ISSN (Print) : 2320 – 3765
ISSN (Online) : 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 2, February 2013

Secure-International Data Encryption Algorithm

Harivans Pratap Singh[1], Shweta Verma[2], Shailendra Mishra [3]

Assistant Professor, Dept. of IT, Galgotias College of Engineering and Technology, Gr. Noida, India[1]

Professor & Head, Dept. of IT, Galgotias College of Engineering and Technology, Gr. Noida, India[2]

Professor & Head, Dept. of C.S.E, Bipin Tripathi Kumaon Institute of Technology, Dwarahat, India[3]

Abstract: There are many security algorithms that are used for security purpose. IDEA is one of them. The block cipher IDEA operates with 64-bit plaintext and cipher text blocks and is controlled by a 128-bit key. The fundamental innovation in the design of this algorithm is the use of operations from three different algebraic groups. The algorithm structure has been chosen such that, with the exception that different key sub-blocks are used, the encryption process is identical to the decryption process. The drawback of IDEA is that the large numbers of weak keys were found in IDEA (International Data Encryption Algorithm). Also a new attack on round 6 of IDEA has been detected. In this paper we are describe the design and implementation of secure data encryption algorithm(S-IDEA) protocol, the size of the key has been increased from 128 bits to 256 bits. This increased key size will increase the complexity of the algorithm. To increase the amount of diffusion two MA blocks (multiplicative additive block) are used in a single round of IDEA as compared to one MA block used previously in a single round, With these modifications in the proposed algorithm will increase the cryptographic strength.

Keywords: International data encryption algorithm(IDEA) ,Secure data encryption algorithm(S-IDEA).Multiple additive(MA)

I. INTRODUCTION

Symmetric encryption, also referred to as conventional encryption or single key encryption was the only type of encryption in use prior to the development of public-key encryption in 1970s.It remains by far the most widely used of the two types of the encryption[5].

A symmetric encryption scheme has five ingredients:

- Plaintext: This is the original intelligible message or data that is fed into the algorithm as input.
- Encryption algorithm: It performs various substitutions and transformations on the plaintext.
- Secret key: It is also an input to algorithm which is shared between the sender and receiver and kept secret between them. The exact substitutions and transformations performed by the algorithm depend on the key.
- Ciphertext: This is scrambled message produced as output. It depends on the plaintext and the secret key. The ciphertext is random stream of data and is unintelligible.
- Decryption algorithm: It is encryption algorithm run in reverse. It takes the ciphertext and secret key and produces the original plaintext.

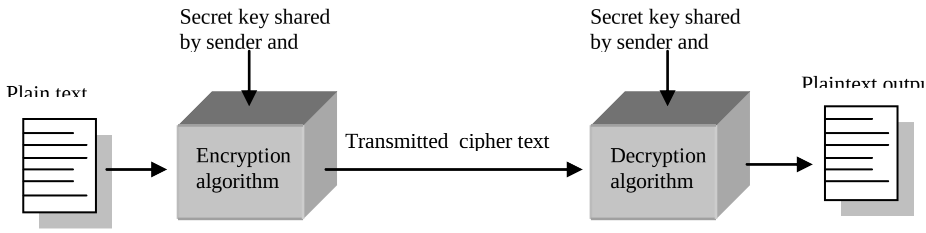


Fig.1 General Model of symmetric encryption[5]

DEFINITION 1.

A cryptosystem is a five -tuple (P, C, K, E, D), where the following conditions are satisfied:

1. P is a finite set of possible plain texts.

2. C is a finite set of possible ciphertexts.

3. K, the keyspace, is a finite set of possible keys.

4. For each K ε k, there is an encryption rule e_K ε E. and a corresponding decryption rule d_K ε D. Each $e_K : P \rightarrow C$ and $d_K : C \rightarrow P$ are functions such that $d_K(e_K(x)) = x$ for every plaintext x ε P.

The main property is property 4. It says that if a plaintext x is encrypted using eK, and the resulting cipher text is subsequently decrypted using dK, then the original plaintext x results.

Cryptography may be divide in two main categorries

- Asymmetric: For Encryption and Decryption using a pair of keys.
- Symmetric: For Encryption and Decryption using the same key (or without key – in the case of Hash function)

All classical cryptosystems (that is cryptosystems that were developed before 1970s) are examples of symmetric keycryptosystems[6]. In addition, most modern cryptosystems are symmetric as well. Some of the most popular examples of modern symmetric key cryptosystems include AES (Advanced Encryption Standard), DES (Data Encryption Standard), IDEA, FEAL, RC5, and many others[2]. All symmetric key cryptosystems have a common property: they rely on a shared secret between communicating parties. This secret is used both as an encryption key and as a decryption key (thus the keyword "symmetric" in the name). This type of cryptography ensures only confidentiality and fails to provide the other objectives of cryptography. Even more importantly the disadvantage of symmetric key cryptography is that it cannot handle large communication networks[2]. If a node in a communication network of n nodes needs to communicate confidentially with all other nodes in the network, it needs n - 1 shared secrets. For large value of n this is highly impractical and inconvenient. On the other hand, an advantage over public key cryptosystems is that symmetric cryptosystems require much smaller key sizes for the same level of security[1]. Hence, the computations are much faster and the memory requirements are smaller (Whitfield et al, 1976)

In public key cryptosystems there are two different keys: a public key, which is publicly known, and the secret key, which is kept secret by the owner. The system is called “asymmetric” since the different keys are used for encryption and decryption– the public key and the private key[1]. If data is encrypted with a public key, it can only be decrypted using the corresponding private key, and vice versa[3]. Today, all public key cryptosystems rely on some computationally intractable problems. For example, the cryptosystem RSA relies on the difficulty of factoring large integers, while El-Gamal cryptosystem relies on the discrete logarithm problem (DLP), which is the problem of finding a logarithm of a group element with generator base in a finite Abelian group Public key cryptosystems do not need to have a shared secret between communicating parties. This solves the problem of large confidential communication network introduced earlier. In addition, public key cryptography opened door for ways of implementing technologies to ensure all goals of cryptography. By means of combining public key cryptography, public key certification, and secure hash functions, there are protocols that enable digital signatures, authentication, and data integrity[1]. Due to the increase in processor speed and even more due to smart modern cryptanalysis, the key size for public key cryptography grew very large. This created a disadvantage in comparison to symmetric key cryptosystems: public key cryptography is significantly slower[2], and requires large memory capacity and large computational power. As an example, a 128-bit key used with DES cryptosystem has approximately the same level of security as the 1024-bit key used with RSA public key cryptosystem (Oorschot P.C. van, et al, 1997).

II. BACKGROUND AND RELATED WORK

The cryptographic schemes are divided into two parts.

1. Classical Cryptography

2. Modern Cryptography

Classical ciphers are often divided into transposition ciphers and ciphers. A substitution cipher replaces the one symbol with another. if the symbol in the plain text are alphabetic character ,we replace one character with another. For example assume the statement to be encrypted is “this is the final chapter”[7].We apply substitution cipher and change each character by its next ASCII character, the resultant statement would be,“uijt jt uif gjomb………”This method is popularly known as Caesar cipher, as it was invited by Roman general Julius Carsar.Whan a character is substituted by another character, it is also known as monoalphabatic cipher.

transposition cipher changes the position of character. As appose of the substitution cipher, the transposition cipher change the order but not changes the character themselves. for example reshuffle the original string by sending the first character to 2nd position ,third character to 4th position ,the fifth character to 3rd position and so on[7].

Modern Cryptography

Public key cryptography(asymmetric)

(a) Encryption

(i) Integer factorization: RSA and Rabin encryption.

(ii) Discrete Logarithm: ElGamal Encryption.

(iii) Elliptic Curve: Elliptic Curve Cryptosystem.

(iv) Chaotic: Fractal Encryption.

(v) Others: NTRU, Gang-Harn.

(b) Key Sharing

(i) Discrete Logarithm: Diffie-Hellman.

(ii) Elliptic Curve: ECDH.

(iii) Chaotic: Fractal Key Exchange.

(c) Digital Signature

(i) Integer factorization: RSA digital signature.

(ii) Discrete Logarithm: DSA.

(iii) Elliptic Curve: ECDSA.

(iv) Chaotic: Fractal Digital signature.

Non public key

(a) Secret key (Symmetric)

(i) Block cipher: DES

(ii) Stream cipher.

(b) Hash function Secure Hash Algorithm (SHA).

The block cipher IDEA operates with 64-bit plaintext and cipher text blocks and is controlled by a 128-bit key. The fundamental innovation in the design of this algorithm is the use of operations from three different algebraic groups. The algorithm structure has been chosen such that, with the exception that different key sub-blocks are used, the encryption process is identical to the decryption process. The drawback of IDEA is that the large numbers of weak keys were found in IDEA (International Data Encryption Algorithm)[4]. Also a new attack on round 6 of IDEA has been detected. For this reason we have proposed secure data encryption algorithm(S-IDEA) protocol

III. SYSTEM DESIGN AND IMPLEMENTATION

(A) Encryption for S-IDEA

The encryption process consists of eight identical encryption steps (known as encryption rounds) followed by an output transformation. The structure of the first round is shown in detail.

In the first encryption round, the first four 16-bit key sub-blocks are combined with two of the 16-bit plaintext blocks using addition modulo 216, and with the other two plaintext blocks using multiplication modulo 216 + 1. The results are then processed further as shown in Figure3.4, whereby two more 16-bit key sub-blocks enter the calculation and the third algebraic group operator, the bit-by-bit exclusive OR, is used. At the end of the first encryption round four 16-bit values are produced which are used as input to the second encryption round in a partially changed order. The process described above for round one is repeated in each of the subsequent 7 encryption rounds using different 16-bit key sub-blocks for each combination. During the subsequent output transformation, the four 16-bit values produced at the end of the 8th encryption round are combined with the last four of the 104 key sub-blocks using addition modulo 216 and multiplication modulo 216 + 1 to form the resulting four 16-bit cipher text blocks.

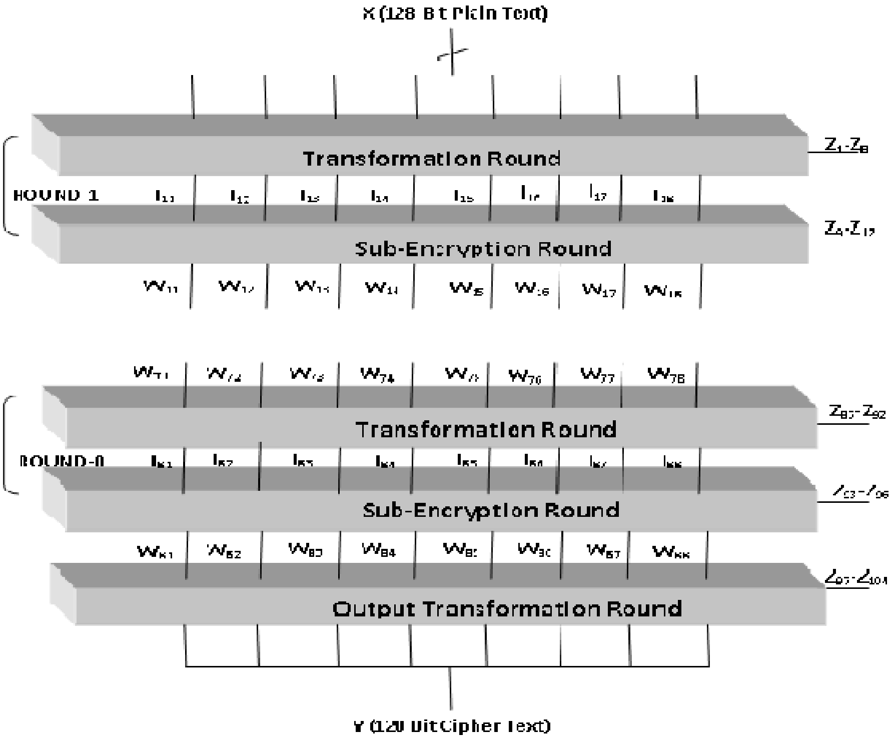

X (128 Bit Plain Text)
Transformation Round
Z_1-Z_8
ROUND 1
I_{11} I_{12} I_{13} I_{14} I_{15} I_{16} I_{17} I_{18}
Z_9-Z_{12}
Sub-Encryption Round
W_{11} W_{12} W_{13} W_{14} W_{15} W_{16} W_{17} W_{18}
W_{71} W_{72} W_{73} W_{74} W_{75} W_{76} W_{77} W_{78}
Transformation Round
Z_{85}-Z_{92}
ROUND-8
I_{81} I_{82} I_{83} I_{84} I_{85} I_{86} I_{87} I_{88}
Sub-Encryption Round
Z_{93}-Z_{96}
W_{81} W_{82} W_{83} W_{84} W_{85} W_{86} W_{87} W_{88}
Z_{97}-Z_{104}
Output Transformation Round
Y (128 Bit Cipher Text)

Fig 2: Block diagram of S-IDEA

Encryption algorithms for S-IDEA

```
Start
unsigned in[9],unsigned out[9],unsigned
z[13][10]
unsigned i1,i2,i3,i4,i5,i6,i7,i8,kk,t1,t2,a,r;
int x;
i1=in [1], i2=in [2], i3=in [3], i4=in [4].
i5=in [5], i6=in [6], i7=in [7], i8=in [8].
for(r=1;r<=8;r++)
i1=mul(i1,z[1][r])
i2=(i2+z[2][r])&one
i3=(i3+z[3][r])&one
i4=mul(i4,z[4][r])
i5=mul(i5,z[5][r])
i6=(i6+z[6][r])&one
i7=(i7+z[7][r])&one
i8=mul(i8,z[8][r])
kk=mul(z[9][r],(i1^i3))
t1=mul(z[10][r],(kk+(i2^i4))&one)
t2=(kk+t1)&one
a=i2^t2
i1=i1^t1
i2=i3^t1
i3=a
i4=i4^tz
kk=mul(z[11][r],(i5^i7))
t1=mul(z[12][r],(kk+(i6^i8))&one)
t2=(kk+t1)&one
a=i6^t2
i5=i5^t1
i6=i7^t1
i7=a
i8=i8^t2
end
```

Detail of single round

(1)The 128 bits text is processed in 8 block of 16 bit each.

(2)The proposed modified version of IDEA(S-IDEA) can be seen as two sub-block of 64 bits running in parallel with each other. Each round in encryption uses two MA block and 12 keys.

(3)Each round consists of two further divisions i.e. Transformation followed by Sub-Encryption, transformation in each round uses 8 keys whereas sub-encryption uses 4 keys.

(4)The former description of keys is valid for round from 1 to 8 where as the 9th round called the Output transformation round uses 8 keys

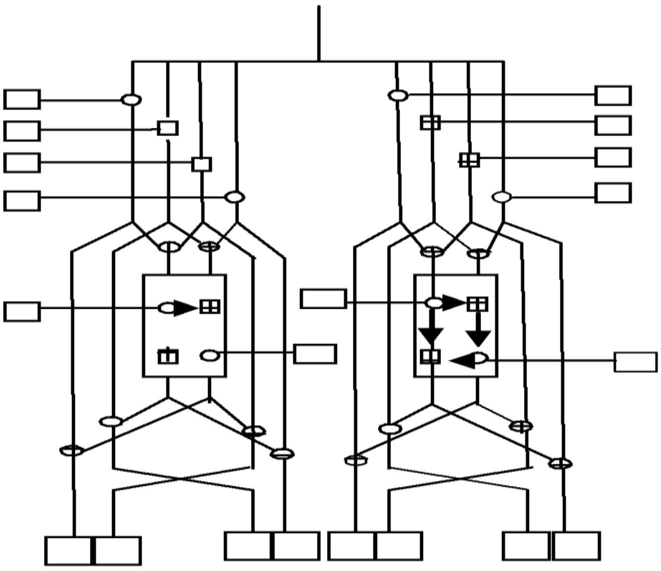


Fig 3. Structural details of round 1

From fig.1and fig.3 following relations can be written:

$W_{11}=I_{11} \oplus MA_{R1} (I_{11} \oplus I_{13}, I_{12} \oplus I_{14})$

$W_{12}=I_{13} \oplus MA_{R1} (I_{11} \oplus I_{13}, I_{12} \oplus I_{14})$

$W_{13}=I_{12} \oplus MA_{L1} (I_{11} \oplus I_{13}, I_{12} \oplus I_{14})$

$W_{14}=I_{14} \oplus MA_{L1} (I_{11} \oplus I_{13}, I_{12} \oplus I_{14})$

$W_{15}=I_{15} \oplus MA_{R2} (I_{15} \oplus I_{17}, I_{16} \oplus I_{18})$

$W_{16}=I_{17} \oplus MA_{R2} (I_{15} \oplus I_{17}, I_{16} \oplus I_{18})$

$W_{17}=I_{16} \oplus MA_{L2} (I_{15} \oplus I_{17}, I_{16} \oplus I_{18})$

$W_{18}=I_{18} \oplus MA_{L2} (I_{15} \oplus I_{17}, I_{16} \oplus I_{18}$

$W_{81}=I_{81} \oplus MA_{R1} (I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{82}=I_{83} \oplus MA_{R1} (I_{81} \oplus I_{83}, I_{82} \oplus I_{84}$

$W_{83}=I_{82} \oplus MA_{L1} (I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{84}=I_{84} \oplus MA_{L1} (I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{85}=I_{85} \oplus MA_{R2} (I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$

$W_{86}=I_{87} \oplus MA_{R2}\ (I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$

$W_{87}=I_{86} \oplus MA_{L2}\ (I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$

$W_{88}=I_{88} \oplus MA_{L2}\ (I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$

The following relations can be seen from fig.4:

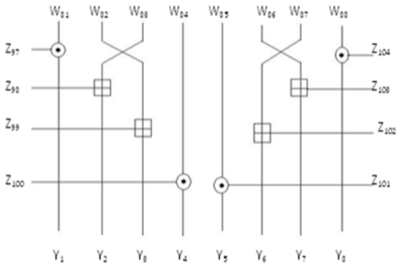


Fig 4. Output Transformation round

Table.1 Generation of encryption sub-keys

$Y_1=W_{81} \odot Z_{97}$

$Y_2=W_{83} \boxplus Z_{98}$

$Y_3=W_{82} \boxplus Z_{99}$

$Y_4=W_{84} \odot Z_{100}$

$Y_5=W_{85} \odot Z_{101}$

$Y_6=W_{87} \boxplus Z_{102}$

$Y_7=W_{86} \boxplus Z_{103}$

$Y_8=W_{88} \odot Z_{104}$

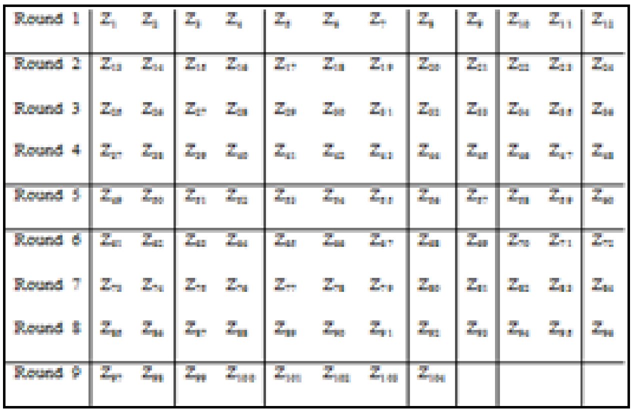
Algorithm of output transformation round

```
Start
out[1]=mul(i1,z[1][round+1])
out[2]=(i3+z[2][round+1])&one
out[3]=(i2+z[3][round+1])&one
out[4]=mul(i4,z[4][round+1])
out[5]=mul(i5,z[5][round+1])
out[6]=(i7+z[6][round+1])&one
out[7]=(i6+z[7][round+1])&one
out[8]=mul(i8,z[8][round+1])
for(x=1;x<9;x++)
out[x]
End
```

(B) Encryption sub key generation for S-IDEA:

The sample user key is fed into key generation for each round module which generates 16 sub-keys in each round.Sample user key is also fed into shift logic module that shifts the key by 25 bit circular left shift and again 16 sub-keys are generated.The process is carried out recursively till all 104 sub-keys are generated.

Table2 for encryption sub key



Round 1	Z_{1}	Z_{2}	Z_{3}	Z_{4}	Z_{5}	Z_{6}	Z_{7}	Z_{8}	Z_{9}	Z_{10}	Z_{11}	Z_{12}
Round 2	Z_{13}	Z_{14}	Z_{15}	Z_{16}	Z_{17}	Z_{18}	Z_{19}	Z_{20}	Z_{21}	Z_{22}	Z_{23}	Z_{24}
Round 3	Z_{25}	Z_{26}	Z_{27}	Z_{28}	Z_{29}	Z_{30}	Z_{31}	Z_{32}	Z_{33}	Z_{34}	Z_{35}	Z_{36}
Round 4	Z_{37}	Z_{38}	Z_{39}	Z_{40}	Z_{41}	Z_{42}	Z_{43}	Z_{44}	Z_{45}	Z_{46}	Z_{47}	Z_{48}
Round 5	Z_{49}	Z_{50}	Z_{51}	Z_{52}	Z_{53}	Z_{54}	Z_{55}	Z_{56}	Z_{57}	Z_{58}	Z_{59}	Z_{60}
Round 6	Z_{61}	Z_{62}	Z_{63}	Z_{64}	Z_{65}	Z_{66}	Z_{67}	Z_{68}	Z_{69}	Z_{70}	Z_{71}	Z_{72}
Round 7	Z_{73}	Z_{74}	Z_{75}	Z_{76}	Z_{77}	Z_{78}	Z_{79}	Z_{80}	Z_{81}	Z_{82}	Z_{83}	Z_{84}
Round 8	Z_{85}	Z_{86}	Z_{87}	Z_{88}	Z_{89}	Z_{90}	Z_{91}	Z_{92}	Z_{93}	Z_{94}	Z_{95}	Z_{96}
Round 9	Z_{97}	Z_{98}	Z_{99}	Z_{100}	Z_{101}	Z_{102}	Z_{103}	Z_{104}				

Algorithms for sub-key generation of S-IDEA

```
1. start
2. short unsigned userkey[17],unsigned z[13][10]
3. int i,r,j.
4. unsigned s[109]
5.  for(i=1;i<17;i++)
6.      s[i-1]=userkey[i]
7.        for(i=16;i<109;i++)
8.          do  if((i+2)%16==0)
9.                s[i]=(s[i-15]<<9)|(s[i-30]>>7)
10.        else if((i+1)%16==0)
11.                s[i]=(s[i-31]<<9)|(s[i-30]>>7)
12.                else
13.        s[i]=(s[i-15]<<9)|(s[i-14]>>7)
14.  for(r=1;r<=round+1;r++)
15.        for(j=1;j<13;j++)
16.              z[j][r]=s[12*(r-1)+j-1]
17. for(j=1;j<13;j++)
18.       for(r=1;r<=round+1;r++)
19.         do  if(r==9&&j>=9)
20.                Z[-][-]
21.         else
22.               z[j][r])
23. end
```

(c) Decryption of S-IDEA:

It implements all the 8+1 rounds of IDEA implementation withall functionalities (Additive modulo, Multiplicative modulo,Exclusive-OR,MA block etc.) of S-IDEA. Decryption .The decryption process is same as encryption process The output ofeach round is denoted by V whereas the intermediate output undergoing transformation in each round denoted by J.

Relation between J and W

The output of transformation round of decryption (J) is related to output of sub-encryption round of encryption (W) and vice versa. Consider the fig.5.We can write the following relations:

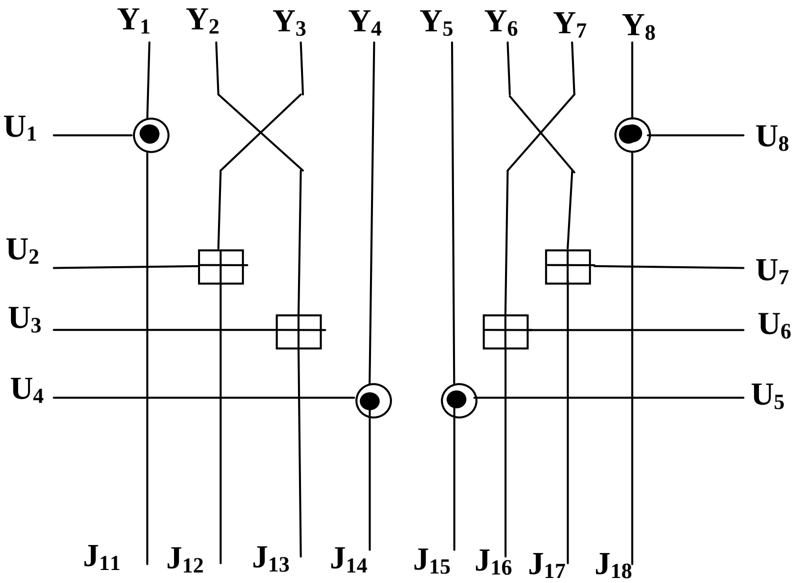


Fig 5. Decryption (transformation round) round 1

Table 2. Relation between encryption and decryption sub key

$$Y_1=W_{81} \odot Z_{97}$$
$$Y_2=W_{83} \boxplus Z_{98}$$
$$Y_3=W_{82} \boxplus Z_{99}$$
$$Y_4=W_{84} \odot Z_{100}$$
$$Y_5=W_{85} \odot Z_{101}$$
$$Y_6=W_{87} \boxplus Z_{102}$$
$$Y_7=W_{86} \boxplus Z_{103}$$
$$Y_8=W_{88} \odot Z_{104}$$

Substituting Y_1 in the equation corresponding to J_{11}:

$$J_{11}=Y_1 \odot U_1$$
$$= W_{81} \odot Z_{97} \odot U_1$$
$$= W_{81} \odot Z_{97} \odot Z_{97}^{-1} \quad (U_1=Z_{97}^{-1})$$
$$= W_{81} \odot 1$$

$\mathbf{J_{11} = W_{81}}$

Substituting Y_2 in the equation corresponding to J_{12}:

$$J_{12}=Y_2 \boxplus U_2$$
$$= W_{83} \boxplus Z_{98} \boxplus U_2$$
$$= W_{83} \boxplus Z_{98} \boxplus -Z_{98} \quad (U_2= -Z_{98})$$
$$=W_{83} \boxplus 0$$

$\mathbf{J_{12} = W_{83}}$

Similarly we can derive the following equivalences:

Table.3 Relation between J and W

$J_{11}=W_{81}$	$J_{15}=W_{85}$
$J_{12}=W_{83}$	$J_{16}=W_{87}$
$J_{13}=W_{82}$	$J_{17}=W_{86}$
$J_{14}=W_{84}$	$J_{18}=W_{88}$

Relation between V and I

We can write V in decryption corresponding to W in encryption as follows:

$$V_{11}=J_{11} \oplus MA_{R1} (J_{11} \oplus J_{13}, J_{12} \oplus J_{14})$$

$$V_{12}=J_{13} \oplus MA_{R1} (J_{11} \oplus J_{13}, J_{12} \oplus J_{14})$$

$$V_{13}=J_{12} \oplus MA_{L1} (J_{11} \oplus J_{13}, J_{12} \oplus J_{14})$$

$$V_{14}=J_{14} \oplus MA_{L1} (J_{11} \oplus J_{13}, J_{12} \oplus J_{14})$$

$$V_{15}=J_{15} \oplus MA_{R2} (J_{15} \oplus J_{17}, J_{16} \oplus J_{18})$$

$$V_{16}=J_{17} \oplus MA_{R2} (J_{15} \oplus J_{17}, J_{16} \oplus J_{18})$$

$$V_{17}=J_{16} \oplus MA_{L2} (J_{15} \oplus J_{17}, J_{16} \oplus J_{18})$$

$$V_{18}=J_{18} \oplus MA_{L2} (J_{15} \oplus J_{17}, J_{16} \oplus J_{18})$$

Consider the following equation:

$$V_{15}=J_{15} \oplus MA_{R2} (J_{15} \oplus J_{17}, J_{16} \oplus J_{18})$$

Substituting the values of J_{15}, J_{16}, J_{17}, J_{18} :

$$V_{15}=W_{85} \oplus MA_{R2} (W_{85} \oplus W_{86}, W_{87} \oplus W_{88})$$

Substituting the values of W_{85} ,W_{86}, W_{87} ,W_{88} :

$$V_{15} = [I_{85} \oplus MA_{R2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88})] \oplus$$

$$MA_{R2} [I_{85} \oplus MA_{R2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$$

$$\oplus I_{87} \oplus MA_{R2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88}),$$

$$I_{86} \oplus MA_{L2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88})$$

$$\oplus I_{88} \oplus MA_{L2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88})]$$

$$V_{15} = [I_{85} \oplus MA_{R2}(I_{85} \oplus I_{87}, I_{86} \oplus I_{88})]$$

$$\oplus MA_{R2} [I_{85} \oplus I_{87}, I_{86} \oplus I_{88}]$$

$$V_{15} = I_{85} \oplus 0$$

$$\mathbf{V_{15}=I_{85}}$$

Similarly we can derive the following equivalences:

Table.4 Relation between V and I

$V_{11}=I_{81}$	$V_{15}=I_{85}$
$V_{12}=I_{83}$	$V_{16}=I_{87}$
$V_{13}=I_{82}$	$V_{17}=I_{86}$
$V_{14}=I_{84}$	$V_{18}=I_{88}$

Algorithms for S-IDEA Decryption:

```
1. Start
2. Unsigned in[9],unsigned out[9],unsigned dk[13][10])
3. unsigned i1,i2,i3,i4,i5,i6,i7,i8,kk,t1,t2,a,r
4. int x
5.  i1=in[1]   i2=in[2] i3=in[3] i4=in[4]
6.  i5=in[5]   i6=in[6] i7=in[7] i8=in[8]
7.for(r=1;r<=8;r++)
8. i1=mul(i1,dk[1][r])
9.i2=(i2+dk[2][r])&one
10.i3=(i3+dk[3][r])&one
11. i4=mul(i4,dk[4][r])
12. i5=mul(i5,dk[5][r])
13.i6=(i6+dk[6][r])&one
14.i7=(i7+dk[7][r])&one
15. i8=mul(i8,dk[8][r])
17.kk=mul(dk[9][r],(i1^i3))
18.t1=mul(dk[10][r],(kk+(i2^i4))&one)
19. t2=(kk+t1)&one
20.a=i2^t2; i1=i1^t1
21.i2=i3^t1;  i3=a;      i4=i4^t2
22.kk=mul(dk[11][r],(i5^i7))
23. t1=mul(dk[12][r],(kk+(i6^i8))&one)
24. t2=(kk+t1)&one
25. a=i6^t2;  i5=i5^t1
26. i6=i7^t1;      i7=a;         i8=i8^t2
27. End
```

Table 5. Relation between encryption and decryption sub key

Round 1

De.Sub-keys	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9	U_{10}	U_{11}	U_{12}
Equivalent to	Z_{97}^{-1}	$-Z_{98}$	$-Z_{99}$	Z_{100}^{-1}	Z_{101}^{-1}	$-Z_{102}$	$-Z_{103}$	Z_{104}^{-1}	Z_{93}	Z_{94}	Z_{95}	Z_{96}

Round 2

De.Sub-keys	U_{13}	U_{14}	U_{15}	U_{16}	U_{17}	U_{18}	U_{19}	U_{20}	U_{21}	U_{22}	U_{23}	U_{24}
Equivalent to	Z_{85}^{-1}	$-Z_{87}$	$-Z_{86}$	Z_{88}^{-1}	Z_{89}^{-1}	$-Z_{91}$	$-Z_{90}$	Z_{92}^{-1}	Z_{81}	Z_{82}	Z_{83}	Z_{84}

Round 3

De.Sub-keys	U_{25}	U_{26}	U_{27}	U_{28}	U_{29}	U_{30}	U_{31}	U_{32}	U_{33}	U_{34}	U_{35}	U_{36}
Equivalent to	Z_{73}^{-1}	$-Z_{75}$	$-Z_{74}$	Z_{76}^{-1}	Z_{77}^{-1}	$-Z_{79}$	$-Z_{78}$	Z_{80}^{-1}	Z_{69}	Z_{70}	Z_{71}	Z_{72}

Round 4

De.Sub-keys	U_{37}	U_{38}	U_{39}	U_{40}	U_{41}	U_{42}	U_{43}	U_{44}	U_{45}	U_{46}	U_{47}	U_{48}
Equivalent to	Z_{61}^{-1}	$-Z_{63}$	$-Z_{62}$	Z_{64}^{-1}	Z_{65}^{-1}	$-Z_{67}$	$-Z_{66}$	Z_{68}^{-1}	Z_{57}	Z_{58}	Z_{59}	Z_{60}

Round 5

De.Sub-keys	U_{49}	U_{50}	U_{51}	U_{52}	U_{53}	U_{54}	U_{55}	U_{56}	U_{57}	U_{58}	U_{59}	U_{60}
Equivalent to	Z_{49}^{-1}	$-Z_{51}$	$-Z_{50}$	Z_{52}^{-1}	Z_{53}^{-1}	$-Z_{55}$	$-Z_{54}$	Z_{56}^{-1}	Z_{45}	Z_{46}	Z_{47}	Z_{48}

Round 6

De.Sub-keys	U_{61}	U_{62}	U_{63}	U_{64}	U_{65}	U_{66}	U_{67}	U_{68}	U_{69}	U_{70}	U_{71}	U_{72}
Equivalent to	Z_{37}^{-1}	$-Z_{39}$	$-Z_{38}$	Z_{40}^{-1}	Z_{41}^{-1}	$-Z_{43}$	$-Z_{42}$	Z_{44}^{-1}	Z_{33}	Z_{34}	Z_{35}	Z_{36}

Round 7

De.Sub-keys	U_{73}	U_{74}	U_{75}	U_{76}	U_{77}	U_{78}	U_{79}	U_{80}	U_{81}	U_{82}	U_{83}	U_{84}
Equivalent to	Z_{25}^{-1}	$-Z_{27}$	$-Z_{26}$	Z_{28}^{-1}	Z_{29}^{-1}	$-Z_{31}$	$-Z_{30}$	Z_{32}^{-1}	Z_{21}	Z_{22}	Z_{23}	Z_{24}

Round 8

De.Sub-keys	U_{85}	U_{86}	U_{87}	U_{88}	U_{89}	U_{90}	U_{91}	U_{92}	U_{93}	U_{94}	U_{95}	U_{96}
Equivalent to	Z_{13}^{-1}	$-Z_{15}$	$-Z_{14}$	Z_{16}^{-1}	Z_{17}^{-1}	$-Z_{19}$	$-Z_{18}$	Z_{20}^{-1}	Z_{9}	Z_{10}	Z_{11}	Z_{12}

Round 9

De.Sub-keys	U_{97}	U_{48}	U_{99}	U_{100}	U_{101}	U_{102}	U_{103}	U_{104}				
Equivalent to	Z_{1}^{-1}	$-Z_{2}$	$-Z_{3}$	Z_{4}^{-1}	Z_{5}^{-1}	$-Z_{6}$	$-Z_{7}$	Z_{8}^{-1}				

IV. EXPERIMENTAL RESULT

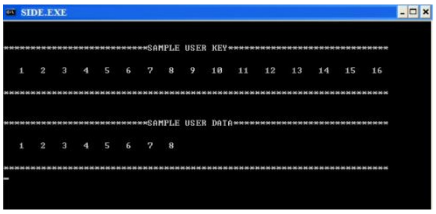
The result displays in fig. 6 shows the sample user key that is generated in the proposed algorithm. From the user key all 104 sub-keys are generated by the algorithm. It also displays the sample user data generated in the algorithm which has to be encrypted using S-IDEA.

Fig.7 depicts 104 encryption sub-keys generated by the proposed algorithm from sample user key. It also shows which sub-keys will be used in which encryption round.

Fig. 8 shows the progress of S-IDEA Encryption process. It shows the data encrypted after each round and finally the cipher text is given as the output

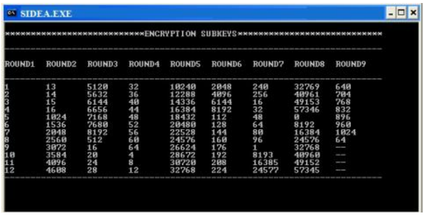
Fig 9, shows the 104 decryption sub-keys generated by the algorithm from encryption sub-keys. It also shows which sub-keys will be used in which decryption round.

Fig 10 ,shows the progress of S-IDEA Decryption process. It shows the data decrypted after each round and finally the plain text is given as the output.



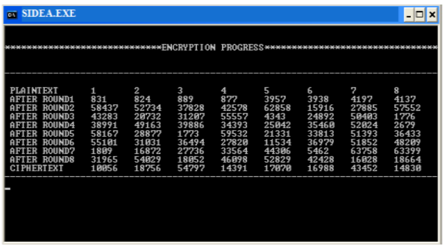
```
****************************SAMPLE USER KEY****************************

   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16

***********************************************************************


****************************SAMPLE USER DATA***************************

   1    2    3    4    5    6    7    8

***********************************************************************
_
```

Fig 6. Simple user data encrypted by S-IDEA



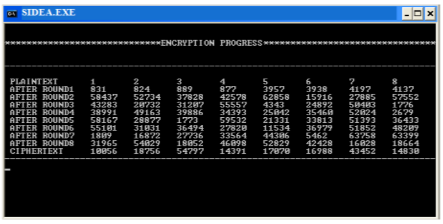
```
***************************ENCRYPTION SUBKEYS***************************

------------------------------------------------------------------------

ROUND1   ROUND2   ROUND3   ROUND4   ROUND5   ROUND6   ROUND7   ROUND8   ROUND9

------------------------------------------------------------------------
1        13       5120     32       10240    2048     240      32769    640
2        14       5632     36       12288    4096     256      40961    704
3        15       6144     40       14336    6144     16       49153    768
4        16       6656     44       16384    8192     32       57346    832
5        1024     7168     48       18432    112      48       0        896
6        1536     7680     52       20480    128      64       8192     960
7        2048     8192     56       22528    144      80       16384    1024
8        2560     512      60       24576    160      96       24576    64
9        3072     16       64       26624    176      1        32768    --
10       3584     20       4        28672    192      8193     40960    --
11       4096     24       8        30720    208      16385    49152    --
12       4608     28       12       32768    224      24577    57345    --
------------------------------------------------------------------------
```

Fig 7. Encryption sub-keys generated by S-IDEA



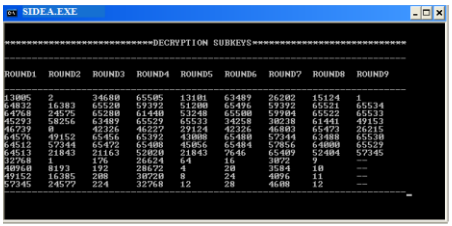
```
******************************ENCRYPTION PROGRESS******************************


-------------------------------------------------------------------------------

 PLAINTEXT      1        2        3        4        5        6        7        8
 AFTER ROUND1   831      824      889      877      3957     3938     4197     4137
 AFTER ROUND2   58437    52734    37828    42578    62858    15916    27885    57552
 AFTER ROUND3   43283    20732    31207    55557    4343     24892    50403    1776
 AFTER ROUND4   38991    49163    39886    34393    25042    35460    52024    2679
 AFTER ROUND5   58167    28877    1773     59532    21331    33813    51393    36433
 AFTER ROUND6   55101    31031    36494    27820    11534    36979    51852    48209
 AFTER ROUND7   1809     16872    27736    33564    44306    5462     63758    63399
 AFTER ROUND8   31965    54029    18052    46098    52829    42428    16028    18664
 CIPHERTEXT     10056    18756    54797    14391    17070    16988    43452    14830
-------------------------------------------------------------------------------
_
```

Fig 8. S-IDEA Decryption process



```
******************************ENCRYPTION PROGRESS******************************


-------------------------------------------------------------------------------

 PLAINTEXT      1        2        3        4        5        6        7        8
 AFTER ROUND1   831      824      889      877      3957     3938     4197     4137
 AFTER ROUND2   58437    52734    37828    42578    62858    15916    27885    57552
 AFTER ROUND3   43283    20732    31207    55557    4343     24892    50403    1776
 AFTER ROUND4   38991    49163    39886    34393    25042    35460    52024    2679
 AFTER ROUND5   58167    28877    1773     59532    21331    33813    51393    36433
 AFTER ROUND6   55101    31031    36494    27820    11534    36979    51852    48209
 AFTER ROUND7   1809     16872    27736    33564    44306    5462     63758    63399
 AFTER ROUND8   31965    54029    18052    46098    52829    42428    16028    18664
 CIPHERTEXT     10056    18756    54797    14391    17070    16988    43452    14830
-------------------------------------------------------------------------------
_
```

Fig 9. Decryption sub key generated by S-IDEA



```
***************************DECRYPTION SUBKEYS***************************

------------------------------------------------------------------------

ROUND1   ROUND2   ROUND3   ROUND4   ROUND5   ROUND6   ROUND7   ROUND8   ROUND9

------------------------------------------------------------------------
13005    2        34680    65505    13101    63489    26202    15124    1
64832    16383    65520    59392    51200    65496    59392    65521    65534
64768    24575    65280    61440    53248    65500    59904    65522    65533
45293    58256    63489    65529    65533    34258    30238    61441    49153
46739    0        42326    46227    29124    42326    46803    65473    26215
64576    49152    65456    65392    43008    65480    57344    63488    65530
64512    57344    65472    65408    45056    65484    57856    64000    65529
64513    21843    21163    52020    21843    7646     65409    52404    57345
32768    1        176      26624    64       16       3072     9        --
40960    8193     192      28672    4        20       3584     10       --
49152    16385    208      30720    8        24       4096     11       --
57345    24577    224      32768    12       28       4608     12       --
------------------------------------------------------------------------_
```

Fig 10. S-IDEA Decryption process

V. CONCLUSION AND FUTURE WORK

The basic aim of this paper is to increase the strength of existing IDEA algorithm. The proposed algorithm International data Encryption algorithm (S-IDEA) has two key features:-increased key size (256 bits) and increased degree of diffusion (two MA blocks are used in a single round instead of one). The 104 sub-keys are being used as compared to 52 sub-keys previously which enhance the complexity of confusion. Therefore the probability of other forms of attack is reduced due to amount of work that has to be carried out when 104 sub-keys are involved. Addition of a new MA block in each round of SIDEA has contributed to an increase in complexity of diffusion. It makes the algorithm more secure and less susceptible to cryptanalysis.

The proposed algorithms increased the cryptographic strength and eliminate the shortcoming of the existing International data Encryption algorithm (IDEA).The future scope of S-IDEA algorithm is that it can also be implemented in hardware using VLSI technology.

REFRENCES

[1]. *Blum M. and Goldwasser S*., "An efficient probabilistic public-key encryption scheme which hides all partial information," Advances in Cryptology-CRYPTO'84, Lecture notes in computer science (Springer-Verlag), pp.289-299, (1995).

[2]. *Bellare M., Desai A., Jokipii E. and Rogaway P.*,"A concrete security treatment of symmetric encryption: analysis of the DES modes of operation", In Proc. 38th Annual Symposium on Foundations of Computer Science,(1997).

[3]. *Bellare M., Desai A., Pointcheval D. and Rogaway P*., "Relations among notions of security for public-key encryption schemes", Advances in Cryptology CRYPTO '98, Lecture Notes in Computer Science, 1462, Springer-Verlag, pp. 26-45, (1998).

[4]*Biryukov, Alex; Nakahara, Jorge Jr.; Preneel, Bart; Vandewalle, Joos*, "New Weak-Key Classes of IDEA", Information and Communications Security, 4th InternationalConference, ICICS 2002.

[5]. *William Stalling* "Cryptography and Network Security".

[6]. *Bruce Schiener* "Applied Cryptography ".

[7].Behrouz A forouzan"Cryptography and network security"

Biography

Harivans pratap singh received the B.Tech. degree in computer science from the Faculty of Engineering and Technology, R.B.S College Agra, India and he is currently working toward the M.Tech. degree in the Department of Computer Science, Uttarakhand technical University. He is an Assistant Professor, Department of Information Technology, Galgotia's college of engineering and technology Mahamaya technical University. His research area include information systems security and privacy.

Dr . Shailendra Mishra received Ph.D degree in CSE & Master of Engineering Degree (M E) in Computer Science & Engineering (Specialization : Software Engineering) from MNREC Allahabad(Now MotiLal Nehru National Institute of Technology (MNNIT)) India, Presently he is Professor & Head , Department of ComputerScience & Engineering , KEC Dwarahat India. His recent research has been in the field of Mobile Computing & Communication, Advance Network Architecture and Software Engineering. He has also been conducting research on Communication System & Computer Networks with Performance evaluation and design of Multiple Access Protocol for Mobile Communication Network. He handled many research projects during the last 5 years; Power control and recourse management for WCDMA System funded and sponsored by UCOST Dehradun Uttrakhand, Code and Time complexity for WCDMA System, OCQPSK spreading techniques for third generation communication system, "IT mediated education and dissemination of health information via Training & e-Learning Platform" sponsored and funded by Oil Natural Gas Commission (ONGC), New Delhi, India (November 2006),"IT based Training and E-Learning Platform", sponsored and funded by UCOST, Department of Science and Technology,Govt. of Uttarakhand, India (December 2006) etc. He received Young Scientist Award in the Yr 2006 and 2008 from DST UCOST Govt.of Uttrakhand.

He had supervised 6 Ph.D and currently guiding five research scholars. He had authored four books in the area of Computer Network and Security and published and presented 60 research papers in international journals and international conferences and wrote more than 10 articles on various topics in national magazines. He is recipient of Young Scientist Award in IIIrd Uttrakhand State Science Congress & Ist Uttaranchal State Science Congress organized by Uttaranchal Council for Science &Technology, Department of Science &Technology, Govt. of Uttarkhand, India (10,11Nov 2008 &11 Nov 2006).He is Member of Institution of Engineers India (IEI) and ISTE.