Journal of Theoretical and Applied Information Technology
20th September 2015. Vol.79. No.2
© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ON THE USE OF OPTIMUM CURVES IN ELLIPTIC CURVE CRYPTOGRAPHY FOR WIRELESS SENSOR NETWORKS

[1]SAEID BAKHTIARI, [1]SUBARIAH IBRAHIM, [1]MAZLEENA SALLEH, [2]MAHDI SHARIFI

[1]Dept. of Computer Science,Universiti Teknologi Malaysia, Johor, Malaysia,

[2]Department of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran

E-mail: [1]bsaeid3@live.utm.my, {subariah, mazleena}@utm.my, [2]m.sharifi@iaun.ac.ir

ABSTRACT

Providing security in Wireless Sensor Networks (WSNs) is a considerable challenge due to the concomitant limitations in processing time, power, area and energy consumption in sensor nodes and the nature of wireless links. A variety of cryptographic methods are available for security establishment. Among them, Elliptic Curve Cryptography (ECC) is the best candidate to accomplish this challenge because it provides high security in spite of a smaller key size. In addition, ECC was subject to many recent studies to optimize the time needed for base point selection and point multiplication operations. In this paper, the points on elliptic curves are analyzed, and an efficient implementation of ECC base point selection is proposed. The proposed implementation can be utilized by extremely constrained devices. We prefer to utilize projective coordinate representations of field elements than to utilize affine coordinates over prime finite field F_p, whereas projective coordinates reduce computational complexity by eliminating the multiplicative inverse in point additions and point doubling. This paper further analyzes non-prime order elliptic curves .The analysis results show that the order of the elliptic curve plays a critical role in determining how fast a base point can be selected.

Keywords: *Elliptic Curve Cryptography, Wireless Sensor, Prime Order, Adhoc Networks, Base Point*

1. INTRODUCTION

A wireless sensor network (WSN) comprises a large number of sensor nodes that are designed for data gathering and propagation in areas that do not lend themselves to ordinary networks due to environmental and/or strategic reasons [1]. A WSN can be applied to a wide spectrum of applications, varying from critical military surveillance applications to evaluating forest fire progress and building security monitoring in the immediate future. To cover such vast fields in which the operational conditions are predominately harsh or even adverse, an abundance of sensors are arranged in these networks. The number of sensors deployed in the aforementioned networks is sufficiently large to enable the network to monitor these vast fields. Deployment in remote places and being left unattended predispose the networks to attacks, including node capture, physical tampering, eavesdropping and denial of service; in this regard, they should correspondingly be provided with highly defensive security mechanisms. Unfortunately, security requirements of resource constricted sensor nodes cannot be assured by classic security mechanisms with high overhead. WSN researchers have recommended various security schemes that are desirable and practical for such resource-constrained networks.

Unlike ad hoc networks, wireless sensor networks exploit a large number of sensor nodes and cover a widespread area, whereas they have inconstant topology due to failure or mobility [2]. Figure 1 demonstrates the architecture of wireless sensor networks and the various constituents of a sensor node. The small circles and red-filled circles are the sensor nodes and the gateways, respectively. Each individual node is qualified to collect data from the environment, execute some computations over its inputs and share the information with other nodes on the network. A network can continue operating by deploying a sensor node just before the battery is depleted of sufficient power. WSNs are surrounded by a diversity of limitations, including energy restrictions, memory limitations, unreliability and high latency in communication.

In wired data networks, a centralized control manipulates the nodes to establish a secure and reliable communication [2]; however, no recognized system exerts its authority on WSN nodes due to their limitations. Therefore, gaining a

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

knowledgeable selection on the proper cryptographic algorithms is substantial.

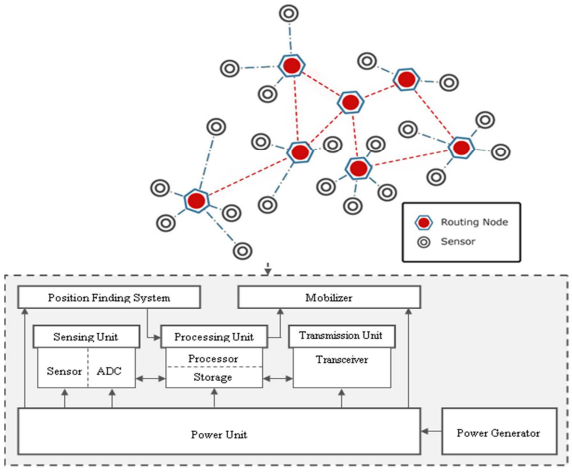

Figure 1. The Architecture Of Wireless Sensor Networks

Elliptic Curve Cryptography (ECC), proposed by Koblitz [3], has been employed in many applications recently because it offers numerous advantages over traditional public key cryptography schemes. Above all (advantages), ECC can provide higher security for equivalent key size in comparison to current asymmetric cryptosystems [4].

The security level in elliptic curve cryptosystems is based on the difficulty of the Elliptic Curve Discrete Logarithm Problem (ECDLP) and secure base point selection. To achieve a time optimization in constrained devices, such as WSNs, radio frequency identification (RFID) and mobiles, researchers have mainly focused on speeding up the following stages:

1) Base point selection
2) Scalar point multiplication

Base point selection is a leading determinant in ECC security level; concomitantly, ECDLP is the other factor to be addressed. Additionally, scalar point multiplication, which is labeled as an underlying operation in ECC, can be performed by finite field arithmetic computations, such as field addition, field multiplication, field squaring, and multiplicative inverse [5-15].

In this paper, we analyze points on elliptic curves considering different features. We also propose an efficient condition to remove the most complicated step of base point selection algorithm known as scalar multiplication; thereby the processing time of ECC base point selection is reduced in WSNs and extremely constrained secure applications .First, related works and the security requirements in wireless sensor networks are explained in Section 2 and 3. Then, an overview of elliptic curve cryptography with the advantage of projective coordinates is described in Section 4. Section 5 is dedicated to base point selection algorithms description. To achieve low computational complexity for base point selection, in Section 6, orders of points on non-prime order elliptic curves are analyzed, and an ultimate relationship between secure points and orders of elliptic curves is obtained. In Section 7, a mathematical deductive argument is provided to prove the achieved relationship. Following this, the efficiency comparison is presented in Section 8. Finally, the paper is concluded in Section 9.

2. RELATED WORKS

Basically, wireless sensor networks are numerously distributed around the monitored areas which are usually far away from human residential areas, e.g. dense forests. The information transmitted between the sensors can be readily intercepted by an intruder. In order to make the information inaccessible to the intruder, WSNs must deploy a highly secure cryptographic algorithm; this algorithm should be asymmetric to overcome the problem of key exchange as well. A number of publications [16-25] have employed such cryptographic algorithms. However, applicability and optimization of these algorithms are still key issues in WSN security since no efficient implementation of asymmetric algorithms on WSNs exists.

Elliptic curve cryptography is the latest asymmetric cryptographic algorithm providing high security level for authentication and encryption. An explanation as to how this cryptosystem meet the security requirements is given in [26]. It also provided an evaluation of ECC efficiency in discovering wormholes in mobile ad-hoc networks. The speed and security level in ECC cryptosystems primarily depends on selecting an appropriate set of public key parameters.

The bulk of research has been devoted to ECC implementation on various AVRs aiming at the performance improvement. The performed implementations differ in the characteristic features including finite field (binary-extension field or modular prime field), elliptic curve group formula, point representation (affine coordinate or projective coordinate), the technique of multiplication (group and fields of arithmetic), and the device type

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(ATmega128, ATmega256 or ATmega328). This diversity therefore, makes a comparison difficult between those researches.

In 2003, ECC was implemented on sensor networks [18, 19], but the employed hardware was almost strong including 16 bit microcontroller with 16 MHz clock frequency. In this way, the results are confined to hardwares with high computational resources while WSNs do not utilize strong CPUs with high clock frequency to avoid exorbitant cost and energy consumption for the network. Kummar [20] used a microcontroller 8051 with the efficiency of 24 MIPS which was three times as fast as ATmega 128. His implementation was performed on a particular finite field which leads to fast computation of scalar multiplication. However, the protection of this particular finite field is uncertain against the Weil Descent Attack and has not been considered in the proposed work.

Gura et al. [27], in 2004, demonstrated the implementation of ECC on ATmega128 using NIST standardized elliptic curves with sizes of 160, 192 and 224 bits. However, the implementation consumes a considerable time to compute scalar multiplication. For instance, it needs 17.52 million clock cycles on a 224-bit curve. Uhsadel et al. [28] implemented ECC on 8-bit AVR using the same device as Gura's implementation in 2007. Their implementation requires 10 million clock cycles for a 160-bit elliptic curve. The proposed implementation is still unsatisfactory, although it is faster than Gura's work.

ATmega128 chips have limited capabilities causing ECC-signature generation takes more than 70 seconds [22, 23]. Considering these constraints, application of elliptic curves with particular characteristics can lead to a faster signature generation. Recently, Chu et al. [29] utilized a particular group of curves called "Twisted Edward curves" in 2013. It leads to an efficient implementation that requires 5.9 million clock cycles for 160-bit curves on ATmega128. However, they used data-dependent branch conditions which are extremely weak against a group of attacks such as timing analysis, fault analysis, differential power analysis and electromagnetic analysis attacks [30].

3. SECURITY REQUIREMENTS IN WIRELESS SENSOR NETWORKS

A large number of vulnerabilities, which are attributed to wireless sensor networks and other sensor network applications, pose several threats to a WSN protocol and thus make security into a critical issue for these networks. Intrusion, interception, modification and fabrication are considered among the most important instances of the aforementioned vulnerabilities [31]. Conceptually, the threats can be listed from diverse aspects. In previous research [28, 32-50], the threats have been listed accordance with the method employed to achieve attacks, the layer of the communication stack on which they are recognized and, finally, whether the malevolent node joins the network whilst the attack is in progress. Security issues in WSNs are categorized as follows: survey, cryptography, secure routing, key management, data aggregation, location aware security and attacks, as demonstrated in Figure 2.

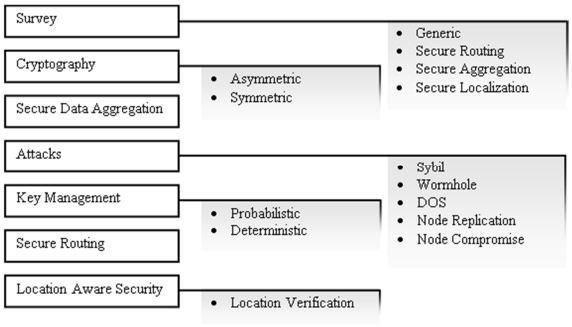

Figure 2. Subcategorized and categorized security areas in WSNs

Different subcategories for security issues in WSNs are illustrated in Figure 2. Security requirements in WSN [51, 52] are divided into the following categories:

1) Data Confidentiality
2) Data Authentication
3) Data integrity
4) Data Freshness
5) Data Availability
6) Time synchronization
7) Secure Group management
8) Secure localization

Consequently, the security issues in wireless sensor networks, which are divided into seven categories, not only control the network operation but also guarantee the availability of the whole network. The wireless environment, as an unguided medium, is more vulnerable to threats and attacks in comparison to wired networks. Deployment of security measures in wireless sensor networks is easier than wireless ad hoc networks because of their architectural aspects, namely centralized base

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

stations or sinks [53]. The number of attacks in WSNs can be considerable, in spite of the security and routing mechanisms.

This paper addresses confidentiality and authentication. To provide these requirements, cryptography is the most common mechanism that has been designed thus far. A variety of cryptographic approaches have been considered, depending on symmetric and asymmetric algorithms. Symmetric cryptographic algorithms efficiently provide (the network with) confidentiality and satisfy the power, space and memory requirements of WSN [54]. However, authenticity and proper key exchange mechanisms are best accomplished by asymmetric algorithms. Asymmetric key algorithms are most commonly used in end devices such as mobile devices, smart cards and servers. There are many asymmetric key algorithms, such as Diffie-Hellman (DH) key exchange, ElGamal Elliptic Curve, ECC, Number Theory Research Unit (NTRU) and Rivest-Shamim-Adleman (RSA).

Elliptic curve cryptography is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields [55]. ECC is a method based on the discrete logarithm problem over the points on an elliptic curve. It is important to consider that ECC is based on Discrete Logarithm Problem (DLP) [56]. Currently, elliptic curve cryptography is used in three different areas in the science of cryptography: key agreement, encryption scheme and digital signature. ECC delivers the highest strength-per-bit of any public-key cryptography [57]. Recently, NIST and many experts have recommended ECC to be used in Secure Socket Layer (SSL) and Transport Layer Security (TLS) protocols [58]. Furthermore, ECC is especially well-suited for constrained environments, such as end devices, because it provides low process time, low storage space, low bandwidth and low power consumption. As part of the effort to promote the widespread use of ECC, Sun Microsystems has donated ECC code to OpenSSL and Network Security Services (NSS) library; this brings ECC to the Apache web server and Mozilla browsers and many other products.

4. OVERVIEW OF ELLIPTIC CURVE CRYPTOGRAPHY

The idea of the deployment of elliptic curves based on finite fields for cryptosystems was first proposed by Kobitz [3]. Occasionally, elliptic curves can be defined on any desirable type of field, for instance, real numbers, rational numbers and complex numbers. However, elliptic curves used for cryptographic purposes are generally defined over finite fields; these fields include a finite number of elements, as their name implies, and are commonly referred to "Galois fields". In the case of cryptosystems that are implemented by elliptic curves, the feasibility, cost and speed of the system is determined mainly by their finite fieldF_q, where $q = p^m$. Moreover, finite fields proposed for elliptic curves that are used as cryptosystems are further classified as two types:

1) Prime finite field $F_p$where p is an odd prime number larger than 3.
2) Binary finite field $GF(2^m)$ or F_{2^m}.

An elliptic curve $E(F_p)$ consists of elements (x, y) and coefficients that satisfy Equation (1).

$$E \to E_{(a,b)} \coloneqq \{x, y, a, b \in F_p = \{1,2,3, \dots, p-1\}; 4a^3 + 27b^2 (mod\ p) \neq 0 | y^2 = x^3 + ax + b\} \quad (1)$$

With constants $a = 0$ and $b = 0$, it is noticeably easier to solve the DLP for the respective curve. These parameter choices are cryptographically weak, and as a result, are vulnerable to attack. To prevent various attacks, such anomalous curve attacks, Weil and Tate pairing attacks, Weil Descent, invalid curve attacks and small subgroup attacks, parameters selection has been a prolific area of ECC research for the last 25 years [56-58]. For each value of x, one needs to prove whether it is a quadratic residue. If it is a quadratic residue, then a couple of values in the elliptic group can be considered for y. Otherwise, the point is not contained in the elliptic group $E_p(a, b)$. The number of points on the elliptic curve $E_p(a, b)$ is represented by the order of elliptic curve, which is denoted by $\#E(F_p)$. By considering that at least 50% of numbers modulo p are quadratic residues, the number of points is predicted to be approximately$p + 1$. Additionally, the precise bound for the order of the elliptic curve has been suggested by Helmut Hasse, and it is known as Hassse's theorem [59]. Equation (2) represents that the order of an elliptic curve satisfies $\#E(F_p) \leq p + 1 - t$ where $|t| \leq 2\sqrt{p}$; t is named as the trace of E.

$$p + 1 - 2\sqrt{p} \leq \#E(Fp) \leq p + 1 + 2\sqrt{p} \quad (2)$$

According to this theorem, there are approximately p points with error bounded that are the sum of two complex numbers by $O(\sqrt{p})$. The order of the group is established to all parties; one can generate a curve randomly and count its order

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

by Schoof's algorithm. This algorithm has methodologies in ECC, especially when it is valuable to judge the difficulty of solving the discrete logarithm problem in the group of points on an elliptic curve, in accordance to the number of points. The total complexity of Schoof's algorithm is $O(Log^8 q)$[60].

Conceptually, a finite field is composed of field elements (finite set of objects) and addition and multiplication operations that can be performed on (x, y) field elements. Addition and multiplication operations in elliptic curve cryptography can be considered as equivalent operations to modular multiplication in common public key cryptosystems, and likewise multiple additions are comparable to modular exponentiation. Section 3.1 and 3.2 are dedicated to describing the point addition and scalar multiplication operations and also the arithmetic analysis of proper operations.

4.1 Sections and Subsections

Fundamentally, one of the basic conditions that should be satisfied by any cryptosystem is that the system should be closed. This implies that any operation on any element of the system has a direct effect on another element of the system [61]. In this regard, non-canonical addition and multiplication operations should be created.

Two types of point representations introduced in elliptic curve cryptography are affine coordinates and projective coordinates. Affine coordinate systems are less complicated because they use the communication between two parties requiring the lowest bandwidth. An affine point on an elliptic curve E can be specified by its respective pair of finite field elements (x, y) known as the affine coordinates for the point. No affine representation is allocated to the point at infinity $\mathcal{O}$[62]. The points located on the curve are represented by upper cases, whereas their lower cases illustrate integers. The addition of two points on the curve generates another point that lies on the curve. This operation is known as point doubling when the points are equivalent. This procedure is explored as pseudo-code in Figure 3.

Affine coordinate prompt division in every addition and every doubling operation requires fewer multiplications in comparison to Jacobian projective coordinates. Briefly, affine coordinates are unfavorable with modular inversion arithmetic operation, whereas Jacobian projective coordinates do not require division in either addition or doubling operations, and merely a single division is executed in the last stage of the elliptic curve exponentiation computation. Because the ratio of the computation amount of division in F_p to that of multiplication in F_p is generally larger than 9, in this particular case, the computation of the elliptic curve exponentiation can be executed faster in Jacobian projective coordinates than in affine coordinates. Here the so-called addition formula in Jacobian projective coordinates is proposed. Let an elliptic curve over F_p where $p > 3$ be as below:

$$E: y^2 = x^3 + ax + b \ (a, b \in F_p \ , 4a^3 + 27b^2 \neq 0)$$

Input: $P_1(x_1, y_1), P_2(x_2, y_2)$
Output: $P_1 + P_2 = P_3(x_3, y_3)$
if ($P_1 == P_2$) then

(Point Addition)
$$\begin{cases} \lambda = \frac{y_2 - y_1}{x_2 - x_1} \\ x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}$$

else

(Point Doubling)
$$\begin{cases} \lambda = \frac{3x_1^2 + a}{2y_1} \\ x_3 = \lambda^2 - 2x_1 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}$$

Figure 3. Point addition pseudo-code

For the elliptic curve, the Jacobian coordinate sets $x = \frac{X}{Z^2}$ and $y = \frac{Y}{Z^3}$, *i.e.*

$$E: Y^2 = X^3 + aXZ^4 + bZ^6 \qquad (3)$$

the addition formulae in Jacobian coordinates is as follows: Let $P = (X_1, Y_1, Z_1)$, $Q = (X_2, Y_2, Z_2)$ and $P + Q = R = (X_3, Y_3, Z_3)$. For point addition of two points in projective coordinates, let $(X_1, Y_1, Z_1) + (X_2, Y_2, 1) = (X_3, Y_3, Z_3)$ then:

$$X_3 = D^2 - (C^3 + 2X_1C^2), Y_3 = D.(X_1C^2 - X_3) - Y_1C^3, Z_3 = Z_1C$$

where $A = X_2Z_1^2$, $B = Y_2Z_1^3$, $C = A - X_1$ and $D = B - Y_1$. For point doubling in Jacobian projective coordinates, let $2(X_1, Y_1, Z_1) = (X_3, Y_3, Z_3)$ then:

$$X_3 = D, Y_3 = C(A - D) - B, Z_3 = 2Y_1.Z_1$$

where $A = 4X_1 + Y_1^2$, $B = 8Y_1^4$, $C = 3(X_1 - Z_1^2)(X_1 + Z_1^2)$ and $D = -2A + C^2$.

The scalar multiplication of a point P by a natural integer k is denoted by $[k]P$. Therefore, kP can be calculated using Equation (4).

$$Q = KP = \underbrace{P + P + \cdots . + P}_{K-Summands} \qquad (4)$$

Here, the point P is an established point that gives rise to an extended prime subgroup of $(F_p).P$ can also be a consummate member in this subgroup.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Suppose that n is the order of elliptic curve $E(F_p)$. Thus, the constant k is an integer that exists in the $[1, n-1]$ interval. A large number of cryptographic protocols have been designated, in which security relies on the hardness of computing the discrete logarithm over the rational points of an elliptic curve, such as ECDH, ECDSA, ECAES and ECEIGamal. In general, the only algorithms known to solve this problem are exponential-time algorithms, and among various computations of all protocols that are based on elliptic curves, the scalar multiplications are mostly responsible for consuming a lot of CPU time [63]. The scalar multiplication is known as the underlying operation of most of these protocols. Efficient scalar multiplication arithmetic is hence considered as a critical issue in cryptography. Fortunately, some characteristics of elliptic curves allow optimization of scalar multiplication. The interested reader is referred to [64] for a good overview of the question.

The inquiry to discover a scalar multiplication algorithm from point addition and point doubling operations is comparable to computing an exponentiation from multiplications and squares. In the context of elliptic curves, binary exponentiation or binary scalar multiplication should be taken into account as an efficient and straightforward algorithm, which is also called the square-and-multiply algorithm or double-and-add algorithm. The binary algorithm performs the process of a loop that scans the bits of the scalar and then executes a point doubling, and whenever the current scalar equals 1, the aforementioned operations are followed by a point addition.

Let k be an integer with binary expansion$(k_{n-1}, \dots, k_1, k_0)_2$, that is to say, $k = \sum_i k_i 2^i$, where $k_i \in \{0,1\}$ for every $i < n-1$ and $k_{n-1} = 1$. The binary scalar multiplication of some point P by k can be perceived as follows. Defining $T_i = [(k_{n-1}, \dots, k_1, k_0)_2]P$, we get a backward sequence where $T_{n-1} = P$, $T_0 = [k]P$ and $T_i = 2T_{i+1} + k_i P$, which is a binary algorithm. (see Algorithm 1).

Algorithm 1. Binary Algorithm

Input: $P \in E(F_p), k = (k_{n-1}, \dots, k_1, k_0)_2 \in \mathbb{N}$
Output: $Q = [k]P$
$R_0 \leftarrow P; R_1 \leftarrow P$
For $i = n-2$ downto 0 do
$R_0 \leftarrow 2R_0$
If $k_i = 1$ then $R_0 \leftarrow R_0 + R_1$
End for
Return R_0

The preceding binary algorithms are efficacious and uncomplicated; however they are not safe in a condition where the scalar is secret and where the implementation is subject to Side-Channel Analysis (SCA) (e.g., a WSN that performs an ECDSA signature). Simple Power Analysis (SPA) can retrieve the secret scalar from a single leakage trace of a binary algorithm computation, even in the existence of data randomization. To oppose SPA successfully, Coron [65] suggested performing a dummy addition in the binary algorithm loop whenever the scalar bit is equal to 0.

Further regular binary algorithms are introduced in the literature and involve attractive features, such as the Montgomery ladder [66]. This algorithm depends on loop invariants and the point registers R_0 and R_1. In the Montgomery ladder, the relation $R_1 - R_0 = P$ is fulfilled at the last stage of every loop iteration (see also [66] for further details).

Algorithm 2. Montgomery Ladder

Input: $P \in E(F_p), k = (k_{n-1}, \dots, k_1, k_0)_2 \in \mathbb{N}$
Output: $Q = [k]P$
set $P_1 \leftarrow P$, $P_2 \leftarrow 2P$
for $i = t-2$ downto 0 do
if $k_i = 1$ then
set $P_1 \leftarrow P_1 + P_2, P_2 \leftarrow 2P_2$;
else
set $P_2 \leftarrow P_1 + P_2, P_1 \leftarrow 2P_1$;
end
End for
$Q = P_1$

Algorithm 3. Montgomery Scalar Multiplication Algorithm using Projective Coordinates

Input: $P \in E(F_p), k = (k_{n-1}, \dots, k_1, k_0)_2 \in \mathbb{N}$
Output: $Q = [k]P$
If $k = 0$ or $x = 0$ then
Output (0,0) and stop;
End
Set $X_1 \leftarrow x, Z\_1 \leftarrow 1, X_2 \leftarrow x_4 + b, Z_2 \leftarrow x^2$;
For $i = t-2$ downto 0 do
If $k_i = 1$ then
$z_3 \leftarrow (x_1.Z_2 + X_2.Z_1)^2, X_3 \leftarrow x.Z_3 + (X_1.Z_2).(X_2.Z_1)$
$Z_4 \leftarrow Z_2^2.X_2^2, X_4 \leftarrow X_2^4 + b.Z_2^4$;
$Z_1 \leftarrow Z_3, X_1 \leftarrow X_3, Z_2 \leftarrow Z_4, X_2 \leftarrow X_4$;
Else
$z_3 \leftarrow (x_1.Z_2 + X_2.Z_1)^2, X_3 \leftarrow x.Z_3 + (X_1.Z_2).(X_2.Z_1)$
$Z_4 \leftarrow Z_1^2.X_1^2, X_4 \leftarrow X_1^4 + b.Z_1^4$;
$Z_1 \leftarrow Z_4, X_1 \leftarrow X_4, Z_2 \leftarrow Z_3, X_2 \leftarrow X_3$;
End
End

As in GF(p), projective coordinates (X, Y, Z) are proposed to eliminate costly inversions by $x = \frac{X}{Z}$

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and $y = \frac{Y}{Z}$ from affine coordinates (x, y). The Montgomery scalar multiplication in projective coordinates is listed below in Algorithm 3.

In Section 4.2, elliptic curve cryptography operations are investigated based on their respective arithmetic. Moreover, an analysis of the point operations algorithm was conducted. Finally, a detailed explanation of the Montgomery ladder algorithm based on such arithmetic is presented.

4.2 Arithmetic Analysis

General-form elliptic curves selected arbitrarily from the set of curves that satisfy$y^2 = x^3 + ax + b$, are the only curves considered in this paper. Note that particular forms of elliptic curves (e.g., Koblitz curves and Edwards curves) are also convenient, the performance advantages of which exceed those of general-form elliptic curves.

Point addition formulae such as in Figure 3 rely on different operations over F_q (e.g., multiplication, inversion, addition, and subtraction), which are characterized by different computational expenses. In the current section, the computational costs of a field inversion, a field multiplication, a field squaring, and a field addition are denoted by I, M, S, and A, respectively. Furthermore, a field doubling and a field subtraction cost the same as a field addition. In the condition where q is a large prime number, it is often supposed that:

1) The inversion cost is I ≈ 100M
2) The squaring cost satisfies S ≈ 0.8M
3) The addition cost is preferred to be ignored.

These assumptions arise from the regular software implementations of the field operations. However, when the second one depends on a hardware coprocessor, for example, in the case of embedded systems, its costs are based on the architecture. Generally, the costs of inversion operations should always total some dozens of multiplications, the squaring should cost the same as one multiplication (possibly a bit cheaper), and the addition should cost substantially loss, but not always negligible [67]. The rest of this section involves the computational cost of different point representations in field operations terms. Furthermore, their memory usage should be taken note of, in terms of field registers, namely memory registers of size $\log_2(q)$bits that can storeF_q elements.

Algorithm 4. Jacobian Doubling				**Algorithm 5.** Jacobian Doubling $(a = -3)$			
Input: $P \equiv (X_1, Y_1, Z_1)$ Output: $2P \equiv (X_3, Y_3, Z_3)$				Input: $P \equiv (X_1, Y_1, Z_1)$ Output: $2P \equiv (X_3, Y_3, Z_3)$			
1. $T_4 \leftarrow T_1^2$	$[X_1^2]$	**10.** $T_6 \leftarrow aT_6$	$[aZ_1^4]$	**1.** $T_4 \leftarrow T_2^2$	$[Y_1^2]$	**10.** $T_3 \leftarrow T_1 + T_1$	$[2(X_1^2 - Z_1^4)]$
2. $T_5 \leftarrow T_2^2$	$[Y_1^2]$	**11.** $T_4 \leftarrow T_4 + T_6$	$[3X_1^2 + aZ_1^4]$	**2.** $T_5 \leftarrow T_1 T_4$	$X_1 Y_1^2 = A$	**11.** $T_1 \leftarrow T_1 + T_3$	$[3(X_1^2 - Z_1^4)]$
3. $T_1 \leftarrow T_1 T_5$	$X_1 Y_1^2 = A$	**12.** $T_4 \leftarrow \frac{T_4}{2}$	$\frac{(3X_1^2 + aZ_1^4)}{2} = B$	**3.** $T_4 \leftarrow T_4^2$	Y_1^4	**12.** $T_1 \leftarrow \frac{T_1}{2}$	$\frac{(X_1^2 - Z_1^4)}{2} = B$
4. $T_5 \leftarrow T_5^2$	Y_1^4	**13.** $T_6 \leftarrow T_4^2$	$[B^2]$	**4.** $T_2 \leftarrow T_2 T_3$	$Y_1 Z_1 = Z_3$	**13.** $T_3 \leftarrow T_1^2$	$[B^2]$
5. $T_6 \leftarrow T_3^2$	Z_1^2	**14.** $T_2 \leftarrow T_1 + T_1$	$[2A]$	**5.** $T_6 \leftarrow T_3^2$	Z_1^2	**14.** $T_3 \leftarrow T_3 - T_5$	$[B^2 - A]$
6. $T_6 \leftarrow T_6^2$	Z_1^4	**15.** $T_6 \leftarrow T_6 - T_2$	$[B^2 - 2A = X_3]$	**6.** $T_1 \leftarrow T_1 + T_3$	$[X_1 + Z_1^2]$	**15.** $T_3 \leftarrow T_3 - T_5$	$[B^2 - 2A = X_3]$
7. $T_3 \leftarrow T_2 T_3$	$Y_1 Z_1 = Z_3$	**16.** $T_1 \leftarrow T_1 - T_6$	$[A - X_3]$	**7.** $T_3 \leftarrow T_3 + T_3$	$[2Z_1^2]$	**16.** $T_5 \leftarrow T_5 - T_3$	$[A - X_3]$
8. $T_2 \leftarrow T_4 + T_4$	$[2X_1^2]$	**17.** $T_4 \leftarrow T_4 T_1$	$[B(A - X_3)]$	**8.** $T_3 \leftarrow T_1 - T_3$	$[X_1 - Z_1^2]$	**17.** $T_1 \leftarrow T_1 T_5$	$[B(A - X_3)]$
9. $T_4 \leftarrow T_4 + T_2$	$[3X_1^2]$	**18.** $T_1 \leftarrow T_4 - T_5$	$B(A - X_3) - Y_1^4 = Y_3$	**9.** $T_1 \leftarrow T_1 T_3$	$[X_1^2 - Z_1^4]$	**18.** $T_1 \leftarrow T_1 - T_4$	$B(A - X_3) - Y_1^4 = Y_3$

When points are revealed in affine coordinates, the addition of two points includes an expensive field inversion. Fortunately, representing points in projective coordinates is a possible approach to prevent this cost expenditure on the intermediate point additions in scalar multiplication. In projective coordinates, a point P = (x, y) is implied by a triplet (X, Y, Z), where $(x, y) = (X/Z^c, Y/Z^d)$ for some given integers c and d. The Jacobian coordinates for which $c = 2$ and $d = 3$ are proved as the most widely utilized among the projective coordinates. These coordinates enable performing a fast point doubling, which is the most frequently used operation in a scalar multiplication algorithm. Let $P = (X_1, Y_1, Z_1)$, the Jacobian doubling of P is defined as $P + P = (X_3, Y_3, Z_3)$ where:

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$X_3 = B^2 - 2A, Y_3 = B(A - X_3) - Y_1^4$ and $Z_3 = Y_1 Z_1$, (5)

with $A = X_1 Y_1^2, B = \frac{1}{2}(3X_1^2 + aZ_1^4)$.

The Jacobian doubling is indicated in Algorithm 4 for the general context and in Algorithm 5 for the case of a = −3. For the latter special case, where a = −3, we have $B = \frac{3}{2}(X_1 + Z_1^2)(X_1 - Z_1^2)$. This equality enables trading 1M+ 2S for 1M+ 1A in the computation of B. The former has a cost of 4M + 6S + 7A and uses 6 field registers, whereas the latter has a cost of 4M + 4S + 9A and uses 5 field registers.

Tetsuya Izu [68] illustrates that a loop iteration of the Montgomery ladder using (X, Z)-coordinates can only be executed in $11M + 4S + 2M_a + 18A$, where M_a denotes the cost of the multiplication by the curve parameter a (which is a small number of additions if a is small, e.g., a = −3).

5. BASE POINT SELECTION ALGORITHM

In context of the elliptic curve cryptosystems, the key security correlates with the rationality of the chosen base points [14]. Base point selection implies choosing a point with a large prime order over a given field. This point is in turn termed as a base point. This section of the paper concentrates mostly on the point choosing algorithm over (2^n). In security terms, the author preferred to adopt the non-super singular curve. Commonly, the approach to base point selection is the random point selection on the elliptic curve, the success rate of which is $\#E(GF(2^n)) / 2^{2n} \approx 2^{-n}$[14]. During the point selection procedure, n is assumed as the order of$G$, where n is a large prime number. The order of elliptic curve is denoted by $\#E(GF(2^n))$. Assume $p_{E(GF(2^n))}$is a large prime factor of $\#E(GF(2^n))$, there is a cofactor which satisfies $h = \#E(GF(2n)) / p_{E(GF(2^n))}$, where h represents a small integer. The optimum base point is achieved when $n = p_{E(GF(2^n))}$.

Definition: If $F = GF(q)$, $K = GF(q^n)$, $q = p^m$(F is the subfield of K), $\alpha \in GF(q^n)$, equation (4) is termed the trace of α, denoted by $Tr(\alpha)$.

$$\mathrm{Tr}_{k/F}(\alpha) = \alpha + \alpha^q + \cdots + \alpha^{q^{n-1}} = \sum_{i=0}^{n-1} \alpha^{q^i} \quad (6)$$

For (α), when $Tr(\alpha) = 0$, $x^2 + x = \alpha$ can be solved, and its solutions are β and $\beta + 1$, θ is the element in K, and $Tr(\theta) = 1$.

Theorem: Let α , θ be the elements in field K,

$\beta = \alpha\theta^q + (\alpha + \alpha^q)\theta^{q^2} + \cdots + (\alpha + \alpha^q + \cdots + \theta^{q^{n-1}})\theta^{q^{n-1}}$, then $\beta - \beta^q = \alpha((Tr(\theta) - \theta) - \theta(Tr(\alpha) - \alpha))$.

The selection of the base point algorithm with the customary method is given in the following succession [69]:

p_i. Randomly pick up an element θ in the field K and compute $Tr(\theta)$. If $Tr(\theta) = 1$, switch to (p_{ii}); else return (p_i) to restart;

p_{ii}. Randomly choose an element in K as x. Let A= x, $= x^3 + ax^2 + b$, and the original equation is rewritten as $y^2 + Ay = B$, let $y = Ax'$. Again, the equation will be transformed to $x'^2 + x' = B / A^2$.Let $\xi = B / A^2$, the equation will be transformed to $x'^2 + x' = \xi$;

p_{iii}. Computing$Tr(\xi)$, if $(\xi) = 0$, switch to (p_{iv}),otherwise switch to (p_{ii});

p_{iv}. Then, β and β +1 are the two solutions of the equation $x'^2 + x' = \xi$, and $y_1 = A\beta$ and $y_2 = A(\beta + 1)$, so (x, y_1) and (x, y_2) are the two points lying on the curve $y^2 + xy = x^3 + ax^2 + b$;

p_v. Randomly adopt a point from the points acquired in (p_{iv}) as P. If$p_{E(GF(2^n))}P = \mathcal{O}$, the point P is one of the base points, otherwise, switch to (p_i) to re-choose.

The preceding algorithm is not an optimal algorithm because there might be a point hP that fulfills$p_{E(GF(2^n))}(hP) = \mathcal{O}$, if $hP \in E(GF(2^m))$ and the point hP is situated on the curve E, and then the point hP is considered as one of the base points of the given curve.

When the base point was selected via the customary method [14], it is apparent that, as the source data seed was chosen at random, there is a substantial number of reiteration calculating (p_i), (p_{iii}), and (p_v), which will in turn provoke an increase in the running costs of the computing time and algorithm complexity.

5.1 YIN Algorithm

In this section an introduction to the parallelization algorithm is presented, which uses $(p \geq 1$ and $pmod\ 2 \equiv 0)$ processors; it also greatly reduces the computing time consumption in the base point selection, base point judgment and consequently enhances the time efficiency of the algorithm.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Initially, a shared area in a system will necessarily be dedicated to save the template outcomes during the interlude of the calculation, before the algorithm begins. For this, three public tables are convenient in this algorithm $Table_\theta$, $Table_P$ and $Table_G$ which are intended to save θ, the random points P and the base points G, respectively.

In the remainder of this section, the parallelization of the base point selection algorithm is explained briefly:

To p processors, each processor p_i, do:

p_i. Scan $Table_\theta$ to perceive if there is data in $Table_\theta$, and if so, switch to p_{ii}, otherwise, adopt an element θ in the field K, and compute $Tr(\theta)$. If $Tr(\theta) = 1$, put θ into $Table_\theta$, and messages the other processors that θ selection is completed so that the processors in running will not do any operations after this round; otherwise, message the other processors that this element cannot be reselected and return p_i.

After the above period is finished, all of the s processors are idle, therefore the p processors should be divided into two groups, each with m ($p = 2m$) processors. The m processors of the first group do as follows:

p_{ii}. Randomly choose an element in K as x. Message the other processors to omit x from the field, then let $A = x, B = x^3 + ax^2 + b$. The original equation is rewritten as $y^2 + Ay = B$, let $y = Ax'$. Again, the equation is transformed to $x'^2 + x' = B / A^2$,let $\xi = B / A^2$. The equation is then transformed to $x'^2 + x' = \xi$;

p_{iii}. Compute $Tr(\xi)$. If $Tr(\xi) = 0$, switch to p_{iv}, otherwise switch to p_{ii};

p_{iv}. The two solutions of the equation $x'^2 + x' = \xi$are β and β +1. Then $y_1 = A\beta$ and $y_2 = A(\beta + 1)$, thus(x, y_1) and (x, y_2) are the two points of the curve $y^2 + xy = x^3 + ax^2 + b$,and (x, y_1) and (x, y_2) are in the public table $Table_P$;

As the first groups of processors are performing the above operations, the other groups of m processors conduct the following operations:

p_v. Scan the public table $Table_P$to discern if it is vacant. If so, the processor will wait until $Table_P$is not empty. Select a point P from $Table_P$. If $p_{E(GF(2^n))}P = \mathcal{O}$, the point P is the base point. Let $G = P$, put G into the public table $Table_G$, then stop the algorithm ,messaging the other processors that the algorithm has been fulfilled; otherwise, continue on;

p_{vi}. Calculate and ascertain if $hP = \mathcal{O}$. If so ,eliminate the point P from $Table_P$; if not ,let $P = hP$, then compute if P is the point lying on the given curve $y^2 + xy = x^3 + ax^2 + b$. If so, let $G = P$, put G into the public table $Table_G$,then stop the algorithm, and message the other processors that the algorithm has been finished; if not, return p_{ii}.

In [69], p_v when P is not a an expected base point, the algorithm can still be continued to p_{vi} in the parallel algorithm, then judge again (the nature of the field computation determines p_{vi} feasibility). If P is still not the base point of the curve, we go to p_{ii}, but not [69]p_i. When the point adopted randomly is not the curve base point, it is not due to inappropriate θ, but it is due to the x. Therefore, the parallel algorithm will go to p_{ii} here, rather than p_i, which will increase the efficiency of the original algorithm by almost twice.

6. ANALYSIS ON NON-PRIME ORDER ELLIPTIC CURVES

Opting for a cryptographically secure elliptic curve in advance of the deployment of an elliptic curve cryptosystem assures the robustness of the cryptosystem against all recognized attacks which were noted in Section 3, e.g., anomalous curve attack, Weil and Tate pairing attacks, and small subgroup attack. If the order of the underlying elliptic curve possesses certain qualities, the cryptosystem can be protected against all of these attacks. Generating cryptographically secure elliptic curves over prime fields is among the most fundamental and complicated problems in elliptic curve cryptography. The approaches most frequently exploited for the generation of ECs over prime fields are the Complex Multiplication (CM) method [70] and the point counting method [71].

Let E be an elliptic curve over a finite field F_p and assume that the order of elliptic curve $\#E(F_p)$ is non-prime n.

$$\#E(F_p) = n = hr ,$$

Where r and h are the divisor of n (for cases of interest, h will be the divisor of n). The set of points on an elliptic curve constitutes a group under a particular addition rule. With this operation, the set of points on the elliptic curve is structurally an abelian group: the sum is associative and commutative, has an identity element (namely O) and every element G has an inverse $-G$ (the inverse

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

of O is itself). While working over a finite field, this group is inevitably finite (because the existing points are all finite).

All of the equations included in Table 1 have respective non-prime order ECs. Table 1 shows the factorization of the elliptic curve order $\#E(F_p)$. According to the information derived from Table 1, the set of all points order is generated via prime factorization of the positive elliptic curve order, along with their multiplicities.

Table 1 illustrates the elliptic curve characteristics, such as equation, finite field, elliptic curve order, set of all points order and its number of elements that was subject to analysis. As well, the order of the appropriate elliptic curve accompanied by its factorization is listed in the fourth column. Among the contents of the fifth column, the main divisors of elliptic curve order are bold and thus they can be readily observed. Table 1 shows the order (number of elements) of all points $\{G_1, G_2, \dots, G_n\}$ of E that divide the order of the elliptic curve $\#E(F_p)$; for instance, let $E: y^2 = x^3 + ax + b$ be elliptic curve defined over finite field p with order $\#E(F_p)$. The factorization of $\#E(F_p)$ is as follows:

$\{r_1^{w_1} \times r_2^{w_2} \times r_3^{w_3} \times \dots \times r_l^{w_l}; r_1 < r_2 < r_3 \dots < r_l,\ w \geq 1\}$.

All points order satisfying $\#E(F_p)$ have prime divisors of elliptic curve order $\{r_1, r_2, \dots, r_l\}$ or together with multiplicities. The number of divisors of $\#E(F_p)$, which are demonstrated in the sixth column of Table 1, can be obtained by Equation (7).

$$(w_1 + 1)(w_2 + 1)(w_3 + 1) \dots (w_l + 1) \quad (7)$$

Suppose that finding a number of divisors of 48 is desired. It is convenient to start with 1 and then continue by working through the set of natural numbers and testing divisibility in each case. Note that divisors can be listed in factor pairs.

$$48 = 1 \times 48 = 2 \times 24 = 3 \times 16 = 4 \times 12 = 6 \times 8$$

It is clear that 48 has exactly ten divisors. It can also be easily perceived that by utilizing this approach (to prove the divisors of a number), it is only ever required to work from 1 up to the square root of the number. This method can be quickly and easily performed with small numbers; however, it is not practical for larger numbers.

Let $\tau(n)$ be the number of divisors for the natural number, n. As an initial step the number is written as a product of prime factors: $n = p^a q^b r^c \dots$ and the number of divisors $\tau(n) = (a + 1)(b + 1)(c + 1)$.

Table 1. Non-prime order ECs and set of all points order.

No	(a, b)	Finite Field p	Elliptic curve Order $\#E(F_p)$	Set of all points order	Number of elements
1	(26,460)	12323	12315 {1,3,5,821}	{1,3,5,15,821,2463,4105, 12315}	8
2	(968,18)	971	970 {1,2,5,97}	{1,2,5,10,97,194,485,970}	8
3	(143,410)	1031	1027 {1,13,79}	{1,13,79,1027}	4
4	(274,355)	1433	1430 {1,2,5,11,13}	{1,2,5,10,11,13,22,26,55, 65,110, 130, 143, 286,715,1430}	16
5	(134,2538)	145177	145157 {1,379,383}	{1,379,383,145157}	4
6	(46521,24508)	11210447	11210413 {1,59,251,757}	{1,59,251,757,14809,44663, 190007,11210413}	8
7	(65179,32293)	530228077	530228063 {1,1327,863,463}	{1,463,863,1327,399569, 614401,1145201,530228063}	8
8	(174,2120)	29753	29749 {1,71,419}	{1,71,419,29749}	4
9	(1008,1825)	104827	104807 {1,311,337}	{1,311,337,104807}	4
10	(573,523)	687893	687891 {1,3,467,491}	{1,3,467,491,1401,1473, 244027, 687891}	8

To prove this, at first, numbers of the form $n = p^a$ are taken into account. The divisors are $1, p, p^2, \dots, p^a$, that is, $\tau(p^a) = a + 1$.

$$n = p^a = \underbrace{1, p, p^2, \dots, p^a}_{a+1}$$

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In this step, consider$n = p^a q^b$. The divisors will be as bellow:

$$\begin{array}{ccccc} 1 & p & p^2 & \dots & p^a \\ q & pq & p^2q & \dots & p^aq \\ q^2 & pq^2 & p^2q^2 & \dots & p^aq^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ q^b & pq^b & p^2q^b & \dots & p^aq^b \end{array}$$

Therefore, it is proved that the function $\tau(n)$ is multiplicative and in this case, is computed by$\tau(p^a q^b) = (a+1)(b+1)$. This can be extended any natural number that is written as a product of prime factors.

Table 2. All possible points order over$E: y^2 = x^3 + 26x + 460$

Point G(x, y)	Order of point #G(x, y)	Set of all points order
$\mathcal{O}$	1	{1,3,5,15,821,2463,4105,12315}
(10829, 5267)	3	{1,3,5,15,821,2463,4105,12315}
(3272, 3635)	5	{1,3,5,15,821,2463,4105,12315}
(1118, 5846)	15	{1,3,5,15,821,2463,4105,12315}
(6, 1154)	821	{1,3,5,15,821,2463,4105,12315}
(3, 3461)	2463	{1,3,5,15,821,2463,4105,12315}
(21, 4629)	4105	{1,3,5,15,821,2463,4105,12315}
(2615, 4332)	12315	{1,3,5,15,821,2463,4105,12315}

Suppose $p = 12323$ and let the elliptic curve be$E: y^2 = x^3 + 26x + 460$. The order of E is $\#E(F_p) = 12315$, which is factorized to $12315 = 3^1 \times 5^1 \times 821^1$. All points order fulfilling E have a prime factor of elliptic curve order or together with multiplicities. Hence, the feasible order probabilities of each point are only (1+1) (1+1) (1+1)=8={1, 3, 5, 15, 821, 2463, 4105, 12315} values. Table 2 illustrates several points on elliptic curve$E: y^2 = x^3 + 26x + 460$ along with their respective order.

7. MATHEMATICAL DEDUCTIVE ARGUMENT

Definition: If *G* is a finite group (or subgroup) then the order of $G$is defined as the number of elements of *G*.

Scientific evidence for the idea will be given by proving that the cosets of a subgroup have the following features:

1) they are disjoint – separate cosets do not have any member in common, and

2) each coset has an exactly identical number of members as the subgroup .

This represents that a subset with $n$elements has $k$cosets and each coset has$n$ elements; and because these cosets do not overlap and together they include every element in the group, the group must have knelements (a multiple of n). Three lemmas will lead to the proof.

Lemma: Suppose that H is one subgroup of a group G, and let $a, b \in G$ Then

(i) $aH = bH \Leftrightarrow b^{-1}a \in H.$

(ii) If $aH \cap bH \neq \phi$, then $aH = bH.$

(iii) $|aH| = |H|$ for all $a \in G.$

Proof:

(i) Let $aH = bH$, then for any $h_1 \in H$ there is $h_2 \in H$ with $ah_1 = bh_2$. This yields

$$b^{-1}a = h_2 h_1^{-1} \Longrightarrow b^{-1}a \in H,$$

because $h_2 \in H$ and $h_1^{-1} \in H$. Let $b^{-1} \in H$. Put $b^{-1}a = h_0$. Then,

$aH \subset bH$, because if $x \in aH$, then $x = ah \Longrightarrow x = b(b^{-1}a)h = b\underbrace{h_0 h}_{h_2} = bh_1 \in bH,$

$bH \subset aH$, becasue if $x \in bH$, then $x = bh \Longrightarrow x = a(b^{-1}a)^{-1}h = a\underbrace{h_0^{-1}h}_{h_2} = bh_2 \in aH.$

so, $aH \subset bH$and $bH \subset aH$, which gives $aH = bH$.

(ii) Let $aH \cap bH \neq \emptyset,$, then there exists an element *x* with

$$x \in aH \cap bH \Longrightarrow ah_1 = x = bh_2 \Longrightarrow b^{-1}a = h_2 h_1^{-1} \in H,$$

so $aH = bH$ by (i).

(iii) Consider that if $h_1$and $h_2$are two distinct elements from *H*, then $ah_1$and $ah_2$are also distinct, because otherwise

$$ah_1 = bh_2 \Longrightarrow a^{-1}ah_1 = a^{-1}ah_2 \Longrightarrow h_1 = h_2,$$

which is a contradiction. Therefore, if all elements of H are multiplied by a, the same number of elements will be attained, namely$|aH| = |H|$.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Suppose that $|G| = t$ and

$$\{a_1H, a_2H, \dots, a_tH\}$$

is the family of all cosets of H in G. Then

$$G = \{a_1H \cup a_2H \cup \dots \cup a_tH\}$$

As $G = \{a_1, a_2, \dots, a_t\}$ and $1 \in H$. By considering part (ii) of the preceding Lemma for any two cosets a_iH and a_jH , only two possibilities exist:

$a_iH \cup a_jH = \emptyset$ or $a_iH = a_jH$.

Furthermore, from part (iii) of the Lemma above, it is concluded that all cosets have exactly $|H|$ number of elements. So

$|G| = |H| + |H| + \cdots + |H| \Longrightarrow |G| = d|H|$,

and the result is accomplished.

If H is a subgroup of a finite group G, then the following statement is correct.

$|H|$ divides $|G|$.

Corollary: If p is a prime, then every group G of order p is cyclic.

Proof: Pick up$a \in G$with $a \neq 1$, and suspect that $H = \langle a \rangle$ is the cyclic subgroup that is generated by a. As a result, $|H|$ is a divisor of $|G| = p$. Because p is a prime and $|H| > 1$, it is achieved that

$|H| = p = |G|$,

and therefore $H = G$, as demanded.

In conclusion, the orders of elements in a finite group are divisors of the group order. Therefore, if and only if the order of group $E(F_q)$ is a prime number$p$, then the two feasible orders are 1 and$p$. In this context, the identity (point at infinity $\mathcal{O}$) has order 1 and no other elements. Therefore, $p$is regarded as the prime order of all remaining elements.

8. EFFICIENCY COMPARISON

This section represents a comparison between the elliptic curves that are generated by respecting the condition that "the order of the curve should be a prime number" and non-prime order elliptic curves, which fail to satisfy the condition, in terms of number of secure base points. According to the principal breakthrough achieved in the investigation and analysis conducted on over 10000 non-prime order elliptic curves ,the number of secure points #N over non-prime order elliptic curves equals $\frac{\#E}{2}$ in the optimum case; however, #N over prime order elliptic curves always follows #N= #E. A comparison between some non-prime order ECs and some prime order ECs is conducted by demonstrating information about elliptic curve characteristics, the order of elliptic curves, the set of all points order and the number of secure points over the proper EC in Table 3. All prime order elliptic curves in Table 3 are highlighted in green.

By considering the information illustrated in Table 3, one can conclude that a greater number of secure points is provided by prime order ECs in comparison to non-prime order ECs such as $y^2 = x^3 + 274x + 355 \pmod{1433}$, with an identical finite field size. Therefore, those elliptic curves with prime order should be predominantly chosen whenever obtaining an optimal number of secure base points is demanded. Moreover, any point on elliptic curves with prime orders can be readily selected as a secure base point by users; this property of ECs with prime orders justifies their greater efficiency in base point selection in comparison to non-prime order elliptic curves.

Basically, many arithmetic operations are involved in ECC, such as point addition, modular inversion and scalar multiplication. Among these operations, scalar multiplication not only consumes a significant portion of time and energy but also plays an important role in determining the speed of ECC implementation. Whenever scalar k in scalar multiplication kG (where G denotes base point) has a large value, a crucial step to a fast implementation of ECC is to choose an efficient algorithm for computing the scalar multiplication from the various algorithms proposed. The scalar multiplication is considered as the most complex part of ECC applications irrespective of the algorithm used for scalar multiplication; because it requires a repetition of point additions and point doublings along with necessary inversions over the finite prime field [64]. Table 4 shows time consumed to calculate n-bit scalar multiplication kG ($n = \lfloor \log_2 p \rfloor$) in different standard ECs where p is finite prime field and the size of scalar k is equal to n. All computations are performed on a 1.6 GHz Core™ i5-4200U CPU with 4 GB RAM and are implemented by using C# programming language (Visual Studio 2010) with no mathematical library. Moreover, the method used for scalar multiplication is left-to-right binary method.

Among the base point selection algorithm steps, scalar multiplication demands longer time to be executed; it takes about 272 ms to compute the scalar multiplication when the size of EC is 512 bits

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

as shown in Table 4. Thus, our study should focus on minimizing this step of the algorithm in order to minimize the time consumption of base point selection algorithms in EC defined in the affine space.

Table 3. Comparison between non-prime order ECs and prime order ECs in terms of secure points number

No.	(a, b, p)	Elliptic curve Order $\#E(F_p)$	Set of all points order	Number of Secure Points(#N)	Percent Point Cover
1	(26, 460, 12323)	12315 {3,5,821}	{3,5,15,821,2463,4105,12315}	3281	26%
2	(999, 520, 135497)	135479 {1,135479}	{1,135479}	135479	100%
3	(968, 18, 971)	970 {2,5,97}	{2,5,10,97,194,485,970}	192	19%
4	(143, 410, 1031)	1027 {13,79}	{13,79,1027}	468	45%
5	(974, 521, 84389)	84377 {1,84377}	{1,84377}	84377	100%
6	(274, 355, 1433)	1430 {2,5,11,13}	{2,5,10,11,13,22,26,55, 65,143,286,715,1430}	241	16%
7	(134, 2538, 145177)	145157 {379,383}	{379,383,145157}	72198	49%
8	(46521, 24508, 11210447)	11210413 {59,251,757}	{59,251,757,14809, 44663,190007,11210413}	5481001	48%
9	(57179, 520, 183467569)	183467539 {1,183467539}	{1,183467539}	183467539	100%
10	(584, 521, 84653)	84649 {1,84649}	{1,84649}	84649	100%

Table 4. The time consumption of scalar multiplication

Size	160 bit	192 bit	224 bit	256 bit	320 bit	384 bit	512 bit
Time (ms)	24.9	35.1	47.1	62.2	99.0	176.5	272.1

In the last step of the base point selection algorithm, we must ascertain that the selected base point is highly secure by proving whether the order of the point equals to the order of the curve or not; this step is performed by computing scalar multiplication and a point with an order equal to the order of the elliptic curve is confirmed as a secure base point. However, not only the time consumption of these operations is considerably high, but also these operations usually should be repeated in a base point selection process. These repetitions increase the time consumption by several folds. It can be seen from the results of last section that every point on the curve has the same order as EC whenever the order of chosen EC is a prime number. In this case, the second phase of the base point selection algorithm can be readily omitted. Figure 4 demonstrates a comparison between two base point selection methods; the random base point selection algorithm comprises the second phase, but this step is omitted in the proposed method. As the size of EC increases, the ratio of the time consumption of current method to the time consumption of proposed method decreases. This reduction is from approximately 6 to 4 in 160- to 512-bit EC.

As illustrated in Figure 4, time consumption of base point selection algorithm dramatically is decreased by omitting the fourth step which requires computation of scalar multiplication. Removing this step reduces the time consumption of the base point selection algorithm by about 24.8 ms in a curve with the size of 160 bits and about 272.1 ms in a 512 bit curve. In other words, choosing an elliptic curve with a prime order (especially having a large size) reduces the time consumed to perform the base point selection algorithm.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

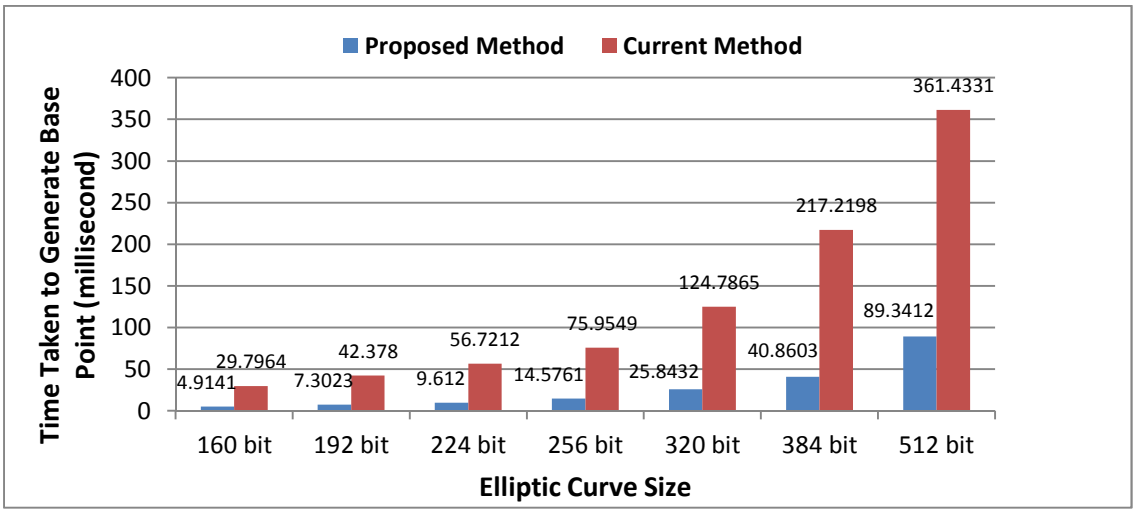

Figure 4. Efficiency comparison

9. CONCLUSION

Elliptic curve cryptography has an advantage over the other asymmetric cryptographic systems because it provides a higher security per bit for extremely constrained applications such as wireless sensor networks. This property of elliptic curve cryptography is exploited in portable constrained devices. Moreover, ECC even surpasses other asymmetric cryptosystems in exhibiting enhanced security with shorter bit sizes. Shorter key length in turn results in: 1) saving power and bandwidth, 2) performance improvements, 3) lower space requirements for key storage and 4) quicker arithmetic operations. In summary, by incorporating the ECC-based algorithms into existing protocols, a constant backward compatibility and security is accomplished with smaller resources. However, these above-mentioned positive features of ECC in WSNs are confined to those elliptic curve cryptosystems that have appropriate underlying elliptic curves with proper parameters (e.g., base point with small order is not cryptographically secure). In this regard, choosing a proper elliptic curve has evolved into a critical issue in WSNs use because the recent sensor devices have restricted computational power and are intended to monitor the security of military facilities. In this paper, an efficient type of elliptic curve for WSNs, termed as prime -order elliptic curve, is proposed. Analysis are applied on secure base point numbers of prime-order elliptic curves and then on non-prime order elliptic curves in succession. According to the results of the analyses, any point on prime order ECs can be selected as base point for secure communication.

Thus, the two following reasons are proposed to justify the suitability of elliptic curves with prime order (cofactor = 1) for WSNs. First, prime order elliptic curves provide quick and uncomplicated base point selection in hardware. Furthermore, these elliptic curves bring low area complexity required for secure resource constrained environments; i.e. in certain cases, even nodes, without considering security aspects, can automatically select any point as a base point randomly. On the other hand, non-prime order elliptic curves are recommended by some security agencies, such as NIST. The next attempt in the paper is an explanation for the mathematical deductive argument to provide reliable evidence to confirm the accuracy of the achievements.

Finally, possible directions for future research are discussed. Elliptic curve cryptosystems (e.g., ECDSA) generally entail modular integer arithmetic in addition to elliptic curve operations. This requirement for two types of arithmetic (binary field and modular integer) creates challenges for all systems using binary curves in constrained environments. Recently some high-speed hardware architectures have been proposed [72-74] but none of them is practical for extremely constrained applications due to conversions between integers and τ-adic representations in Koblitz curves. Extremely constrained applications typically require careful fine-tuning (e.g., by fixing certain parameters) of the cryptosystems that are used in the application, and these aspects should be taken into account when considering different cryptosystems for the application.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their valuable comments and to Universiti Teknologi Malaysia (UTM) for the FRGS Grant Vote numbers 4F220, 4L108 and 4L127 that are sponsored by Ministry of Education (MOE) and Research Management Centre, Universiti Teknologi Malaysia, Skudai, Johor.

REFRENCES:

[1] X. Guo and J. Zhu, "Research on security issues in Wireless Sensor Networks," in *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, 2011, pp. 636-639.

[2] Y. Wang, G. Attebury, and B. Ramamurthy, "A survey of security issues in wireless sensor networks," 2006.

[3] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation,* vol. 48, pp. 203-209, 1987.

[4] N. Koblitz, A. Menezes, and S. Vanstone, "The state of elliptic curve cryptography," in *Towards a Quarter-Century of Public Key Cryptography*, ed: Springer, 2000, pp. 103-123.

[5] R. Azarderakhsh and A. Reyhani-Masoleh, "High-Performance Implementation of Point Multiplication on Koblitz Curves," *Circuits and Systems II: Express Briefs, IEEE Transactions on,* vol. 60, pp. 41-45, 2013.

[6] V. S. Dimitrov, K. U. Jarvinen, M. Jacobson, W. Chan, and Z. Huang, "Provably sublinear point multiplication on Koblitz curves and its hardware implementation," *Computers, IEEE Transactions on,* vol. 57, pp. 1469-1481, 2008.

[7] D. Hein, J. Wolkerstorfer, and N. Felber, "ECC is Ready for RFID–A Proof in Silicon," in *Selected Areas in Cryptography*, 2009, pp. 401-413.

[8] Y. Hu, Y.-y. Cui, and T. Li, "An Optimization Base Point Choice Algorithm of ECC on GF (p)," in *Computer Modeling and Simulation, 2010. ICCMS'10. Second International Conference on*, 2010, pp. 103-105.

[9] K. Järvinen, "Optimized FPGA-based elliptic curve cryptography processor for high-speed applications," *Integration, the VLSI Journal,* vol. 44, pp. 270-279, 2011.

[10] U. Kocabas, J. Fan, and I. Verbauwhede, "Implementation of binary Edwards curves for very-constrained devices," in *Application-specific Systems Architectures and Processors (ASAP), 2010 21st IEEE International Conference on*, 2010, pp. 185-191.

[11] E. S. Kumar and C. Paar, "Are standards compliant elliptic curve cryptosystems feasible on RFID?," in *In Proc. of RFIDSec'06*, ed Austria, July 2006.

[12] H. Mahdizadeh and M. Masoumi, "Novel Architecture for Efficient FPGA Implementation of Elliptic Curve Cryptographic Processor Over," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 21, pp. 2330-2333, 2013.

[13] G. D. Sutter, J. Deschamps, and J. L. Imana, "Efficient Elliptic Curve Point Multiplication Using Digit-Serial Binary Field Operations," *Industrial Electronics, IEEE Transactions on,* vol. 60, pp. 217-225, 2013.

[14] X. Yin and J. Zou, "A parallel base point choosing algorithm of ECC on binary field," in *Systems and Informatics (ICSAI), 2012 International Conference on*, 2012, pp. 2425-2428.

[15] L. Yong Ki, K. Sakiyama, L. Batina, and I. Verbauwhede, "Elliptic-Curve-Based Security Processor for RFID," *Computers, IEEE Transactions on,* vol. 57, pp. 1514-1527, 2008.

[16] R. Anderson, F. Bergadano, B. Crispo, J.-H. Lee, C. Manifavas, and R. Needham, "A new family of authentication protocols," *ACM SIGOPS Operating Systems Review,* vol. 32, pp. 9-20, 1998.

[17] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong, "Talking to Strangers: Authentication in Ad-Hoc Wireless Networks," in *NDSS*, 2002.

[18] Q. Huang, J. Cukier, H. Kobayashi, B. Liu, and J. Zhang, "Fast authenticated key establishment protocols for self-organizing sensor networks," in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, 2003, pp. 141-150.

[19] Q. Huang, H. Kobayashi, and B. Liu, "Energy/security scalable mobile cryptosystem," in *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC*

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2003. 14th IEEE Proceedings on, 2003, pp. 2755-2759.

[20] S. Kumar, M. Girimondo, A. Weimerskirch, C. Paar, A. Patel, and A. S. Wander, "Embedded end-to-end wireless security with ECDH key exchange," in *Circuits and Systems, 2003 IEEE 46th Midwest Symposium on*, 2003, pp. 786-789.

[21] D. Liu, P. Ning, and R. Li, "Establishing pairwise keys in distributed sensor networks," *ACM Transactions on Information and System Security (TISSEC),* vol. 8, pp. 41-77, 2005.

[22] K. Lorincz, D. J. Malan, T. R. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder*, et al.*, "Sensor networks for emergency response: challenges and opportunities," *Pervasive Computing, IEEE,* vol. 3, pp. 16-23, 2004.

[23] D. J. Malan, M. Welsh, and M. D. Smith, "A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography," in *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, 2004, pp. 71-80.

[24] Y. Qiu, J. Zhou, J. Baek, and J. Lopez, "Authentication and key establishment in dynamic wireless sensor networks," *Sensors,* vol. 10, pp. 3718-3731, 2010.

[25] A. Weimerskirch and D. Westhoff, "Zero common-knowledge authentication for pervasive networks," in *Selected Areas in Cryptography*, 2004, pp. 73-87.

[26] F. Tellez and J. Ortiz, "Behavior Elliptic Curve Cryptosystem for the Wormhole Intrusion in MANET," ed: IJCSo, 2011.

[27] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," in *Cryptographic hardware and embedded systems-CHES 2004*, ed: Springer, 2004, pp. 119-132.

[28] L. Uhsadel, A. Poschmann, and C. Paar, "Enabling full-size public-key algorithms on 8-bit sensor nodes," in *Security and Privacy in Ad-hoc and Sensor Networks*, ed: Springer, 2007, pp. 73-86.

[29] D. Chu, J. Großschädl, Z. Liu, V. Müller, and Y. Zhang, "Twisted Edwards-form elliptic curve cryptography for 8-bit AVR-based sensor nodes," in *Proceedings of the first ACM workshop on Asia public-key cryptography*, 2013, pp. 39-44.

[30] S. Mangard, E. Oswald, and T. Popp, *Power analysis attacks: Revealing the secrets of smart cards* vol. 31: Springer Science & Business Media, 2008.

[31] U. Sabeel, N. Chandra, and S. Dagadi, "A Novel Scheme for Multiple Spoof Attack Detection and Localization on WSN-based Home Security System," in *Computational Intelligence and Communication Networks (CICN), 2013 5th International Conference on*, 2013, pp. 360-367.

[32] J. Deng, R. Han, and S. Mishra, "A performance evaluation of intrusion-tolerant routing in wireless sensor networks," in *Information Processing in Sensor Networks*, 2003, pp. 349-364.

[33] Y.-C. Hu and A. Perrig, "A survey of secure wireless ad hoc routing," *IEEE Security & Privacy,* vol. 2, pp. 28-39, 2004.

[34] C. Karlof, N. Sastry, and D. Wagner, "TinySec: a link layer security architecture for wireless sensor networks," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004, pp. 162-175.

[35] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," in *Security and Privacy, 2005 IEEE Symposium on*, 2005, pp. 49-63.

[36] Y. Sang, H. Shen, Y. Inoguchi, Y. Tan, and N. Xiong, "Secure data aggregation in wireless sensor networks: A survey," in *Parallel and Distributed Computing, Applications and Technologies, 2006. PDCAT'06. Seventh International Conference on*, 2006, pp. 315-320.

[37] D. Westhoff, J. Girao, and M. Acharya, "Concealed Data Aggregation for Reverse Multicast Traffic in Sensor Networks: Encryption, Key Distribution, and Routing Adaptation," *Mobile Computing, IEEE Transactions on,* vol. 5, pp. 1417-1431, 2006.

[38] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks," *ACM Transactions on Sensor Networks (TOSN),* vol. 2, pp. 500-528, 2006.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[39] R. Muraleedharan, Y. Yan, and L. A. Osadciw, "Detecting sybil attacks in image senor network using cognitive intelligence," in *Proceedings of the First ACM workshop on Sensor and actor networks*, 2007, pp. 59-60.

[40] A. Srinivasan and J. Wu, "A survey on secure localization in wireless sensor networks," *Encyclopedia of Wireless and Mobile communications,* 2007.

[41] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Distributed software-based attestation for node compromise detection in sensor networks," in *Reliable Distributed Systems, 2007. SRDS 2007. 26th IEEE International Symposium on*, 2007, pp. 219-230.

[42] W. Zhang, M. Tran, S. Zhu, and G. Cao, "A random perturbation-based scheme for pairwise key establishment in sensor networks," in *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, 2007, pp. 90-99.

[43] T. Zia and A. Zomaya, "Secure localization in wireless sensor networks," in *Proc. of the 4th IASTED Asian Conference on Communication Systems and Networks (AsiaCSN'07), Phuket, Thailand. ACTA Press*, 2007, pp. 177-180.

[44] H. Alzaid, S. Abanmi, S. Kanhere, and C. T. Chou, "BANAID: A sensor network test-bed for wormhole attacks," 2008.

[45] H. Alzaid, E. Foo, and J. G. Nieto, "Secure data aggregation in wireless sensor network: a survey," in *Proceedings of the sixth Australasian conference on Information security-Volume 81*, 2008, pp. 93-105.

[46] S. Capkun, K. Bonne Rasmussen, M. Cagalj, and M. Srivastava, "Secure Location Verification with Hidden and Mobile Base Stations," *Mobile Computing, IEEE Transactions on,* vol. 7, pp. 470-483, 2008.

[47] Q. Dong, D. Liu, and P. Ning, "Pre-authentication filters: providing dos resistance for signature-based broadcast authentication in sensor networks," in *Proceedings of the first ACM conference on Wireless network security*, 2008, pp. 2-12.

[48] A. Liu and P. Ning, "TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks," in *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on*, 2008, pp. 245-256.

[49] P. Ning, A. Liu, and W. Du, "Mitigating DoS attacks against broadcast authentication in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN),* vol. 4, p. 1, 2008.

[50] W. Wang, J. Kong, B. Bhargava, and M. Gerla, "Visualisation of wormholes in underwater sensor networks: a distributed approach," *International Journal of Security and Networks,* vol. 3, pp. 10-23, 2008.

[51] J. P. Walters, Z. Liang, W. Shi, and V. Chaudhary, "Wireless sensor network security: A survey," *Security in distributed, grid, mobile, and pervasive computing,* vol. 1, p. 367, 2007.

[52] E. Yoneki and J. Bacon, "A survey of Wireless Sensor Network technologies," UCAM-CL-TR2005.

[53] A.-S. Pathan, M. Alam, M. Monowar, and M. Rabbi, "An efficient routing protocol for mobile ad hoc networks with neighbor awareness and multicasting," in *E-Tech 2004*, 2004, pp. 97-100.

[54] G. Xiaowang and Z. Jianyong, "Analysis and design of energy-oriented security protocols for Wireless Sensor Networks," in *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International Conference on*, 2011, pp. 2298-2301.

[55] J. M. Bahi, C. Guyeux, and A. Makhoul, "Secure data aggregation in wireless sensor networks: homomorphism versus watermarking approach," in *Ad Hoc Networks*, ed: Springer, 2010, pp. 344-358.

[56] J. Meng and L. Ye, "Secure mobile payment model based on WAP," in *Proceedings of 4th IEEE International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM apos' 08)*, 2008, pp. 1-4.

[57] L. Mou, T. Huang, L. Huo, W. Li, W. Gao, and X. Chen, "A secure media streaming mechanism combining encryption, authentication, and transcoding," *Signal Processing: Image Communication,* vol. 24, pp. 825-833, 2009.

[58] V. Gupta, D. Stebila, and S. C. Shantz, "Integrating elliptic curve cryptography into the web's security infrastructure," in *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, 2004, pp. 402-403.

© 2005 - 2015 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[59] T. Tao, "Algebraic combinatorial geometry: the polynomial method in arithmetic combinatorics, incidence combinatorics, and number theory," *arXiv preprint arXiv:1310.6482,* 2013.

[60] R. Schoof, "Elliptic curves over finite fields and the computation of square roots mod " *Mathematics of computation,* vol. 44, pp. 483-494, 1985.

[61] T.Revathi and P.Sumathi, "Distributed Data Mining based on Random Projection with Optimal Communication," *International Journal of Soft Computing & Engineering,* 2013.

[62] M. J. J.-J. Quisquater, "Cryptographic Hardware and Embedded Systems–CHES 2004," 2004.

[63] H. Rifa-Pous and J. Herrera-Joancomartí, "Computational and energy costs of cryptographic algorithms on handheld devices," *Future internet,* vol. 3, pp. 31-48, 2011.

[64] H. Darrel, M. Alfred, and V. Scott, "Guide to elliptic curve cryptography," *Hankerson Darrel, Menezes Alfred J., Vanstone Scott—Springer-Verlag Professional Computing Series.—2004.—311 p,* 2004.

[65] J.-S. Coron, "Resistance against differential power analysis for elliptic curve cryptosystems," in *Cryptographic Hardware and Embedded Systems*, 1999, pp. 292-302.

[66] P. L. Montgomery, "Speeding the Pollard and elliptic curve methods of factorization," *Mathematics of computation,* vol. 48, pp. 243-264, 1987.

[67] C. Giraud and V. Verneuil, "Atomicity improvement for elliptic curve scalar multiplication," in *Smart Card Research and Advanced Application*, ed: Springer, 2010, pp. 80-101.

[68] T. Izu, B. Möller, and T. Takagi, "Improved elliptic curve multiplication methods resistant against side channel attacks," in *Progress in Cryptology—INDOCRYPT 2002*, ed: Springer, 2002, pp. 296-313.

[69] L. Shengli, Z. Dong, and W. Yumin, "FINDING SECURE ELLIPTIC CURVES OVER GF(2n) AND THEIR BASE POINTS " *JOURNAL OF ELECTRONICS,* vol. 22, pp. 824-830, 2000.

[70] A. O. L. Atkin and F. Morain, "Elliptic curves and primality proving," *Mathematics of computation,* vol. 61, pp. 29-68, 1993.

[71] R. Schoof and P. R. E. Schoof, "Counting points on elliptic curves over finite fields," 1995.

[72] J. Adikari, V. S. Dimitrov, and K. U. Jarvinen, "A Fast Hardware Architecture for Integer to tauNAF Conversion for Koblitz Curves," *Computers, IEEE Transactions on,* vol. 61, pp. 732-737, 2012.

[73] B. B. Brumley and K. U. Jarvinen, "Conversion Algorithms and Implementations for Koblitz Curve Cryptography," *Computers, IEEE Transactions on,* vol. 59, pp. 81-92, 2010.

[74] K. Järvinen, J. Forsten, and J. Skyttä, "Efficient Circuitry for Computing τ-adic Non-Adjacent Form," in *ICECS'06*, ed, 2006, pp. 232-235.