# A KIND OF DESIGN OF KNAPSACK PUBLIC KEY CRYPTOSYSTEM BASED ON CHAOTIC SYSTEM

Zhijuan DENG[1, 2], Shaojun ZHONG[1, 2]

*In this article, we first introduced the importance of the information security and the knapsack cryptosystem, and then we summarized several methods of generating the pseudo-random variables by chaos. After that, we put forward a kind of knapsack system based on chaotic system and addition, and added the pseudo-random variables generated by chaos to such knapsack system. The analysis indicates that the chaos has improved the security of the addition knapsack system. At last, we had a discussion on the algorithm performance, which turned out that this algorithm has several characteristics, for instance, it is very simple and easy, its security is high, and etc. Furthermore, this algorithm can be used to encrypt the binary data and can also be used to encrypt all kinds of files in the computers in theory, including figures and sounds. However, since the visual effects of the figures haven't been considered in the process of the encryption, the efficiency of encrypting those files with special format, such as figures, sounds, and etc., is lower than that of the dedicated encryption algorithm.*

**Keywords:** Cryptography, Chaos, Information Security, Threshold Method

## 1. The Importance of the Information Security

The twenty-first century is the era of information. On one hand, all kinds of information technologies emerge one after another and are updated rapidly, on the other hand, the information security has been threatened constantly. Since the information security matters to the national and social security and stability, we must take harsh measures to ensure the information security [1].

The information security involves many aspects, such as management, technology, law and other related issues. The key technologies are password and network security. Only if we ensure the security of the underlying hardware and software in the information system can we ensure the security of the whole information system [2].

To analyze from the technical point of view, the following reasons mainly cause such serious problems of the information security [3].

[1] School of Information Management & Engineering, Shanghai University of Finance and Economics, Shanghai, China

[2] Faculty of Science, Jiangxi University of Science and Technology, Ganzhou, China, E-mail: 66162815@qq.com

(1) The security structure of the microcomputer is too simple. At the end of the 20th century, the integrated circuit technique was developed rapidly, hence, the microcomputer, which is also called the personal computer, was invented. As the rapid development and popularization of the microcomputer, the manufacturers have removed many mature security mechanisms (such as the application security protection mechanism) in order to lower the cost, which caused the following problems: The program can be executed without certification, and the program and system data can be modified at random. Many viruses, Trojans, and other malicious programs take advantage of these loopholes exactly [4]. Information security is guaranteed by software systems.

(2) The development of the computer networks. With the development of the network, the range of the computer information exchange has expanded to the entire network. Since there are big potential security problems in the Internet network design, the security of the computers in the network world will be threatened to a greater extent. Because of the complexity of the networking protocol, it is difficult to keep the network safe and have it verified. At present, only some simple networking protocols can be proved and verified, therefore, the potential safety hazard is inevitable. Besides, even if the networking protocols are correct, there are still many attacks taking advantage of the correct networking protocols exactly. For instance, according to the philosophical principle of "quantitative change to qualitative change", the attackers will initiate lots of normal accesses, which will cause the exhaustion of computer or network resources, and thus paralyze the computers. The well-known DoS attack is a clear proof [5]. In order to ensure network security, changing the information transmission mechanism seems to do more harm than good, but improving the software system to ensure security is more feasible.

(3) There are security flaws in the operating system. Since personal computers rely on their operating systems, the security of operating systems is the foundation of computers. Nevertheless, there is a vast collection of programs in the operating system, so it cannot be one hundred percent correct. We can usually ignore the functional faults caused by the defects of the operating systems, for instance, if the Windows system is halted, it can be rebooted. However, if the attackers take advantage of the defects of the operating systems for attacks, it will cause serious potential safety hazard [6].

Knapsack plays an important role in public key cryptosystem. After introducing the basic knapsack scheme, we propose a chaotic encryption system based on chaotic system proposed by Matthews, which integrates chaotic and additive operations. After that, we analyze the scheme, which has the characteristics of fast encryption and decryption speed and high security. However, due to the particularity of image format, this encryption mechanism is not suitable for the application of image encryption.

## 2. The Development History of the Knapsack Cryptosystem

In 1976, Diffie and Hellman put forward the new concept of the public key cryptosystem in their article [7-10], which laid the foundation of the public key cryptography. In this article, Diffie and Hellman put forward the concept of the trapdoor one-way function, and in the meanwhile, they indicated that some NP problems had the properties of the trapdoor one-way function, which can be used to construct information security system based on public key encryption.

Under the guidance of Diffie and Hellman's thoughts, many public key encryption systems were established. They can be roughly divided into two categories. One is based on the number theory problem, such as the big prime number factorization; while the other is based on the knapsack problem. The encryption system on the basis of the number theory problem has high security, and yet its encryption and decryption speed is relatively slow. While the speed of the encryption system on the basis of the knapsack problem is relatively fast, and yet its security is relatively low. With the emergence of a variety of decoding algorithms [11], many 0-1 knapsack systems have been decoded. However, the advantage of its speed still attracts people to make continuous improvement.

### 2.1. The Description of the Knapsack Problem Escape

From the description above, we can get to know the general description of the knapsack problem: To take out several pieces from M items and to put them into the knapsack whose space is W, the weight of each item is $w_1, w_2, \ldots, w_n$, and their corresponding value is $p_1, p_2, \ldots, p_n$. Find out the option of the selected item with the maximum value ($\sum p_i$) and the total weight not exceeding W ($\sum w_i < W$) [12].

With the continuous study of knapsack problem, in the subsequent knapsack public key system, the problem description adopted, which is different from the general description. It is based on its subproblem—subset sum problem, that is to say, whether the specific subset, where the sum (or other operations) of elements is exactly some certain natural number, can be selected in a given and definitive set of natural numbers. In other words, presetting a natural number vector $(a_1, a_2, \ldots, a_n)$ (since the weight is generally defined as the value here, $a_i$ can be called as weight or value), and an integer $S$, and looking up a binary vector $\langle x_1, x_2, \ldots, x_n \rangle$, which can satisfy the specific conversion. For example

$$\sum_{i=1}^{n} a_i \cdot x_i = S \tag{1}$$

### 2.2. The Basic Idea of the Knapsack Cryptosystem

The basic idea of the knapsack system is constituted as the following steps [13-15]:

Step1. Find a simple knapsack (generally can be solved by linear time).

Step2. Look for a trapdoor to convert the simple knapsack into the difficult knapsack.

Step3. Announce the difficult knapsack.

Stip4. Ensure that the decryption methods for legitimate decryptors and cryptanalysts are inconsistent. So, the cipher analysts have to solve the NP difficult problem, while the legal code breakers can turn to solve the simple knapsack problem by means of the trap door, that is to say, they only have to solve the linear solvable problem.

### 2.3. MH System

The MH system is a relatively mature knapsack public key cryptosystem, which describes two basic methods of constructing the knapsack system.

(1) The MH system based on the modular multiplication

The basic idea of this method is to select a simple knapsack first, and then choose trapdoor information, so that the easy-to-solve knapsack can be turned into the difficult-to-solve knapsack problem. If the trapdoor information is known, the difficult-to-solve knapsack problem becomes the easy-to-solve knapsack problem. Therefore, MH method is also called trapdoor knapsack cryptography method.

The design idea of trapdoor knapsack is almost always constructed from a simple knapsack problem. The easy-to-solve knapsack problem is disguised as a seemingly difficult knapsack problem, and the disguised method is trapdoor information. The legitimate recipient Alice can restore the problem to a knapsack problem because he has the trapdoor information. By solving the knapsack problem, Alice can reconstruct plaintext. For the illegal recipient Eve, restoring plaintext from ciphertext means solving a difficult knapsack problem.

(2) The MH system based on the logarithmic module

Since the method 1 is relatively easy to be understood, it is usually the only method referred to as the MH system in the books. As a matter of fact, another method is also described in the MH system for constructing the knapsack sequence, and the construction method is as follows. At first, we use the natural numbers which are pairwise relatively prime to construct the simple knapsack and use the logarithmic module instead of the strong modular multiplication to change the simple knapsack into the difficult knapsack. In the process of encryption, we use the difficult knapsack to calculate the subset sum, which can be changed into the vector product of the simple knapsack by conversion in the process of decryption, and we make use of the relatively prime property of the simple knapsack for solution.

In order to make sure that the legal code breakers can restore the ciphertexts back to the plaintexts in linear time, we use the knapsack sequences which are pairwise relatively prime as the simple knapsack in this type of

knapsack algorithm. When the ciphertext information S is the subset product of this sequence, we only have to verify the divisibility of S for the knapsack sequence vector elements one by one. If $S = 0 \mod b_i$, the place value of $b_i$ is 1, otherwise, the place value of $b_i$ is 0. So, we can get the plaintext of the information in a single pass of the simple knapsack.

In order to improve the security and the difficult level of the decryption, those two methods above can both be iterated for many times. And we use new m and w for each change in the iteration. Actually, if we use the strong modular multiplication in the iteration every time, the information rate of the ciphertexts and the density of the difficult knapsacks will be decreased gradually [21].

## 3. The Pseudo-random Vectors Generated by the Matthews Chaotic System

The British mathematician Matthews studied the chaotic encryption method, and put forward the general expression of a kind of chaotic system:

$$y_{n+1} = f(y_n) = (\beta + 1)\left(1 + \frac{1}{\beta}\right)^{\beta} y_n (1 - y_n)^{\beta} \tag{2}$$

When satisfying $1 \le \beta \le 4$, it can be called as the Matthews chaotic system [16].

Definition: The vector $Y = (y_1, y_2, \ldots, y_n)$ is called as the chaos random vector, which means that $y_i (i = 1,2,\ldots,n)$ is some iterative value $y_i$ of the chaotic system $y_{n+1} = f(\beta, y_n)$, and some kind of pseudo-random number determined according to the specific rules.

The sequence of iterations determined by the chaotic system usually has better randomness than the general pseudo-random vectors. Let us discuss the method of each component in the chaos random vectors constructed based on the Matthews chaotic system next.

### 3.1. The Method of the Decimal Part Interception

As for the given $\beta, y_0$, we calculate the chaos sequence $\{y_n\}_{n=1}^{\infty}$ by iteration; after truncating the first $n$ iterative values, we will get the new sequence $\{y_i\}_{i=1}^{\infty}$, if $y_i$ has decimal part, we might as well have it recorded as

$$y_i = \alpha_0^{(i)} \alpha_1^{(i)} \alpha_2^{(i)} \cdots \alpha_j^{(i)} \cdots \alpha_k^{(i)} \cdots \tag{3}$$

In which $\alpha_j^{(i)}$ represents the $j^{th}$ decimal of $x_i$, therefore, we can take several digits of the decimal part to form the decimal number:

$$x_i = \alpha_j^{(i)} \alpha_{j+1}^{(i)} \cdots \alpha_k^{(i)} \tag{4}$$

In which $\alpha_j^{(i)}$ represents the jth decimal of $x_i$.

If we take $y_0 = 0.35760000$, $\beta = 1.2$, we can get $y_3 = 0.39735923$ by the Matthews chaotic system, and the random number generated by such method will be 3973.

### 3.2. Complementation

As for the fixed $\beta$ and the initial value $y_n$, the chaotic system $y_{n+1} = f(\beta, y_n)$ generates the sequence $\{y_n\}_{n=1}^{\infty}$ by iteration. After truncating the first n iterative values, we will get the new sequence $\{y_i\}_{i=1}^{\infty}$. As for the fixed $i$, if $y_{n+1}$ is decimal fraction, we will have it converted into the two's complement requiring digit in advance:

$$y_i = \left(y_1^{(i)} y_2^{(i)} \cdots y_k^{(i)}\right)_2 \tag{5}$$

For example: $y_3 = 0.39735923$ is a value generated by the Matthews chaotic system in the chaotic sequence $x_n$, and the process of converting it into the two's complement is shown in the Table 1.

*Table 1*

**The Process of Converting into the Two's Complement by the Chaotic Sequence**

| ***The Digit of the Binary Code*** | ***Calculated Value*** | ***The Corresponding Binary Code*** |
|---|---|---|
| 1 | 0.39735923*2=0.79471846 | 0 |
| 2 | 0.79471846*2=1.58943692 | 1 |
| 3 | 0.58943692*2=1.17887384 | 1 |

Therefore, the two's complement of $y_3 = 0.39735923$ is 011…, and the length can be taken as required. If some value in the $y_n$ sequence is minus, we can convert it into the absolute value first, and then calculate its complement. As a result, $x_i = (011)_2 = (3)_{10}$.

### 3.3. The Threshold Method

As for the fixed $\beta$ and the initial value $y_0$, the chaotic system $y_{n+1} = f(\beta, y_n)$ generates the sequence $\{y_n\}_{n=1}^{\infty}$ by iteration. After truncating the first $N_i$ iterative values, we will get the new sequence $\{y_i\}_{i=1}^{\infty}$. As for the length $k$ required in advance, we can select $y_1, y_2, \cdots, y_k$, and thus calculate the pseudo random number.

$$x = (x_1, x_2, \cdots x_k)_2, \qquad x_k = \begin{cases} 1 & y_k \geq \mu \\ 0 & y_k < \mu \end{cases} \tag{6}$$

For example, after taking $y_0 = 0.35760000$, $\beta = 1.2$, we can get $y_1 = 0.09888611, y_2 = 0.39735923, y_3 = 0.98529418$, $y_4 = 0.02837039$, and $y_5 = 0.12478965$ by the Matthews chaotic system. After taking $\mu = 0.5$, we can get the random number $x = (00100)_2 = 4$.

### 3.4. The Method of the Vector of the Parameter $\beta$

As for the given initial value $y_0$ and the vector $(\beta_1, \beta_2, \cdots, \beta_m)$, $\beta_i$ of the parameter $\beta$, therefore, we can get the sequence $\{y_n^{(\beta_i)}\}_{n=1}^{\infty}$ iterated respectively by m systems

$$y_{n+1}^{(\beta_i)} = f(\beta_i, y_n^{(\beta_i)}), \qquad y_0^{(\beta_i)} = y_0, \qquad (i = 1,2,\ldots,m) \tag{7}$$

As for the fixed $i$, we can get the corresponding pseudo-random sequence adopting the method 3.1, 3.2 and 3.3. After taking $y_0 = 0.35760000$ here, we can get the following data by the Matthews chaotic system. The random vector can be generated combining with the method 3.1, the results are shown in the Table 2, and the vector generated from it is $(x_1, x_2, x_3) = (3973, 0878, 6)$.

*Table 2*

**In which $x_i$ is the random vector achieved by the first method**

| $\beta$ | $y_3$ | $x_i$ |
|---|---|---|
| 1.2 | 0.39735923 | $(3973)_{10}$ |
| 1.4 | 0.08785342 | $(0878)_{10}$ |
| 1.6 | 0.00068256 | $(0006)_{10}$ |

### 3.5. The Method of the Initial Value Vector

As for the fixed $\beta$ and the initial value vector $(y_0^{(1)}, y_0^{(2)}, \cdots, y_0^{(n)},)$, therefore, we can get the sequence $\{y_n^{(i)}\}_{n=1}^{\infty}$ iterated respectively by m systems $y_{n+1}^{(i)} = f(\beta, y_n^{(i)})$. As for the fixed $i$, we can get the corresponding pseudo-random numbers adopting the methods in 3.1, 3.2 and 3.3. And the chaos random sequence vectors generated by combining this method with the method in 3.1 are shown in the Table 3.

*Table 3*

**The Chaos Random Vectors Determined by the Method of the Initial Value Vector**

| $i$ | $y_0^{(i)}$ | $y_5^{(i)}$ | $x_i$ |
|---|---|---|---|
| 1 | 0.35760000 | 0.00460101 | $(0046)_{10}$ |
| 2 | 0.46214624 | 0.99868842 | $(9986)_{10}$ |
| 3 | 0.26435789 | 0.31034400 | $(3103)_{10}$ |
| 4 | 0.75646132 | 0.00189032 | $(0018)_{10}$ |
| 5 | 0.84235775 | 0.95576876 | $(9557)_{10}$ |

## 4. The Public Key Knapsack System Implemented Based on the Matthews Chaotic System

The idea of MH cryptosystem is valuable; Merkle-Hellman used this idea to transform super-incremental sequence into pseudo-random sequence by multiplication in 1978. So, another powerful encryption algorithm has been developed, called MK-KPKC. Although this algorithm is more secure, but it is unavoidable to compute multiplication inverse elements and modular operations when using this system, which will lead to a large amount of computation. In order to overcome the massive calculation of determining the multiplicative inverses in the MK-KPKC, we can convert the super increasing sequences into the pseudo-random sequences based on the additive operation. Therefore, we can get the new knapsack public key cryptosystem ADD-KPKC implemented based on the additive operation.

### 4.1. The Generation of the Secret Key

(1) The recipients select a super increasing sequence $B=(b_1,b_2,\ldots,b_n)$, a random vector $X=(x_1,x_2,\ldots,x_n)$, and an integer $P$ at random, which satisfy $P \succ \sum_{i=1}^{n} b_i$.

(2) The sender converts a super-increasing sequence $B$ into a pseudo-random sequences $A=(a_1,a_2,\ldots,a_n)$, such that $A=B+PX$, that is to say, $a_i$ satisfies $a_i=b_i+Px_i(i=1,2,\cdots,n)$.

(3) The sender publishes $A$ as a Personal Public Key, and publishes $B$ and $P$ as the private keys, and destroys the random vector $X$.

Where the random vector $X$ is a chaos sequence. This is mainly because chaotic maps have good stochastic properties. The ontology system uses the iteration formula of chaotic maps proposed by Matthews to obtain random sequences. Since the encrypted quantity is digital, a method has to be used to map the sequence $\{X_n\}$ consisting of real numbers to a pseudo-random sequence consisting of integers. The simplest kind of mapping is to select several significant digits after the decimal point of $\{X_n\}$ to form an integer.

### 4.2. The Encryption Operation

If $B$ wants to send the plaintext $M=[m_1,m_2,\ldots,m_n]$to $A$, $B$ will find out $A$'s public key $A=[a_1,a_2,\ldots,a_n]$ first, will work out $S=\sum_{i=1}^{n} m_i a_i$, and then will send it to $A$ as the ciphertext.

### 4.3 The Decryption Operation

(1) After receiving the ciphertext $S$, $A$ will convert $S$ into $S'$ with the private key $P$,

$$\begin{aligned} S' &\equiv s \bmod p \equiv \sum_{i=1}^{n} m_i a_i \bmod p \equiv \sum_{i=1}^{n} m_i (b_i + x_i P) \bmod p \\ &\equiv \sum_{i=1}^{n} (m_i b_i + m_i x_i P) \bmod p \equiv \sum_{i=1}^{n} m_i b_i \bmod p \end{aligned} \tag{8}$$

(2) Since $B = (b_1, b_2, \ldots, b_n)$ is the super increasing sequence, the plaintext $M = [m_1, m_2, \ldots, m_n]$ can be easily worked out with the greedy algorithm by $S'$. And the greedy algorithm herein is also called the super increasing sequence algorithm.

## 5. The Algorithm Example and Comparison

### 5.1. The Algorithm Example

Suppose the super increasing sequence selected by the recipient $A$ is $B = (b_1, b_2, b_3, b_4, b_5) = (132, 173, 313, 641, 1279)$. And the sender $B$ will send the plaintext $M = (1,0,0,1,1)$ to $A$. The procedures of implementing the above algorithms are as follows:

(1) The generation of the random vector

We select the random vector Y= (46, 9986, 3103, 18, 9557) generated in the previous Table 3.

(2) The generation of the secret key

Step1. The recipient selects the integer $P = 6311 > \sum_{i=1}^{n} b_i$ at random, and makes use of the random vector $Y = (y_1, y_2, y_3, y_4, y_5) = (46, 9986, 3103, 18, 9557)$ generated before at the same time. Then

$$\begin{aligned} PY &= (6311 \cdot 46, 6311 \cdot 9986, 6311 \cdot 3103, 6311 \cdot 18, 6311 \cdot 9557) \\ &= (833052, 1091803, 1975343, 4045351, 8071769) \end{aligned}.$$

Step2. $A$ converts the super increasing sequence $B$ into the pseudo-random sequence $A$.

$$A = (a_1, a_2, a_3, a_4, a_5) = B + PY = (833184, 1091976, 1975656, 4045992, 8073048)$$

Step3. A announces the pseudo-random sequence $A$, hides the super increasing sequence $B$ and the integer $P$ privately, and destroys the random vector $Y$.

(3) The encryption operation

$$S = \sum_{i=1}^{5} m_i a_i = 12952224$$

At present, the plaintext is $M = (1,0,0,1,1)$, therefore, the ciphertext $S = \sum_{i=1}^{5} m_i a_i = 12952224$ can be calculated only if the public key $A$ is found.

(4) The decryption operation

Step1. When receiving the ciphertext $S = 12952224$ , the recipient $A$ finds out the secret key $P = 6311$ hidden privately, and calculates

$S' \equiv S \bmod p \equiv 12952224 \bmod 6311 \equiv 2052$ .

Step2. The plaintext $M = (1,0,0,1,1)$ can be worked out with the greedy algorithm by using the super increasing sequence $B$ hidden privately.

### 5.2. Comparison

In this section, we present a comparison between our proposed scheme and few other code-based and knapsack-based cryptosystems.

We compared our scheme with 4 encryption/decryption schemes. They are LDLC-Based [17], QC-LDPC [18], Bernstien [19], and NPKC-MHKC [20], respectively. The comparative indicators includes: public key size, private key size, encryption speed, decryption speed and message expansion, which are the main indicators to measure the encryption/decryption algorithm.

In terms of key space (public and private), LDLC-Based [17] and QC-LDPC [18] use matrix to storage the keys, Bernstien [19] uses communication datagram, NPKC-MHKC [20] and ours use character string. Hence, the key size of NPKC-MHKC [20] and ours is minimum. In terms of speed (encryption and decryption), except for our scheme, all other schemes use multiplication. Hence, the speed of our scheme is minimum. However, LDLC-Based [17] and QC-LDPC [18] provided message expansion, and only a few bytes are used. Other schemes are not provided, including ours. This comparison is given in table 4.

*Table 4*

**Comparing some code-based and knapsack-based cryptosystems (key size is given in bits).**

| | **LDLC-Based [17]** | **QC-LDPC[18]** | **Bernstien [19]** | **NPKC-MHKC [20]** | **New Scheme** |
|---|---|---|---|---|---|
| Public key size | $O(n^2)$ | $O(n^2)$ | $O(n \log n)$ | $O(n)$ | $O(n)$ |
| Private key size | $O(n^2)$ | $O(n^2)$ | $O(n \log n)$ | $O(n)$ | $O(n)$ |
| Encryption Speed | $O(n^2)$ | $O(n^2)$ | $O(n^2 \log n)$ | $O(n^2)$ | $O(n)$ |
| Decryption Speed | $O(n^2)$ | $O(n^2)$ | $O(n^2 \log n)$ | $O(n^2)$ | $O(n)$ |
| Message expansion | Provided | Provided | Not provided | Not provided | Not provided |

## 6. The Discussion on the Algorithm Application in the Image Encryption

From the above algorithm example, we can notice that the ADD-KPKC has the following advantages:

(1) Simplification: Unlike MK-KPKC, it doesn't need to solve the multiplicative inverse or to implement large amount of modular arithmetic, and it also doesn't need to spend plenty of time in selecting $W, P$ to make $\gcd(W, P)=1$, which considerably reduces the operation time of encryption and decryption beyond doubt.

(2) Security: When constructing the pseudo-random sequence, since we use the n-dimensional random vector $X=(x_1, x_2, \ldots, x_n)$ generated by the chaos function, which can cover the "trace" of the super increasing sequence well, it is safer than MK-KPKC.

From the principle of this algorithm, we get to know that this algorithm can be used to encrypt the binary system. That is to say, it can encrypt all computer files in theory, including images, sounds, and other files. However, the various types of files have their own characteristics, for instance, the images have the characteristic of visualization, the sounds are meaningful composition of words, and etc. Even though this algorithm can encrypt the images, it doesn't take the visual effects of images into consideration. Therefore, its encryption efficiency of images is lower than that of the specialized algorithms.

## R E F E R E N C E S

[1]. *Iregbenu P C, Uzonwanne M C.* Security Challenges and Implications to National Stability. Social Science Electronic Publishing, 6(4):169-175. 2015

[2]. *Siewe F.* Towards the modelling of secure pervasive computing systems: A paradigm of Context-Aware Secure Action System. Journal of Parallel & Distributed Computing, 87(43):121-144.2016

[3]. *Pissanetzky S.* On the Future of Information: Reunification, Computability, Adaptation, Cybersecurity, Semantics. IEEE Access, 4(18):1117-1140.2016

[4]. *Muthumanickam K, E. I E.* CoPDA: Concealed process and service discovery algorithm to reveal rootkit footprints. Malaysian Journal of Computer Science, 28(1):1-15. 2015

[5]. *Saied A, Overill R E, Radzik T.* Detection of known and unknown DDoS attacks using Artificial Neural Networks. Neurocomputing, 172(C):385-393. 2016

[6]. *Burk R A, Kallberg J.* Cyber Defense as a part of Hazard Mitigation: Comparing High Hazard Potential Dam Safety Programs in the United States and Sweden. Journal of Homeland Security & Emergency Management, 13(1):77-94. 2016

[7]. *Konheim A G.* The Impetus to Creativity in Technology. Cryptologia,39(4):291-314. 2015

[8]. *Choo K K R, Domingo-Ferrer J, Zhang L.* Cloud Cryptography: Theory, Practice and Future Research Directions. Future Generation Computer Systems, 62:51-53. 2016

[9]. *Cash D, Küpçü A, Wichs D.* Dynamic Proofs of Retrievability via Oblivious RAM. Journal of Cryptology, 30(1):22-57. 2017

[10]. *Hart,Hedtke S, Müller-Hannemann I.* A fast search algorithm for (m,m,m)Triple Product Property triples and an application for 5×5 matrix multiplication. Groups Complexity Cryptology, 7(1):31-46. 2015

[11]. *Choi D, Matni Z, Shah C.* What social media data Should I use in my research?: A comparative analysis of twitter, youtube, reddit, and the new york times comments. Proceedings of the Association for Information Science & Technology, 53(1):1-6. 2016

[12]. *Ghofrani A, Gaba S, Payvand M, et al.* A Low-Power Variation-Aware Adaptive Write Scheme for Access-Transistor-Free Memristive Memory. Acm Journal on Emerging Technologies in Computing Systems, 12(1):1-18. 2015

[13]. *Huang Y Z, Hsueh C T, Chiang Y J.* A new cryptosystem based on three knapsacks with plaintext encoding. Frontiers in Artificial Intelligence & Applications, 274:770-781. 2015

[14]. *Fujioka A, Suzuki K, Xagawa K, et al.* Strongly secure authenticated key exchange from factoring, codes, and lattices. Designs Codes & Cryptography, 76(3):469-504. 2015

[15]. *Zheng M, Hu H, Wang Z.* Generalized cryptanalysis of RSA with small public exponent. Science China Information Sciences, 59(3):32108-032108. 2016

[16]. *Vaidyanathan S,Volos C.* Analysis and adaptive control of a novel 3-D conservative no-equilibrium chaotic system . Archives of Control Sciences, 25(3):333-353. 2015

[17]. *Hooshmand R, Aref M.* Public Key Cryptosystem Based on Low Density Lattice Codes. Wireless Personal Communications, 92(3):1-17, 2016

[18]. *Zhang S, Cao W, Li A, et al.* A new decryption algorithm of the quasi-cyclic low-density parity-check codes based McEliece cryptosystem. Proceeding of 8th IEEE International Conference on Communication Software and Networks (ICCSN), 8(1):53-57, 2016

[19]. *Sasi S, Jyothi L S.* Robustic public key cryptosystem for space data communication. Proceeding of International Conference on Communication & Electronics Systems.17 (3):35-39, 2017

[20]. *Thangavel M, Varalakshmi P*. A novel public key cryptosystem based on Merkle-Hellman Knapsack Cryptosystem. Proceeding of 8th International Conference on Advanced Computing. 17(1):117-122, 2017.

[21]. *Berce, Petru; Pacurar, Razvan; Balc, Nicolae*. Virtual Engineering for Rapid Product Development. INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH. Book Series: Mathematics and Computers in Science and Engineering Pages: 195, 2008.