Avrupa Bilim ve Teknoloji Dergisi
Sayı 14, S. 334-336, Aralık 2018

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
No. 14, pp. 334-336, December 2018

Research Article

The Knapsack Cryptography with Bell Number using Python Programming

Setenay Doğan[1*], Nilgun DEMİR[2], Furkan TOKAÇ[3]

[1]Bursa Uludag University, Science and Arts Faculty, Mathematics Department,16059, Bursa-Turkey
[2] Bursa Uludag University, Science and Arts Faculty, Physics Department,16059, Bursa-Turkey
[3] Bursa Uludag University, Science and Arts Faculty, Physics Department, 16059, Bursa-Turkey

(First received 26 November 2018 and in final form 10 December 2018)

(DOI: 10.31590/ejosat.487931)

Abstract

The cryptology is consisted of Kryptos (hidden) and logos (word) terms in the Greek. It also means that "secrecy science" at the communication. In the present days, the expansion of the electronic comminucation network has more increased the importance of cryptology. In this work, we have focused on Knapsack cryptosystem. In this purpose, the Bell numbers in the form of 'super- increasing sequence', which constitute the hypotenuse of the Bell triangle, are generated in the Python programming. The Knapsack encryption and decryption of these numbers are modeled using the Python program. As an example, "ULUDAG UNIVERSITY" was considered, a 12-bit encryption was performed. It was observed that the Bell numbers are suitable for Knapsack encryption.

Key words: Bell number, Knapsack encryption, Python

1. Introduction

Cryptology is a mathematical science based on number theory. It consist of Krypto's and Logo's words in the Greek. Cryptology examine in two main branches as cryptography and cryptanalysis. While cryptography is used to provide security and privacy in cyber systems and electronic communications, cryptanalysis reveals the weak and strong aspects of cryptographic systems. Cryptology is important in many areas such as health services, banking, communications, govermental secrecy, military security and security of the large companies.

The Knapsack cryptosystem which was one of the public key cryprosystem, was formulated by Ralph Merkle and Martin Helmann in 1978 [1].The most important feature of this cryptosystem is to encrypt with a public key and decrypt with a private key. Nowadays, the public key cryptosystem techniques are used especially as a lot of areas the marketing, banking security, dijital signiture and authentication control at the electronic media. The studies on the development of Knapsack public key cryptosystem are available in the literature [2-5].

In this work, it was shown that the super-increasing sequence which is consisting of numbers on the hypotenuse of the bell triangle, is appropriate to the Knapsack encryption. At this purpose, Bell number sequence was generated using the Python program. The text of 'ULUDAG UNIVERSITY' was encrypted using these numbers. In order to test the encryption process, decryption was performed with the Python program.

2. Materials and Methods

2.1 Bell Number

The Bell numbers, which were come into use in the 19th century and based on mediaeval Japan, have been used scientifically by Eric Temple Bell in 1930. The number of partitions of a set with n elements is called n. Bell number and

$$B_n = \sum_{k=0}^{n} S(n,k) \quad (1)$$

where S (n,k) is Stirling number of second kind. It was given,

$$S(n,k) = \left\{ {n \atop k} \right\}. \quad (2)$$

Thus,

$$B_n = \sum_{k=0}^{n} \binom{n}{k}. \quad (3)$$

B_0 is defined as one. There is a possible partition of a set with one element, so $B_1 = 1$. There is not a simple formula that gives a B_n, but we can calculate from the recurrence relation $\binom{n}{k}$ is binomial coefficient),

$$B_{n+1} = \binom{n}{k} B_k \quad (4)$$

First a few Bell number are,

[1] Corresponding Author: Bursa Uludag University, Science and Arts Faculty, Mathematics Department,16059, Bursa-Turkey, setenay@uludag.edu.tr

1,1, 2, 5, 15, 52, 203, 877, 4140, 21147, … or we can find it simply using a Bell Triangle. Its construction is similar to the Pascal's triangle [6,7],

1

1 2

2 3 5

5 7 10 15

15 20 27 37 52

52 67 87 114 151 203

2.2 Knapsack Cryptology Method

Merkle-Hellman is a public key encryption method. There is a public key and private key at this method. This encryption operation runs to one-way different from RSA encryption method. While the public key is used only encryption, the private key is used only decryption. The procedure of Knapsack encryption method is;

1- We need to super-increasing sequence. Let

$s_i = (s_1, s_2, s_3, \ldots, s_n)$ (5)

is non-empty sequence of positive integers. If each s_i is greater than the sum of all previous elements, S is called to a super-increasing sequence. In this case, the S must satisfy,

$s_i > \sum_{j=1}^{i-1} s_j$ (6)

for each $s_i \in S$.

2- We choose modulus; it is satisfy,

$m > \sum_{i=1}^{n} s_i$ (7)

3- A random t number is choosing. It is invertible modulo m, t and m must be relatively prime (gcd (t,m)=1).

4- A new set of numbers to use as a public key, is calculated from;

$n_i = s_i \times t \pmod m$ (8)

5- Every elements $n_1, n_2, n_3, \ldots, n_n$ of the n_i sequence are multiplied with the corresponding elements of the binary set b. The generated numbers are added and then the encrypted message is created.

For n- bit message, $m_i = \{m_1, m_2, m_3, \ldots, m_n\}$

Public key, $n_i = \{n_1, n_2, n_3, \ldots, n_n\}$

Encrypted message, $C = \sum m_i . n_i$ (9)

for $1 \leq i \leq n$, with $0 \leq C < q$

Decryption is also;

6- Private key is (s_i, m, t).

7- We compute $C' = Ct^{-1} \pmod m$ (10)

8- The binary numbers of corresponding to the sequence of C′ are obtained and the message is decrypted.

3. The Results of Encryption with Python Program

The Python program is an object-oriented program. It was created by Guido van Rossum in the late 1980s [8]. Python programming language struggles to provide a simple but powerful syntax contrary to the other popular languages such as C, C++, Java, and C#.

In the present work, the Bell numbers were used which form the hypotenuse of the Bell triangle. These numbers were generated from the Stirling numbers with the Python 3.5 version. As an example, 'ULUDAG UNIVERSITY' was chosen as the text to be encrypted. The computes were performed for n=12 bits. The generated Bell number sequence is,

s_i = (1,2,5,15,52,203,877,4140,21147,115975,678570,4213597).

The sequence is satisfying the condition (6). Thus it is a super-increasing sequence.

The m modulo was chosen which satisfy to (7) condition,

m=5034591.

t= 91 was taken and gcd (91, 503459)=1.

The n_i number sequence was computed using the equation (8),

n_i = (91,182,455,1365,4732,18473,79807,376740,1924377,
484543,1334778,808411).

The ASCII codes corresponding to each element of the text to be encrypted are matched with the sequence of n_i as can be seen below;

U=000001010101 →

$C_1 = \sum U_i \; n_i$=18473+376740+484543+808411

L = 000001001100 → $C_2 = \sum L_i \; n_i$= 18473+1924377+484543

D = 000001000100 → $C_3 = \sum D_i \; n_i$= 18473+484543 etc.

C_1 =1688167

C_2 = 2427393

.
.
.

C_{17} = 3128001

numbers were obtained for 17-character 'ULUDAG UNIVERSITY' including space. Then the encrypted text is

C= (1688167, 2427393, 1688167, 503016, 826884, 2646205, 79807, 1688167, 3762171, 2751261, 2214534, 1311427, 1729991, 2538402, 2751261, 879756, 3128001).

In order to test the accuracy of the encryption process, decryption was done with the Python program. The inverse of t number was computed that,

t^{-1} = 940528.

Then C′ values were obtained from equation (10);

C′= (4333915, 137325, 4333915, 116178, 4213800, 5008345, 877, 4333915, 815895, 4234947, 798888, 4329775, 682913, 4896510, 4234947, 120318, 4239087).

$C_1' = 1688167.940528 \pmod{503459} = 4333915$

s_i = (1,2,5,15,52,203,877,4140,21147,115975,678570,4213597)

The largest element of $s_i \leq C_i'$ is 4213597 → U_{12} = 1

$C_U' = 4333915 - 4213597 = 120318 \rightarrow U_{10} = 1$

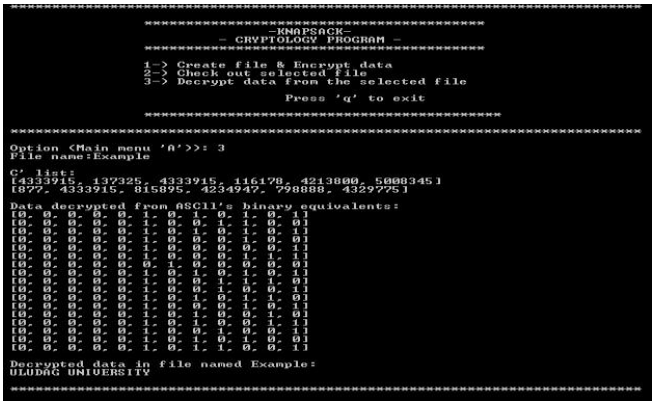
$C_U' = 120318 - 115975 = 4343 \rightarrow U_8 = 1$

$C_U' = 4340 - 4140 = 203 \rightarrow U_6 = 1.$

Then the ASCII code of U character for 12 bits was computed as

U = 000001010101.

An output of the running program for decryption of the text was given below;



4. Results and Discussion

In the present work, we have investigated that suitability at the Knapsack encryption method of the Bell numbers located to the hypotenuse of a Bell triangle. In this purpose all the number of sequence at the hypotenuses 1., 2., 3.,… were tested by the Python program and was observed that all the Bell number sequence were provided to Knapsack cryptology method.

References

Merkle R.C., Hellman M.E., 1978. Hiding Information ans Signatures in Trapdoor Knapsacks. IEEE Transaction on Information Theory 24, 525-530.

Thangavel M., Varalakshmi P., 2016. A Noval Public Key Cryptosystem Based on Merkle-Hellman Knapsack Cryptosystem. IEEE Eighth International Conference on Advanced Computing, IEEE Xplore, 117-122.

Zhang W., Wang B., Hu Y., 2009. A New Knapsack Public-Key Cryptosystem. Fifth International Conference on Information Assurance and Security, IEEE Xplore, 53-56.

Stallings W., 2010. Knapsack Public-Key Algorithm. ISBN-10: 0136097049, Supplement to Cryptography and Network Security, Fifth Edition.

Resource. http://mathworld.wolfram.com/BellTriangle.html

Venners B., 2003. The Making of Python. Artima Developer. Retrived 22 March 2007.

Jain A., Chaudhari N.S., 2015. Analysis of the improved knapsack cipher.Eighth International Conference on Contemporary Computing, IEEE Xplore.

Guichard D. February 4, 2018. Combinatorics and Graph Theory https://www.whitman.edu/mathematics/cgt_online/book/section01.04.html

Weisstein, Eric W. Bell Triangle. MathWorld–A Wolfram Web