Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 08, No. 01, March 2022.

ED_{25519}: A NEW SECURE COMPATIBLE ELLIPTIC CURVE FOR MOBILE WIRELESS NETWORK SECURITY

Mausam Das[1] and Zenghui Wang[2]

(Received: 7-Nov.-2021, Revised: 9-Jan.-2022, Accepted: 24-Jan.-2022)

ABSTRACT

Wireless Sensor Networks (WSNs) create various security threats, such as application variance in different sectors along with the model of cryptographic primitivity and necessity. Despite modernistic evolution, the skillful utilization of Elliptic Curve Cryptography (ECC) for WSNs is a very progressive investigation topic and approaches to reduce the time and intensity cost. Security of ECC commits on the hardness of the Elliptic Curve Discrete Logarithm Problem. Many elliptic curve standards are available, such as ANSI X9.62, NIST FIPS 186-2 …etc. Due to some drawbacks in NIST curves associated with security matters, it is important to investigate for secure alternatives. In our work, we will select ED_{25519} (Edwards curve) at the 128-bit security level and contrast it with Weierstraß curve (also known as Weierstrass curve). To complete the field-calculation functions, we utilize a radix-2^4 , which illustrates the operands with MoTE-ECC for Memsic's MICAz motes over Optimal Prime Fields (OPFs) of variable size; e.g. 160, 192, 224 and 225 bits. We take ECDH (Elliptic-curve Diffie–Hellman) key interchange among two nodes where every node needs two scalar multiplications to execute. The scalar multiplication over twisted Edwards curve utilizes a comb technique to establish base point and utilizes extended projective coordinates for point summation. Our implementation shows that an ECDH takes 18.20 mJ energy consumption over 160-bit OPF, which is performing better than AVR-based sensor node. The advantages of our proposed method will grant advance security and power consumption and diminish communication burden through key management.

KEYWORDS

Domain name system security extension (DNSSEC), Secure real-time transport protocol (SRTP), Secure/Multipurpose internet mail extension (S/MIME), Spectrum-aware degree-ranking-based energy-efficient clustering (SDEC), Advance virtual RISC (Reduced instruction set computing (AVR)), Twisted Edwards curve (TE), Weierstrass curve (WEI).

1. INTRODUCTION

Elliptic Curve Cryptography (ECC) was recommended separately by Victor Miller and Neal Koblitz in 1985 and it started to benefit in cryptographic standards [1]. Cryptographic essential protocols (Transport Layer Security (TLS) protocols, key exchange protocols, public key encryption, digital signatures) that use ECC became very famous due to their small key sizes, exceptional computational performance. Using ECC often yields Perfect Forward Secrecy (PFS), as compared to RSA(Rivest, Shamir, Adleman). In this work, we will consider cryptographic primitives with so-called Optimal Prime Fields (which grant for capable modular rebate), where security builds on Elliptic Curve Discrete Logarithm Problem (ECDLP). To obtain the achievement, we also desire for a lightweight application with small amount of RAM and ROM. Different investigators and associations have suggested many elliptic curves (ECs), such as Weierstraß and NIST; those are used mainly for key exchange and digital signatures. Few prominent examples are Elliptic Curve Diffie-Hellman Key Exchange (ECDHE) and Elliptic Curve Digital Signature Algorithm (ECDSA). For different security levels, NIST has suggested a few prime and binary elliptic curves [5]. Nonetheless, the research community raised ambiguity on the security of Weierstraß or NIST recommended curves for complexity on scalar multiplication within Dual Elliptic Curve Deterministic Random Bit Generator [7], [20] and did not replicate the present state-of-the-art of ECC in terms of efficiency. Due to this reason, we select alternative elliptic curves with better performance and greater security level [31]. Some suggest Brain pool curves developed by Teletrust [8]. Bernstein recommended Montgomery curve; Curve25519 [9]. A set of elliptic curves have been proposed by J.-W.-Bos et al. of Microsoft Research with performance and security perspectives [6]. In this work, we have selected a new elliptic curve ED_{25519} at the 128-bit security level and shown

1. M. Das (ORCID: 0000-0002-0259-7440) is with Department of Information Systems, Madda Walabu University, Ethiopia and with School of Computing, University of South Africa, Florida, South Africa. Email: mausamdas2010@gmail.com
2. Z. Wang (ORCID: 0000-0003-3025-336X) (Corresponding Author) is with Department of Electrical Engineering, University of South Africa, Florida, South Africa. Email: wangz@unisa.ac.za

field-calculation through a radix-2^4 that demonstrates the quantities with MoTE-ECC over Optimal Prime Fields (OPFs) of variable size; e.g. 160, 192, 224 and 225 bits. We take ECDH (Elliptic-curve Diffie–Hellman) key interchange among two nodes. Our implementation shows that an ECDH takes much less energy consumption over 160-bit OPF and we compare it to a Weierstraß curve with regard to ECDLP, energy consumption and "ECC security" [16]. Based on our experiment and calculation, we can say that our selected curve may perform better in wireless networks. Our implementation result takes less energy consumption over 160-bit OPF. The remainder of the paper is organized as follows: in Section 2, we explain Elliptic Curve including ED_{25519} and related work. Section 3 introduces motivation, Section 4 shows the methodology and Section 5 provides implementations. Section 6 shows fixed-base comb method for point multiplication. Section 7 performs security analysis and Section 8 exhibits execution time. Section 9 exemplifies energy consumption and performance and Section 10 represents the conclusion.

2. ELLIPTIC CURVE AND RELATED WORK

2.1 Elliptic Curve

According to Euler and Gauss entirety, Edwards popularized ordinary form of elliptic curve in 2007 [2]. The curve is explained as:

$$y^2 + x^2 = a^2(1 + x^2y^2) \tag{1}$$

over the field K, where $a \in K$, such that: $a^5 \neq a$. As Edwards declared in his paper, each curve of the form given in (1) is bi-rationally identical to an elliptic curve in Weierstraß [3]. Because of an established field K of different distinctive and erratic integers c, $d \in K$ so that $cd(1 - dc^4) \neq 0$, the curves are popularized as:

$$y^2 + x^2 = c^2(1 + dx^2y^2) \tag{2}$$

The aforementioned explanation covers higher than $1 = 4$ of entire isomorphism classes of elliptic curves over a restricted field. It is illustrated that each elliptic curve on a non-binary field is birationally equivalent to a curve in Edwards structure over an expansion of the field and in several facts over the innovative field [4]. In [6], Bernstein et al. established a simplification of Edwards curves called twisted Edwards curves. These combine elliptic curve in Montgomery form [10]. As interpreted in [6], the set of twisted Edwards curves is constant under quadratic flourish, whereby a quadratic twist of an Edwards curve is not naturally an Edwards curve. A quadratic flourish of a curve is an isomorphic curve on a field expansion of scale two. In a field K of different distinctive and non-zero components a, $d \in K$, the twisted Edwards curve $E_{T,a,d}(K)$ is described as:

$$E_{T,a,d}(K): ax^2 + y^2 = 1 + dx^2y^2 \tag{3}$$

If $a = 1$, then $E_{T,a,d}$ is an Edwards curve with $c = 1$. Moreover, $E_{T,a,d}$ is a quadratic flourish of the Edwards curve $E_{\mathcal{O},1,d/a} = a$ with the map: $(\bar{x}, \bar{y}) \rightarrow (x, y) = (\frac{\bar{x}}{\sqrt{a}}, y)$ over the field expansion $K(\sqrt{a})$:

$$\bar{x}^2 + \bar{y}^2 = 1 + (d = a)\bar{x}^2\bar{y}^2 \tag{4}$$

Twisted Edwards curves and Montgomery curves are intently relevant. As explained in [4], each twisted Edwards curve $E_{T,a,d}$ on the ground K with char$(K) \neq 2$ is birationally identical to a Montgomery curve $E_{M,A,B} : Bv^2 = u^3 + Au^2 + u$ using the map:

$$(x, y) \rightarrow (u, v) = (\frac{(1+y)}{(1-y)'}\frac{(1+y)}{(1-y)}) \tag{5}$$

where $A = \frac{(a+d)}{(a-d)'}$ and $B = \frac{4}{(a-d)}$

Whether a is a square in K, therefore such curves are isomorphic through K itself. Since the function enumerates of the point computation in [11], it is simple to watch that twisted Edwards curves surpass curves in Weierstraß shape in fast condition (although the binary form of Edwards curve is a bit-delay from compared with its Weierstraß equivalent [10]). Twisted Edwards curve's cluster rules are standardized and perfect; that carries to secure fulfilments over specific kinds of offensives [4]. The best relevant implementation of twisted Edwards curves is Edwards-curve Digital Signature Algorithm (EdDSA). The ED_{25519} is a twisted Edwards curve utilized for EdDSA, elsewhere particular parameters are determined like [12]: $a = -1,\ d = \frac{121665}{121666'},\ p = 2^{255} - 19.$

The respective Montgomery curve of ED_{25519} is Curve25519 that is specified as [13]:

$$y^2 = x^3 + 486662x^2 + x \quad (6)$$

Point propagation is speedy and capable upon Montgomery curves. This successfully utilizes unlike point supplement and point folding [11] as well as regular Montgomery ladder computation to execute a point addition[14]. The constant Montgomery ladder algorithm [40] is executed in fixed rate, which leads to timing attack. Various activities have utilized Curve25519 since its presentation by Bernstein in 2006 [9]. Moreover, because of its 128-bit security stage and effective computation [44], it has also an optimistic application for Internet of Things (IoT) demand. Currently, an amount of hardware fulfilments have been established [15]-[19] over a concentrate on IoT demand. All these functions use FPGA (Field-Programmable Gate Array) DSP (Digital Signal Processing) segments to execute modular factors. High-efficiency cryptographic converters that may be initiated on affordable FPGAs or ASICs (Application Specific Integrated Circuits) are in order in favour of mobile uses similar to the Internet of Things (IoT) and Intelligent Transport Systems (ITSs) [14]. Affordable FPGAs (containing negative-consolidated-established FPGAs) are specifically limited in several hardware funds. Utilization of hardware assets will minimize with movable low energy without wasting achievement [43]. In consequence, we propose a zone-competent, low-energy hardware execution of the ED_{25519} on FPGA. The DSP parts of FPGA assets will not utilize our exploit. We establish a great race interlace modular factor adjusted in favour of such implementation.

2.2 Related Work

Security: Studies specifically associated to the security of ECDSA P-256 and ED_{25519} have been performed before. Nystrom [21] observed through an examination of an initial RFC8080 (Request for Comments) design with no citation or confirmation that ED_{25519} would provide enhanced security assets and enforcement features comparable to RSA and ECDSA algorithims, causing such declaration to be eliminated through RFC8080. However, there are motivations to trust that Ed25519 gives better security compared with ECDSA P-256; e.g. while monitoring Lange and Bernstein's security roster in favour of elliptic curves in comprehensive called security curves [16], it is observed that Weierstraß is examined to be unreliable, while ED_{25519} (which is related to Curve25519 [38]) is considered to be safe. An ECDSA P-256 particular assault has been outlined as well. Brumley et al. [22] discovered that such ECDSA P-256 in the newest form of OpenSSL 1.0.1 (which is OpenSSL 1.0.1u) is exposed to reserve-schedule attacks, permitting themselves to restore the individual for TLS and SSH. That might be appropriate in favour of DNSSEC, whereas DNSSEC package may trust OpenSSL, considering that enforcements of ED_{25519} might be protected against reserve-schedule attacks[23]. The importance of these security inconvenience and achievable alleviates is evaluated in this work.

3. MOTIVATION

In this study, we choose ED_{25519} curve and its extended twisted Edwards coordinates at the 128-bit security level. A famous Weierstraß elliptic curve is presently obtained, though it contains a few disadvantages due to that we select another curve ED_{25519}. Additionally, we focus on high-speed signature validity, achieving SPA (Simple Power Analysis) attack and high-speed scaler computations. Security has become a major concern due to high benefit through IoT equipment which we are using in our daily life. Particular types of IoT equipment are source-compelled; in favour of particular cases, these contain less storage capacity as well as lengthy battery life. Due to this, encoded algorithms such as ECC are appropriate here.

3.1 Some Drawbacks of Weierstraß Curves

Weierstraß mathematical expression is $y^2 + ax + b$ over $\mathbb{F}_p$. The straightforward calculation in Weierstraß curve is hard. Magma supplies small methods in favour of calculation on elliptic curves shown in small as well as in large (difficult) Weierstraß designs [15]. The circumstance is much intricate in the constant case: the majority of quality algorithms may effectively move into exceptions. Weierstraß curve's quantifiable characteristics place into 3 and twisted Edwards curve's quantifiable characteristics place into -1. Two alternative curves are chosen in a fixed way and provide twist-security; this characteristic is benefited from in some works. Clock period of Weierstraß curve's point supplement as well as point multiplication are high over 160-bit OPFs (Optimal Prime Fields). This type of curves is

of less value to assist Twisted Edwards curves over prime fields and might be combined with past equipment by exchanging the curve's variables as well as the field computation.

3.2 Problems in Existing WSN Encryption System

Our current task is relevant to high-speed ECDH key shares and less energy consumption. Attackers may estimate the existing record from encoded media [24]. The problem that we have recognized is that the effective use of PKC(Public-Key Cryptography) is the serious restrict assets of cell-voltage sensor nodes. Also, ECDH key exchange energy ingest is high [41]. For illustration, the predominant MICAz mote from (the ATmega128 [25]) as well as distinctive attributes are 4 KB of RAM and 128 KB flash storage area. Tiny ECC [26] is the presently installed ECC software package for WSNs; this is a favourable arrangement with several times observations to establish famous curves across 160- and 192-bit primary grounds. However, we have noticed in [27] that ECC on compelled equipment is not a self-loading occurrence; for this reason, the advanced function currently does not fulfil the majority of the software. So, the effective performance of ECC on sensor node is still a demandable research matter and new methods are necessary to enhance the performance rate (i.e., energy cost) as well as memory.

3.3 Suitable Encryption Scheme for WSNs

Encoding and decoding expressions are normally asset-vigorous security techniques [42], but wireless equipment is attribute-restricted. Therefore, thin-security algorithms are considered relatively less material- absorbing. In this statement, measured to other non-symmetrical key algorithms, ECC is a more desirable act, because the size of key length is very small as well as it needs a smaller amount of power absorption. Another important thing is that detector node equipment is generally a fixed mechanism which is composed of technique-on-chip, micro devices, memory chips, energy-control ICs and additional similar types of chips. In favour of security-specified jobs, varieties of ICs are deployed in this scheme. Whether a particular concern security plan is preferred for wireless sensor node equipment, for a particular case where ECC arrangement is in favour of key shares, we decide another possibility to diminish energy consumption. Whether only ECC is deployed for together key shares and encoding functions, then the complete security process might be applied through a restricted amount of chips to reduce the quantity of gates over circuits and diminish energy absorption. Through this experiment, we have seen high-speed ECDH key shares with less energy consumption.

4. METHODOLOGY

We carefully choose ED_{25519} curve in relation to EdDSA and extended coordinates aspect. Particularly, this curve's expression is $E: \{(x, y) \in F_q \times F_q : -x^2 + y^2 = 1 + dx^2y^2\}$. ED_{25519}-SHA (Secure Hash Algorithm)-512 is EdDSA accompanied by these arguments: $b = 256$; H is $SHA - 512$; q is the prime $2^{255} - 19$; the 255-bit encrypting of $F_{2^{255}-19}$ is the regular small-endian encoding of $\{0,1\ldots, 2^{255} - 20\}$; ℓ is the prime $2^{252} + 27742317777372353535851937790883648493$ from [28], $d = -121665/121666 \in F_q$; as well as B is the special point $(x, 4/5) \in E$ for which x is optimistic. This Edwards curve is corresponding to $-x^2 + y^2 = 1 - \left(\frac{121665}{121666}\right)x^2y^2$, because -1 is multiplied in F_q. Additionally, the security of ED_{25519}-SHA-512 is not harmed, because r is not able to be seen by the attacker. The most vital job in favour of elliptic prime curve procreation is to select a prime number. Brainpool curves achieve pseudo-random prime digits to produce the prime curves, but due to lack of capabilities, such kind of curve is not as good as Edwards curves. Weierstraß curves make progress on random prime branches. To enhance performance, TinyECC or MICAz restrain better situations in favour of proficiency 128-, 160- and 192-bit fields. We have deployed coordinates-twisted Edwards curve and have selected the additional quantifiable characteristics of the curve in a particular way. In favour of field-computing process, we deploy a radix-2^4 model in relation to MoTE-ECC adjacent to Optimal Prime Fields (OPFs). In MoTE-ECC, RAM size in favour of 256-bit OPFs is 556 bytes and for 160-bit OPFs, RAM size is 380 bytes, which is smaller than in AVR-based sensor nodes. Our selected curve's parameters and other details are explained in the next parts.

4.1 Elliptic Curve Discrete Logarithm Problem (ECDLP)

Elliptic curve discrete logarithm problem (ECDLP) specifies the security of ED_{25519} and Curve25519.

Assume E is the elliptic curve stated on the prime field $\mathbb{F}_p$ and assume the combination of logical dots over curve E indicated through $E(\mathbb{F}_p)$. At the present time, consider a point $P \in 2E(\mathbb{F}_p)$ of order n as well as the closed ring subgroup of $E(\mathbb{F}_p)$ produced by point $\langle P\rangle = \{\mathcal{O}, P, 2P, \ldots, (n-1)P\}$. Acquire a random number $k \in [1, n-1]$ and let $Q = k \cdot P$, where the point Q is stated through joining spot P to emphasize $k-1$ times.

$$Q = k.P = \underbrace{\{P + P \ldots + P.\}}_{k\ times} \quad (7)$$

Specifying particular parameters of an area Q, the problem of establishing a particular number k is specified (ECDLP) [15]. The dot Q is able to be quickly calculated in relation to k deploying particular identical-aspect task $Q = k \cdot P$ (declared elliptic curve point doubling or scalar multiplication). However, to determine analytically k from recognized points Q and P is absolutely complex.

4.2 Parameters of Ed25519

Parameter	Value
p	p of ED_{25519} in [RFC7748] (i.e., 2^{255}-19)
b	256
encoding of	255-bit little-endian encoding of $\{0,1,\ldots,p-1\}$
H(x)	SHA-512(dom2(phflag, context)‖ x) [RFC6234]
c	base 2 logarithm of cofactor of ED_{25519} (i.e., 3)
n	254
d	-121665/121666=37095705934669439343138083508754565189542113898432190163887855330859402835555
a	-1
b	(X(P),Y(P)) of ED_{25519} in [RFC7748] (i.e., 15112221349535400772501151409588531511454012693041857206046113283949847762202, 46316835694926478169428394003475163141307993866256225615783033603165251855960))
L	Order of ED_{25519} in [RFC7748] i.e., 2^252+27742317777372353535851937790883648493).
PH(x)	x (i.e., the identity function)

4.3 Edwards-curve Digital Signature Algorithm (EdDSA)

EdDSA is a digital authorization arrangement. ED_{25519} understands EdDSA authorization.

Algorithm 1. EdDSA key establishment as well as authorization generation

Key setup.
1: Hash k such that $H(k) = (h_0, h_1, \ldots, h_{2b-1}) = (a, b)$
2: $a = (h_0, \ldots, h_{b-1})$ perform with integer in little-endian symbols
3: $b = (h_b, \ldots, h_{2b-1})$
4: Compute public key: $A = aB$
Signature generation.
5. Compute ephemeral private key: $r = H(b, M)$.
6. Compute ephemeral public key: $R = rB$.
7. Compute $h = H(R, A, M)$ and convert to integer.
8. Compute: $S = (r + ha) \bmod \ell$.
9. Signature pair: (R, S).

Establishing the key first four sequences is deployed and implemented through a private key. Element $(x, \ldots, y)$ indicates addition of the constituent part. We specified an individual scalar and $b = (h_0, h_1, \ldots, h_{2b-1})$ the auxiliary key. Particular ephemeral key r is established in Step 5 . To justify an authorization (R, S) over a message M accompanied by public key A, a justifier observes the method explained in Algorithm 2. ECDSA acts this way: it substitutes F_q^* accompanied by an order$-\ell$ subdivision of an elliptic-curve set in contrast with F_q and describes $x(R)$ even though $x -$ is the element of R. ECDSA in addition to substituting A accompanied by $-A$, exchanges the authorizer's

Algorithm 2. EdDSA signature establishment

1: Compute $h = H(R, A, M)$ and convert it into an integer.
2: Check if the group equation $8SB = 8R + 8hA$ in E holds.
3: If the group equation holds, the signature is correct.

calculation directed to sum and acquires the establishment of equation $H(M)B + x(R)A = SR$. ECDSA substitutes specifically three-scalar mathematical expressions accompanied by the function of two-scalar mathematical expressions $S^{-1}H(M)B + S^{1}x(R)A = R$ at the cost of needing S to be altered modulo ℓ.

The $pmuldq/pmuludq$ directions accomplish two quantities of 32-bit numbers, manufacturing 64-bit output in each sequence. The $pmuldq$ direction is explained in [23].

4.4 Elliptic-curve Diffie–Hellman (ECDH)

A direction by Hisil [29] of ECDH on an Edwards curve is accompanied by identical security characteristics to Curve25519. Lin and Scott [30] of ECDH conducted an investigation on an Edwards curve in addition to an endomorphism. Bernstein's Curve25519 programme was used in favour of Diffie-Hellman key shares.

Such curve is described by $\frac{\varepsilon}{F_p}: y^2 = x(x^2 + 48662_x + 1) where\ p = 2^{255} - 19$.

We identify $\#\varepsilon(F_p) = 8r$ and $\#\varepsilon'(F_p) = 4r'$, in which r as well as r' are 253-bit primes. Recent performance of this system uses x-coordinate on a Montgomery display of the curve, for together mathematical reduction, side-channel security and little effort to establish.

4.5 Extended Twisted Edwards Coordinates

The important mathematical relationship in favour of point calculation on twisted Edwards curves was nominated by Hisil et al [29], constituting points in the utmost twisted Edwards coordinates: a point $P = (x, y)$ is appointed through the quadruple $(X: Y: T: Z)$; for example $x = X/Z$, $y = Y/Z$, $xy = T/Z$ and $Z \neq 0$. The additional coordinate T homogeneous coordinates were derived $(X: Y: Z)$ in relation to the multiplication of x as well as y, with a characteristic $T = XY/Z$. The group affinity element is demonstrated through $(0: 1: 0: 1)$, undesirable of an element $(X: Y: T: Z)$ which is $(-X: Y: -T: Z)$. A point in affine elements (x, y) be able to change into extended twisted Edwards coordinates by $X = x$, $Y = y$, $T = xy$ as well as $Z = 1$. To change rear to affine, T is disregarded, in addition to an inversion and two multiplications are needed: $x = X/Z$ and $y = Y/Z$. Likewise, it is possible to convert a point into identical projective coordinates $(X: Y: Z)$ merely *via* dumping T. Hisil et al projected an extensive coordinate scheme to facilitate a supplementary coordinate $t = xy$ [29]. As a substitute of signifying a point over twisted Edwards curve E_T through association with x and y coordinate solitary, we can utilize the extended affine coordinates (x, y, t). The consequent developed coordinates of that point are $(X: Y: T: Z)$, by which the supplementary coordinate T has the assets $T = XY/Z$ through $Z \neq 0$. In appreciation of these coordinates, Hisil et al. invented the proficient point addition method, particularly under the constraint $a = -1$. Behind implementation of clear-cut resources [32], the mathematical calculation rate of an assorted point addition over a curve is done through $a = -1$ total to $7M + 6A$, whereas a doubling needs $3M + 4S + 6A$.

Algorithm 3. Point multiplication in assorted homogeneous and extended twisted Edwards coordinates

Input: $\boldsymbol{P_1 = (X_1; Y_1; Z_1)}$ **in homogeneous projective coordinates.**
Output: $\boldsymbol{P_3 = 2P_1 = (X_3; Y_3; T_3; Z_3)}$ **in extended twisted Edwards coordinates.**
1: $A \leftarrow X_1^2;\ B \leftarrow Y_1^2;\ C \leftarrow 2Z_1^2$
2: $D \leftarrow -A;\ E \leftarrow (X_1 + Y_1)^2 - A - B;\ G \leftarrow D + B$
3: $F \leftarrow G - C;\ H \leftarrow D - B;\ X_3 \leftarrow E.F$
4: $Y3 \leftarrow G.H;\ T_3 \leftarrow E.H;\ Z_3 \leftarrow F.G$

Algorithm 4. Point addition in extended twisted Edwards coordinates

Input: $\boldsymbol{P1 = (X_1, Y_1, T_1, Z_1)}$ **and** $\boldsymbol{P_2 = (X_2, Y_2, T_2, Z_2)}$ **in extended twisted Edwards coordinates; constant** $\boldsymbol{k = -2d}$**, where** $\boldsymbol{d = -121665/121666}$**.**
Output: $\boldsymbol{P_3 = (X_3, Y_3, T_3, Z_3)}$ **in extended twisted Edwards coordinates.**
1: $A \leftarrow (Y_1 - X_1).(Y_2 - X_2);\ B \leftarrow (Y_1 + X_1).(Y_2 + X_2);\ C \leftarrow k.T1.T_2;$
2: $D \leftarrow 2Z_1Z_2;\ E \leftarrow B - A;\ F \leftarrow D - C;$
3: $G \leftarrow D + C;\ H \leftarrow B + A;\ X_3 \leftarrow E.F;$
4: $Y3 \leftarrow G.H;\ T_3 \leftarrow E.H;\ Z_3 \leftarrow F.G;$

4.6 Projective Coordinate Randomization

We put the arbitrary projective coordinates countermeasure to the extended twisted Edward coordinates $(X{:}\,Y{:}\,T{:}\,Z)$ in Joye's Double-Add, Goundar's Signed-digit and FLS (Fuzzy Logic System) algorithms [33]. In Joye's Double-Add and Goundar's Signed-digit algorithms, we arbitrarily produce $\lambda \in F_p \backslash 0$ and execute $X' \leftarrow \lambda x,\ Y' \leftarrow \lambda y,\ T' \leftarrow xY'\ Z \leftarrow \lambda$, where $P = (x; y)$ is the enter position in affine organization and particular consequential point $P' = (X'{:}\,Y', T', Z')$ is worn in position of P within particular rest of algorithms. During FLS algorithm [31], we indiscriminate the coordinates for initial point which is filled from the chart of previous calculated points, $P_0 = (X{:}\,Y{:}\,T{:}\,Z)$, as pursue: produce arbitrary $\lambda \in F_p \backslash 0$ and do $X' \leftarrow \lambda x,\ Y' \leftarrow \lambda Y,\ T' \leftarrow \lambda$ T and $Z' \leftarrow \lambda Z$. Particular consequential point $P'_0 = (X'{:}\,Y'{:}\,T'{:}\,Z')$ is adopted in position of P_0. While this countermeasure is implemented, the ideals of the coordinates of the accumulator point Q are randomized, altering from single implementation of the scalar multiplication to the additional.

Since particular significance of P'_0 is allocated to Q in the foundation of assessment phase, the extended twisted Edwards coordinates of every point $P_i j$ are governed to accumulate in the table, by means of a random λ produced in favour of every point, similar to how P_0 was governed. The authorization generation utilizes the FLS algorithm through $(v = 1; w = 4)$ (8 points, 1 chart) with an exclusive table search for safeguarded and governed coordinates countermeasures. The accomplishment effect of EdDSA-ED_{25519}-SHA512 progress particular state of the art capabilities [34] needs 19047706 sequences for authorizing; an enhancement of 17.9% and 30776942 cycles in favour of authentication; an enhancement of 5.7%. The transparency of the table search security (action taken for threat) as well as governed projective coordinates to particular FLS algorithm is merely 1.0%. Likewise, while these actions are implemented over the signature creation role, transparency is too little (0.9%). To calculate shared secret utility, overhead of the coordinate randomization is only 0.04%. Particular point replication algorithm (Algorithm 2) stands on the enthusiastic doubling principle, most effective in favour of $a = -1$ (the case for ED_{25519}) and charges $4M + 4S$. We examine stand on Faz-Hernandez et al.'s customized LSB (Lower Side Band)-set comb algorithm, called FLS [33].

4.7 Prime Field $\mathbb{F}_2^{255-19}$

A constituent of $\mathbb{F}_p$ is an integer modulo 2^{256-38} through field process. This surplus illustration favourably permits additional proficient decline than sinking straight modulo p. Simply, at the last part of scalar multiplication computation, where an integer is not previously existent in $\mathbb{F}_p$, we deduct p in steady time.

Table 1. Standard outcome on ATmega328P.

Operation Class	Operation/Algorithm	Cycles
Fixed-base ECSM ED_{25519}	FLS(v=1,w=3)	21 553 188
	FLS(v=2,w=4)	26 661 293
	FLS(v=1,w=3) lookup prot.	21 658 857
	FLS(v=1,w=4)	18 119 234
	FLS(v=2,w=3)	19 170 150
	FLS(v=1,w=4)lookup prot.	18 264 710
	FLS(v=2,w=3) lookup+rand. coord.	18298387

4.8 Field Multiplication and Squaring

Field squaring is executed as a 3-level subtractive Karatsuba, in which there is no provisional exclusion of M. The 32-bit multiplier from the multiplication is reprocessed here, along with a function name, at the foundation level.

4.9 Field Inversion

We utilize Fermat's theorem, $x^{-1} \equiv x^{p-2} (mod\ \ p)$, to calculate inversion in $\mathbb{F}_p$ in steady time. Addition sequence is composing of 254 squares at 11 multiplications, although we diminish the amount of provisional field variables; those needed are 10 to just 5.

4.10 Arithmetic Modulo Ed25519 Group Order

EdDSA ED_{25519} authorization method needs accumulation as well as multiplication modulo ED_{25519} cluster order (N). We execute decline modulo N in C utilizing a stable rate edition of the Barret algorithm achieved through unfolding the ultimate subtraction circle into pair facsimiles of its body (the highest digit of computational process) as well as utilizing provisional shifts executed in steady time. We computed the mutual of the modulus $R = \lfloor b^{2n}/N \rfloor = \lfloor 256^{64}/N \rfloor$, where a constraint of the Barret algorithm [15] is accumulated into program memory. The development identifies the 256-bit multiplier and afterword diminishes completely. The addition is also reduced as well as executed in assembly.

4.11 Optimal Prime Fields

The prime fields that we utilize in MoTE-ECC fit in a unique class of restricted fields recognized as Optimal Prime Fields (OPFs) [39]. The OPF library establishes a great extent of resilience as single and an identical role can process to operate every extent. An additional significant characteristic of the library is its flexibility over SPA attacks, since entire mathematics functions are applied in a normal manner and perform forever the identical series of guidelines, irrespective of the real worth of the operands. These grounds are described through primes and mentioned as $p = u.2^k + v$ by which u as well as v are a little contrasted to 2^k; henceforth they robust into single or double registers for targeted podium. MoTE-ECC sustains OPFs through $2^{15} \leq u < 2^{16}$ (i.e. u is 16 bits lengthy) as well as $v = 1$. An actual illustration is $p = 65356.2^{144} + 1$ (i.e., $u = 65356$ and $k = 144$), which occurs through a 160-bit prime and hex notation is as pursued. $p = 0xFF4C0000000000000000000000000000000000001$. Low hamming weight characterized these prime structures, while only extremely important bytes as well as slightest-importance bytes are non-zero; while other bytes are zero. The small power of p permits favourably precise resources of modular calculation, since just non-zero bytes of p require to be progressed in the decline function. We apply the inversion in OPFs founded on Fermat's little theorem $a^{p-2} \equiv a^{-1} \bmod p$, the straight idea of the appearance $u.2^k - 1$, needing n squarings as well as n multiplications, by which n indicates the bit-span of p. To diminish the bit-span, we choose an algorithm in favour of effective enhancement-supported inversion with regard to OPFs.

In appreciation about such an algorithm, it is feasible to reduce the entire number of functions to n squaring addition just $HW(k) + HW(u-1) + 1$ multiplications, whereby $HW(x)$ implies the hamming weight for x. During the initial period, $a^{2^{0.6ex}}$k-1 is measured through the exponentiation technique. During the next stage, a right-to-left square-and-multiply algorithm is conducted.

Algorithm 5. Optimized exponentiation-based inversion for OPFs

Input: Element a of $\mathbb{F}_p$ with $p = u.2^k + 1$.**Output:** $r \equiv a^{u.2^k-1} \equiv a^{-1} \bmod p$.

1: $u' \leftarrow u - 1$
2: $r \leftarrow a$, $b \leftarrow \lceil ld(k) - 2 \rceil$, $i \leftarrow 1$
3: while $b > 0$ do
4: $t \leftarrow r^2$
5: for $j = 1$ to $i - 1$ do
6: $t \leftarrow t^2$
7: end for
8: $r \leftarrow r.t$
9: $i \leftarrow i \ll 1$
10: if $k\&b > 0$ then
11: $r \leftarrow r^2.a$
12: $i \leftarrow i + 1$
13: end if
14: $b \leftarrow b \gg 1$
15: end while
16: $t \leftarrow r.a$
17: $b \leftarrow 1$
18: while $b < 0x8000$ do
19: if $u'\&b > 0$ then
20: $r \leftarrow r.t$
21: end if
22: $t \leftarrow t^2$
23: $b \leftarrow b \ll 1$
24: end while
25: if $u'\&b > 0$ then
26: $r \leftarrow r.t$
27: end if

5. IMPLEMENTATIONS

A radix-2^r illustration a component f within b-bit prime area like $(f0, f1, \ldots, f_{\lceil b/r \rceil - 1})$ is given as:

$$f = \sum_{i=0}^{\lceil \frac{b}{r} \rceil - 1} fi2^{\lceil ir \rceil}$$

This is termed a radix-2^r illustration. We utilize radix-2^4, indication in favour of field components. We diminish intermediary consequences modulo $2^{256} - 38$ within the complete implementation of the scalar propagation and merely diminish the ultimate result modulo $2^{255} - 19$. We completed n- bit quantities, an entire of $\lceil n/2 \rceil$ limited results is produced, where the outcome of extreme altitude for limited output array is $\lceil n/2 \rceil + 1$ components to be combined. A radix-2^4 numeral is ullustrated through numbers, since the particular set $D = \{0,1,2\ldots,14,15\}$ within an identical radix-2^4 illustration utilizing a zero-discharge number set in the shape of $D' = \{\pm 1, \pm 3, \ldots, \pm 13, \pm 15\}$. $\mathbb{F}_p$ implies an OPF established through a prime structure $p = u \cdot 2^k + 1$; therefore, u is within the order $[2^{15}, 2^{16} - 1]$; i.e., u contains extent of 16 bits. Despite the aforementioned topic, the bit range n for primes is a product of 32; e.g. n = 160, 192, 224 or 256 bits. Field components are mentioned as $a \in \mathbb{F}_p$. The proper partitioning application is selected to the equilibrium of the digit of odd as well as uniform directions, diminishing the entire digits of the needed round. We track investigation: odd n-bit numeral k specified by $k = \sum_{i=0}^{n-1} k_i \cdot 2^i$ through $k_i \in \{0,1\}$ for $0 < i < n - 1$ and $k_{n-1} = k_0 = 1$ may be mentioned as conventional Binary Signed-Digit(BSD), since $k = 2^{n-1} + \sum_{i=0}^{n-2} (2k_{i+1} - 1) \cdot 2^i$; i.e., entire numbers of BSD illustration of k are non-null. For confirmation, we monitor:

$$\begin{aligned} k &= 2^{n-1} + \textstyle\sum_{i=0}^{n-2} (2k_{i+1} - 1) \cdot 2^i = 2^{n-1} - \sum_{i=0}^{n-2} 2^i + \sum_{i=0}^{n-2} 2k_{i+1} \cdot 2^i \\ &= 1 + \textstyle\sum_{i=0}^{n-2} k_{i+1} \cdot 2^{i+1} = 1 + \sum_{i=0}^{n-1} k_i \cdot 2^i = \sum_{i=0}^{n-1} k_i \cdot 2^i \ \ with \ \ k_0 = 1 \end{aligned} \tag{8}$$

This formula is utilized to change an odd numeral provided in regular binary shape into a BSD illustration including merely non-zero numerals; specifically -1 and 1.

6. FIXED-BASE COMB METHOD FOR POINT MULTIPLICATION

We transfer the entire binary illustration of k single bit as appropriate and introduce a "1" within empty MSB (Most- Significant Bit) place. Presently, this transferred bit-series is accurately transferred in the structure of BSD of k to clarify all zero bits as -1. Radix-2^4 illustration may be acquired by splitting the bit-series within a cluster of 4-bit numerals, every single one communicating with an odd digit in the area $[-15, 15]$. This way, w indicates the digit of bits (i.e., size of bit-series) treated in every replication of the curve and $d = \lceil n/w \rceil$. Our alternative composes an off-heritage moment (Step 1) as well as an online stage. During the beginning stage, 2^{w-1} items are computed in advance and accumulated, including all straight associations of P. Our execution computes in advance eight points as we utilize $w = 4$ to obtain an equalization among implementation times as well as accumulator demands. A demonstration of the structure $(2a_i - 1)$ in stage 1 output is one way 1 (while $a_i = 1$) or the other -1 (if $a_i = 0$), therefore implementing the numeral-set alteration explained previously. In every recurrence, the online form includes a basic curve that performs duality followed through summation. Anyhow, compared to established comb technique, $w - 1$ bits (in place of w bits) through k are utilized to establish a certain kind of 2^{w-1} advance calculated points, which are to be added, when an additional bit (specifically $K_{(w-1)d+i}$ within stage 5 of Algorithm 6) is specified, since such point is truly joined or deducted. To accomplish a normal implementation, we require an operation that is subject to the significance of a bit, allocating a spot R or the adverse of that spot (i.e., $-R$) to a target. The pessimistic matter of R in prolonged refined coordinates is $-R = (-x, y, -t)$. MoTE-ECC executes the revocation of a component $x \in \mathbb{F}_p$ confiding on the quality of a bit b as pursue. We compute $x' = p - x$ through deductions after we carry bit b to extract a cover m, that is likewise an all-1 byte (if $b = 1$), otherwise an $all - 0$ byte (if $b = 0$). Furthermore, a second mask is required: m' is the bit-smart supplement of m; i.e., m' is 0 when m is an all -1 byte and *vice versa*. Next, we assess $(x'_i \& m) | (x_i \& m')$ for total byte of x' as well as x (whereby & and | express a bit-smart and and or function). Particular field is whether $-x = p - x$ (if $b = 1$; i.e., the negation is truly performed) rather (if $b = 0$; i.e., no contradiction). Comb procedure follows $d - 1$ point addition and $d - 1$ multiplication of the real worth of scalar bits.

Algorithm 6. Regular w-bit comb method for fixed-base scalar multiplication

Input: $\boldsymbol{n - bit\ scalar\ k = (k_{n-1}, \ldots, k_1, k_0)_2\ with\ k_0 = 1, point\ P \in E(\mathbb{F}_p)}$.
Output: $\boldsymbol{Q = k \cdot P}$

1: Pre-compute $R[j] = R[a_{w-2}, \ldots, a_1, a_0] = 2^{dw}P + (2a_{w-2} - 1)2^{(d-1)w}P + \ldots + (2a_1 - 1)2^wP + (2a_0 - 1)P\ for\ all\ bit - strings\ j = (a_{w-2}, \ldots, a_1, a_0)\ of\ length\ w - 1$
2: $Q \leftarrow R[k_{dw}, \ldots k_{2d}, kd]$
3: $for\ i = d - 1\ down\ to\ 1\ do$
4: $Q \leftarrow 2Q$
5: $Q \leftarrow Q + (2k_{(w-1)d+i} - 1) \cdot R[k_{(w-2)d+i}, \ldots, k_{d+i}, k_i]$
6: end for

7. SECURITY ANALYSIS

MoTE-ECC acquires the prolonged coordinate procedure in favour of twisted Edwards curves. It is possible to convert each twisted Edwards curve into a Montgomery curve conversely. ECDH procedure is to utilize the ridiculous comparison among Montgomery and twisted Edwards curves. Every L-detector is pre-installed through a single individual secret key. Next to key setup, every couple of connected L-detectors has various mutual keys. Therefore, yielding L-detector does not influence the security of transmissions within different L-detector. The DH private key is merely calculated among two transmit stakeholders, After that, it is utilized in favour of its progressive communication. Scalar multiplication protocols generally include three instances: established base point (kG), while G is a determined point (generally subset creator) and k is a scalar; varying foundation point (kP), while P is a point which is not previously known. Suppose two detector nodes A and B to determine a mutual private key, while the group domain arguments (a, d, A, B, G, p) are concurred above. This way, a and d are the arguments of twisted Edwards curve E_T, when A and B distinguished to be bi-rationally identical to Montgomery curve E_M. G exists at a point of prime rule over E_T and p specifies the essential OPF. Single turn ECDH key sharing protocol can be split into three phases:

First; node A produces a private key d_A and produces the respective public key $Q = d_A \cdot G$. Such scalar multiplication is completed through twisted Edwards curve E_T utilizing originator G. Afterwards, node A transforms the point $Q = (x_q, y_q)$ into spot $M = (x_m, y_m)$ over the bi-rationally identical Montgomery curve E_M and dispatches x-coordinate x_m of M to node B. Node B executes the identical stairs through private key d_B and transmits particular x-coordinate to A.

Second; thereafter, x-coordinate is obtained by node A from B; it starts to measure the scalar multiplication $S = d_A \cdot M$(M includes merely an x coordinate) over the Montgomery curve E_M. Node B performs similarly through the x-coordinate, obtained from node A. Together node A as well as node B need to perform dual scalar multiplication to acquire the mutual private key $S = d_A \cdot d_B \cdot G$. Considering that the foundation point G is steady and previously aware, we utilize quick scalar multiplication through fixed-base comb procedure utilizing a window diameter $w = 4$ as well as eight points quantified in advance.

Third; ECDH key interchange is mainly established by the calculated energy W_c in favour of two scalar multiplications; the correspondence energy W_t is mainly insignificant.

Table 2. Execution time (clock cycle) of field arithmetic function for operands of a measurement of 160,192, 224 and 256 bits.

Operation	160 bits	192 bits	224 bits	256 bits
mod_add	530	631	732	833
mod_sub	530	631	732	833
mod_mul	3237	4500	5971	7650
mod_sqr	2901	3909	5058	6347
mod_inv	571916	830823	1163655	1491839

Executing a fixed-base scalar multiplication over twisted Edwards curve $E_T: -x^2 + y^2 = 1 - 121665/121666x^2y^2$ is in contrast to $\mathbb{F}_2^{255} - 19$ and the outcome is reversed compared to the Mongomery curve in terms of single inversion. The curve points are represented as $E_T(F_2^{255} - 19)$. A

proficiently quantifiable bi-rational correspondence exists between E_T and E_M, hence the curves exchange similar cluster framework. Twisted Edwards curve is ideal in favour of $a = -1$ (for ED_{25519}), the input point is truly illustrated in twisted Edward complements and point Q outcomes are based on quick scalar multiplication in extended projective coordinates. Alteration of point Q over a twisted Edwards curve E_T within a point M on the birationally-corresponding Montgomery curve E_M can be performed in this method. Initially, we alter the projective point $Q = (X_q, Y_q, T_q, Z_q)$ on E_T associated with affine illustration $Q = (x_q, y_q)$ and work out $M = (x_m, y_m)$ on E_M by the use of $x_m = (1 + y_q)/(1 - y_q)$ along with $y_m = (1 + y_q)/((1 - y_q) \cdot x_q)$. We scamper an inversion within affine-to-projective alteration to acquire $1/Z_q$ as well as other reversal for the element of Edwards-to-Montgomery alteration ($to\ obtain\ 1/[(1 - y_t) \cdot x_t]$). To diminish the computational transparency reasoned through two inversions, we straightforwardly alter the point $Q = (X_q, Y_q, Tq, Z_q)$ to the point $M = (x_m, y_m)$ as follows:

$$x_m = (1 + y_q)/(1 - y_q) = (1 + Y_q/Z_q)/(1 - Y_q/Z_q) = (Z_q + Y_q)/(Z_q - Y_q) \quad (9)$$

$$y_m = (1 + y_q)/(x_q \cdot (1 - y_q)) = (Z_q^2 + Y_q Z_q)/(X_q Z_q - X_q Y_q) \quad (10)$$

Now, we will obtain one inversion to calculate $1/(X_q Z_q - x_q Y_q)$, which is multiplied by X_q to get $1/(Z_q, -Y_q)$.

8. EXECUTION TIME

We applied the OPF inversion from scrape and utilized OPF documentation from additional arithmetic functions.

We utilize the role of ANSI C and establish the performance time of point addition and point multiplication on twisted Edwards curve. The point addition and point multiplication on twisted Edwards curve are quicker than on Weierstraß curve. The supremacy of scalar multiplication through the performance period of all cryptographic activities is clear. Regarding signature authentication performance on ED_{25519}, the implementation time rises dramatically while the marvellous characteristic is activated since this presents an additional multiplication. Particular arithmetic functions replicate the carry transmission sequence even without transmission. Particular regular-time implementation characteristic (not obligatory) makes stronger the execution in contrast to the aggressor's capability to utilize side-channels within the structure of executing timing assault.

Table 3. Implementation time (in clock cycles) of point arithmetic functions over 160-, 192-, 224- and 256-bit OPFs.

Operation	160 bits	192 bits	224 bits	256 bits
TE point add	27355	36903	47907	60367
TE point dbl	25421	33848	43463	54262
WEI Point add	40222	N/A	N/A	N/A
WEI Point dbl	31536	N/A	N/A	N/A

Table 4. Implementation time (in clock series) of scalar multiplication over 160-, 192-, 224- and 256-bit OPFs.

Operation	160 bits	192 bits	224 bits	256 bits
Scalar mul. TE curve	2767454	4412519	6603888	9420788
Scalar mul. WEI curve (R.Int)	7384579	N/A	N/A	N/A
ECDH	9044084	14377068	21460334	30539566

The ECDH protocol is implemented to $2.76 \cdot 10^6$ clock series over a 160-bit OPF, which contains Edwards-to-Montgomery alteration. MoTE-ECC utilizes Montgomery curve with an execution time of $6.27 \cdot 10^6$ cycles over 160-bit OPF. The complete computation cost of an ephemeral ECDH key exchange amounts to about $9.04 \cdot 10^6$ clock cycles while utilizing a 160-bit OPF, with an implementation time of 1.22 s at 7.37 MHz.

9. ENERGY CONSUMPTION AND PERFORMANCE

MoTE-ECC, the slightest ECC is executed in favour of Memsic's MICAz motes as well as other 8-bit AVR-established sensor nodes. Energy is the highest valuable source for battery-driven detector nodes. Thus, it is essential to maximize the achievement of ECC application due to the reason the energy utilization of scalar multiplication increases consecutively over the implementation time. Based on [37], the ATmega128 processor of a MICAz mote relies on a medium current of 8 mA (at a delivered voltage of 3.0 V) while it is operating. Due to the reason that the clock rate for a mote is familiar to be 7.3728 MHz, we may obtain the energy utilization of single scalar multiplication through the execution of a basic computation like $W = U \cdot I \cdot t$, thus U implies the delivered power (i.e., 3 V while utilizing two traditional 1.5 V AA power cells). I is the medium current worn through the processor (i.e., 8 mA in our case), and t is the performance time. In our execution, we get a medium implementation time of 2767454 clock cycles, substantiating the energy charges of computing an individual scalar multiplication figure to $W_c = U \cdot I \cdot t = 3v \cdot 8mA \cdot (2767454/7.3728 \cdot 10^6) = 9.008mJ$. ECDH key interchange needs every node to calculate two scalar multiplications as well as to transmit a message (including the public key) to another node. Based on the energy pattern explained in [35], the energy value of sending an agreement message is $W_t = P \cdot t = 0.185$ mJ. Thus, the entire energy expenditure of ECDH key interchange is mainly rigid by the computation energy W_c for two scalar multiplications; the correspondence energy W_t is basically insignificant. Complete energy utilization to achieve an ECDH key interchange is $W = 2 \cdot W_c + W_t = 18.20$ mJ for each node. Piotrowski et al. declare in [36] that the assessed capability of a 1.5 V AA alkaline power cell is around 2500 mAh and two AA batteries may technically release an energy of 21600 Ws. The node energized by two AA alkaline power cells utilizes just 31.25 % of the entire ability. ECDH key interchanges can execute prior to the delivered voltage of the MICAz mote falling below 2.7 V.

Table 5. Energy utilization of TE curve, WEI curve and ECDH over 160-, 192-, 224- and 256-bit OPFs.

Operation	160 bits	192 bits	224 bits	256 bits
TE Curve	9.00 mJ	14.36 mJ	21.49 mJ	30.66 mJ
WEI Curve	24.03 mJ	N/A	N/A	N/A
ECDH	18.20 mJ	28.91 mJ	43.17 mJ	61.51 mJ

Beyond achievement, execution-time memory utilization is a significant factor for WSN utilization, since a standard AVR-supported detector node characterizes merely 4 kB RAM. Our comb technique in favour of scalar multiplication on a twisted Edwards curve needs to reserve eight points provided in enlarged affine coordinates. Like that, we merely require to shift the point which is needed for the present replication of the comb technique from ROM or flash memory to RAM. Our selection of $w = 4$ with eight previously assessed points illustrates a fair trade-off between performance and code size. The total ROM/flash footprint of MoTE-ECC supporting Montgomery as well as twisted Edwards curves is 14.7 kB, which establishes about 11.5% of 128 kB flash storage which exists on an ordinary AVR-deployed sensor node.

9.1 Signature Verification

The speed of Edwards-curve summation, particularly through -1 twist, makes such methods especially proficient. The prime $q = 2^{255} - 19$ is identical to 5 modulo 8; therefore, every square $\alpha \in F_q$ fulfils $\alpha^2 = \beta^4$, whereby $\beta = \alpha^{(q+3)/8}$, i.e.,$\pm\alpha = \beta^2$. The regular assessment is an individual elaboration to measure β, pursued by a fast propagation of β by $\sqrt{-1}$ if $\beta^2 = -\alpha$. In expansion, α is a percentage u/v, while $u = y^2 - 1$ and $v = dy^2 + 1$. Beginning from u and v requires only some multiplication compared with single exponent.

9.2 Defeating SPA Attacks

Within a model, SPA attacks attempt first to identify the energy utilization of a series of commands performed on a tool identical to the aimed tool. A determined couple (key, data) is refined and replicates for such various pairs oppose a single track acquired from the objective (pattern identical stage). A greatly routine execution of the comb technique for permanent-base scalar multiplication is utilized to diminish the SPA-leakage.

9.3 Fast Scalar Multiplication

To obtain the finest speed, we presume that $(a = -1)$. An n-bit scalar multiplication contains of absolutely $d = \lceil n/w \rceil$ point doublings as well as effectively d point additions. Therefore, w-bit comb technique reduces the amount of point doublings by an element of w in contrast to the binary method in spite of that the respective w bits of k are all 0. We can estimate the value of $\varepsilon^e \leftarrow 2\varepsilon$ as $3M + 4S$ by pressing an additional multiplication to the function measure of $\varepsilon \leftarrow \varepsilon^e + \varepsilon^e$.

10. CONCLUSION

This work suggested a novel ECDH key exchange technique through little energy utilization. First, we select ED25519's extended coordinates through point addition as well as point-doubling algorithm. Second, we determine radix-2^4 in favour of optimal execution through a Fermat-established inversion that is powerful over SPA attacks. Third, we represent the achievement of ECDH key interchange by combining Montgomery as well as twisted Edwards curves. Fourth, we compute and illustrate energy exhaustion of TE Curve, ECDH and contrast it with WEI curve. MoTE-ECC in favour of Memsic's MICAz motes is utilized for quick ECDH key interchange on 160-bit OPF and RAM footmark of OPF 256 bits is 556 bytes. We obtained the implementation time by joining quick $\mathbb{F}_p$ mathematically (grateful to utilizing an OPF) with extremely competent group mathematical twisted Edwards curve. MoTE-ECC assists 160-bit OPF with merely 380 bytes in RAM. We applied multiplication as well as other arithmetic functions required for ECC in a framework model to obtain great expandability and little code size. In favour of additional effective field arithmetics, we utilize quick point addition/doubling equation of twisted Edwards curves. An ECDH key interchange needs just one-third of energy of the ECDH execution. In the future, we will extend our work through Montgomery and Edwards curves for secure monitoring of low-power wireless devices by leveraging the security modules, like ECDH and ECDSA, which will support more ECC security features and will perform better than the existing curves.

ACKNOWLEDGEMENTS

This research is partially supported by the South African National Research Foundation (Grant Nos. 137951 and 132797), South African National Research Foundation incentive grant (No. 114911), and South African Eskom Tertiary Education Support Programme.

REFERENCES

[1] Rehana, Jinat, "Security of Wireless Sensor Network," Seminar on Internetworking, [Online], Available: http://www.cse.tkk.fi/en/publications/B/5/papers/Rehana_final.pdf, 2009.

[2] D. J. Bernstein and T. Lange, "Faster Addition and Doubling on Elliptic Curves," Proc. of the International Conference on the Theory and Application of Cryptology and Information Security, pp. 29-50, Springer, Berlin, Heidelberg, December 2007.

[3] D. J. Bernstein, P. Birkner, M. Joye, T. Lange and C. Peters, "Twisted Edwards Curves," Proc. of the International Conference on Cryptology in Africa, pp. 389-405, Springer, Berlin, Heidelberg, June 2008.

[4] P. L. Montgomery, "Speeding the Pollard and Elliptic Curve Methods of Factorization," Mathematics of Computation, vol. 48, no. 177, pp. 243-264, 1987.

[5] A. Verri Lucca, G. A. Mariano Sborz, V. R. Quietinho Leithardt et al., "A Review of Techniques for Implementing Elliptic Curve Point Multiplication on Hardware," Journal of Sensor and Actuator Networks, vol. 10, no. 1, pp. 3-17, 2021.

[6] D. J. Bernstein, T. Lange and R. R. Farashahi, "Binary Edwards Curves," Proc. of the International Workshop on Cryptographic Hardware and Embedded Systems, pp.244-265, Springer, Berlin, Heidelberg, August 2008.

[7] O. Reyad, M. Karar and K. Hamed, "Random Bit Generator Mechanism Based on Elliptic Curves and Secure Hash Function," Proc. of the IEEE International Conference on Advances in the Emerging Computing Technologies (AECT), pp. 1-6, arViv:2002.09239, 2020.

[8] Brainpool, "ECC Brainpool Standard Curves and Curve Generation," v. 1.0, [Online], Available: https://www.teletrust.de/fileadmin/files/oid/oid_ECC-Brainpool-Standard-curves-V1.pdf, October 2005.

[9] D. J. Bernstein, "Curve25519: New Diffie-Hellman Speed Records," Proc. of the International Workshop on Public Key Cryptography, pp. 207-228, Springer, Berlin, Heidelberg, April 2006.

[10] P. Sasdrich and T. Güneysu, "Efficient Elliptic-curve Cryptography Using Curve25519 on Reconfigurable Devices," Proc. of the International Symposium on Applied Reconfigurable Computing, pp. 25-36, DOI:10.1007/978-3-319-05960-0_3, Springer, Cham, April 2014.

[11] P. Koppermann, F. De Santis, J. Heyszl and G. Sigl, "X25519 Hardware Implementation for Low-latency Applications," Proc. of the IEEE Euromicro Conference on Digital System Design (DSD), pp. 99-106, Limassol, Cyprus, August 2016.

[12] P. Koppermann, F. De Santis, J. Heyszl and G. Sigl, "Low-latency X25519 Hardware Implementation: Breaking the 100 Microseconds Barrier," Microprocessors and Microsystems, vol. 52, pp. 491-497, 2017.

[13] F. Turan and I. Verbauwhede, "Compact and Flexible FPGA Implementation of Ed25519 and X25519," ACM Transactions on Embedded Computing Systems (TECS), vol. 18, no. 3, pp. 1-21, 2019.

[14] T. Schütze, "Automotive Security: Cryptography for Car2X Communication," Proc. of Embedded World Conference, vol. 3, pp. 4-24, Nürnberg, Germany, March 2011.

[15] D. Hankerson, A. J. Menezes and S. Vanstone, Guide to Elliptic Curve Cryptography, ISBN: 978-0-387-21846-5, Springer Science & Business Media, 2006.

[16] D. J. Bernstein and T. Lange, "SafeCurves: Choosing Safe curves for Elliptic-curve Cryptography," [Online], available: https://cr.yp.to/talks/2014.01.18/slides-dan+tanja-20140118-a4.pdf, 9 April 2019.

[17] V. Bunimov and M. Schimmler, "Area and Time Efficient Modular Multiplication of Large Integers," Proc. of the IEEE International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp. 400-409, The Hague, Netherlands, 2003.

[18] N. Takagi and S. Yajima, "Modular Multiplication Hardware Algorithms with a Redundant Representation and their Application to RSA Cryptosystem," IEEE Transactions on Computers, vol. 7, pp. 887-891, 1992.

[19] M. A. Nassar and L. A. El-Sayed, "Efficient Interleaved Modular Multiplication Based on Sign Detection," Proc. of the IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), pp. 1-5, Marrakech, Morocco, 2015.

[20] M. Scott, "Backdoors in NIST Elliptic Curves," MIRACL, [Online], Available: https://miracl.com/blog/backdoors-in-nist-elliptic-curves/, 2013.

[21] M. Nystrom, "Last Call Review of draft-ietf-curdle-dnskey-eddsa-02," [Online], Available: https://datatracker.ietf.org/doc/review-ietf-curdle-dnskey-eddsa-02-secdir-lc-nystrom-2016-12-15/, 2016.

[22] C. P. García and B. B. Brumley, "Constant-time Callees with Variable-time Callers," Proc. of the 26th USENIX Security Symposium (USENIX Security 17), pp. 83-98, 2017.

[23] D. J. Bernstein, N. Duif, T. Lange, P. Schwabe and B.-Y. Yang, "High-speed High-security Signatures," Journal of Cryptographic Engineering, vol. 2, pp. 77-89, 2012.

[24] P. Gupta and V. Shmatikov, "Security Analysis of Voice-over-IP Protocols," Proc. of the 20th IEEE Computer Security Foundations Symposium (CSF'07), pp. 49-63, Venice, Italy, July 2007.

[25] Atmel, "8-bit ARV Microcontroller with 128K Bytes In-System Programmable Flash: ATmega128, ATmega128L, Datasheet," [Online], Available: https://datasheet.ciiva.com/26814/atmega128l-8au-26814613.pdf, June 2008.

[26] A. Liu and P. Ning, "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," Proc. of the 7th IEEE International Conference on Information Processing in Sensor Networks (IPSN 2008), IEEE Computer Society Press, pp. 245-256, St. Louis, MO, USA, 2008.

[27] P. Szczechowiak, L. B. Oliveira, M. Scott, M. Collier and R. Dahab, "NanoECC: Testing the Limits of Elliptic Curve Cryptography in Sensor Networks," Proc. of European Conference on Wireless Sensor Networks (EWSN 2008), Part of the Lecture Notes in Computer Science Book Series, vol. 4913, pp. 305-320, 2008.

[28] J. Großschädl, M. Hudler, M. Koschuch, M. Krüger and A. Szekely, "Smart Elliptic Curve Cryptography for Smart Dust," Proc. of the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine 2010), vol. 74, pp. 623-634, Springer, Berlin, Heidelberg, 2010.

[29] H. Hisil, K. K.H. Wong, G. Carter and E. Dawson, "Twisted Edwards Curves Revisited," Proc. of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2008), vol. 5350, pp.326-343, Springer, Berlin, Heidelberg, 2008.

[30] S. D. Galbraith, , X. Lin and M. Scott, "Endomorphisms for Faster Elliptic Curve Cryptography on a Large Class of Curves," Proc. of Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 518-535, Springer, Berlin, Heidelberg, 2009.

[31] A. Faz-Hernández, P. Longa and A. H. Sánchez, "Efficient and Secure Algorithms for GLV-based Scalar Multiplication and Their Implementation on GLV-GLS Curves," Topics in Cryptology – CT-RSA 2014 Conf., pp.1-27, DOI:10.1007/978-3-319-04852-9_1, Springer, Cham, 2014.

[32] M. Hamburg, "Fast and Compact Elliptic-curve Cryptography," IACR Cryptology ePrint Archive: Report 2012/309, [Online], Available: https://ia.cr/2012/309, 2012.

[33] E. Nascimento, J. López and R. Dahab, "Efficient and Secure Elliptic Curve Cryptography for 8-bit AVR Microcontrollers," Proc. of the 5th International Conference on Security, Privacy, and Applied Cryptography Engineering (SPACE 2015), vol. 9354, pp. 289–309, Springer, Cham, October 2015.

[34] M. Hutter and P. Schwabe, "NaCl on 8-bit AVR Microcontrollers," Proc. of the International Conference on Cryptology in Africa, pp. 156-172, Springer, Berlin, Heidelberg, 2013.

[35] G. De Meulenaer, F. Gosset, F. X. Standaert and O. Pereira, "On the Energy Cost of Communication and Cryptography in Wireless Sensor Networks," Proc. of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, pp. 580-585, Avignon, France, October 2008.
[36] K. Piotrowski, P. Langendoerfer and S. Peter, "How Public Key Cryptography Influences Wireless Sensor Node Lifetime," Proc. of the 4th ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '06), pp. 169-176, DOI: 10.1145/1180345.1180366, October 2006.
[37] Crossbow Technology Inc., "MICAz Wireless Measurement System," Data Sheet, [Online], Available: http://courses.ece.ubc.ca/494/files/MICAz_Datasheet.pdf, April 2015.
[38] S. Ullah and R. Zahilah, "Curve25519 Based Lightweight End-to-End Encryption in Resource Constrained Autonomous 8-bit IoT Devices," Cybersecurity, vol. 4, no. 1, pp. 1-13, 2021.
[39] Z. Liu, E. Wenger and J. Großschädl, "MoTE-ECC: Energy-scalable Elliptic Curve Cryptography for Wireless Sensor Networks," Proc. of International Conference on Applied Cryptography and Network Security (ACNS), Lecture Notes in Computer Sciences, vol. 8479, pp. 361-379, 2014.
[40] Z. Liu, J. Weng, Z. Hu and H. Seo, "Efficient Elliptic Curve Cryptography for Embedded Devices," ACM Transactions on Embedded Computing Systems (TECS), vol. 16, no. 2, pp. 1-18, 2016.
[41] S. Nimbhorkar and L. Malik, "Comparative Analysis of Authenticated Key Agreement Protocols Based on Elliptic Curve Cryptography," Proc. of the International Conference on Information Security & Privacy (ICISP2015), pp. 826-827, Nagpur, India, Elsevier, December 2015.
[42] M. Elhoseny, H. Elminir, A. Riad and X. Yuan, "A Secure Data Routing Scheme for WSN Using Elliptic Curve Cryptography and Homomorphic Encryption," Journal of King Saud University-Computer and Information Sciences, vol. 28, no. 3, pp. 262-275, 2016.
[43] F. De Rango, G. Potrino, M. Tropea and P. Fazio, "Energy-aware Dynamic Internet of Things Security System Based on Elliptic Curve Cryptography and Message Queue Telemetry Transport Protocol for Mitigating Replay Attacks," Pervasive and Mobile Computing, vol. 61, pp. 101-105, 2020.
[44] M. Düll, B. Haase, G. Hinterwälder et al., "High-speed Curve25519 on 8-bit, 16-bit and 32-bit Microcontrollers," Designs, Codes and Cryptography, vol. 77, no. 2, pp. 493-514, 2015.

ملخص البحث:

تَخلق شبكات المجسّات اللاسلكية تهديداتٍ تتعلّق بالأمان، الى جانب مدى بساطة وضرورة نموذج التّشفير. وعلى الرغم من حداثة ظهورها، فإنّ الاستفادة من التّشفير باستخدام المنحنيات البيضوية موضوع بارز للبحث ومنهج يهدف الى تقليل كلفة الوقت والكثافة في تلك الشبكات. ويرتكز أمان التّشفير باستخدام المنحنيات البيضوية على صعوبة مشكلة الخوارزميات المجردة المرتبطة بالمنحنيات البيضوية المعيارية مثل: X9.62 ANSI و 186.2 FIPS NIST وغيرها. وبسبب بعض العيوب في منحنيات NIST فيما يتعلق بالأمان، فإنّ من الضروري استقصاء بدائل آمنة. وفي هذا البحث، نختار ED_{25519}، وهو منحنى إدواردز (Edwards) على مستوى أمان قدرُهُ 128 بِت، ونقارنه بمنحنى وايرستراس (Weierstrass). ولإتمام وظائف الحساب الميداني، نستخدم ($radius\text{-}2^4$) الذي يمثل العوامل الداخلة في الحساب مع [MoTE-ECC] كنظام تشفير على المجالات الأساسية المثالية (OPFs) بأحجام مختلفة مثل: 160 و 192 و 244 و 255 بِت. كما نستخدم نظام (ECDH) لتبديل المفاتيح بين عُقدتيْن، حيث تحتاج كلّ عُقدة الى عمليّتي ضرب بين كمّيات غير متجهة لكي يتمّ تنفيذ العملية المتعلقة بتبديل المفاتيح. وتَستخدم عملية ضرب الكميات غير المتجهة مُنحنى إدواردز المُزاح وتقنية المشط لإنشاء نقطة الأساس، وتَستغل المحاور ذات الإسقاط الممتدّ لجمع النّقط. ويُظهر تطبيقنا على نظام (ECDH) استهلاكاً للطاقة قدرُهُ 18.20 ملي جول (mJ) عند استخدام مجالٍ أساسي مثالي بحجم 160 بِت. وهذا أفضل من عُقد المجسّات المستندة الى نظام (AVR). وتجدر الإشارة الى أن إيجابيات الطّريقة المقترحة تحقّق درجةً متقدمةً من الأمان والطّاقة المستهلكة، مع التّقليل من عوائق الاتصال عبر إدارة المفاتيح.