# A New Cryptography Algorithm Based on ASCII Code

**Yaser M.A. Abualkas**

Department of Computer Science and Systems Engineering, Andhra University College of Engineering (A)
Andhra University, Visakhapatnam-530003, India
*319206415031[at]andhrauniversity.edu.in*

**Abstract:** ***Encryption and Decryption is the encoding of information so that only those who have access to a password or encryption key can access it. Encryption protects data content, rather than preventing unauthorized interception of or access to data transmissions. It is used by intelligence and security organizations and in personal security software designed to protect user data. This paper introduces a new algorithm for cryptography to achieve a higher level of security. In this algorithm it becomes possible to hide the meaning of a message in unprintable characters. The main issue of this paper is to make the encrypted message undoubtedly unprintable using several random number key of ASCII conversions.***

**Keywords:** Cryptography, Encryption and Decryption, Higher Level of Security, Unprintable Encrypted Message, ASCII Conversion

## 1. Introduction

The cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication and data origin authentication [1]. Cryptography is a big deal in the present era of information and communication technology. Though it has been used for thousands of years to hide secret messages, systematic study of cryptology as a science (and perhaps an art) just started around one hundred years ago [2]. Also it means hidden writing, and it refers to the practice of using encryption to conceal text [3]. The major security goals that are of concern to the Cryptography are confidentiality, authentication, integrity, non-repudiation and access control [4]. Among the available three modern security offering techniques namely cryptography, steganography and watermarking, cryptography is the base to understand and also easy to implement ensuring a higher level of security in the real-time security systems. In this paper, we proposed a new cryptographic algorithm which follows a different methodology from the traditional symmetric-key cryptography, asymmetric-key cryptography or hashing function. In this paper, we proposed an efficient algorithm which is different from the traditional symmetric-key cryptography, asymmetric-key cryptography or hashing function. ASCII and number system conversions have also been used in this paper which makes the cipher different from other algorithms.

## 2. Literature Survey

The security of information has become more important since the introduction of the Internet. Sensitive information like credit cards, banking transactions and social security numbers need to be protected. Encryption is the process of manipulating and collecting that information which can be identify and processed. Encryption is a word that is associated with secrecy. The straightforward definition of encryption describes it as a procedure that scrambles regular text or data into an illegible form. For encryption and decryption we use the key concept that can be provided to each end, there is two types of key which can used according to the user authorization.

Symmetric encryption scheme ingredients:

- Plaintext: This is the original message or data that is fed into algorithm as input
- Encryption algorithm: The encryption algorithm performs various substitution and transformation on the Plaintext.
- Secret key: The secret key is also input to algorithm. The exact substitution and transformation performed by the algorithm depend on the key.
- Cipher text: This is the scrambled message produced as output.it depend on the plaintext and the secret key. For given message, two different keys will produce two different cipher text.
- Decryption algorithm: This is essentially the encryption run in reverse. It takes the cipher text and the same secret key and produce the original plaintext.

Types of Keys

• Private Key

Private Key cryptography contains the same key for sender and the receiver. The sender sends the key along with the data for receiver to decrypt the data using the same key.

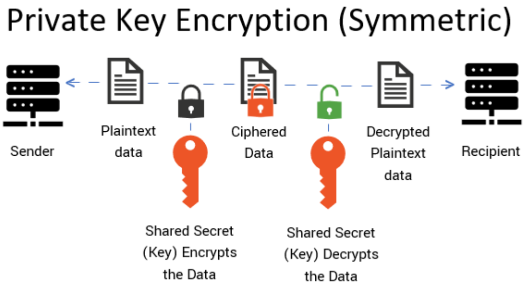

Private Key Encryption (Symmetric)
Sender
Plaintext data
Ciphered Data
Decrypted Plaintext data
Recipient
Shared Secret (Key) Encrypts the Data
Shared Secret (Key) Decrypts the Data

**Figure 1:** Symmetric encryption scheme

• Public Key

Public key cryptography will used both private and public keys. Public key will be send to all authorized user which can be enabled for all and one private key which can be known by only receiver. Public key is used for encryption and private key for decryption.

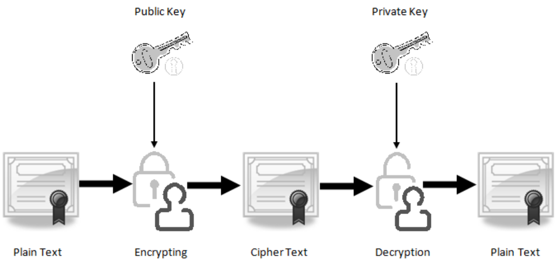


**Figure 2:** Asymmetric encryption scheme

Asymmetric encryption scheme ingredients:
Same of asymmetric encryption scheme ingredients furthermore it Public and private key this is a pair of keys that have been selected so that if one is used for encryption, the other is used for decryption. The exact transformations performed by the encryption algorithm depend on the public or private key that is provided as input.

## 3. Proposed Algorithm

### Introduction

The objective of this paper is to propose a new technique of encryption. It changes the data into its respective ASCII values and then converts these ASCII values to cipher text using the random number. The method also uses mathematical operation to produce final ASCII code, which makes data more secure from getting encrypted by intruders. At the receiver side encrypted data is received along with key and random number set. This key is then decrypted and compared with receiver id. Original data is encrypted only if those two id matches.

### Encryption Phase:

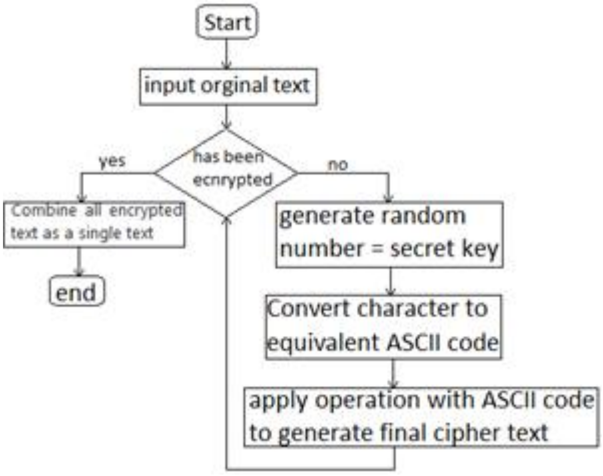

Start
input orginal text
has been ecnrypted
yes
no
Combine all encrypted text as a single text
end
generate random number = secret key
Convert character to equivalent ASCII code
apply operation with ASCII code to generate final cipher text

**Figure 3:** Flowchart of the Encryption Algorithm

In the encryption phase of the proposed algorithm, at first the input characters of the text to be converted to ASCII code it is a first phase of cipher text, then generate the random number that mean is a secret key, then apply the operation, to obtain the final cipher text.

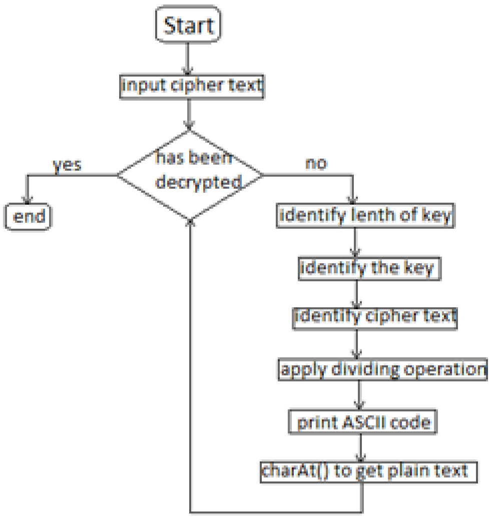
**Example:**
Plain text is: YaSeR
intascii = (int) character; method used to obtain the ASCII code number.
ASCII code of Y is: 89
ASCII code of a is: 97
ASCII code of S is: 83
ASCII code of e is: 101
ASCII code of R is: 82
Secret key is: 655
Length of secret key is: 3
Operation is: * (multiplication)
655*89=58295
655*97=63535
655*83=54365
655*101=66155
655*82=66155
Cipher text will be as the follow
I justify the cipher text blocks to equal indexes using if condition statement
Encrypt Text for Y is: 058295
Encrypt Text for ais: 063535
Encrypt Text for S is: 054365
Encrypt Text for e is: 066155
Encrypt Text for R is: 053710
Concatenate all cipher text
All final cypher text is: 6553655058295063535054365066155053710

**Code of Encryption:**

```
Scanner scan=new Scanner(System.in);
Random randnu = new Random();
System.out.println ("Encryption Algorithm ---->");
System.out.print ("Enter Plain Text : ");
String plaintext=scan.nextLine();
intlen=plaintext.length();
int rand =randnu.nextInt(1000);
String slenrn = Integer.toString(rand);
intlenrn=slenrn.length();
String key=slenrn;
System.out.println ("Key number is : "+key);
System.out.println ("Length of random number is : "+lenrn);
intenctext;
String et="";
for (inti = 0; i<len; i++){
char character = plaintext.charAt(i);
intascii = (int) character;
enctext=rand*ascii;
String tlenct = Integer.toString(enctext);
intlenct=tlenct.length();
if(lenct<6){
String s="0"+enctext;
et+="0"+enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text is : "+s);
}else if(lenct<5){
String s="00"+enctext;
et+="00"+enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text is : "+s);
}else if(lenct<4){
String s="000"+enctext;
et+="000"+enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text is : "+s);
}else if(lenct<3){
String s="0000"+enctext;
et+="0000"+enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text
is : "+s);
}else if(lenct<2){
String s="00000"+enctext;
et+="00000"+enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text
is : "+s);
}else{
et+=enctext;
System.out.println(character+" = "+ ascii+"     Encrypt Text
is : "+enctext);
}
}
String altext=key+"3"+key+et;
System.out.println ("All cypher text is :"+altext);
```

**Decryption Phase:**

In the decryption phase of the proposed algorithm, at first will compare the cipher text with plain text if it equal, print the plain text, if not apply decryption algorithm starting to identify the length of key, then identify the key, then identify the number of index for cipher text, then remove any extra zeros before the first number of cipher text, then start decryption using the opposite of operation that use in encryption by dividing the cipher text and key to obtain the ASCII code, then use the method to convert ASCII code to character to obtain plain text, so we can see the original text as we put it.



**Example:**

All cypher text is:
655365505829506353505436506615505371 0
Secret key is: 655
Length of secret key is: 3
Remove useless zeros from cipher text
Before is: 058295 after: 58295
Before is: 063535 after: 63535
Before is: 054365 after: 54365
Before is: 066155 after: 66155
Before is: 053710 after: 53710
Operation is: / (dividing)
58295/655=89
63535/655=97
54365/655=83
66155/655=101
53710 /655=82

charAt(i); method used to obtain the character from ASCII code valu.
charAt(89)=Y, the original text is : Y
charAt(97)=a, the original text is : a
charAt(83)=S, the original text is : S
charAt(101)=e, the original text is : e
charAt(82)=R, the original text is : R
Plain Text is :YaSeR

**Code of Decryption:**

```
System.out.println ("\n\n\nDecryption Algorithm ---->");
String number =altext ;
String input = key;
String x = "";
intloopp=0;
for (inti = 0; i<number.length(); i++) {
x += number.charAt(i);
loopp++;
if (x.equals(input)) {
System.out.println("key is :"+input);
System.out.println("len is :" +  loopp);
int k = 0;
String y = "";
for (k = i + 2; k < (i * 2) +3; k++) {
y += number.charAt(k);
}
System.out.println("key is :"+y);
int l = i + 5;
String sixDigits = "";
String everysix="";
String allpt="";
for (int index = l; index <number.length(); index += 6) {
for (int j = index; j < index + 6; j++) {
sixDigits += number.charAt(j);
}
everysix=sixDigits;
String c = "";
IntnumLengthsix=everysix.length();
for (int t = 0; t <numLengthsix; t++) {
if (everysix.charAt(0) == '0' && t == 0) {
t++;
}
c += everysix.charAt(t);
}
intept=Integer.parseInt(c);
intpt= ept/rand;                                    char
cc=(char)pt;                               allpt+=cc;

          System.out.println("before  is:  "+everysix+"   after
:"+c+"  Plain text is : "+pt+"  the orginal text is : "+cc);
sixDigits = "";
}

System.out.println ("Plain Text is : "+allpt);
}
}
}
}
```

## 4. Conclusion

This paper presented a new symmetric cryptography algorithm. The proposed algorithm can be used to encrypt and decrypt text messages based on the ASCII code of characters in most sensitive data and critical position like bank account details. The main idea of the proposed algorithm is to ensure higher security and to hide data in effective way. In future, we increase the security technique and extending the proposed idea to work for asymmetric cryptography as well.

## References

[1] E. Cole, R. Krutz and J. W. Conley, Network Security Bible, Wiley Publishing Inc, 2005.

[2] Sidhpurwalahuzaifa. A Brief History of Cryptography. [Online]. Available: https://securityblog.redhat.com/2013/08/14/a-briefhistory-of-cryptography/

[3] A. Menezes, V. Oosrschot and A. Vanstone, Handbook on Applied Cryptography, CRC Press Inc., NY, USA, 2000.

[4] D. Stinson, Cryptography Theory and Practice, CRC Press Inc., NY, USA, 1995.