.pdf

https://doi.org/10.30534/ijeter/2020/11872020

A Survey on Efficient and Secure Implementation of ECDSA against Fault Attack

J Kaushalya[1], R.Vijay Sai[2]

[1]PG Scholar, School of Computing, SASTRA Deemed to be University,India,gowsalyajayaraman29@gmail.com

[2]Assistant Professor, School of Computing, SASTRA Deemed to be University, India, vijaysai@it.sastra.edu

ABSTRACT

Public key cryptography provides fundamental protection components in contemporary cryptosystems, packages and protocols guaranteeing privacy, credibility and non-reputability of electronic correspondences and information stockpiling. Elliptic Curve Cryptography (ECC) gives high security and preferable execution over other public key techniques and these algorithms strengthen against various attacks. This paper presents a precise and complete review of an update of the Elliptic Curve Digital Signature (ECDSA) algorithm and fault attack and its countermeasures and describes about the future work to be done.

Key words: Digital signature, Elliptic curve cryptography, fault attack, point multiplication, protection.

1. INTRODUCTION

In cryptography systems, these curves are considered as a superior decision than RSA on such stages, since it gives identical security at very smaller key size. Smaller key size deduces lower equipment costs and lesser force utilization [1]. Nowadays, ECC related researches have attractive attention as they are used in many areas such as electronic commerce, mobile context including cellular phones and the Internet of Things. In order to upgrade the security of IoT ECC has been used in this paper [73].

Due to their effectiveness, it has been used in the constrained environment sources such as radio frequency identification tags, wireless sensor networks, smart cards etc. are always been challenging, since they provide security properties[2].ECDSA offers better performance in these Rivest Shamir Adleman, traditional digital signature algorithm over other cryptographic algorithms. Likewise, many approaches have been done for several other things such as to improve efficiency, to reduce cost, energy, memory, power and to check the capabilities of processors [3], [4].

In ECC, point multiplication operation plays a major role in encryption and decryption part and it is a time consuming module also. In ECDSA, point multiplication is used to generate and verify the signature by introducing various changes in point representation (affine, projective, jacobian-coordinates), by elliptic curve model changing, and by using different point multiplication methods (such as double and add method, Non-adjacent form (NAF), comb, window etc. Many researchers have done work related to countermeasures against various attack. But selecting a proper countermeasure is very important without providing any tradeoff between performance and security. It is very imperative to use prime or binary fields from an approved organization. This will lead to a secure implementation of ECDSA.

2. PRELUDE OF ECDSA

In this area, the fundamental ideas of ECC and ECDSA are clarified.

2.1 ECC

To give confidentiality in the communication network, ECC has been utilized to encrypt data and it relies upon the discrete logarithmic problem (DLP) because it's very difficult to attack and to get an integer k from the points on the curve P and Q [5]. ECC provides encryption, signal generation and key exchange [6].

Two finite fields have been used by ECC, they are prime and binary fields. Binary fields are used mostly in hardware part [7].In this field type, let Fq be the field, if p is greater than 3 then q is equal to p, then it uses prime field else if p is equal to 2, then it uses binary field [8].

By Weierstrass equation, an elliptic curve E over a field F is characterized as:

$$E: y_2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \qquad (1)$$

$$a_{1,} \ldots\ldots\ldots a_6 \in F.$$

For a binary field, (1) can be streamlined as:

$$E_b: y^2 + xy = x^3 + ax^2 + b \qquad (2)$$

For a prime field, (1) can be streamlined as:

$$y^2 = x^3 + ax + b \qquad (3)$$

In this curve, to add 2 points they use chord and tangent law. If 2 points are P and Q, then their point co-ordinates are (x_1,y_1) and (x_2,y_2) , then 2 points sum produces another point $R(x_3,y_3)$. In these curve, they use 2 operations point addition (P+Q-) and point doubling (P+P) for scalar multiplication.

If there should be an occurrence of the point addition where P and Q ∈ E (Fp): By using the slope $\lambda = y_2 - y_1/x_2 - x_1$

$$x_3 = \lambda^2 - x_1 - x_2,\ y_3 = \lambda^2(x_1 - x_3) - y_1 \qquad (4)$$

In there should be an occurrence of the point doubling, where P ∈E (Fp): by using the slope $\lambda = 3x_1^2+a / 2y_1$

$$x_3 = \lambda^2-2x_1,\ y_3 = \lambda^2(x_1-x_3) -y_1 \tag{5}$$

When binary field is used, on the basis of streamlined equation, the point addition where P and Q ∈ E uses the slope

$$\lambda = (y_1+y_2)/(x_1+x_2),\ x_3 = \lambda^2 +\lambda+x_1 +x_2 +a,$$

$$y_3 = \lambda(x_1 +x_3) + x_3 +y_1 \tag{6}$$

For point doubling, where P ∈ E: $x_3 = x_1^2 + b /x_1^2$,

$$y_3 = x_1^2 + ((x_1 + y_1)/(x))\ x_3 + x_3 \tag{7}$$

These are the basics of operations about the fields, which we are using in the elliptic curves explained in [9] and Figure.1 shows arithmetic operations in ECDSA. In [10], they explained all operations of ECC.

2.2 ECDSA

For digital signature and verification many domain parameters are used in ECDSA. They are,

q, the field size and a, b parameters for elliptic curve

G = ($x_{G,}$ y_G), known as base point

N, the base point G order

Generally, the curve equation are written as,

$$y^2=x^3+ax+b \bmod q \tag{8}$$

For the prime fields of P-256, a = q-3 so the equation is written as the one given in FIPS 184-4 [11],

$$y^2 = x^3-3x+b \bmod q \tag{9}$$

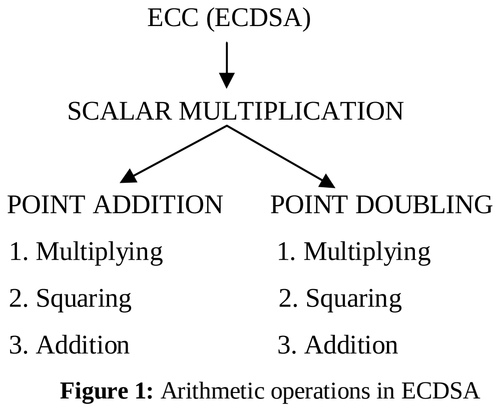

Figure 1: Arithmetic operations in ECDSA

A. ECDSA signature generation

For signal generation, some inputs such as curve domain parameters, a message M, hash function private key d are needed. Then, the output is (r, s) both are integers and their interval is [1, (order of G) -1].

1) An integer k is selected which is the per message secret number, $-1 \leq k \leq 1$

2) Generate (k, k^{-1}) and k^{-1} is it's modulo n.

3) Evaluate R= kG = (x_R , y_R)

4) Evaluate r = x_R mod n

5) Evaluate H = Hash (M)

6) Transform them in to an integer e

7) Evaluate s = (k^{-1} *(e + d*r)) mod n

8) Signature of m is(r, s).

B. ECDSA verification

For signal verification, some inputs such as received message M', received signature (r', s'), curve domain parameters, hash function and public key Q are needed. Mainly, output is to check whether the signature is valid or not.

1) Confirm that r', s' are in the intervals [1, n-1].

2) Evaluate hash of the received message H' and then convert it into integer e'.

3) Evaluate w = $(s')^{-1}$ mod n

4) Evaluate u1=(e'*w)mod n and u2= (r' *w) mod n

5) Evaluate the elliptic curve point R= (x_R,y_R)=u_1G+u_2Q

6) Evaluate v = x_R mod n

7) Differentiate v and r', if both are equal, it is valid else invalid.

C. Implementation of ECDSA

ECDSA algorithm is suitable in many environments such as WSN, smart cards etc. because of their better performance and security rather than RSA and DSA**.** In the sub-section, we will explain the implementation in different fields

1) Smart cards

Nowadays, technology uses more algorithms such as ECDSA, RSA, DSA etc. to provide authentication. ECDSA implementation and design is used most widely in this environment.

Jin-Hee Han and Young-Jin Kim [6] used java cards by including a light-weight Java byte code mediator to smart card, operating system and downloading Java class records, which are changed over to a littler, exclusive arrangement. The first OS was utilized to download, oversee, and execute the application code and its information. At last in the result, it can be used by other wireless devices also. In this [12], join two algorithms (ECDSA and ECDH) and use in java cards, which require only simple request or information to transfer between them. In [1], it is concluded that same keys are used for many times, in order to improve the randomness in ECDSA. In [13], it is concluded that they provide efficient output on power, storage and energy etc. than RSA on smart cards.

2) Wireless Sensor Network (WSN)

To assemble information about a specific domain, it comprises gathering of hubs that discuss remotely with one another in different applications [14],[74]. They regularly have confined sources, for example, vitality and memory. Along these lines, a WSN need efficient calculations to decrease the unpredictability of the calculation so as to build the stretch of the system over a long period of time. To forestall attacks, it requires raised level of security. In WSN, many analysts examined about the ECDSA and informed that it is very helpful for WSN.

For instance, in [15] they found that the accepting expense is half the transmission cost. They dissected signature in ECDSA (160 bit) and RSA (1024) bit, signature generation rate of ECDSA is beneath a signature generation rate of RSA. From this analysis, they clarified that ECDSA requires low vitality cost than RSA. In WSN, it demonstrated the chance of utilizing RSA and ECDSA [16]. ECDSA is superior to RSA because it utilizes short keys (160-bit) which deduces vitality, memory and so on.

In [17] on WSN, it examined the vitality cost through the Kerberos key dispersion convention and ECDH-ECDSA key understanding convention. In [18], Micaz bits with the finite field 163-bit actualized ECDSA. By the collaboration of their nearby hubs, the signal verification was improved.

In [19], it has been received to upgrade security and protection in radio frequency identification. Creators brought up that trivial procedures carried out by Elliptic curve digital signature in information signing are notably effective in radio frequency identification. Likewise, in [20] to ensure about RFID development they utilized ECDSA with the Shamir plan. They mainly used this to decrease the expense of two point multiplication to one. In [21], they implemented to secure RFID on IoT applications.

3. ASPECTS OF ECC/ ECDSA

In this segment, we will concentrate on the current studies of ECC/ECDSA basically on their performance, efficiency, security, countermeasures in brief manner.

Driessen et al. [22] looked at various signature plans (ECDSA, XTR-DSA, and NTRUSing) as far as vitality utilization, space, execution and so on. Based on this they came to conclusion that NTRUSing is high quality in terms of period of execution and memory. But it tolerate from security shortcoming against attacks. Much research called attention to utilizing hardware quickening agent's prompts superior, but it offering malleability, in which decreasing circuits need to be utilized to get better the malleability.

In [23], they discussed about the attacks such as active and passive attacks and countermeasures in detail. They explained that we should select the countermeasures in perfect way without producing any tradeoff. A few overviews have contemplated open cryptography calculations as far as calculation of difficult issue DLP and cross sections in quantity and old style PCs [24]. From these ideas, they came to a conclusion that ECC gives an excessive protection than other cryptosystem. It represents points of interest in fast, less capacity, and littler keys size.

In [25], [26], SCA and FA are the two main physical attacks they focused. They explained some countermeasures to overcome such faults like SPA, DPA etc. In [27], the author focused on the different standard curves (such as ANSI X9.63, safe curves etc.), from these the most common one is safe curves with SECP256k1 through using ECDSA, which is the strongest curve standards.

4. IMPROVEMENTS ON ECDSA

This section discusses about different kinds of improvement in scalar multiplication and curve operations.

4.1 Improvements on scalar multiplication

This segment provides representation of scalar multiplication. It requires lot of time [28], [29] in ECDSA algorithm and Elliptic curve cryptography algorithms. It takes over 80% for working time in sensor gadgets [5], [30].It is a fixed one of point addition and doubling that produces kP [31], [32].

The efficient and quick execution of elliptic curve cryptography algorithm and its subordinates are expected to quicken PM usage [33]. In ECDSA algorithm signature verification requires 2 scalar increases (u1G + u2Q) on step 5 [34] that require activity of a multifaceted nature calculation. SM utilizes three activities: inversion, doubling and multiplication but they are costly for an ECC algorithm [35]. When the Affine co-ordinates are utilized, they use both point doubling and addition which devour 2 multiplications, 1 squaring and 1 division activity in field [34]. Because of enhancing SM, cost of the algorithm and their execution time are getting deduced and but it improves the efficiency of execution in ECC algorithm. The conventional technique in point multiplication utilizes base 2 in point multiplying [34]. In this strategy, point multiplying is actualized in all bits in k, while point addition is executed with bit equivalent one in k [35].

4.2Double base chain

To create doubling (2P) which is significantly increasing (3P) i.e. tripling is one of the techniques. They proposed this method to utilize bases 2 and 3 so that it can deduce the carrying out time of the PM. In [36] to improve significantly, point tripling is collapsed with the point multiplying to get two (P) + Q when 1 reversal is in excess of 6 multiplications. They found that two (P) + Q are quicker than P + (P+Q) but two (P) +Q requires extra rate. They utilized the possibility of [37] in expelling y_3 from conditions when registering two (P) + Q; the creators utilized this thought with three (P) + Q and evacuated y_4 when 1 reversing is extra than six multiplications to deduce the algorithm rate. From this, they understand that it increases the point multiplication efficiency in elliptic curve algorithms such as ECC, ECDSA, etc.

The condition for double base chain is

$$k = \sum_i s_i \, 2^{b_i} 3^{c_i} \quad (10)$$

Where s_i is ±1, and ($2^{b_i}3^{c_i}$) are whole number numbers and bi and ci diminished monotonically ($b_1 > b_2 \ldots b_m > 0$ and $c_1 > c_2 \ldots c_m > 0$).

4.3 Multi base representation (2, 3, 5)

Scalar multiplication enhancement is done by using a point quintupling (5P).Since, it uses three bases (2, 3, 5) it is called as step multi base representation. When multi base representation algorithms is compared with DBNS, it provides small terms, more unnecessary things and have more meagerness. The representation of 160-bit in bases representation (2, 3) requires cost 23 terms whereas fifteen terms in representation bases (2, 3, and 5). The equation of Step multi base representation is presented as follows:

$$k = \sum_i s_i 2^{b_i} 3^{c_i} 5^{d_i} \quad (11)$$

Where s_i is ±1, and ($2^{b_i}3^{c_i}5^{d_i}$) are integer numbers and b_i, c_i and d_i decreased monotonically ($b_1 > b_2 \ldots b_m > 0, c_1 > c_2 \ldots c_m > 0$ and $d_1 > d_2 \ldots d_m > 0$).

4.4 Multi base representation (2, 3, 7)

To quicken point multiplication, it relies on triple bases. This 3 bases (2, 3, and 7) is called as multi base number representation.

This representation is the advancement of past two representation Double base representation and Step multi base representation. In [38] suggested three base technique (binary, ternary and septenary) with addition and subtraction. This is used mainly to change over an integer to the three base (2, 3, 7), by searching the nearest whole number to a scalar. They called attention to that septupling (seven P) costs is under two equations (two (2P) + 3P and two (3P) + P). Additionally, the squaring cost was disregarded. Multi base number representation accompanying the condition:

$$k = \sum_i s_i 2^{b_i} 3^{c_i} 7^{d_i} \quad (12)$$

4.5 Different methods of curve operations

To improve the performance of PM algorithm, many approaches are introduced such as double and add method, Non-adjacent form (NAF), comb, window etc.

Algorithm 1: The double and add algorithm

```
Input:  Let P be the point on E (Fp); $k = (k_{l-1} \ldots k_0)^2$
Output: Q = kP
Q = P
 for i = 0 upto l – 2 do
Q = 2(Q)
 if k (i) = 1 then
Q = Q + P
return Q
```

A. Double and add method

It is a crucial method for PM and has been utilized for two activities: point addition and multiplying. This is same as like the square-and-double algorithm [5], and it relies scalar bit k, where when k = 0, it will execute point multiplying, then when k = 1, it will execute both activities in each circle. In this point multiplying is twice that of point addition.

B. Non Adjacent Form (NAF) Method

This method is speeder than double and add method. Because, they reduce the execution time in PM and the number of point addition to one-third. It doesn't take into consideration any two nonzero bits in scalar k [5], [35] and this prompts decreases hamming weight based on the quantity of zeros and length of the bits in a scalar. It is represented as:

$$\text{NAF}(k) = l \sum_{i=0}^{-1} k_i 2_i \quad (13)$$

where $k \in \{0, \pm 1\}$, in [39].

Algorithm 2: NAF algorithm

```
Input: k as a positive integer
Output: NAF (k)
i=0
while $k \leq 1$ do
if k is odd then
$k_i = 2 - ((k) \bmod (4))$
$k = k - k_i$
else $k_i = 0$
k = k / (2),
i = i +1.
return $(k_{i-1}, k_{-i-2}, \ldots, k_1, k_0)$.
```

C. Window method (WM)

In order to improve the execution time in D&A method here they use particular window size. The window method is equivalent to D&A method, when window size is 1. For fixed point multiplication, they utilized this method and for furthermore diminished point addition better than D&A [5], [28]. In Wang and Li [40], to improve execution for PM, it utilized NAF and window method. They observed that window method is extra efficient than NAF. Instead of multi-precision multiplication, they used hybrid multiplication to reduce the memory size.

Algorithm 3: Window algorithm

```
Input: let P be the point on the field E (Fp), Window width w,
$d = [l/w]$, $k = (k_{d-1}, \ldots, k_0)2^w$.
Output: Q = kP.
$P_0 = P$
Evaluate in advance: for i = 1 to $2^{w-1}$ do:
$P_i = P_{i-1} + P$
Q=0
for i = d-1 down to 0 do
$Q = 2^w(Q)$
$Q = Q + P_{kd}$
return Q
```

D. Montgomery method

During scalar multiplications, in this method it uses only one co-ordinate which saves both computation and storage in hardware. In this [37], they achieved the enhancement of PM cost from 3.8% to 8.5%, since they used left to right algorithm where they eliminate y co-ordinate. Dimitrov et al [41] used Eisentrager's idea to improve the PM efficiency. Therefore, it offers better performance than other method.

Algorithm 4: Montgomery ladder algorithm

```
Input: A P be a point on the field E and a positive integer k = $(k_{i-1}...k_0)2$.
Output: The point kP.
P2 = P and P1 = 2(P)
for i = l - 2 down to 0 do
if ki = 0 then
P2 = 2(P2)
P1 = P1+P2
else
P1 = P1+P2
P1 = 2(P1)
return P2
```

5. CATAGORIES OF ATTACK

In this section, we will learn different types of attack and the attack which is concentrated in this research ie. Fault attack and their counter measures. Basically, the attacks are classified into two types, physical attack and non-physical attack. They both have active and passive attacks.

5.1 Non-physical attack

A. Passive attack

During this attack, the attacker has to analyze the data about their sender and receiver IP address, TCP protocol, location, data size etc. and this will help them for authentication operations and so on [42]. Some of the most widely used passive attacks are Eavesdropping, key logger and snooping, tracking, guessing etc.

B. Active attacks

This attack occurs by inserting, forging, modifying, replacing the user message while transferring through the network [42]. In many research projects in [43],[47] to prevent many attacks such as moderation, making fun of, contradiction and cyber. ECDSA is a best security option. ECDSA provides a security for many attacks. The security requirements are authentication, non-repudiation and integrity.

5.2 Physical attacks

In this attack, they require some problems to attack, so as to retrieve the key. The problems to retrieving the key are Electromagnetic radiation, Power consumption and Errors.

A. Passive attacks

In this attack, we can retrieve the private key through some leakage of information or data. With that leakage, whole key is got, for this also, power consumption is used and electromagnetic radiation problem is studied. There are different ways to retrieve the key, they are Simple power analysis, Timing attacks, Template attacks and Differential power analysis.

The countermeasure to overcome is to eliminate the relationship between the data and the leakages and also the relationship between the fake and real data.

B. Active attacks

In this attack, some errors are used to retrieve the secret key k, called as fault attack. There are four main attacks and they are Algorithmic specific attacks, Differential fault analysis, Safe error based attacks and Tampering with program flow. The description of the attacks are as follows:

1) Algorithmic Specific Attack

In order to circumvent, the attacker can inject some problem in to the data and so that it will be easy for them to solve easily. This kind of injection may be different based on the algorithm**.**

For example, by choosing the wrong base point, the scalar multiplication makes to shift the original curve to be a weak curve, because of this the ECDLP can be performed easily [48]. Another one is by using wrong parameters to achieve their goal[48].

2) Differential Fault Analysis

In this, through taking the distinction between the right and inaccurate cipher text, the secret key can be retrieved. Based on amassing, a number of misguided plaintext-cipher textual content pairs and acquiring the collisions, it is feasible to make the most secret [49].

3) Safe Error Attacks

For this attack, there is no need of faulty output, due to the fact it solely makes use of to take a look at whether or not the error happens in the output or not. There are two types, Computational safe errors and Memory safe errors. C- Safe error utilizes the weak spot of the algorithm and the M-safe error utilizes the implementation [50], [51].

6. NATURE, TARGETS AND COUNTER MEASURES OF FAULT ATTACK

6.1 Physical Nature of fault attack

There are four basic natures of fault attacks**,**

A. Clock glitches

This can be applied if the device require external clock. Clock glitches is a deviation to the supplied clock signal because of this deviation, the processor execute the next instruction before the current instruction execution. For internal clock, this attack cannot be possible.

B. Under-powering and energy spikes

Interfering with the strength provided of the device is a very low value fault injection method. Another method is by inducing excessive versions in the power supply. This may be additionally motive to misinterpret or to skip the instruction.

C. Temperature attack

By varying the temperature too high or too low faults can be induced [52], [53], [54], but particular portion of data should be focused.

D. Electromagnetic fault injection

This may cause the chip to damage or change the memory content in that, or it may induce single bit change [55].

6.2 Fault targets

Generally, the processor may have many components such as I/O port, data path, and storage and control part. Based on those parts attacks are performed .Table 1 shows fault targets.

A. Input parameters attack

Depending on the device, its utilization can also be achievable to set off exponentially the faults with the aid of controlling the input parameters. Sometimes the attacker's selection of input may cause the implementation to fail; only some extent, it may be true. It is not a compulsion that the input must be sending from I/O port, it may also from a memory (ROM) [56], [57].

B. Data processing part attack

This attack carried out while computation transferring is carried out by means of bus or in the register or reminiscence [52], [58], [59].

C. Storage part attack

Errors in unstable storage exploit to alter the intermediate results, while computational error happens in the non-volatile memory, which will have an effect on the machine parameters fully.

Table 1: Fault Targets

Attacks	**Target**			
Components	**Input**	**Data path**	**Memory**	**Control**
Algorithm Specific	Yes	yes	Yes	yes
Control	No	no	Yes	yes
Safe error	No	yes	Yes	yes
DFA	No	yes	Yes	yes

D. Instruction processing part attack:

This attack mainly takes place on the application waft which primarily goals the manipulate phase of the gadget as an alternative of data path.

6.3 Countermeasures against fault attacks

As countermeasures include some significant pitfalls, they are picked to give a decent tradeoff among hardware and execution expenses and security level. Practically speaking, countermeasures are planned to make an attack adequately costly, not unthinkable, despite everything keeping an ensured plan and performing [60]. Hardware and design to be driven are two primary concepts for safety of cryptographic devices. In hardware, countermeasures are passive and active fields. Passive fields are metallic layer which may additionally cowl the chip from various attacks such as temperature attack, optical attacks etc. [61]. An active field consists of wire mesh which detects the interruption on wire. But the main drawback is its cost. In design to be driven, to construct this, two principles are used, 1) inducing redundancy to check whether the fault is occurred or not 2) layout implementation to be inherently inclined to the attacks. But it will be suitable for particular attack only. Since, it cannot use only one and solve all attack; it can use parallel protection mechanism. There are different abstraction levels to perform and achieve the protection. Figure.2 shows the abstraction levels. They are

A. Protocol level

In this, it can design something so that certain attacks may not be possible. For example fresh-rekeying i.e. new will be generated for every encryption [62].

B. Cryptographic primitive level

In this level, the end result of the process can be checked earlier than execution. Example in DSA implementation, end result is tested whether or not it is fault-free end result or now not [63]. Since same data path is used for encryption and decryption it is convenient to observe the fault.

C. Algorithmic level

Based on the algorithm, fault detection technique can be applied for instance in MPL, consistency test is higher way to observe these blunders efficaciously [64].

D. Arithmetic level

In this, it knows many operations are used and they are protective also. E.g. error detection codes [65]. Another protection way is fault detection, which plays a major role, because by detecting the error, it is able to correct. Figure 3 shows the countermeasures for FA.

E. Input parameters protection

If implementation checker is no longer used, the attacker can use some input to attack the device [48]. Even if the inputs are now not provided backyard and they are using non-volatile memory, they can also able to attack and reveal the secret key. So, the input parameters have to be checked before using for computation. Cyclic redundancy check is used to check the system parameter. But it requires more storage.

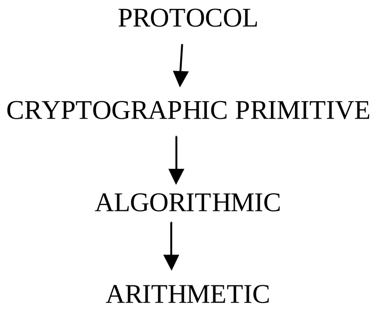

Figure 2: Abstraction levels

F. Processing part protection

While processing, if the fault is injected, it may cause leakage of sensitive message, it may change registers, program flow etc. so some instructions can miss. To prevent that,

1) Redundant computations:
Concurrent error detection can be used to prevent the fault. In this same operation may be computed twice, either in parallel or consequently [66]. Time redundancy based CED involves double encryption data and comparing the results. This reduces the overhead [67].

2) Checking algorithmic specific properties:
Checking whether the output of the scalar multiplication belongs to the curve or not, is the best fault detection method [68], [48]. Another one is checking the correspondence between the middle variables and if the output is exactly the base point, then this will be one of the fault detection method [69], [70].

3) Program flow protection:
When attacking the program flow, it may miss some instruction or execution problem may occur. To overcome that, by checking and calculating its fingerprint is done, using An + B codes [71], [72]

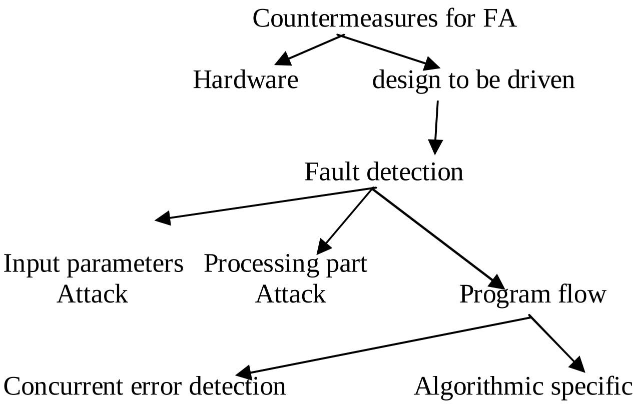

Figure 3: Countermeasures for FA

7. CONCLUSION AND FUTURE WORK

This present paper makes a precise learn about the safety and enhancements of the ECDSA algorithm. This learning will allow us to see the current enhancements of protection and special strategies of curve operations. This up to date enhancement can also perhaps encourage some of the strategies in the protection of the algorithm.

From this paper, we believe that it will be advisable for an analyst to hit upon search for thoughts procuring and enhancing the safety of the digital signature algorithm. In this paper, it is also narrated about some countermeasures of fault attacks. In existing works, they mostly used error correction techniques like coherent redundant error checker to check the fault and to prevent from key retrieving. In our proposed work, our work focus on both error detection and error correction in order to give superior protection and it mainly uses only point doubling and addition than other functions which reduces computation timing and complexity.

ACKNOWLEDGMENT

Thanks for Ms. A. Suganya, for helping us to focus on this domain, which helped us to learn a lot about ECC.

REFERENCES

1. J. W. Bos, J. A. Halderman, N. Heninger, J. Moore, M. Naehrig, and E. Wustrow. **Elliptic curve cryptography in practice, in International Conference on Financial Cryptography and Data Security**. Springer, 2014, pp. 157–175 https://doi.org/10.1007/978-3-662-45472-5_11
2. M. Aydos, T. Yanik, and C. Koc. **High-speed implementation of an ecc-based wireless authentication protocol on an arm microprocessor,** IEE Proceedings-Communications, vol. 148, no. 5, pp. 273–279, 2001.
3. H. Cohen, A. Miyaji, and T. Ono. **Efficient elliptic curve exponentiation using mixed coordinates**, in International Conference on the Theory and Application of Cryptology and Information Security. Springer, 1998, pp. 51–65.
4. M. Varchola, M. Drutarovsky, M. Repka, and P. Zajac. **Side channel attack on multiprecision multiplier used in protected ecdsa implementation**,in 2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig). IEEE, 2015, pp. 1–6.
5. M. B. O. Rafik and F. Mohammed. **The impact of ecc's scalar multiplication on wireless sensor networks**, in Programming and Systems (ISPS), 2013 11th International Symposium on. IEEE, 2013, pp. 17–23. https://doi.org/10.1109/ISPS.2013.6581488
6. J.-H. Han, Y.-J. Kim, S.-I. Jun, K.-I. Chung, and C.-H. Seo. **Implementation of ecc/ecdsa cryptography algorithms based on java card**,in Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on. IEEE, 2002, pp. 272–276.
7. G. Nabil, K. Naziha, F. Lamia, and K. Lotfi. **Hardware implementation of elliptic curve digital signature algorithm (ecdsa) on koblitz curves**,in Communication Systems, Networks & Digital Signal Processing (CSNDSP), 2012 8th International Symposium on. IEEE, 2012, pp. 1–6.
8. K. Eisentr¨ager, K. Lauter, and P. L. Montgomery.**Fast elliptic curve arithmetic and improved weil pairing evaluation,** in Cryptographers' Track at the RSA Conference. Springer, 2003, pp. 343–354
9. D. Johnson, A. Menezes, and S. Vanstone. **The elliptic curve digital signature algorithm (ecdsa)**,International Journal of Information Security, vol. 1, no. 1, pp. 36–63, 2001
10. H. Zhong, R. Zhao, J. Cui, X. Jiang, and J. Gao. **An improved ecdsa scheme for wireless sensor network,** International Journal of Future Generation Communication and Networking, vol. 9, no. 2, pp. 73–82, 2016.

11. NIST, **Digital Security Standard (DSS**) ,FIPS PUB 186-4 July2013.
12. L. Batina, J.-H. Hoepman, B. Jacobs, W. Mostowski, and P. Vullers. **Developing efficient blinded attribute certificates on smart cards via pairings**,in International Conference on Smart Card Research and Advanced Applications. Springer, 2010, pp. 209–222. https://doi.org/10.1007/978-3-642-12510-2_15
13. M. Savari and M. Montazerolzohour **All about encryption in smart card,** in Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on. IEEE, 2012, pp. 54–59
14. W. Osamy and A. M. Khedr. **An algorithm for enhancing coverage and network lifetime in cluster-based wireless sensor networks**,International Journal of Communication Networks and Information Security (IJCNIS), vol. 10, no. 1, 2018.
15. A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz. **Energy analysis of public-key cryptography for wireless sensor networks**, in Third IEEE international conference on pervasive computing and communications. IEEE, 2005, pp. 324–328
16. P. Trakadas, T. Zahariadis, H. Leligou, S. Voliotis, and K. Papadopoulos. **Analyzing energy and time overhead of security mechanisms in wireless sensor networks**, in 2008 15th International Conference on Systems, Signals and Image Processing. IEEE, 2008, pp. 137–140. https://doi.org/10.1109/IWSSIP.2008.4604386
17. G. De Meulenaer, F. Gosset, F.-X. Standaert, and O. Pereira. **On the energy cost of communication and cryptography in wireless sensor networks**,in 2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications. IEEE, 2008, pp. 580–585.
18. X. Fan and G. Gong,**Accelerating signature-based broadcast authentication for wireless sensor networks**,Ad Hoc Networks, vol. 10, no. 4, pp. 723–736, 2012.
19. M. I. Younis and M. H. Abdulkareem. **Itpmap: An improved three-pass mutual authentication protocol for secure rfid systems,**Wireless Personal Communications, vol. 96, no. 1, pp. 65–101, 2017
20. A. Ibrahim and G. Dalkılıc**. An advanced encryption standard powered mutual authentication protocol based on elliptic curve cryptography for rfid, proven on wisp**,Journal of Sensors, vol. 2017, 2017. https://doi.org/10.1155/2017/2367312
21. R. Li, T. Song, N. Capurso, J. Yu, J. Couture, and X.Cheng. **Iot applications on secure smart shopping system**, IEEE Internet of Things Journal, 2017.
22. B. Driessen, A. Poschmann, and C. Paar. **Comparison of innovative signature algorithms for wsns**,in Proceedings of the first ACM conference on Wireless network security. ACM, 2008, pp. 30–35.
23. J. Fan, X. Guo, E. De Mulder, P. Schaumont, B. Preneel, and I. Verbauwhede. **State-of-the-art of secure ecc implementations: a survey on known side-channel attacks and countermeasures**,in Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on. IEEE, 2010, pp. 76–87
24. A. S. Abdouli, J. Baek, and C. Y. Yeun. **Survey on computationally hard problems and their applications to cryptography,** in Internet Technology and Secured Transactions (ICITST), 2011 International Conference for. IEEE, 2011, pp. 46–52.
25. J. Fan and I. Verbauwhede. **An updated survey on secure ecc implementations: Attacks, countermeasures and cost,**in Cryptography and Security: From Theory to Applications. Springer, 2012, pp. 265–282. https://doi.org/10.1007/978-3-642-28368-0_18
26. J.-L. Danger, S. Guilley, P. Hoogvorst, C. Murdica, and D. Naccache. **A synthesis of side-channel attacks on elliptic curve cryptography in smart-cards**,Journal of Cryptographic Engineering, vol. 3, no. 4, pp. 241–265, 2013
27. **H. Mayer. Ecdsa security in bitcoin and ethereum: a research survey**,2016.
28. H. Li, R. Zhang, J. Yi, and H. Lv. **A novel algorithm for scalar multiplication in ecdsa**, in Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on. IEEE, 2013, pp. 943–946
29. J.-L. Danger, S. Guilley, P. Hoogvorst, C. Murdica, and D. Naccache. **A synthesis of side-channel attacks on elliptic curve cryptography in smart-cards**, Journal of Cryptographic Engineering, vol. 3, no. 4, pp. 241–265, 2013
30. M. Hutter, M. Feldhofer, and J. Wolkerstorfer. **A cryptographic processor for low-resource devices: Canning ecdsa and aes like sardines**,in IFIP International Workshop on Information Security Theory and Practices. Springer, 2011, pp. 144–159. https://doi.org/10.1007/978-3-642-21040-2_10
31. A. M. Ismail, M. R. M. Said, K. M. Atan, and I. S. Rakhimov. **An algorithm to enhance elliptic curves scalar multiplication combinigmbnr with point halving**,Applied Mathematical sciences, vol. 4, no. 1, pp. 259–1, 2010
32. F. Morain and J. Olivos. **Speeding up the computations on an elliptic curve using addition-subtraction chains**, Information Theory Applications, vol. 24, no. 6, pp. 531–543, 1990.
33. G. Purohit, A. S. Rawat, and M. Kumar**. Elliptic curve point multiplication using mbnr and point halving**,Int. J. Advanced Networking and Applications, vol. 3, no. 2, pp. 1329–1337, 2012.
34. A. M. Ismail, M. R. M. Said, K. M. Atan, and I. S. Rakhimov. **An algorithm to enhance elliptic curves scalar multiplication combinigmbnr with point halving**, Applied Mathematical sciences, vol. 4, no. 1, pp. 259–1, 2010
35. K.-W. Wong, E. C. Lee, L. Cheng, and X. Liao. **Fast elliptic scalar multiplication using new double base chain and point halving,** Applied mathematics and computation, vol. 183, no. 2, pp. 1000–1007, 2006.
36. M. Ciet, M. Joye, K. Lauter, and P. L. Montgomery. **Trading inversions for multiplications in elliptic curve cryptography**, Designs, codes and cryptography, vol. 39, no. 2, pp. 189–206, 2006.

37. K. Eisentr¨ager, K. Lauter, and P. L. Montgomery. **Fast elliptic curve arithmetic and improved weil pairing evaluation**,in Cryptographers' Track at the RSA Conference. Springer, 2003, pp. 343–354. https://doi.org/10.1007/3-540-36563-X_24
38. G. Purohit and A. S. Rawat. **Fast scalar multiplication in ecc using the multi base number system**, International Journal of Computer Science Issues, vol. 8, no. 1, pp. 131–137, 2011.
39. D. Hankerson, A. J. Menezes, and S. Vanstone. **Guide to elliptic curve cryptography,** Springer Science & Business Media, 2006.
40. H. Wang and Q. Li. **Efficient implementation of public key cryptosystems on mote sensors (short paper)**, in International Conference on Information and Communications Security. Springer, 2006, pp. 519–528
41. V. Dimitrov, L. Imbert, and P. K. Mishra. **Efficient and secure elliptic curve point multiplication using double-base chains**, in International Conference on the Theory and Application of Cryptology and Information Security. Springer, 2005, pp. 59–78.
42. A. Nadeem and M. P. Howarth. **A survey of manet intrusion detection & prevention approaches for network layer attacks**, IEEE communications surveys & tutorials, vol. 15, no. 4, pp. 2027–2045, 2013 83, 99, 100, 20, 101
43. W. Pan, F. Zheng, Y. Zhao, W.-T. Zhu, and J. Jing. **An efficient elliptic curve cryptography signature server with gpu acceleration**, IEEE Transactions on Information Forensics and Security, vol. 12, no. 1, pp. 111–122, 2017 https://doi.org/10.1109/TIFS.2016.2603974
44. M. Masud and M. S. Hossain. **Secure data-exchange protocol in a cloud-based collaborative health care environment**," Multimedia Tools and Applications, pp. 1–15, 2017.
45. A. T. Lo'ai, T. F. Somani, and H. Houssain. **Towards secure communications: Review of side channel attacks and countermeasures on ecc**,in Internet Technology and Secured Transactions (ICITST), 2016 11th International Conference for. IEEE, 2016, pp. 87–91.
46. M. Franekov´a, P. Holeˇcko, E. Buben´ıkov´a, and A. Kana´likov´a. **Transport scenarios analysis within c2c communications focusing on security aspects**,in Applied Machine Intelligence and Informatics (SAMI), 2017 IEEE 15th International Symposium on. IEEE, 2017, pp. 000461–000466.
47. C. Mann and D. Loebenberger. **Two-factor authentication for the bitcoin protocol**, International Journal of Information Security, vol. 16, no. 2, pp. 213–226, 2017.
48. M. Ciet and M. Joye. **Elliptic curve cryptosystems in the presence of permanent and transient faults**, Designs, Codes Cryptography, vol. 36, no. 1, pp. 33–43, 2005.
49. J. Blömer and V. Krummel. **Fault based collision attacks on AES**, in Fault Diagnosis and Tolerance in Cryptography (Lecture Notes in Computer Science), L. Breveglieri, I. Koren, D. Naccache, and J.-P. Seifert, Eds. Berlin, Germany: Springer-Verlag, 2006, pp. 106–120.
50. J. Fan, X. Guo, E. De Mulder, P. Schaumont, B. Preneel, and I. Verbauwhede. **State-of-the-art of secure ecc implementations: a survey on known side-channel attacks and countermeasures**,in Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on. IEEE, 2010, pp. 76–87. https://doi.org/10.1109/HST.2010.5513110
51. P.-A. Fouque, S. Guilley, C. Murdica, and D. Naccache, **Safe-errors on spa protected implementations with the atomicity technique**, in The New Codebreakers. Springer, 2016, pp. 479–493.
52. D. Boneh, R. A. DeMillo, and R. J. Lipton. **On the importance of checking cryptographic protocols for faults**," in Proc. 16th Annu. Int. Conf. Theory Appl. Cryptograph. Tech., 1997, pp. 37–51.
53. I. Peterson. **Chinks in digital armor: Exploiting faults to break smartcard cryptosystems**, Sci. News, vol. 151, no. 5, pp. 78–79, 1997.
54. S. Skorobogatov. **Low Temperature Data Remanence in Static RAM [Online]**. Available: http://www.cl.cam.ac.uk/techreports/ UCAM-CL-TR-536.pdf, Jun 2002.
55. J.-J. Quisquater and D. Samyde. **Eddy current for magnetic analysis with active sensor**, in Proc. Esmart, Nice, France, Sep. 2002, pp. 1–8.
56. E. Brier, B. Chevallier-Mames, M. Ciet, and C. Clavier. **Why one should also secure RSA public key elements**, in Proc. Int. Workshop Cryptograph. Hardware Embedded Syst., 2006, pp. 324–338.
57. M. Kara-Ivaniov, E. Iceland, and A. Kipnis. **Attacks on authentication and signature schemes involving corruption of public key (modulus),** in Proc. Workshop Fault Diagnosis Tolerance Cryptography, Aug. 2008, pp. 108–115.
58. J.-M. Schmidt and M. Medwed. **Fault attacks on the montgomery powering ladder,** in Proc. Int. Conf. Inf., Security Cryptol., 2011, pp. 396–406.
59. J. Carrijo, R. Tonicelli, and A. C. A. Nascimento. A fault analytic method against HB+, IEICE Trans., vol. E94, no. 2, pp. 855–859, 2011. https://doi.org/10.1587/transfun.E94.A.855
60. J. Van Woudenberg, M. Witteman, and F. Menarini. **Practical optical fault injection on secure microcontrollers**, in Proc. Workshop Fault Diagnosis Tolerance Cryptography, Sep. 2011, pp. 91–99.
61. H. Handschuh, P. Paillier, and J. Stern. **Probing attacks on tamperresistant devices**,in Cryptographic Hardware and Embedded Systems (Lecture Notes in Computer Science), C. KoC and C. Paar, Eds. Berlin, Germany: Springer-Verlag, 1999, p. 727
62. M. Medwed, F.-X. Standaert, and F. Regazzoni. **Fresh re-keying: Security against side-channel and fault attacks for low-cost devices**, in Progress in Cryptology (Lecture Notes in Computer Science), Berlin, Germany: Springer-Verlag, 2010, pp. 279–296
63. B. Kaliski and M. Robshaw. **Comments on some new attacks on cryptographic devices**, in Proc. RSA Laboratories' Bulletin, Jul. 1997, pp. 1–5.

64. C. Giraud. **An RSA implementation resistant to fault attacks and to simple power analysis**, IEEE Trans. Comput., vol. 55, no. 9, pp. 1116–1120, Sep. 2006
65. M. Medwed and J.-M. Schmidt.**Coding schemes for arithmetic and logic operations-how robust are they?** in Information Security Applications (Lecture Notes in Computer Science), Berlin, Germany: SpringerVerlag, 2009, pp. 51–65.
66. A. Dominguez-Oviedo and M. Hasan. **Error detection and fault tolerance in ECSM using input randomization**,IEEE Trans. Dependable Secure Comput., vol. 6, no. 3, pp. 175–187, Jul.–Sep. 2009. https://doi.org/10.1109/TDSC.2008.21
67. R. Karri, K. Wu, P. Mishra, and Y. Kim. **Concurrent error detection schemes for fault-based side-channel cryptanalysis of symmetric block ciphers**, IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 21, no. 12, pp. 1509–1517, Dec. 2002
68. I. Biehl, B. Meyer, and V. Müller. **Differential fault attacks on elliptic curve cryptosystems** ,in Proc. 20th Annu. Int. Cryptology Conf. Adv. Cryptol., 2000, pp. 131–146.
69. A. Dominguez-Oviedo. **On fault-based attacks and countermeasures for elliptic curve cryptosystems**, Ph.D. dissertation, Dept. Electr. Comput. Eng., Univ. Waterloo, Waterloo, ON, Canada, 2008.
70. D. Karaklajic, J. Fan, J.-M. Schmidt, and I. Verbauwhede.**Low-cost fault detection method for ECC using montgomery powering ladder**, in Proc. Design, Autom. Test Eur. Conf. Exhibit., Mar. 2011, pp. 1–6.
71. I. Proudler. **Idempotent AN codes**, in Proc. Colloq. Signal Process. Appl. Finite Field Mathem, Jun 1989, pp. 1–5.
72. M. Medwed and J.-M. Schmidt. **Generic fault countermeasure providing data and program flow integrity**, in Proc. Workshop Fault Diagnosis Tolerance Cryptography, Aug. 2008, pp. 68–73 https://doi.org/10.1109/FDTC.2008.11
73. Muzammil Parvez M , R. S. Ernest Ravindran, Syed Inthiyaz , Ch. Tejkumar , K. Veera Ram Sai , K. Ashok Shiva Reddy **Network Security using Notable Cryptographic Algorithm for IoT Data**, International Journal of Emerging Trends in Engineering Research, Volume 8. No. 5, May 2020. https://doi.org/10.30534/ijeter/2020/111852020
74. Amer Abu Salem, Tareq Alhmiedat Energy-Efficient Clustering **WSN System for Environment Monitoring Applications**, International Journal of Emerging Trends in Engineering Research,Volume 8. No. 5, May 2020. https://doi.org/10.30534/ijeter/2020/106852020