CRYPTOGRAPHIC TECHNIQUE FOR COMMUNICATION SYSTEM

*Mohd Nabeel, **Lalu Kumar

* Malout Institute of Management and Information Technology

Computer Science and Engg. Department

Malout,Punjab

Abstract

In Today's world Sensitive data is increasingly used in communication over the internet. Thus Security of data is the biggest concern of internet users. Best solution is use of some cryptography algorithm which encrypts data in some cipher and transfers it over the internet and again decrypted to original data. The field of cryptography deals with the procedure for conveying information securely. Cryptography includes a set of techniques for scrambling or disguising data so that it is available only to someone who can restore the data to its original form. Cryptography or communication by using secret code was used by the Egyptians some 4000 years ago. However, the science of cryptography was initiated by Arabs since 600s. Cryptography becomes vital in the twentieth century where it played a crucial role in the World War I and 2. This paper focuses on the analysis of the two types of key cryptography exists, based on the availability of the key publicly: Private key Cryptography, and Public Key Cryptography. Both the sender and the recipient share a key that must be kept private.

Keywords: Encryption Techniques, Cryptography, Algorithm, information security, data protection

1 .Introduction

Cryptography is technique of securing information and communications through use of codes so that only those people for whom the information is intended can understand it and process it. Thus preventing un authorized access to information. The prefix "crypt" means "hidden" and suffix graphy means "writing".

Techniques used For Cryptography: In today's age of computers, cryptography is often associated with the process where an ordinary plain text is converted to cipher text which is the text made such that intended receiver of the text can only decode it and hence this process is known as encryption.

The process of conversion of cipher text to plain text this is known as decryption.Cryptography can be classifies into Symmetric and asymmetric encryption algorithms as shown in Figure

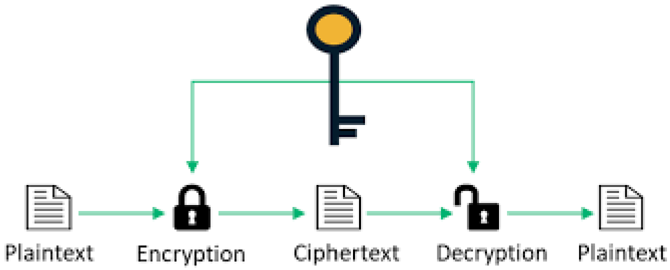


Figure . Symmetric Encryption

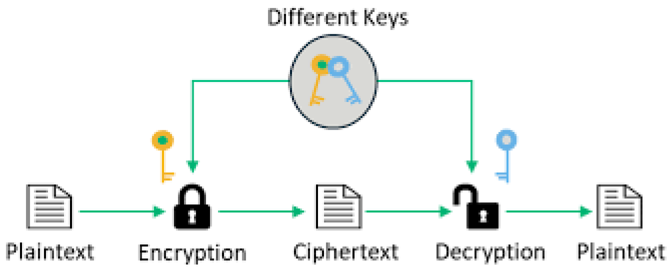


Figure . Asymmetric Encryption

2. Cryptography Objectives

Cryptography is the science of writing in secret code and is an ancient art; the first documented use of cryptography in writing dates back to circa 1900 B.C. when an Egyptian scribe used non-standard hieroglyphs in an inscription. cryptography is necessary when communicating over any untrusted medium, which includes just about any network, particularly the Internet. Within the context of any application-to-application communication, there are some specific security requirements, including:

- Confidentiality: Information can only be accessed by the person for whom it is intended and no other person except him can access it.

- Integrity: Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.

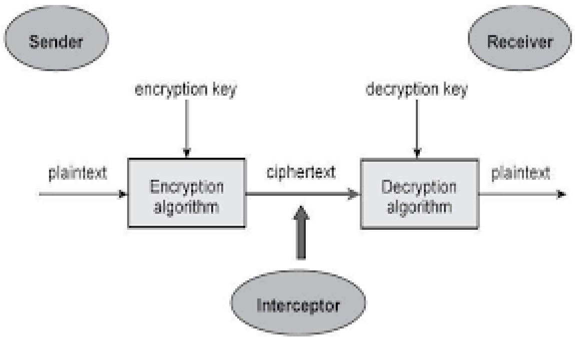
- Non-repudiation: The creator/sender of information cannot deny his or her intention to send information at later stage.

- Authentication: The identities of sender and receiver are confirmed. As well as destination/origin of information is confirmed

Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication. There are, in general, three types of cryptographic schemes typically used to accomplish these goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions, each of which is described below.

Secret Key Cryptography (symmetric cryptography): uses a single key to encrypt data. Both encryption and decryption in symmetric cryptography use the same key, making this the easiest form of cryptography. Secret Key Cryptography can be used on both in-transit and at-rest data, but is commonly only used on at-rest data, as sending the secret to the recipient of the message can lead to compromise. Example: AES, DES, Caeser Chiper.

Public Key Cryptography (asymmetric cryptography): uses two keys to encrypt data. One is used for encryption, while the other key can decrypts the message. Unlike symmetric cryptography, if one key is used to encrypt, that same key cannot decrypt the message, rather the other key shall be used. One key is kept private, and is called the “private key”, while the other is shared publicly and can be used by anyone, hence it is known as the “public key”. The private key should not be distributed and should remain with the owner only. The public key can be given to any other entity. Example: ECC, DSS, Diffie-Hellmen. *Hash functions*: Hash functions are irreversible, one-way functions which protect the data, at the cost of not being able to recover the original message. Hashing is a way to transform a given string into a fixed length string. A good hashing algorithm will produce unique outputs for each input given. The only way to crack a hash is by trying every input possible, until you get the exact same hash. A hash can be used for hashing data (such as passwords) and in certificates.



Components of a Cryptosystem: A basic cryptosystem includes the following components:

Plaintext- This is the data that needs to be protected.

Encryption algorithm- This is the mathematical algorithm that takes plaintext as the input and returns ciphertext. It also produces the unique encryption key for that text.

Ciphertext- This is the encrypted, or unreadable, version of the plaintext.

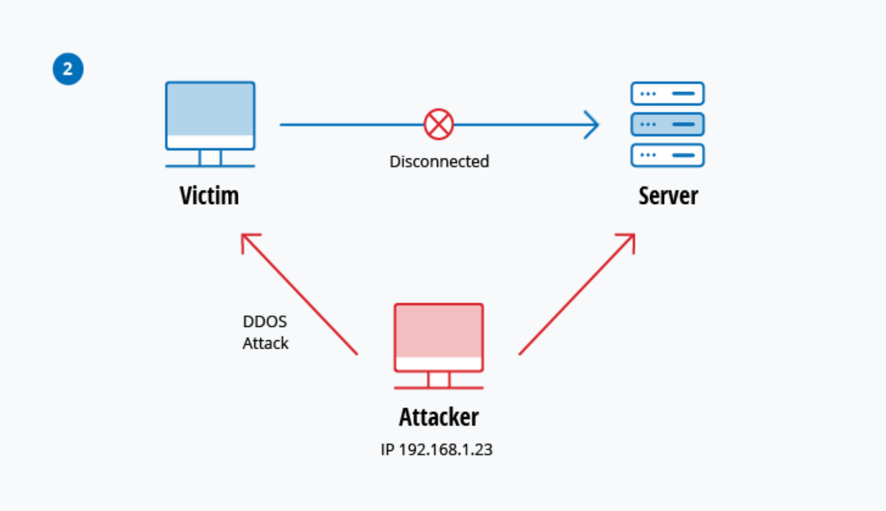
Decryption algorithm- This is the mathematical algorithm that takes ciphertext as the input and decodes it into plaintext. It also uses the unique decryption key for that text.

Encryption key- This is the value known to the sender that is used to compute the ciphertext for the given plaintext.

Decryption key- This is the value known to the receiver that is used to decode the given ciphertext into plaintext.

Cryptography Attacks: A cryptographic attack is a method for circumventing the security of a cryptographic system by finding a weakness in a code, cipher, cryptographic protocol or key management scheme.
Based on the methodology used, attacks on cryptosystems are categorized as follows –

Dictionary Attack: This attack has many variants, all of which involve compiling a 'dictionary'. In simplest method of this attack, attacker builds a dictionary of ciphertexts and corresponding plaintexts that he has learnt over a period of time.
Brute Force Attack (BFA): In this method, the attacker tries to determine the key by attempting all possible keys. If the key is 8 bits long, then the number of possible keys is 28 = 256.
BirthdayAttack: This attack is a variant of brute-force technique. It is used against the cryptographic hash function. When students in a class are asked about their birthdays, the answer is one of the possible 365 dates. Let us assume the first student's birthdate is 3rd Aug. Then to find the next student whose birthdate is 3rd Aug, we need to enquire 1.25*• √365 ≈ 25 students.
Man in Middle Attack (MIM): The targets of this attack are mostly public key cryptosystems where key exchange is involved before communication takes place.
Buffer flow attack: A buffer is a temporary space for data storage. Buffer overflow occurs if the data is stored by a program or process in a buffer is greater than the maximum capacity of the buffer. The extra data can overflow into adjacent buffer corrupting or overwriting the valid data held in them.
Ping of death attack: Ping of death attack takes advantage of a weakness in TCP-IP protocol. The weakness is that many computer system can not handle an IP packet larger than the maximum IP packet size of 65535 bytes. Buffer overflow is occure in ping of death attack.
DoS Attack (Denial of Service Attack): A Denial-of-Service attack or DoS is an attack targeting the availability of web applications. Unlike other kinds of attacks, DoS attacks' primary goal is not to steal information but to slow or take down a web site.
Teardrop attack: A teardrop attack is a denial-of-service (DoS) attack that involves sending fragmented packets to a target machine. Since the machine receiving such packets cannot reassemble them due to a bug in TCP/IP fragmentation reassembly, the packets overlap one another, crashing the target network device.
Ciphertext Only Attacks (COA): In this method, the attacker has access to a set of ciphertext(s). He does not have access to corresponding plaintext.
Side Channel Attack (SCA): this type of attack is not against any particular type of cryptosystem or algorithm.
Fault analysis Attacks: In these attacks, errors are induced in the cryptosystem and the attacker studies the resulting output for useful information.



Cipher: Cipher is an algorithm which is applied to plain text to get ciphertext. It is the unreadable output of an encryption algorithm. The term "cipher" is sometimes used as an alternative term for ciphertext. Ciphertext is not understandable until it has been converted into plain text using a key.

There are mainly Two Traditional Ciphers such as:-

Substitution Cipher Technique: In Substitution Cipher Technique plain text characters are replaced with other characters, numbers and symbols as well as in substitution Cipher Technique, character's identity is changed while its position remains unchanged. Example – Caesar Cipher, Polybius Cipher, Vigenere Cipher.

Transposition Cipher Technique: Transposition Cipher Technique rearranges the position of the plain text's characters. In transposition Cipher Technique, The position of the character is changed but character's identity is not changed. Example – Rail fence Cipher.

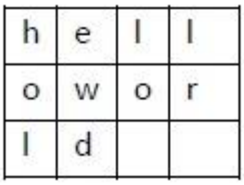
Literature review: This paper the security for web keeping money, account passwords, messages accounts secret word, etc requires content protection in mechanized media. It shows the security besides; pressure for the information with the move encryption standard. The age of key has been done with the assistance of the Polybius square. The extension in number of rounds it will require increasingly computational speculation and will end up irksome for the software engineer to break the system.

Caesar Cipher technique is one of the earliest and simplest method of encryption technique. It's simply a type of substitution cipher, i.e., each letter of a given text is replaced by a letter some fixed number of positions down the alphabet. For example with a shift of 1, A would be replaced by B, B would become C, and so on. The method is apparently named after Julius Caesar, who apparently used it to communicate with his officials.

Transposition Cipher is a cryptographic algorithm where the order of alphabets in the plaintext is rearranged to form a cipher text. In this process, the actual plain text alphabets are not included.

Example: A simple example for a transposition cipher is columnar transposition cipher where each character in the plain text is written horizontally with specified alphabet width. The cipher is written vertically, which creates an entirely different cipher text.

Consider the plain text hello world, and let us apply the simple columnar transposition technique as shown below



h	e	l	l
o	w	o	r
l	d		

The plain text characters are placed horizontally and the cipher text is created with vertical format as : holewdlo lr. Now, the receiver has to use the same table to decrypt the cipher text to plain text.

The vigenere cipher is an algorithm that is used to encrypting and decrypting the text. The vigenere cipher is an algorithm of encrypting an alphabetic text that uses a series of interwoven caesar ciphers. It is based on a keyword's letters. It is an example of a polyalphabetic substitution cipher.

Example: Plain Text: GIVE MONEY
Key: LOCK

G	I	V	E	M	O	N	E	Y
L	O	C	K	L	O	C	K	L

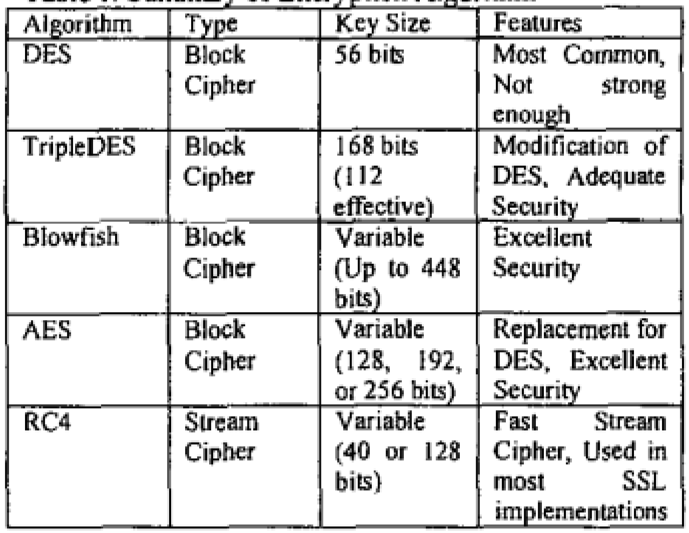
Cipher →R W X O X C P O J

For Decryption,

Cipher→	R	W	X	O	X	C	P	O	J
Key→	L	O	C	K	L	O	C	K	L
Plain→	G	I	V	E	M	O	N	E	Y

Cryptographic Algorithms: Cryptography has several differences from pure mathematics. While a mathematician may use A and B to explain an algorithm, a cryptographer may use the fictious names Alpha and Beta Suppose Alpha wants to send a message to his bank to transfer money. He would like the message to be private, since it includes information such as his account number and transfer amount. One solution is to use a cryptographic algorithm, a technique that would transform his message into an encrypted form, unreadable except by those for whom it is intended. When encrypted, the message can only be interpreted through the use of the corresponding secret key. Without the key the message is useless: good Cryptographic algorithms make it so difficult for intruders to decode the original text that it isn't worth their effort . Some of Encryption Algorithm is shown in Table.1.

Table 1. Summary of Encryption Algorithm



Algorithm	Type	Key Size	Features
DES	Block Cipher	56 bits	Most Common, Not strong enough
TripleDES	Block Cipher	168 bits (112 effective)	Modification of DES, Adequate Security
Blowfish	Block Cipher	Variable (Up to 448 bits)	Excellent Security
AES	Block Cipher	Variable (128, 192, or 256 bits)	Replacement for DES, Excellent Security
RC4	Stream Cipher	Variable (40 or 128 bits)	Fast Stream Cipher, Used in most SSL implementations

3. Polybius Square Cipher

In this problem, we are given a string and we have to find its integer encryption using the *Polybius Square Cipher*.It is a table that is used for the conversion of letters into numbers. The table for English encryption is a 5X5 table i.e. contains 25 cells for 26 alphabets of an English dictionary. The letters i and j are kept together in a single cell.

The following table shows a *Polybius square Cipher* –

	1	**2**	**3**	**4**	**5**
1	A	B	C	D	E
2	F	G	H	I,J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

The letter of the tables can be randomized. Also, the size of the table can be changed based on the number of alphabets of the language. Let's take an example to understand the problem,

Input – Hello

Output – 2315313134

Methadology: The message as plaintext and Key is send through sender in two phase for execution and working of System as in first phase it will proceed through Vigenere Cipher and then the new instructed and disputed encrypted cipher comes and then in second phase it became the input of Polybius cipher which result as output as Numerical encrypted Cipher that is confusing and scrambled mix numerical. This Output from Polybius at last phase is numerical and the Input that proceed in first phase was alphabetic letters this all confuses and doesn't allow the intruders, detectors, thefts, hackers and cyber crime to commit any assaults and attacks on system and doesn't allow them to steal Information. A python programming is written and executed for the working of System. Google Colab as Online and Sypder IDE on Independent System are taken for Execution of process. Flowchart of Hybrid Algorithm-

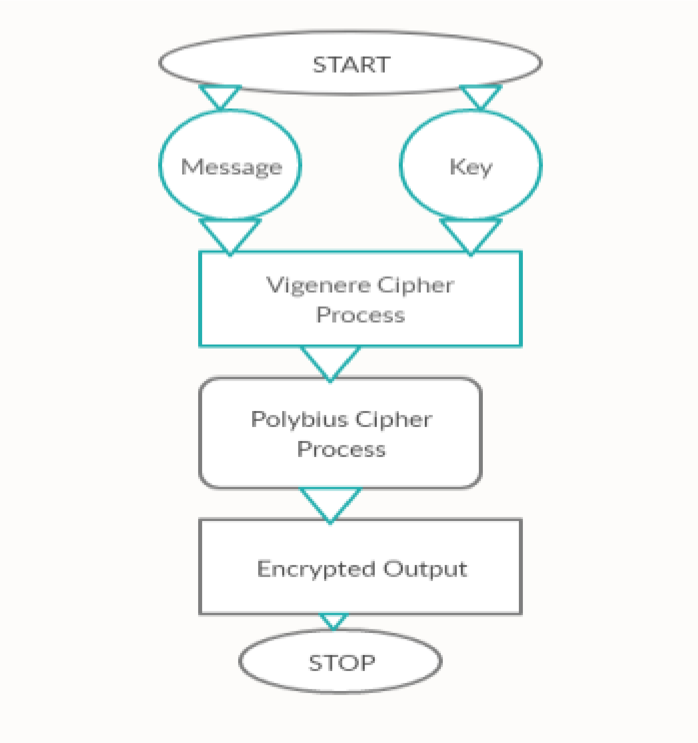


Fig: Hybrid System

Implementation: Python programming is written for Implementation of Hybrid System.

Step 1 : Vigenere Cipher:

```
alphabet = 'abcdefghijklmnopqrstuvwxyz'

letter_to_index = dict(zip(alphabet, range(len(alphabet))))
index_to_letter = dict(zip(range(len(alphabet)), alphabet))

def encrypt(message, key):
    encrypted = ''
    # split the message to the length of the key
    split_message = [message[i:i + len(key)] for i in range(0, len(message), len(key))] # (start, end, stop)

    # want to convert the message to index and add the key (mod 26)
    for each_split in split_message:
        i = 0
        for letter in each_split:
            number = (letter_to_index[letter] + letter_to_index[key[i]]) % len(alphabet)
            encrypted += index_to_letter[number]
            i += 1
    return encrypted

def decrypt(cipher, key):
    decrypted = ''

    # split the cipher to the length of the key
    split_cipher = [cipher[i:i + len(key)] for i in range(0, len(cipher), len(key))]  # (start, end, stop)

    # convert cipher to index and subtract key (mod 26)
    for each_split in split_cipher:
        i = 0
        for letter in each_split:
            number = (letter_to_index[letter] - letter_to_index[key[i]]) % len(alphabet)
            decrypted += index_to_letter[number]
            i += 1
    return decrypted

def main():
    key = 'nabeel'
    message = 'laluisbestprogrammer'
    encrypt_message = encrypt(message, key)
    decrypted_message = decrypt(encrypt_message, key)

    print('Original message: ' + message)
    print('Encrypted message: ' + encrypt_message)
    print('Decrypted message: ' + decrypted_message)

main()
```

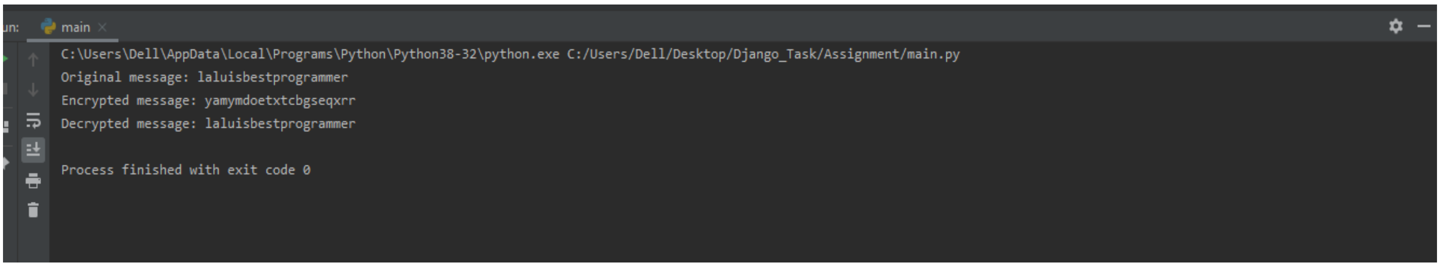
Step 2 : Polybius Cipher:

```
import string

class Polybius:

    def encipher(plain, keyword):

        # create secret alphabet => 5x5 square
        square = []
        for c in keyword + string.ascii_lowercase:
            if c not in square and c != "j":
                square.append(c)
        square = ''.join(square)

        # loop through plain text to encipher
        cipher = []
        for i in plain:
            n = square.find(i) + 1
            row,col = divmod(n,5)
            cipher.append(str(row+1)+str(col))

        # return
        return cipher

    def decipher(cipher, keyword):

        # create secret alphabet => 5x5 square
        square = []
        for c in keyword + string.ascii_lowercase:
            if c not in square and c != "j":
                square.append(c)
        square = ''.join(square)

        # loop through cipher text to decipher
        plain = []
        for i in range(len(cipher)):
            row = int(cipher[i][0])
            col = int(cipher[i][1])
            letter = square[(row-1)*5 + col-1]
            plain.append(letter)

        # return
        return "".join(plain)

encp = Polybius.encipher("laluisbest", "nabeel")
decp = Polybius.decipher(encp,"nabeel")
print(encp)
print(decp)
```

The output will be Encrypted text as Cipher text will be generated from the system. This two combination of cipher program will be executed back to back to get cipher text. It can be implemented on any System, IDE, Interpreter, and Compiler or on Cloud System such as Jupyter, Anaconda, Google collaboratory, etc.

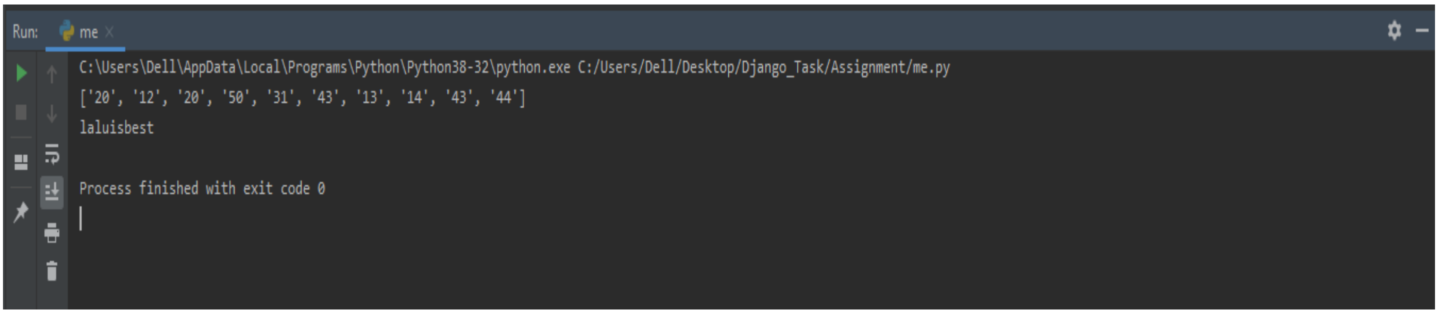
Output:

Vigenere cipher encryption:



```
C:\Users\Dell\AppData\Local\Programs\Python\Python38-32\python.exe C:/Users/Dell/Desktop/Django_Task/Assignment/main.py
Original message: laluisbestprogrammer
Encrypted message: yamymdoetxtcbgseqxrr
Decrypted message: laluisbestprogrammer

Process finished with exit code 0
```

Polybius cipher encryption:



```
C:\Users\Dell\AppData\Local\Programs\Python\Python38-32\python.exe C:/Users/Dell/Desktop/Django_Task/Assignment/me.py
['20', '12', '20', '50', '31', '43', '13', '14', '43', '44']
laluisbest

Process finished with exit code 0
```

4. Discussions and Conclusions

Cryptography is a particularly interesting field because of the amount of work that is, by necessity, done in secret. The irony is that today, secrecy is not the key to the goodness of a cryptographic algorithm. Regardless of the mathematical theory behind an algorithm, the best algorithms are those that are well-known and welldocumented because they are also well-tested and well-studied! In fact, time is the only Vue test of good cryptography; any cryptographic scheme that stays in use year after year is most likely a good one. The strength of cryptography lies in the choice (and management) of the keys; longer keys will resist attack bener than shorter keys. The basic concepts, characteristics, and goals of various cryptographic have been discussed. The essential parts of Cryptography in communications systems are shown. How this makes them especially attractive as a potential platform to implement cryptographic algorithms.

5. References

[l] 1. Vajda, Extraction of random bits for cryptographic purposes, Tatra Mountains Mathematical Publications, 2002, vol. 25, pp. 83-49
[2] Ferguson, N. and B. Schneier, Proclicol Cryptography. New York John Wiley & Sons, 2003
[3] Barr, T.H. hitation to Cryptologv. Upper Saddle River (NJ): Prentice Hall, 2002.
[4] Bauer, F.L. Decrypted Secrets: Methods and Maxims of Cryptology. 2nd ed. New York Springer Verlag, 2002.

[5] D.E.R. Denning, Cryptography and Data Securi@, Addison-Wesley, 1982.
[6] E. Kranakis, Primality and Cryptography, Wiley, 1986.
[7] A.G. Konheim, Cryptogrophy: A Primer, John Wiley, 1981.
[8] J. Seberry and J. Pieprzyk, Cryptography: An Introduction to Computer Security, Prentice-Hall, 1989.
[9] D. Welsh, Codes and Cryptography, Oxford Science Publications, 1988.
[I0] D. R. Stinson, Cryptography: Theory and Practice. CRC Press, 1995.
[11] B. Schneier, Applied Cryptograp&, Wiley, 1994.
[12] M. Y. Rhee, cryptography and .%cure Communications, McGraw-Hill, 1994.