BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **20**, No 3
Sofia • 2020 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2020-0030

# Key Generation Using Generalized Pell's Equation in Public Key Cryptography Based on the Prime Fake Modulus Principle to Image Encryption and Its Security Analysis

*K. R. Raghunandan*[1]*, Aithal Ganesh*[2]*, Shetty Surendra*[1]*, K. Bhavya*[3]

[1]*Department of Computer Science and Engineering, N.M.A.M Institute of Technology, Nitte, Affiliated to Visvesaraya Technological University, India*
[2]*Department of Electronics and Communications Shri Madhwa Vadiraja Institute of Technology and Management, Bantakal, Udupi, Affiliated to Visvesaraya Technological University, India*
[3]*Department of Mathematics, N.M.A.M Institute of Technology, Nitte, Affiliated to Visvesaraya Technological University, India*
*E-mails: raghunandan@nitte.edu.in ganeshaithal@gmail.com hsshetty@nitte.edu.in bhavyak@nitte.edu.in*

**Abstract**: *RSA is one among the most popular public key cryptographic algorithm for security systems. It is explored in the results that RSA is prone to factorization problem, since it is sharing common modulus and public key exponent. In this paper the concept of fake modulus and generalized Pell's equation is used for enhancing the security of RSA. Using generalized Pell's equation it is explored that public key exponent depends on several parameters, hence obtaining private key parameter itself is a big challenge. Fake modulus concept eliminates the distribution of common modulus, by replacing it with a prime integer, which will reduce the problem of factorization. It also emphasizes the algebraic cryptanalysis methods by exploring Fermat's factorization, Wiener's attack, and Trial and division attacks.*

**Keywords**: *Public Key Cryptography, Fermat's Factorization, Standard Deviation, Pell's Equation, Wiener's Attack, Trial and Division.*

## 1. Introduction

There are two approaches in cryptography which are based on the usage of keys; they are Private or Symmetric Key and Public or Asymmetric key. In Symmetric key, both sending end and receiving end use the same key while in asymmetric, different key will be used. This paper concentrates on asymmetric key, especially on RSA which is explained briefly in next section.

### 1.1. Asymmetric Key Cryptography

RSA is a public key system which is also referred to as Asymmetric key. In this algorithm, couple of related keys are used for enciphering and deciphering. The public key used for enciphering algorithm and key for deciphering algorithm is private. Fig. 1 explains asymmetric key cryptography which uses different keys for enciphering and deciphering algorithm.

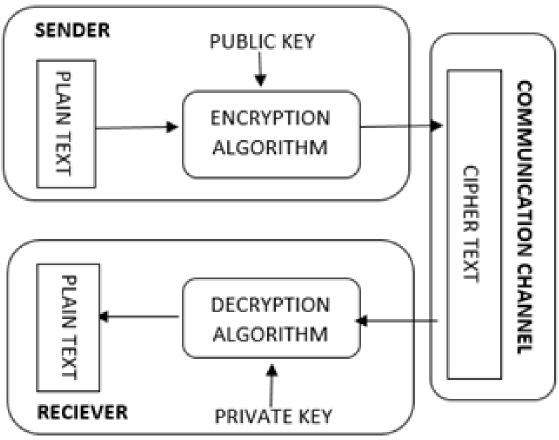

Fig. 1. Asymmetric Key Cryptography process

Any user wishing to send any information must initially use recipient's public key information. Using this key, the message is encrypted and sent. It is hard to decipher the enciphered data by anyone who knows only the public key. The legitimate operator only, who has the private key can decipher the original data [1].

RSA is considered to be one of the most secure public key system. This common RSA eventually stands on the effort of making judgement over the known $e$-th root on to the product of two or more prime value $n$ [2]. RSA key generation needs a substantial quantity of calculation to acquire prime elements, hence increase in time complexity for the system [3].

RSA uses three processes for the system to operate, which are key generation, encipher and decipher. Key generation process deals with computing secret key exponent from the public key based on the Euler's algorithm. This exponent is computed with a condition that, $\gcd(e, \emptyset(n)) = 1$, where $\emptyset(n)$ is Euler's totient function. Since the deciphering key $d$ and enciphering key $e$ are different, the system makes it difficult to generate one key from the other.

## 1.2. RSA methodology

Let $p_i$ and $q_i$ be large prime numbers and product of these numbers form modulus $n$. Calculate $\emptyset(n)$ using $(p_i - 1)\times(q_i - 1)$. Algorithm computes public key $e$ such that $1<e<\emptyset(n))$ and $\gcd(e, \emptyset(n)) =1$. Secret key $d$ is computed by choosing $d$ where $d\times e \equiv 1 \bmod \emptyset(n)$. After key generation process, $e$ and modulus $n$ is announced publicly while $d$ is kept secret. On receiving public key, sender encrypts the plaintext ($M$) by using equation $C=M^e \bmod n$. Encrypted text is deciphered using secret key by $M=C^d \bmod n$.

### 1.3. Literature survey

Eavesdroppers always try to break any cryptographic system. This section includes the work proposed by different analysts in RSA cryptosystem. The conversation is based on the Amendment of RSA algorithm through the recent past.

The following part primarily focuses on differentiating between the types of attacks, their effects and their counter measures to provide insight in order to develop a variant. Literature survey is classified into three sections. Section one gives survey of forgeries a hacker can implement on RSA. Section two summarizes different RSA variants. In Section three, different properties of Pell's equation used in public key approaches are dealt in detail.

### 1.3.1. RSA Attacks

RSA attacks are broadly classified into Mathematical attack and Elementary attack. Mathematical attack includes those attacks that directly target the mathematical function and Elementary attack includes those attacks that exploit weakness in its implementation. Since RSA algorithm is prone to mathematical attack, this work concentrates only towards mathematical attacks of RSA by using generalized Pell's equation.

### I.3.2. Mathematical Attacks on RSA:

In this attack, attacker emphasis is on breaking the fundamental arrangement of the mathematical function. Main attacks under this category are low public key exponent attack [4, 5], Hastad broadcast attack [6, 7], Coppersmith's Short Pad Attack [8], Factorization attack [9], chosen cipher text attack [10], common modulus attack [11], low private key exponent attack [12] and Blömer and May in [13] are presented as an extension work of Wiener's attack on small RSA secret decryption exponents. All these attacks are summarized as follows:

Thuc D. Nguyen, Than D. Nguyen and Long D. Tran [14] suggested in his work that using a small public-key exponent *e* helps in reducing the signature-verification time. If the public key *e* is small, using the value of secret key *d*, attacker can factorize *n* using

$$e \times d = 1 \bmod \emptyset(n). \qquad (1)$$

The common modulus *n* can be factorized after finding the value of $\emptyset(n)$. Hence all private keys can be calculated using the value of $\emptyset(n)$ and their public exponent *e*, and then all messages corresponding to them can be decrypted. If the sender makes use of a larger public key exponent, this attack can be prevented and sending messages can be more secure. Coppersmith in [15] presents how to find a small integer solutions to a polynomial in a single variable modulo *N*, and to a polynomial in two over the integers. Coppersmith's theorem [16, 17] for padded messages is the basis of most of the powerful attacks on low public exponent RSA. In this attack, the receiver is constantly sent padded messages until the actual message reaches him. The whole attack is based on the fact that if the original padded message fails to reach the receiver due to interception of a hacker, the sender tries to resend the same message with a different pad, which is also intercepted by the hacker. A randomized padding scheme that is cryptographically secure can be used. Boneh [18] suggested factorization attack, here using modulus $n$, the hacker can find out $\emptyset(n)$ from which attacker can find the decryption exponent

(2) $$d = e^{-1} \bmod \emptyset(n).$$

He recommended General Number Field Sieve (GNFS) deliberated as the strongest factoring process. The main aim is to look into attacks on RSA that enable decryption of messages without having to factor RSA modulus $n$. Factorization of $n$ gives $\emptyset(n)$. Once $e$ is discovered, $d$ can be easily computed. It is relatively easy to factor $n$ once the value of $d$ is recovered. Mumtaj and Ping in [19] describes a brief survey of past findings and detailed descriptions about specific attacks and also showcased that a well implemented algorithm is unbreakable and it is survived against a number of cryptanalytic attacks from last forty years. In [13] authors proposed an attack constructed on continued fractions procedure on RSA public key pair ($n$, $e$) with $e \in Z\times \emptyset(n)$, which satisfies

(3) $$e \times d^{-1} = 1 \bmod \emptyset(n),$$

for some, $d<(1/3)n^{0.25}$ that produces factorization of $n=p\times q$. In this, attacker makes use of sequence of continuous fraction so that decryption key is exposed when key is of smaller value. This attack can be reduced if exponent $e$ is replaced by exponent $e^1$ where

(4) $$e^1 = e + Z\,\emptyset(n),$$

for some large value of $Z$. when $e^1 > n^{1.5}$, Wiener attack becomes insufficient even if $d$ is small. Boneh and Durfee [20] experimentally demonstrated that private exponent $d$ can be recovered using lattice attack which is extra operational than Wiener Attack which uses continued fraction expansion $\frac{e}{n}$. Durfee and Nguyen [21] proposed RSA variants with short secret exponent extended version. They also proposed extended version of Boneh and Durfee [20] attack which is constructed on Coppersmith's lattice-based procedure.

In the above sections attacks have been discussed. Since several authors have worked on RSA variants also, it is necessary to consider this in the survey. The next section gives a brief summary of variants of RSA cryptosystem.

### 1.3.3. RSA variants

In the dual RSA proposed by Sun et al. [22], two distinct key pairs are generated. The RSA key pair has same encryption key and decryption key. Hence they are known as Dual RSA. Dual RSA reduces the key storage requirements. Dual RSA has application in authentication or secrecy and blind signature. RSA presented in paper [23] uses three prime numbers and the time required for enciphering and deciphering is the same as original RSA. Thangavel et al. [24] came up with a modified RSA approach which uses four prime factors instead of two, by doing its complexity greater with respect to time increases to find prime factors. Tuteja and Shrivastava [25] introduce new algorithm to change the modulus $n$ of RSA. In this, the original modulus $n$ is changed to fake modulus $F_n$. The fake modulus $F_n$is sent as public key parameter for encryption of plain text on sender side. Jaju and Chowhan [26] presented improved RSA where they used three prime numbers to form modulus. Instead of modulus $n$, new parameter $\varepsilon$ is sent publicly. If $p> q$ then $x$ is calculated as

(5) $$n - p < \varepsilon < n \text{ and } \gcd(\varepsilon, n) = 1;$$
else if $p<q$ then $\varepsilon$ is considered as
(6) $$n - q < \varepsilon < n \text{ and } \gcd(\varepsilon, n) = 1.$$

Public key exponent of modified RSA is of the form $(e, \varepsilon)$ and secret key $(d, \varepsilon)$. Thus the security is increased by three levels instead of two levels in RSA making it difficult for attackers. Segar and Vijayaragavan [27] developed a new approach where generation of keys is based on Pell's equation, which makes use of the roots of Diophantine equation. They also proposed and analysed its complexity with RSA variants. Further, they analyzed cryptanalysis of Fermat's attack [28, 29], Weiner's continued fraction [30] and extended Euclidean method and these are compared along with RSA using numerical examples. A novel public key cryptography technique is proposed by Raghunandan et al. [31] which makes use of using Pell's quadratic case for key generation process and these are compared along with RSA using numerical examples. Raghunandan et al. [32] also introduces the concept of fake modulus which is to overcome the limitations of Integer factorization attack. Nag Amitava et al. in [33] proposed a general $(k, n)$ secret image sharing scheme using low reconstruction complexity and preservation of the fault tolerance property.

After RSA variants, it is also noted that numerous mathematical equations are used in the field of cryptography. This paper concentrates predominantly on the improvement of RSA using generalized Pell's equation in public key cryptography based on fake modulus principle and its security analysis. As a survey of literature, in the next section enhancements made in the field of RSA using Pell's equation in public key cryptography is discussed.

### 1.3.4. Pell's Equation

Pell's Equation has been used in the area of number theory for numerous applications from ancient times since it comes under cyclic group and has multiple solutions. Based on these features, many cryptography algorithms have been designed. Barbeau [34] suggests the usage of Pell's equation for higher order. Chen, Chang and Yang [35] introduced fast RSA which is established on Pell's equation. Using this, he showed that encryption speed is 1.5 times faster and decryption is two times faster than standard RSA. Padhye [36] projected a new operation using solution space of Pell's Equation and proposed three RSA variants. Rao et al. [37] proposed an identity based encryption algorithm which makes use of Pell's equation. Burton [38] says clearly that Pell's equation can be used in addition to RSA to foil some of the above attacks. Raghunandan et al. [39] proposed dual RSA approach using Pell's equation to hide public key component, and in paper [40] they used Pell's cubic equation for securing media information. Hence in this work, it has been shown that in addition to RSA, generalized Pell's equation gives an added advantage.

In the next section, the mathematical background for the work is dealt in detail. Section 3 deals with methodology of the proposed system. In Section 4, all the experimental details are elaborated. Section 5 describes analysis of the results. The paper concludes in Section 6.

## 2. Mathematical preliminaries

A group $G$ is denoted as $\{G, *\}$, where * is binary operation on set of elements on $G$. The binary operations are performed on ordered pairs $(m, n)$ of the set to get an element using $(m*n)$ in $G$. To form group $m$ set, $G$ must satisfy the following axioms:

- Closure: If $m, n \in G$, then $(m*n)$ is also in $G$.
- Associative: $(m*n) * y = m*(n*y)\ \forall\ m, n, y \in G$.
- Identity element: Any $e \in G$ so that $m*e = e*m = m\ \forall m$ in $G$.
- Inverse element: For each $m$ in $G$, then an inverse element $m'$ in $G$, i.e., $m*m' = m'*m = e$.

Number of components in a group forms order of the group and if the group contains finite components, is Finite Group. If a group fulfils the above axioms with a following additional axioms, we can refer to it as Abelian groups.

- Commutative: For all $m, n \in G$, $m*n = n*m$.

If a group is generated by a single element, then we refer to it as cyclic group. Group $G$ is having an element $m$, called the generator of the group, such that all the elements of the group are powers of the element $m$,

(7) $$G = \{m^n : n \in \mathbb{Z}\},$$

where $m$ is the generator of $G$.

Cyclic groups are said to be isomorphic if the groups have same order. Also, if order of the group is a prime number, then it is cyclic. Cyclic groups are the simplest abelian groups when they have an order = 1 or the order is prime.

From Fermat's Little Theorem, if $p$ is a prime number and $a$ is an integer, then

(8) $$a^p \bmod p = a.$$

Furthermore, if $\gcd(a, p) = 1$ then

(9) $$a^{p-1} \bmod p = 1.$$

Let $M$ be the original data to be enciphered using Equation (9) then, enciphering is done by raising plaintext to the $e$-th power modulo $p$ to obtain cipher text $C$, then to obtain $M$ again, decipher is done by raising the cipher text to the $d$-th power modulo $p$

(10) $$C \equiv E(M) \equiv M^e (\bmod p),$$

(11) $$M \equiv D(C) \equiv C^d (\bmod p).$$

Original message obtained by deciphering an enciphered message $(D(E(M)) = M)$,

(12) $$(M^e \bmod p)^d \bmod p = M,$$

(13) $$M^{e.d} \bmod p = M.$$

Further we extended using Pell's equation for more rigidity.

Let $x^2 - Ry^2 = 1$, linear Diophantine equation, where $R$ is a positive integer, for any $x, y$ belongs to integer set $Z$.

**Proposition 1.** If the Pell's equation $x^2 - Ry^2 = 1$ has nontrivial solutions, then $R$ is a positive integer, which is not a perfect square.

*Proof*:

**Case I ($R = -1$).** The equation $x^2 + y^2 = 1$ has four trivial solutions: (±1, 0), (0, ±1).

**Case II ($R < -1$).** Then $x \neq 0 \Longrightarrow x^2 - Ry^2 \geq 2$, so (1) has only the solutions (±1, 0).

**Case III (*R* = *N*$^2$).** Then $x^2 - Ry^2 = (x + Ny)(x - Ny) = 1$, and this imposes either: $x + Ny = x - Ny = 1$ in which case $x = 1$, $y = 0$; or $x + Ny = x - Ny = -1$, 1 in which case $x = -1$, $y = 0$ and there are only trivial solutions.

Since the proposed scheme uses Case I, we are not using Case II and Case III in our scheme

Finding solutions of the Pell's equation using the next

**Theorem.** Let *R* be a positive, non-square integer.

a) There exists a positive integral result $(x, y)$ to $x^2 - Ry^2 = 1$,

b) There exists a unique positive integral result $(x_1, y_1)$ with $(x_1, y_1)\sqrt{R}$ is minimal. Put $u=(x_1+ y_1)\sqrt{R}$ . Then every positive integral solution is of the form

$$\left(\frac{u^n+u'^n}{2}, \frac{u^n-u'^n}{2\sqrt{R}}\right) = \left(\frac{u^n}{2}, \frac{u^n}{2\sqrt{R}}\right) \text{ for a unique } n \in Z^+,$$

c) Every result to the Pell's equation is of the form ± $(x_n, y_n)$ for $n \in Z$.

Let $x+y\sqrt{R}$ be the selected number, the multiplicative inverse of that is $x-y\sqrt{R}$ subsequently,

(14) $$x + y\sqrt{R},\ x - y\sqrt{R}\,(x + y\sqrt{R}) = x^2 - Ry^2 = 1,$$

here $(x, y)$ is a solution of $x^2 - Ry^2 = 1$. Thus the solutions $(x_i, y_i)$ form a group, and the positive number $x+y\sqrt{R}$ is a subgroup. As given in axioms given in (b) the $x + y\sqrt{R}$ is a unique solution of $(x_i, y_i)$. Since it is a unique solution, it is shown that solutions of $x^2 - Ry^2 = 1$, which is infinite cyclic group.

Let $(x_1, y_1)$ be a solution of $x^2 - Ry^2 = 1$. Then we can generate another solution given in the Equation (14)

(15) $$(x_1^2 + Ry_1^2)^2 - R\left(2x_1y_1\sqrt{R}\right)^2 = 1.$$

Let $(x_1, y_1)$ be the ultimate result of $x^2 - Ry^2 = 1$. Then pair $(x_n, y_n)$ is defined by

(16) $$(x_{n,} + y_n\sqrt{R}) = (x_{1+}y_1\sqrt{R})^n,$$

$$(x_{n,} + y_n\sqrt{R}) = (x_{1+}y_1\sqrt{R})^n,$$

which has also a positive solution, $n = 1, 2, 3\ldots$

Suppose *R* is a positive integer and *x* and *y* are integers satisfying the Cubic power of Pell's equation $x^3 - Ry^3 = 1$. Security features of RSA by cubic power of Pell's equation is discussed in [41]. This Pell's equation is generalization of *m*-th order, which is used in this work and gives one more level of security abstraction using a novel concept in public key cryptographic technique. Barbeau [34] proved that there will be set of roots in higher order of Pell's equation.

In our work, the integer roots of *m*-th order of Pell's equation is taken. Pell's equation is

(17) $$x^m - Ry^m = 1,$$

where *R* and *m* belongs to the integer set.

Consider a solution set $C_p$, set of all solutions of the equation

(18) $$x^m - Ry^m = 1(\bmod n).$$

Let *n* be the common modulus, calculated by multiplying prime numbers *p, q*, *r* and *s*, and totient function $\emptyset(n)$ calculated as

(19) $$\emptyset(n) = (p - 1) \times (q - 1) \times (r - 1) \times (s - 1).$$

Private key component $d$ can be computed as per RSA. Select an integer $m$ and $g$enerate $R$ and $(x, y)$ pair from the Equation (17).

Let us consider $m$-th order for equation $x^m – Ry^m = 1$. This equation can be extended based on Euler's totient function of $n$ to get the value $\beta$, which is

$$\beta = [y + \emptyset(n)]^m - R[x + e]^m. \tag{20}$$

Based on the Binomial expansion, this equation can be simplified as

$$\beta + m_{C_1}Rx^{m-1}e - m_{C_2}Rx^{m-2}e^2 + \cdots + Re^m \equiv 1 \bmod \emptyset(n). \tag{21}$$

The Equivalence (21) can also be written as

$$\beta = \sum_{k=0}^{m} \frac{m!}{(m-k)!k!} x^{m-k}\emptyset(n)^k - R\sum_{k=0}^{m} \frac{m!}{(m-k)!k!} y^{m-k}e^k. \tag{22}$$

Using $\beta$ the computation of public key $S$ as

$$S\text{=}[\beta + R[m_{C_1}y^{m-1}e + m_{C_2}y^{m-2}e^2 + \ldots + m_{C_m}e^m]] * d^m * \text{mod } \emptyset(n). \tag{23}$$

Calculate fake modulus $z$ to replace $n$ using

$$z = \frac{((e\times d)-1+k)}{k}, \tag{24}$$

where $k > 1$ and $z$ should be prime number.

Computation of private key exponent $E$ using

$$E = e^m \text{ mod } \emptyset(n). \tag{25}$$

The above mathematical technique is used in a public key cryptographic system and the methodology for it is explained in the next section.

## 3. Proposed methodology

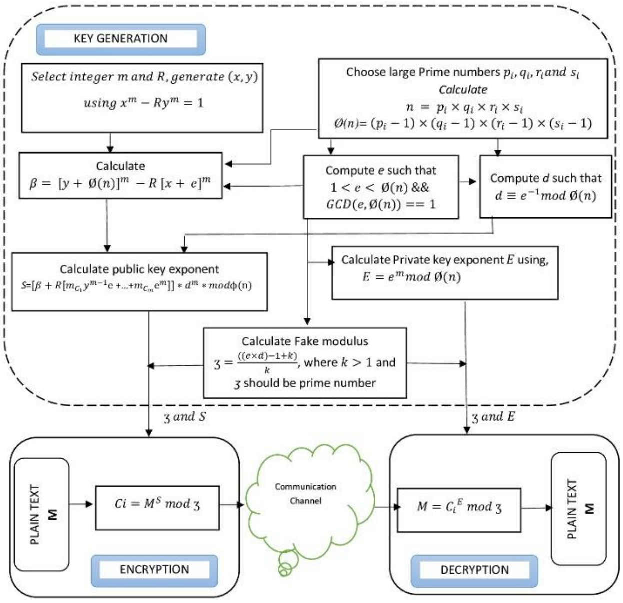
KEY GENERATION
Select integer $m$ and $R$, generate $(x, y)$ using $x^m - Ry^m = 1$
Choose large Prime numbers $p_i, q_i, r_i$ and $s_i$
Calculate
$n = p_i \times q_i \times r_i \times s_i$
$\emptyset(n) = (p_i - 1) \times (q_i - 1) \times (r_i - 1) \times (s_i - 1)$
Calculate
$\beta = [y + \emptyset(n)]^m - R[x + e]^m$
Compute $e$ such that $1 < e < \emptyset(n)$ && $GCD(e, \emptyset(n)) == 1$
Compute $d$ such that $d \equiv e^{-1} mod\, \emptyset(n)$
Calculate public key exponent
$S$=$[\beta + R[m_{C_1}y^{m-1}e + \ldots + m_{C_m}e^m]] * d^m * mod\phi(n)$
Calculate Private key exponent $E$ using,
$E = e^m mod\, \emptyset(n)$
Calculate Fake modulus
$\mathfrak{z} = \frac{((e\times d)-1+k)}{k}$, where $k > 1$ and $\mathfrak{z}$ should be prime number
$\mathfrak{z}$ and $S$
$\mathfrak{z}$ and $E$
PLAIN TEXT M
$Ci = M^S mod\, \mathfrak{z}$
ENCRYPTION
Communication Channel
$M = C_i^E mod\, \mathfrak{z}$
PLAIN TEXT M
DECRYPTION

Fig. 2. Block-diagram of Generalized Pell's equation algorithm for both key generation and encryption

The above-generalized equation can be easily adapted to the extension of RSA. This section deals with the methodology of using Generalized Pell's equation in addition to RSA, which has three parts Key generation, Encryption and Decryption.

Block-diagram of Generalized Pell's equation algorithm for both key generation, encryption and decryption is explained using Fig 2.

**1. Key generation.** Here also, as in the case of RSA, instead of two prime numbers, choose four large prime numbers $p_i,\ q_i,\ r_i$ and $s_i$.

Let $n = p_i \times q_i \times r_i \times s_i$ and $\emptyset(n) = (p_i - 1) \times (q_i - 1) \times (r_i - 1) \times (s_i - 1)$, evaluated as per Equation (19). Select an integer $m$ and generate $R$ and ($x$, $y$) pair from the Equation (17). Calculate $\beta$ using Equations (20), (21) or (22). Calculate Public Key Exponent $S$ using Equation (23). Calculate fake modulus $z$ to replace $n$ using Equation (24). Private Key exponent $E$ can be calculated using Equation (25).

**2. Encryption.** Plain texts are encrypted using Public key exponent $S$ using the equation

(26) $$C_i = M^S \bmod z.$$

**3. Decryption.** Using Private Key receiver obtaining plaintext back using the equation

(27) $$M = {C_i}^E \bmod z.$$

The above said work is explained by the following example.

**Example.** In Equation (17), Let $m$ be the exponent integer and $R$ will be an integer, obtaining $x$, $y$ pairs which is required for generation of encryption and decryption keys. Let an integer $m$=5 and by keeping $R$=31, obtained ($x, y$) pair as (2, 1) from the Equation (17). Let four prime numbers be $p$ = 61, $q$ = 89, $r$ = 29 and $s$ = 31 computed modulus $n$ = 4880671 and using Equation (19) obtained $\emptyset(n)$ = 4435200. Select $e$ = 13 where gcd(13, 4435200) = 1. Using $e$ obtained $d$ = $13^{-1}$mod4435200 = 1364677. By Equations (20) and (22) obtained, $\beta$ = 17161965404586733344609045813434 88. Substituting in Equation (23) obtained public key exponent $S$ = 3130357. Fake modulus $z$ = 8870401 can be estimated using Equation (24).

Private Key exponent is computed by applying Equation (25) as $E = 13^5$mod 4435200 = 371293. During encryption, sender selects plain text $M$ = "$c$", ASCII value of "$c$" = 99. Sender generates Cipher text using Equation (26) as, $C_i$=$99^{3130357}$mod 8870401= 5453252. At the receiving end receiver can obtain original message $M$ using Equation (27) as, $M$ = $(5453252)^{371293}$mod 8870401 = 99 = "$c$".

## 4. Results and analysis

This section is organized as follows. Comparisons of proposed scheme with standard RSA are discussed in Section 1. Computational intricacy of the proposed algorithm is analysed and compared with standard RSA in Section 2. Complexity to break proposed scheme using different attacks are summarized and compared against RSA in Section 3. Experimental results conducted using proposed algorithm is summarized in Section 4.

### 4.1. Comparison with RSA

The paper compares the proposed algorithm with standard RSA for security in terms of mathematical attacks. Standard RSA faces the problem of factorization, because intruder can forge the secret key $d$ by using public key exponent $e$ and $n$, here common modulus $n$ is dependent on two factors which can be factored using minimal computation time. But in case of proposed scheme public key $S$ depends on several parameter $(x, y, R, m, e, \beta)$, all the parameters included in $S$ are depend on each other so obtaining all the parameter itself is a big challenge, since $z$ cannot be factorised.

### 4.2. Complexity analysis

The time complexity is the computational factor that describes the volume of time it takes to run an algorithm. Table 1 summarizes number of elementary operations performed in each step of RSA compared against the proposed scheme using Big-O notation.

Table 1. Complexity analysis comparison RSA/ Generalized Pell's equation

| Parameters | Normal RSA | Generalized Pell's equation | | |
|---|---|---|---|---|
| | | $M$=2 | $M$=3 | $M$=5 |
| $\emptyset(n)$ | $O(n^2)$ | $O(n^3)$ | $O(n^3)$ | $O(n^3)$ |
| $E$ | $O(\log n)$ | $O(\log n)$ | $O(\log n)$ | $O(\log n)$ |
| $D$ | $O(n^2)$ | $O(n^2)$ | $O(n^2)$ | $O(n^2)$ |
| Encryption | $O((\log n)^2)$ | $O((\log n)^2)$ | $O((\log n)^2)$ | $O((\log n)^2)$ |
| Decryption | $O((\log n)^2)$ | $O((\log n)^2)$ | $O((\log n)^2)$ | $O((\log n)^2)$ |

This indicates that there is no growth in time for enciphering and deciphering processes. The time used for the generation of key will increase exponentially as per the value of $m$, however there is no change in the timing during generation of key in case of $e$ and $d$. Time complexity in case of evaluation of $\emptyset(n)$ is $O(n^3)$.

### 4.3. Mathematical attacks

Mathematical attacks emphasise on forging the principal arrangement of the mathematical function. Since RSA is prone to mathematical attack, major attacks under this category are Fermat's factorization attack, Trial and Division attack and Wiener's attack which is explained in the following subsections with explanations for the same.

#### 4.3.1. Fermat's attack

Fermat's factorization method uses the fact that the difference between two squares expresses any number. Fermat's factorization method factors $n$ if the gap between $p_i$ and $q_i$ is below the square root of $p_i$. In standard RSA algorithm, Fermat's factoring method which heuristically splits a composite number $n$ in $O(n^{\frac{1}{4}+e})$ steps. In this $n$ is represented as the difference of two rational squares for an integer approximation to $\sqrt{\frac{q_i}{p_i}}$ which provides an algorithm with complexity $O(n^{\frac{1}{2}+e})$. If a prime $m$ divides a square, then $m^2$ will also divide that square to achieve a heuristic speed-up to $O(n^{\frac{1}{4}+e})$ steps [42].

In RSA, the factorization time purely depends on difference between the prime numbers, and does not depend on the size of $n$. Factorization time increases with the increase of difference between the prime factors.

In proposed algorithm, we replace $n$ by fake modulus $z$. Since $z$ is a prime number, its factors are always $z$ and 1. In order to get $X + k = z$ and $X - k = 1$ values of $X$ and $k$ will be $\frac{n}{2} + 1$ and $\frac{n}{2}$, respectively. Thus, the value of $k$ starts with $\sqrt{n}$ and ends with $\frac{n}{2}$.The number of loops or steps it takes to terminate the execution is $\frac{n}{2} - \sqrt{n}$. Therefore, the complexity of Fermat's algorithm when $n$ is substituted by a prime number $z$, is $O(n\ )$.

Let $E_1$ be the event to factorise the prime factors of $z$ in case of proposed system, $n$ in case of RSA. It is evident that in case of RSA factorization of $n$ takes $O(n)$, however in the proposed system, factorizing $z$ is impossible since it is a prime number, therefore $E_1$ is infinity. Let $E_2$ be the factorization of $\emptyset(n)$ in case of RSA and $\emptyset(z)$ in case of proposed system. $\emptyset(n)$ is defined as $(p_i - 1)\times(q_i - 1)$, which is less than $z – 1$ where $\emptyset(z)$ is defined as $z – 1$, since $z$ is a prime number which indicates to factor the proposed Euler's totient function is harder than RSA.

### 4.3.2. Trail and Division

In RSA, the Trial division method varies from 1 to $\sqrt{n} - 1$, since $n$ can be divided. The proposed method uses $z$ in place of $n$ where $z$ is a prime number, the division of prime number is not possible. Hence,trial division method fails.

### 4.3.3. Wieners Attack

Polynomial-time attack on a RSA cryptosystem is considered as Wiener's attack which uses a small secret deciphering exponent $d$, which works if $d < n^{1/4}$, where the modulus of the cryptosystem is $n = pq$, to exploit the loophole of RSA continued fraction method is used.

Let $n$=$p_i \times q_i$ with $q_i$<$p_i$<$2q_i$ and $d$ < ($n^{0.25}$)/3) by sharing public key component ($n$, $e$) with $ed$≡1mod($\emptyset(n)$), which leads to recovering $d$ very easily. In that case, $d$ is convergent and the denominator $\frac{p_i - m}{q_i - m}$ of the continued fraction expansion of $\frac{e}{n}$ and therefore computation of $d$ from the public key$(n, e)$ is effective. When $d$ is few bits longer than $n^{0.25}$ numerous variants of Wiener's attack are proposed which breaks the RSA cryptosystem which contains the run-time complexity $O(D^2)$, where

$$d = D * (\ n^{0.25}),\ D = \frac{d}{n^{0.25}}.$$

The complexity is $O(D^2)$and can also be written as $O\left(\frac{d^2}{n^{0.125}}\right)$.

In Proposed model, $n$ is replaced by $z$ and it is the fake modulus, since $n$ is not related to $z$, and $z$ is also a prime number, it is imposible to find the factors of $z$, hence Wieners Attack can be foiled.

### 4.4. Experimental results

Experiment is carried out and tested by using Lenna monographic image by taking integer exponent key $m$ = 5 pair of keys are generated. Public key $S$ = 3,130,357 and fake modulus $z$ = 8,870,401 encryption is done and obtaining plain text back by using private keys $d$ = 1,364,677 and fake modulus $z$ = 8,870,401. Plain image and cipher image is exposed in Fig 3a and b. It is evident by the visual observation that no trace of plain image is available in the cipher image.

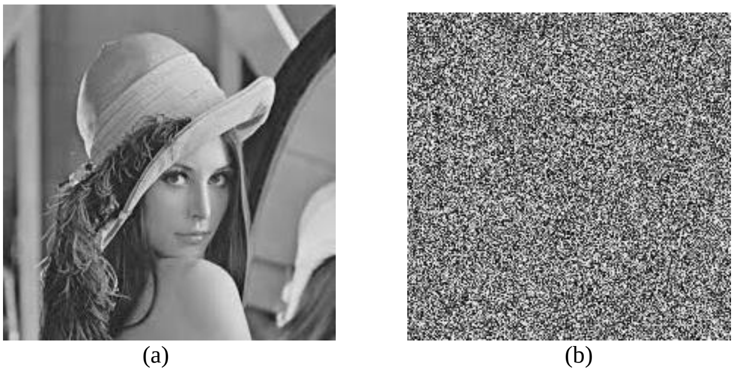

Fig. 3. Plain image (a); Encrypted image (b)

The number of occurrences of pixels of plain image is plotted against all the values of the plain image pixel value. This is shown in Fig. 4a. After encryption for the cipher image, the same is plotted and histogram is shown in Fig. 4b. In cipher image histogram, it is observed that occurrences of all the pixels are equiprobable indicating resistance to immunity.

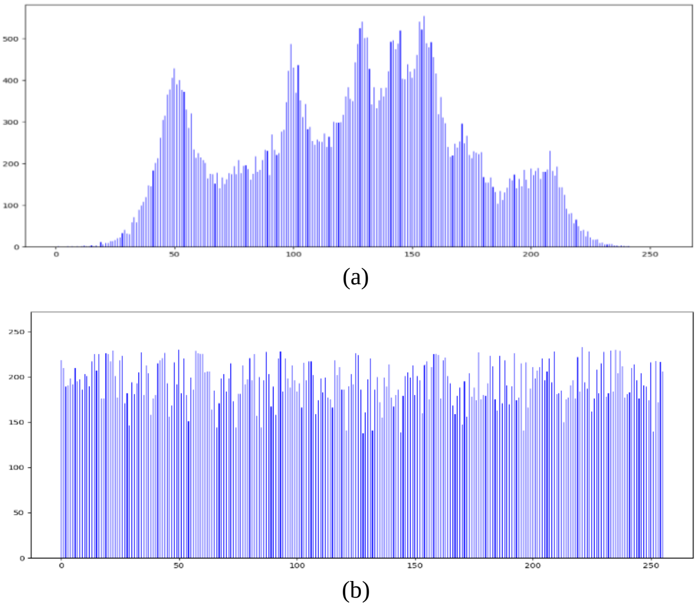

Fig. 4. The number of occurrences of pixels of plain image (a); The number of occurrences of pixels of cipher image (b)

Similarly, the standard deviation of the number of occurrence of the plain image and cipher image for modulus *n* and fake modulus *z* is evaluated and plotted in Fig. 5.

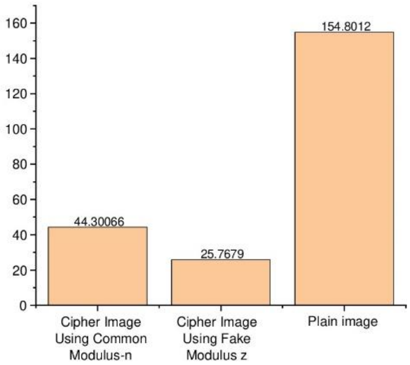

Fig. 5. Standard Deviation of plain image and cipher image for modulus n and fake modulus *z*

This indicates each pixel value has equal amount of distribution with flat spreading which makes the algorithm protected from intruders by different attacks.

### 4.5. Avalanche effect

The avalanche effect is a necessary property for all purposes of cryptographic algorithms. It causes dynamically increasing significant changes as the information spreads in the structure of the algorithm. Consequently, a piece or bit of the original image, obtaining huge rate of change in the encrypted image [43]. It is explained using the equation

$$\text{Avalache Effect(AE)} = \left(\frac{\sum_i \text{bit change}}{\sum_i \text{bit total}}\right) \times 100. \tag{28}$$

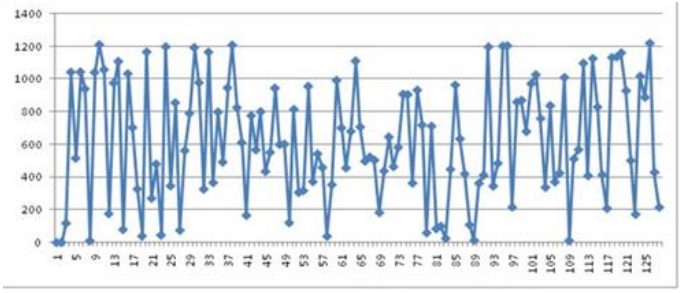

Fig. 6. Avalanche effect of proposed scheme

Fig. 6 shows a small change in the original image which leads to a tremendous change in the cipher text, which in turn makes it hard to decrypt the image and obtain the original image back.

## 5. Conclusion

Protection of information plays a vital role in day to day life. In this paper, new variant of RSA using generalized Pell's equation is experimented and explained. It is explored in the results that RSA is prone to factorization problem, hence intruder can break the private key $d$ using minimal computation time. But in case of proposed scheme, public key $S$ depends on several parameter $(x, y, R, m, e,\ \beta)$, all the parameters included in *S* are dependent on each other parameter, hence obtaining all the parameters itself is a big challenge.Computational complexity of the anticipated scheme is compared and analysed with RSA and indicates that there is no increase in time for encipher and deciphering process. Proposed model proved to be better compared to RSA as it uses Fake modulus *z*, which makes the system immune against mathematical attacks. Experimental results conducted using Standard Deviations, Histogram analysis which indicates occurrences of each pixel value have equal occurrence with flat distribution making the algorithm immune from attacks such as Fermat's factorization, Trial and division and Wiener's attack. Finally, this article shows that proposed cryptosystem is indeed as intractable as the factorization problem.

## References

1. Ranjan, K. S., S. P. Fathimath, G. Aithal, S. Shetty. A Survey on Key(s) and Keyless Image Encryption Techniques. – Cybernetics and Information Technologies, Vol. **17**, 2017, No 4, pp. 134-164.
2. Das, S. B., S. K. Mishra, A. K. Sahu. A New Modified Version of Standard RSA Cryptography Algorithm. – In: Smart Computing Paradigms: New Progresses and Challenges. Advances in Intelligent Systems and Computing. Vol. **767**. Singapore, Springer, 2020, pp. 281-287.
3. Raghunandan, K. R., G. Aithal., S. Shetty. Comparative Analysis of Encryption and Decryption Techniques Using Mersenne Prime Numbers and Phony Modulus to Avoid Factorization Attack of RSA. – In: Proc. of International Conference on Advanced Mechatronic Systems, Kutsugu, Japan, 2019, pp. 152-157.
4. Zheng, Y.-H., Y.-F. Zhu, H. Xu. An Application of Low Private Exponent Attack on RSA. – In: Proc. of 4th International Conference on Computer Science & Education, Nanning, 2009, pp. 1864-1866.
5. May, A. Secret Exponent Attacks on RSA-Type Schemes with Moduli $N$=$p^rq$. – In: F. Bao, R. Deng, J. Zhou, Eds. Public Key Cryptography – PKC 2004. PKC 2004. Lecture Notes in Computer Science. Vol. **2947**. Berlin, Heidelberg, Springer, 2004, pp. 352-360.
6. Diffie, W., M. Hellman. New Directions in Cryptography. – In: IEEE Transactions on Information Theory, Vol. **22**, November 1976, No 6, pp. 644-654.
7. Plantard, T., W. Susilo. Broadcast Attacks against Lattice-Based Cryptosystems. – In: M. Abdalla, D. Pointcheval, P. A. Fouque, D. Vergnaud, Eds. Applied Cryptography and Network Security. ACNS 2009. Lecture Notes in Computer Science. Vol. **5536**. Berlin, Heidelberg, Springer, 2009, pp. 456-472.
8. Hinek, M. Jason, M. K. Low, E. Teske. On Some Attacks on Multi Prime RSA. – In: Proc. of 9th Annual International Workshop at Selected Areas in Cryptography, Lecture Notes in Computer Science. Vol. **2595**. St. John's, Newfoundland, Canada. 2002, pp. 385-404.
9. Nitaj, A., Y. Pan, J. Tonien. A Generalized Attack on Some Variants of the RSA Cryptosystem. – In: C. Cid, J. M. Jacobson, Eds. Selected Areas in Cryptography – SAC 2018. SAC 2018. Lecture Notes in Computer Science. Vol. **11349**. Cham, Springer, pp. 3-26.
10. Bleichenbacher, D. Chosen Ciphertext Attacks against Protocols Based on the RSA Encryption Standard PKCS #1. – In: H. Krawczyk, Ed. Advances in Cryptology CRYPTO'98. CRYPTO 1998. Lecture Notes in Computer Science. Vol. **1462**. Berlin, Heidelberg, Springer, 1998, pp. 1-12.
11. Hinek, M., J., C. C. Y. Lam. Common Modulus Attacks on Small Private Exponent RSA and Some Fast Variants. – Journal of Mathematical Cryptology, 20 January 2009.
12. Zhao, Y. D., W. F. Qi. Small Private-Exponent Attack on RSA with Primes Sharing Bits. – In: J. A. Garay, A. K. Lenstra, M. Mambo, R. Peralta, Eds. Information Security. ISC 2007. Lecture Notes in Computer Science. Vol. **4779**. Berlin, Heidelberg, Springer, 2007, pp. 221-229.

13. Blömer, J., A. May. A Generalized Wiener Attack on RSA. – In: F. Bao, R. Deng, J. Zhou, Eds. Public Key Cryptography – PKC 2004. Lecture Notes in Computer Science. Vol. **2947**. Berlin, Heidelberg, Springer, 2004, pp. 1-13.
14. Nguyen, T. D., T. D. Nguyen, L. D. Tran. Attacks on Low Private Exponent RSA: An Experimental Study. – In: Proc. of International Conference on Computational Science and Its Applications, 2013, pp 162-165.
15. Coppersmith, D. Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities. – Journal of Cryptology, Vol. **10**, 1997, No 4, pp. 233-260.
16. Nitaj, A., M. R. K. Ariffin, D. I. Nassr, H. M. Bahig. New Attacks on the RSA Cryptosystem. – In: D. Pointcheval, D. Vergnaud, Eds. Progress in Cryptology – AFRICACRYPT'2014. Lecture Notes in Computer Science. Vol. **8469**. 2014, pp 178-198.
17. Coppersmith, D., M. Franklin, J. Patarin, M. Reiter. Low-Exponent RSA with Related Messages. – In: U. Maurer, Ed. Advances in Cryptology – EUROCRYPT'96. EUROCRYPT. Lecture Notes in Computer Science. Vol. **1070**. Berlin, Heidelberg, Springer, 1996, pp. 1-9.
18. Boneh, D. Twenty Years of Attacks on the RSA Cryptosystem. – In: Notices of the American Mathematical Society, Vol. **46**, 1999, pp. 203-213.
19. Mumtaj, M., L. Ping. Forty years of attacks on the RSA cryptosystem: A Brief Survey. – Journal of Discrete Mathematical Sciences and Cryptography, Vol. **22**, 2019, No 1, pp. 9-29.
20. Boneh, D., G. Durfee. Cryptanalysis of RSA with Private Key d Less than $N^{0.292}$. – Proc. of EUROCRYPT'99, IACR, Springer-Verlag, LNCS, Vol. **1592**, 1999, pp. 1-11.
21. Durfee, G., P. Q. Nguyen. Cryptanalysis of the RSA Schemes with Short Secret Exponent from Asiacrypt '99. – In: T. Okamoto, Ed. Advances in Cryptology – ASIACRYPT 2000. ASIACRYPT 2000. Lecture Notes in Computer Science. Vol. **1976**. Berlin, Heidelberg, Springer, 2000, pp. 14-29.
22. Sun, H.-M., M.-E. Wu, W.-C. Ting, M. J. Hinek. Dual RSA and Its Security Analysis. – Information Theory, IEEE Transactions, Vol. **53**, 2007, pp. 2922-2933.
23. Al-Hamami, A. H., I. A. Aldariseh. Enhanced Method for RSA Cryptosystem Algorithm. – In: Proc. of International Conference on Advanced Computer Science Applications and Technologies (ACSAT'12), Kuala Lumpur, 26-28 November 2012, pp. 402-408.
24. Thangavel, M., P. Varalakshmi, M. Murrali, K. Nithya. An Enhanced and Secured RSA Key Generation Scheme (ESRKGS). – Journal of Information Security and Applications, Vol. **20**, 2015, pp. 3-10.
25. Tuteja, A., A. Shrivastava. Implementation of Modern RSA Variants. – International Journal of Computer Science and Information Technologies (IJCSIT), Vol. **5**, 2014, No 6, pp. 7493-7495.
26. Jaju, S. A., S. S. Chowhan. A Modified RSA Algorithm to Enhance Security for Digital Signature. – In: Proc. of International Conference and Workshop on Computing and Communication (IEMCON'15), 2015, pp. 1-5.
27. Segar, T. C., R. Vijayaragavan. Pell's RSA Key Generation and Its Security Analysis. – In: Proc. of 4th International Conference on Computing, Communications and Networking Technologies (ICCCNT'13), Tiruchengode, 2013, pp. 1-5.
28. Ya, S. Y. Integer Factorization Attacks. – In: Cryptanalytic Attacks on RSA. Boston, MA, Springer, 2008, pp. 91-110.
29. Zhang, H., T. Takagi. Attacks on Multi-Prime RSA with Small Prime Difference. – In: C. Boyd, L. Simpson, Eds. Information Security and Privacy. ACISP 2013. Lecture Notes in Computer Science. Vol. **7959**. Berlin, Heidelberg, Springer, 2013, pp. 41-56.
30. Maitra, S., S. Sarkar. Revisiting Wiener's Attack – New Weak Keys in RSA. – In: T. C. Wu, C. L. Lei, V. Rijmen, D. T. Lee, Eds. Information Security. ISC 2008. Lecture Notes in Computer Science. Vol. **5222**. Berlin, Heidelberg, Springer, 2008, pp. 228-243.
31. Raghunandan, K. R., A. Ganesh, S. Surendra, K. Bhavya. Image Encryption Scheme in Public Key Cryptography Based on Cubic Pells Quadratic Case. – Indonesian Journal of Electrical Engineering and Computer Science, Vol. **20**, 2020, No 1, pp. 385-394. DOI: 10.11591/ijeecs.v20.i1.pp385-394.

32. Raghunandhan, K. R., S. Shetty, G. Aithal, N. Rakshith. Enhanced RSA Algorithm Using Fake Modulus and Fake Public Key Exponent. – In: Proc. of International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT'18), Msyuru, India, 2018, pp. 755-759.

33. Amitava, N., B. Sushanta, S. Debasree, S. Partha. Secret Image Sharing Scheme Based on a Boolean Operation. – Cybernetics and Information Technologies, Vol. **14**, 2014, No 2, pp. 98-113.

34. Barbeau, E. J. Pell's Equation Problem Books in Mathematics. – In: Springer, 2003, XII, 212. New York, Springer-Verlag, 1999. ISBN: 0-387-95529-1, DOI: 10.1007/b97610.

35. Chen, C. Y., C. C. Chang, W. P. Yang. Fast RSA Type Cryptosystem Based on Pell Equation. – In: Proc. of International Conf. on Cryptology and Information Security, Taiwan, 1-5 December 1996, pp. 1-5.

36. Padhye, S. A Public Key Cryptosystem Based on Pell Equation. Eprint Archive2005/109. **http://eprint.iacr.org/2006/191.pdf**

37. Rao, K. M., P. S. Avadhani, D. L. Bhaskari, K. Ssarma. An Identity Based Encryption Scheme based on Pell's Equation With Jacobi Symbol. – International Journal of Research in Engineering and Science (IJRES), Vol. **1**, 2013, No 1, pp. 17-20.

38. Burton, D. M. Elementary Number Theory. Sixth Edition. International Series in Pure and Applied Mathematics, University of New Hampshire, McGraw-Hill Higher Education, 2007.

39. Raghunandan, K. R., R. R. Dsouza, N. Rakshith, S. Shetty, G. Aithal. Analysis of an Enhanced Dual RSA Algorithm Using Pell's Equation to Hide Public Key Exponent and a Fake Modulus to Avoid Factorization Attack. – In: N. Chiplunkar, T. Fukao, Eds. Advances in Artificial Intelligence and Data Engineering. Advances in Intelligent Systems and Computing. Vol. **1133**. Singapore, Springer, pp. 809-823. **https://link.springer.com/chapter/10.1007%2F978-981-15-3514-7_60**

40. Raghunandan, K. R., S. N. Nireshwalya, S. Sudhir, M. S. Bhat, H. M. Tanvi. Securing Media Information Using Hybrid Transposition Using Fisher Yates Algorithm and RSA Public Key Algorithm Using Pell's Cubic Equation. – In: N. Chiplunkar, T. Fukao, Eds. Advances in Artificial Intelligence and Data Engineering. Advances in Intelligent Systems and Computing. Vol. **1133**. Singapore, Springer, pp. 975-993. **https://link.springer.com/chapter/10.1007%2F978-981-15-3514-7_73**

41. Raghunandan, K. R., R. Shetty, G. Aithal. Key Generation and Security Analysis of Text Cryptography Using Cubic Power of Pell's Equation. – In: Proc. of International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT'17), Kannur, 2017, pp. 1496-1500.

42. Wu, M.-E. R. Tso, H.-M. Sun. On the Improvement of Fermat Factorization. – In: Proc. of 6th International Conference on Network and System Security (NSS'12), Berlin, Heidelberg, Springer-Verlag, 2012, pp. 380-391.

43. Rao, R., A. Ganesh, S. Surendra. Secure RSA Variant System to Avoid Factorization Attack Using Phony Modules and Phony Public Key Exponent. – International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol. **8**, 2019.

*Received: 25.01.2020; Second Version: 07.06.2020; Third Version: 16.06.2020; Accepted: 29.06.2020*