Advances in Computer Science and Information Technology (ACSIT)
p-ISSN: 2393-9907; e-ISSN: 2393-9915; Volume 2, Issue 14; October-December, 2015 pp. 30-32
© Krishi Sanskriti Publications
http://www.krishisanskriti.org/Publication.html

An Ingenious Enciphering Approach based on Caesar Cipher

Kanchan Bisht[1] and Ashutosh Ghildiyal[2]

[1,2]Department of Information Technology College of Technology, GBPUAT, Pantnagar
E-mail: [1]kanchanbisht.rocks@gmail.com, [2]ghildiyal.ashutosh@yahoo.com

Abstract—*In the era of modern communication, e-communication and e-transactions have become a vital part of our day-to-day lives. Thus, protection of data via secure cryptography algorithms is taking strides too. In this paper, we propose a Caesar cipher based algorithm that will subdue the limitations of the traditional Caesar cipher. Our algorithm will provide high-end security by employing the technique of random key generation. Each letter of the message will be encrypted by a separate key that will be generated randomly. Also, if there are multiple recipients, then each would receive a different copy of the cipher text corresponding to the same message. This will encumber the attacker's intentions further. Thus, it will become next to impossible for the attacker to decipher the original message, ensuring the decoding of the message, only by the genuine recipient(s).*

1. INTRODUCTION

Cryptography, a principal method of protecting valuable electronic information, has become the soul of all trending e-communications and e-transactions today. Cryptography comes from the Greek word, 'kryptos' (hidden) and 'graphein' (to write). It is the science of encoding information in a way that it becomes incomprehensible to all except the intended recipients. With cryptography, we can protect our data from unauthorized access. Following are the basic terminologies associated with cryptography:

- Plain text: It is the non-encoded original message.
- Cipher text: It is the encrypted form of the original message.
- Key: The value pertaining to which the cipher text is obtained.
- Encryption: The process of obtaining the cipher text from plain text.
- Decryption: The process of obtaining the plain text from the cipher text.
- Cipher: The algorithm that takes the plain text and key as input and produces the cipher text; or, takes the cipher text and the key as input and produces the plain text.

There are two basic categories of ciphers-Classical and Modern.

- Classical: It includes Substitution and transposition ciphers. Substitution ciphers may be mono-alphabetic or polyalphabetic.
- Modern: It includes Asymmetric and Symmetric ciphers. Symmetric ciphers make use of the same key for both encryption and decryption whereas Asymmetric ciphers make use of different keys for the same.

The following are the primary objectives of cryptography:

- Confidentiality: The sent information is only understood by the intended recipients.
- Integrity: The information is not tampered with on the way from the sender to the receiver.
- Non-repudiation: The sender nor the receiver can deny having sent the information or having received the information.
- Authentication: The sender/receiver can confirm each other's identity as well as the origin/destination of the information.

2. CAESAR CIPHER

In this paper, we throw light only upon the Caesar Cipher. As stated in the abstract, Caesar cipher is the most common and most simple classical substitution cryptography technique. It is also one of the earliest known ciphers. This cipher is named after the famous Roman general and statesman, 'Julius Caesar' who used it in his private correspondence. In this cipher, each character is replaced by another character that comes generally 3 places after it, however, the key may differ.

CT=PT + Key

PT=CT - Key

Let us have a look at a few examples:-

Plain text: CAESAR

Key: 3

Cipher text: FDHVDU

Plain text: ADORE

Key: 5

Cipher text: FITWJ

This method uses modular 26 arithmetic therefore if the key is 3 then XYZ will be encoded to ABC.

Because of its extreme simplicity and the presence of only 25 possible keys, it is immensely vulnerable to attacks. The attacker can try all the 25 keys (brute-force attack) to obtain the original message. Thus, the traditional Caesar cipher fails to provide communication security.

3. OUR APPROACH (KASHU CIPHER)

This cipher is based on Caesar cipher, but unlike traditional Caesar cipher, it provides high-end security. Instead of using one key for the entire message, we make use of a key array. Each character of the plain text (original message) is extracted one by one starting from index 0 (i.e. the beginning). A random number is generated between 0-999 (the upper limit can be changed). This number is wrapped to an integer between 0-255. This is the key corresponding to the i^{th} character and is stored at i^{th} index in the key array. Cipher text is then obtained by adding the i^{th} key to the plain text character at i^{th} index. The array CT contains the encrypted text.

CT[i] = PT[i] + Key[i]

We make use of modular 256 arithmetic instead of modular 26 arithmetic. This enables us to use the entire ASCII character set. Thus, each character can now be encoded to any of the 256 characters. The key array is encrypted and is send along with the encrypted text to the intended recipient. The recipient then decodes the key. The key is encrypted using a symmetric algorithm that is known both to the sender as well as receiver. After obtaining the original key array the plain text can be obtained in a way similar to the Caesar cipher.

PT[i] = CT[i] - Key[i]

3.1. Encryption Algorithm Sample Code

```
String encrypt(String message)
{
char pt[]=message.toCharArray();
int random=0;
char ct[]=new char[pt.length];
for(int i=0; i<pt.length; i++)
{
random =(int)(Math.random()*1000);
key[i]=(short)(random%256);
ct[i]=(char)(pt[i]+key[i]);
}
return(new String(ct));
}

short[] encrypt(short key[])
{
for(int i=0;i<key.length;i++)
{
if(i%2==0) key[i]*=2;
else key[i]*=3;
}
return key;
}
```

3.2. Decryption Algorithm Sample Code

```
short[] decrypt(short key[])
{
for(int i=0;i<key.length;i++)
{
if(i%2==0) key[i]/=2;
else key[i]/=3;
}
return key;
}

String decrypt(String msg,short k[])
{
String PT="";
char ct[]=msg.toCharArray();
for(int i=0; i<ct.length; i++)
PT+=(char)(ct[i]-k[i]);
return PT;
}
```

3.3. Experimental Results

0 1 3 4 5 6 7 8 9 10 11 12 13 14 15

A B C D E F G H I J K L M N O

16 17 18 19 20 21 22 23 24 25 26

P Q R S T U V W X Y Z

```
Encryption:
Plain text: ABCDEFG
Key generated: {226, 21, 122, 131, 52, 172, 20}
Corresponding Cipher text: ?W½Çyò[
Encrypted key: {452, 63, 244, 393, 104, 516, 40}

Decryption:
Cipher text: ?W½Çyò[
Encrypted key: {452, 63, 244, 393, 104, 516, 40}
Decrypted key: {226, 21, 122, 131, 52, 172, 20}
Decrypted text: ABCDEFG

Encryption:
Plain text: ENIGMA
Key generated: {94, 47, 100, 2, 155, 10}
Corresponding Cipher text: £}-IèK
```

Encrypted Key: {188, 141, 200, 6, 310, 30}

Decryption:
Cipher text: £}-IèK
Encrypted key: {188, 141, 200, 6, 310, 30}
Decrypted key: {94, 47, 100, 2, 155, 10}
Decrypted text: ENIGMA

From the above examples, it is clear that it is impossible for the attacker to break the code. Thus, Kashu cipher provides remarkable security to data.

4. SUGGESTED APPLICATION AREAS

- Password
- ATM pin
- Credit/Debit card number
- OTP (One time password)

5. CONCLUSION

Although this cipher uses more memory than the traditional Caesar cipher as we need to send the key array to the receiver too, yet, the service it provides is exceptional. Moreover, memory is no longer an issue today. The key for each letter of the encrypted text is different and it is not possible to predict it beforehand. Each letter can be encrypted using any of the 256 keys. For the next letter there are again 256 possibilities. Thus, as can be inferred from the illustrated examples, the hacker will not be able to hack the key in order to decrypt the message. The randomness and the unpredictability of the key values make this method unbreakable.

REFERENCES

[1] http://en.wikipedia.org/wiki/Caesar_cipher

[2] http://practicalcryptography.com/ciphers/caesar-cipher

[3] Java-2 Complete Reference - Patrick Haughton.

[4] William Stallings(2005), Cryptography and Network Security Principles and Practices, Fourth Edition, Prentice Hall

[5] Henk C. A. van Tilborg, “Fundamentals of Cryptology”, pp. 9 - 21.

[6] Sharad Kumar Verma, Dr. D.B. Ojha, ‘An innovative Enciphering Scheme based on Caesar Cipher’, International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 5, July 2014, ISSN 2348-7968

[7] Mr. Vinod Saroha, Suman Mor, Anurag Dagar, ‘Enhancing Security of Caesar Cipher by Double Columnar Transposition Method’, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2 Issue 10, October 2012, pp. 86-88, ISSN 2277 128X

[8] Kashish Goyal, Supriya Kinger, ‘Modified Caesar Cipher for Better Security Enhancement’, International Journal of Computer Applications (0975- 8887), Vol. 73-No. 3, July 2013, pp. 26-31

[9] Sailakshimi.S, Shashikala.G, ‘Caesar Cipher with Complement Approach’, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 5 Issue 5, May 2015, pp. 398-400, ISSN 2277 128X

[10] Yashpalsingh Rajput, Dnyaneshwar Naik, Charudatt Mane, ‘An Improved Cryptographic Technique to Encrypt Text using Double Encryption’, International Journal of Computer Applications (0975- 8887), Vol. 86-No. 6, January 2014, pp. 24-28

[11] Ochoche Abraham, Ganiyu O. Shefiu, ‘An Improved Caesar Cipher(ICC) algorithm’, International Journal of Engineering Science and Advanced Technology, Vol. 2 issue 5, pp. 1199-1202, ISSN 2250 3676