Block TEA based on Chaotic Systems

Ch.Udaya Bhaskar,
Associate Professor,CSE,
Aditya Engineering College,
Surampalem

S.Rama Sree,
Associate Professor,CSE,
Aditya Engineering College,
Surampalem

ABSTRACT
This Paper proposes a new Symmetric Block Cipher based on chaotic systems. This cipher has improved security, realizes the idea behind one time pad and works with different block sizes of plaintext. It also has characteristics like fast operation, precise restoration and non-extra distortion.

Keywords
Encryption; decryption; block cipher; TEA; chaotic equation; security;

1. INTRODUCTION
Today, due to widespread applications of Internet a large amount of sensitive information is transmitted over networks. This made Information security more important than what it was. Encryption is an important method to keep information secure. Chaotic encryption [1] was developed basing upon the random and non-periodical characteristics of chaotic equations. It has disadvantages like large cipher storage, inefficiency and requiring floating point computations. The combination of chaotic equations and other encryption techniques can produce good ciphers [2]. This paper presents a new such cipher – Block TEA based on chaotic systems.

2. CHAOTIC SYSTEMS
Chaotic equations are used to produce highly random phenomena. They are good examples for determined non-linear systems. They are very sensitive to initial values. Two isomorphic chaotic systems with tiny distinctions in initial values will produce two entirely different sequences with in a short period of time. As the sequences produced by these equations are highly random and regenerative they can be used as encryption sequences in cryptography.

In this paper a chaotic equation or map is used. It is the Chebyshev map of order 4, whose formal equation is,

$$A_{n+1} = \cos(4 \times \cos^{-1} A_n) \qquad A_n \in [-1, 1] \qquad (1)$$

This equation is highly complex like noise and sensitive to initial value.

3. Block TEA – A VARIANT OF TEA
TEA stands for Tiny Encryption Algorithm [4].The basic Version encrypts 64-bits of plaintext using a 128-bit key through 32 iterations by using XOR, ADD and SHIFT operations to provide non-linearity.

In Block TEA [5] the number of word(32 bit)s to be encrypted can be changed conveniently. The following two sub sections contain routines for Block TEA Encryption and Decryption.

3.1 Encoding Routine
```
Encode (long *v, long n, long *k)
{
unsigned long z = v [n-1], sum = 0,e;
delta = 0x9e3779b9;
long m, p, q;

q = 6 + 52/n;

while (q - - > 0)
{
sum + = delta;
e = sum >> 2 & 3;
for ( p = 0 ; p <  n; p ++)
z = v[p] + = ( z<<4  ^  z>>5 )  + z ^ k[ p&3^e] + sum ;
}
}
```

3.2 Decoding Routine
```
Decode( long *v, long n, long *k)
{
unsigned long z = v[n-1], sum = 0, e;
delta = 0x9e3779b9;
long m, p ,q;

q = 6+52/n;
sum = q*delta;

while (sum!=0)
{
e = sum>>2&3;
for ( p=n-1; p>0 ; p - -)
{
z = v[p-1];
v[p]-= (z<<4 ^ z>>5) + z ^ k[ p&3^e] + sum ;
}
z=v[n-1];
v[0] - = (z<<4 ^ z>>5) + z ^ k[ p&3^e] +sum ;
sum - =delta;
}
}
```

Where n represents number of blocks or words to be encrypted and is greater than 1
v is the n word data vector
k is the 4 word key

Even though the above algorithm is faster than TEA for n>4, it is highly susceptible to dictionary crypto analysis like TEA. This is because the same key is used to encrypt the entire

plaintext. Due to the same reason it can't resist attacks such as insertion and deletion.

4. COMBINING BLOCK TEA WITH CHAOTIC SYSTEM

4.1 Working Principle

We Encrypt n blocks of a plaintext at a time at the sender using a random key generated by quantifying the Chebyshev sequence using an initial key. The key is altered for every n blocks of plaintext or after every 8 iterations realizing the idea behind one time pad there by increasing security and resisting dictionary crypto analysis.

At the receiver n blocks of cipher text is decrypted at a time Using the key generated by quantifying Chebyshev sequence with the same initial key used at the sender.

4.2 Encryption

The process of encryption is as shown in Figure 1,

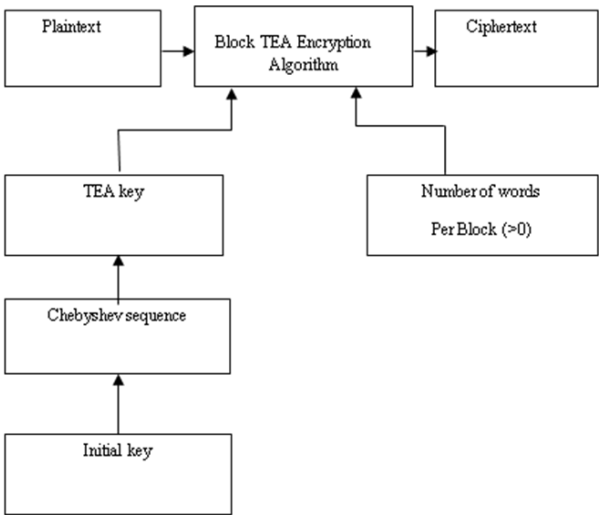

Figure 1: Encryption Mechanism

Steps involved in Encryption are,

Step 1: Use the initial key as A_n in equation (1) to obtain Chebyshev sequence.

Step 2: Repeat equation (1) for R times , as computed by equation (2) of Step 3, to get better chaotic sequence.

Step 3: R=((key mod 3) × (Length mod 3)+1)×1000 (2)

Where Length is the length of the plaintext.
key is the initial key that was used in step 1

Step 4: Quantify Chebyshev sequence into key. This is done by considering bits between 22 and 6^{th} positions (16 bits) in the mantissa of the binary representation of the Chebyshev map and considering it as, k, one eighth part of the total key.

Total key, KEY, is generated as,

KEY = k×8+0 k×8+1 k×8+2 …. K×8+7 (3)

Step 5: Now read the plaintext along with block size and give them as inputs to Block TEA along with the Dynamic key to get encryption done.

Step 6: To encrypt next n blocks of plaintext or after 8 iterations in the encryption process , change the KEY by repeating Steps 2 and 4.

4.3 Decryption

The decryption part is completely symmetric to the encryption process as shown in Figure 2,

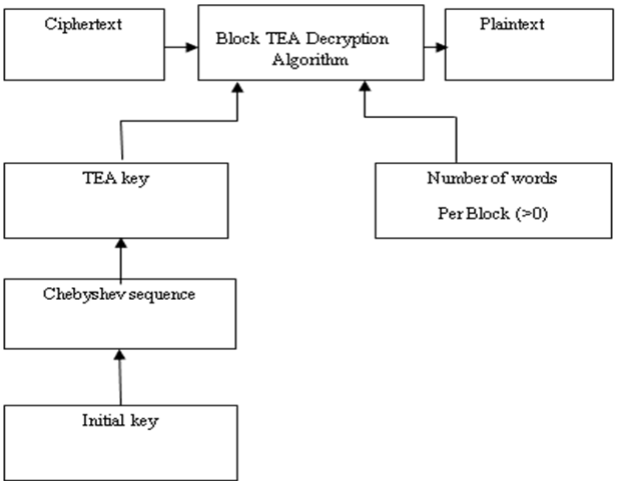

Figure 2: Decryption Mechanism

4.4 Simulation

We tested the algorithm on a plain text file to check it's correctness. The results are shown in the following figures.

4.4.1 File before Encryption

```
#include<stdio.h>
int main()
{
int a;

/*reading value of a*/
printf("Enter a value");
scanf("%d",&a);

/*outputting a value*/
printf("%d is a value", a);

return 0;
}
```

Figure 3: Original File

4.4.2 File after Encryption

```
&¥]_@]z¤Å'    ,n€Ä'}‹¡__XHûÚ&_å®ÿ¶
       _"Ý"_œc¿¿HJ$¤f_ÍÈ‰Ò[²„_Ôdi_4□Ac:}P
¬pÈ_;œ29~_á¼$ÜZ_b¿;n,¯]í(¾]Ô_uá¯
HÍm«5!B^Æ_ÂÃð"ÇË1_˜X§yÙ□ Õö‹©Ñ¿ÃW[äjUï ª HÑj
T«□ _™ƒqòÝ_½d¡î                …Ž¶æ"Ûd¢'tp┖
¬_^¾_V,à,?47>‰U3‡³/³DT_x_`_ã7Á_ïû
       Q)__ûj'Ýv¥p±î
```

Figure 4: Encrypted File

4.4.3 File after Decryption:

```
#include<stdio.h>
```

```
int main()
{
int a;

/*reading value of a*/
printf("Enter a value");
scanf("%d",&a);

/*outputting a value*/
printf("%d is a value", a);

return 0;
}
```

Figure 5: Decrypted File

5. PERFORMANCE ANALYSIS

We tested this algorithm on a text file containing around 800 different words and the result has shown that it has achieved complete diffusion and confusion.

The range of floating point number as per IEEE floating point definition [7] is $\pm 3.4 \times 10^{38}$. This gives a range of 10^{38} to the key space.

With the use of random key and key change for every block of plaintext the algorithm obtains resistance to dictionary crypto-analysis. It also resists attacks like insertion and deletion.

6. CONCLUSIONS

With any initial value between -1 and 1 the Chebyshev sequence can produce a good key. With key being changed for every n blocks of plaintext or after 8 iterations the idea behind one time pad is achieved. Resistance to differential as well as linear cryptanalysis, fast operation than normal TEA, random and dynamic key makes this cipher ideal for encryption.

7. REFERENCES

[1] Ya Li,Jiang-Feng Xu. The New Development Of Based-Chaos Image Encryption Technology[J] . Journal of Henan Normal University(Natural Science) .2005, 33(3): 150-151.

[2] QIU Shui-sheng , Chen Yan-feng , WU Min , MA Zai-guang ,LONG Min. A Novel Scheme of Chaotic Encryption System [J].Journal of Circuits and Systems, 2006,(11): 98-103.

[3] Liao Ni-huan Gao Jin-feng .The Chaotic Spreading Sequences Generated by the Extended Chaotic Map and Its Performance Analysis[J] . Journal Of Electronics & Information Technology.2006 28(7) 1255-1257.

[4] David Wheeler,Roger Needham. TEA, a Tiny Encryption Algorithm[J]. England : Computer Laboratory,Cambridge University,1994.

[5] David Wheeler,Roger Needham. TEA extensions. England : Computer laboratory,Cambridge University,1997

[6] C.E. Shannon Communication theory of secrecy system[J] Bell System Technical Journal 1949 27(4) 656 -715.

[7] W. J. Cody et al. IEEE standards 754 and 854 for Floating-Point Arithmetic[J]. The IEEE Magazine MICRO, Aug. 1984: 84 - 100.

[8] Lu Kai-cheng .Cryptography computer - the computer network data confidentiality and security (3rd edition)[M] Beijing:Tsinghua University Press,2003.50-150.