ISSN:0975-9646

Enchanced of Network Security Using Simplified Encryption Standard (SES):

K. Bommi Mayuri-M.Sc., M.Phil[1]
Assistant Professor,
Department of Computer Science,
Park's College
Tirupur – 641 605

E.Rajalakshmi M.Sc., M.Phil[2]
Assistant Professor,
Department of Computer Science,
Park's College
Tirupur – 641 605

Abstract- With the fast progression of data exchange in electronic way, information security is becoming more important in data storage and transmission. While storing and transmitting multimedia data are not easy and they need large storage devices and high bandwidth network systems. Compression and encryption technologies are important to the efficient solving of network bandwidth and security issues. This paper focus on dual approach of compression and security where compression is achieved through lossless algorithm (Huffman coding) according to size of compressed data is encrypted using traditional SES Algorithm.

Cryptography is a important part of preventing data and which is deal with secret transmission of message between two or more parties in world. Encryption techniques ensure security of data during transmission. However in most cases this increases the length of the data, thus increasing the cost. When it is desired to transmit, it is customary to first compress the data and then encrypt it. The Compression to removing the excessive information in file. The new compression and encryption algorithm is proposed to more protection. The techniques compress the data to reduce its Average length of the code in data. Afterward compressed data is encrypted and then further compressed using a new encryption algorithm without compromising the compression efficiency and the information security. This Huffman algorithm provides a higher compression ratio and enhanced data security. The SES provides more confidentiality and authentication between two communication systems.

Indexed Terms: **Compression, key-expansion, AddRoundKey, Encryption, Decompression, Decryption.**

1. INTRODUCTION

In this papers evaluate the compression and then encrypt the large amount of data such as multimedia based on text document, images, audio, video and database of file storage. This process reduces the data size, removing the excessive information and redundancy, although which is possible for given test set. The compression method broadly divided into two categories: lossless and lossy.

Lossless method: This can reconstruct the original data exactly from the compress data.

HUFFMAN CODING: Huffman Coding compression Technique is a primitive data resulting in volume of intellectual double-talk. Huffman patent has long since expired and no license required. There are how many variation of this method still being patented. The code can easily implement in very high speed compression system. The Huffman codes assume "Prior knowledge" of the relative character frequencies stored in a table.

A secret table only available to authorized user for encryption of compressed data. A more sophisticated and efficient lossless compression technique is "Huffman coding", where the most commonly character in the file have the shortest binary codes, and least common have the longest. The Huffman compression algorithm is an algorithm used to compress the files. It does this by assuming smaller codes are frequently used characters and longer codes longer codes for characters that less frequently used.

Simplified ES (SES): Simplified version of Advanced Encryption Standard is designed is required where the data security on open network. It has all functions of SES and made suitable for information secrecy maintains requirement. Encryption part of the SES has 5 sub functions: Key-Expansion, Adding Key Round, Substitution Function, Row transformation function, and Mix column function. As well as decryption process has inverse of the encryption.

Compression scheme invented by *HUFFMAN* in1952. This algorithm is named after its inventor, David Huffman, formerly professor at MIT. This code has become a favorite of mathematicians and academics,

Lossy method: This can only reconstruct an approximation of the original data.

So most commonly used technique as "lossless data compression" for data security and lossy through open network. Since, encryption and decryption section provide data security in conversation using cipher key. This key is considering as secret key between them as Alice and Bob. When the compression technique is done to give information, then did the encryption using simplified- ES(SES).

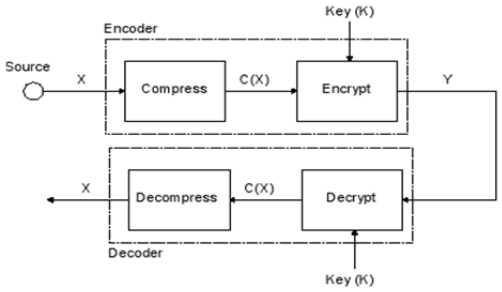


Fig 1 Data Encoder and Decoder

1.1 Existing system:

Currently compression and encryption methods are done separately. In some case compressed the encrypted data without knowledge of cipher key. Accordingly, it not secures during the transmission, and easily hacked the plain text after the decompression. **1.2 Proposed system:**

To lessen the problem, the approaches combine into one process. i.e. proposed new approach will perform both compression and then encryption at the same time. Here cipher key is selected from the frequency of symbols and code value. Hence processing time also less and more secure.

2. SYSTEM ANALYSIS AND DESIGN

2.1 Problem definition

Currently compression and encryption methods are done separately. The major problem is compressed the encrypted data without knowledge of cipher key. If large amount of data to compression and then encrypt separately required more time. Thus, not secures during the transmission, and easily hacked the plain text after the decompression. Therefore, our approaches combine the two processes into one. In these new approach both encryption of compressed data takes fewer processing time and more speed. When compression is done, using S-Box for state transmission in encryption time.

3. RELATED WORK

Last few decades have been seen lot of schemes using proposed for data encryption using keys afterwards it would be compressed, some of the prominent ones have been here. In this paper evaluate first data should be compressed. The first step in encryption part as using ARK_i is applying on the compressed text, state will be interchanged. At the time derived the ARK'_i to decryption process for each rounds. In addition clock-wise rotation moreover performed. The second step is compressed result had numeric and non-numeric values. So, related S-Box should be created for state transmission. The Existing SESS-Box had values of 0 to 9 and A to F only, where as these new table has 0 to 9 and A to Z values. So, more suitable to Encryption of compressed data using Simplified Advanced Encryption Standard algorithm.

4. PROPOSED ALGORITHM

4.1 CRYPTOGRAPHY

Cryptography comes from Greek words meaning "hidden writing". Cryptography is an important part of preventing private data from eavesdropper. However, which is a science of writing Secret code using mathematical techniques. The many schemes used for enciphering constitute the area of study known as cryptography. These technologies used to protect the network communication and data transmission over wireless network. Data Security is the main aspect of secure data transmission over unreliable network. The conventional methods of encryption/decryption can only maintain the data security. In process the art of protecting information by transforming it (*encrypting* it) into an unreadable format, called cipher text. Only those who possess a secret *key* can decipher (or *decrypt*) the message into plain text.

Types of Cryptography

There are two types of Cryptography: Symmetric key Cryptography and Asymmetric key Cryptography.

- Symmetric Key Cryptography

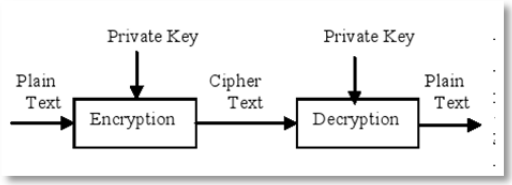


Fig 2. Symmetric Key Cryptography

It is also known as Secret key cryptography (private key). Symmetric-key algorithms are the algorithms under cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of cipher text. The keys represent a shared secret between two or more parties that can be used to maintain a private information link.

- Asymmetric Key Cryptography:

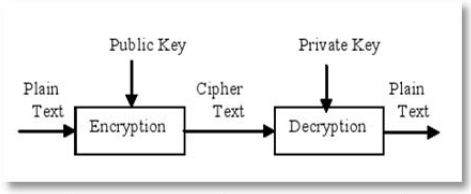


Fig 3. Asymmetric Key Cryptography

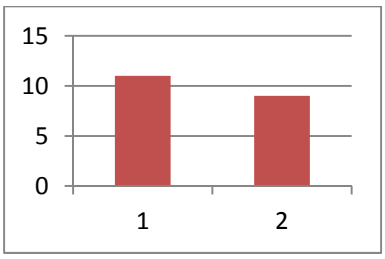
Asymmetric Key is known as Public key Cryptography. Public-key cryptography refers to a cryptographic system requiring two separate keys, one to encrypt the plaintext, and one to decrypt the cipher text. One of these keys is published or public and the other is kept private.

4.2 Huffman Compression

The compression of data may view as a branch of information theory in which primary objective is reducing size of data or file to be transmitted. More than compression techniques are available such as Arithmetic coding, Run-Length Encoding, LZW, Back Propagation, and so on. One of the compression methods is Huffman code, which is a simple but effective code. Data compression has important application in the area of data transmission and storage. Many data processing application is constantly increasing the use of computer extends to new disciplines. At the same time the proliferation of computer communication networks is resulting in massive transfer of data over the communication links. Compressing data to be stored or transmitted reduces storage and/or communication costs amount of compression that can be obtained with lossess compression. Lossless compression ratios are range 2:1 to 8:1. Compression ratio is calculated using below formula when it will be done.

$$\text{Compression Ratio} = \frac{\text{Size of Before compression}}{\text{Size of After compression}}$$

The chart describe before compression ratio as above formula. When file is compression using Huffman technique, which file is surely reduced compared to before size of file.



Compression Method	Huffman
Compression ratio	Very good
Compression speed	Fast
Decompression speed	Fast

Comparison of Huffman coding methodologies

Data compression often referred to as coding is a very general term encompassing special representation of data. Suppose we have a random variable X, which may take on the values x_1, x_2,…..x_i and that the corresponding probabilities of each

outcome are $P(x_j)$, $P(x_1)$, …. $P(x_i)$. In the sequence of K occurrences X, that outcome x_i will average be selected KP (x_i) times. Therefore average number of information obtained from K outcomes is [using P_i as an abbreviation for $P(X_i)$]. In data the symbols is dividing by K(size of data) to the probability values. Afterward applying the Huffman code word to each Pi.. The average length of the bits/symbols of function H(X) is often expressed as an enumeration of the probability of the possible outcomes as.

$$H(X)= \sum^{9}_{i=1} \mathbf{P_i L_i}$$

ALGORITHM FOR HUFFMAN ENCODING

STEP 1: Select the data and determine the bytes in original file size. [Ex: Transmission data as, "AKRWKRNETWORRKNESRWOKERWOSK" explained below].

STEP 2: Find each symbols in given data how many times frequently occurred for probability calculation. The probability defined as P(Xi).

STEP 3: Calculate the Probability = Symbol frequency/ File size.

STEP 4: The 1st-step in Huffman coding is values are arranged in descending probabilities. Then two probability symbols at the bottom of list are combined to form a compound symbol.

STEP 5: The step 4 repeated until reduced source with two symbols is reached.

STEP 6: The 2nd-step is code-word (0 and 1) is assigned to each symbol, which is used to average length computation.

For Example, the transmission of data as," AKRWKRNETWORRKNESRWOKERWOSK" between them before it must compressed using, Huffman compression Techniques as below…

Symbols				Probability
A	-	1/27	=	0.038
N	-	2/27	=	0.074
E	-	3/27	=	0.112
T	-	1/27	=	0.038
W	-	4/27	=	0.148
O	-	3/27	=	0.074
R	-	6/27	=	0.223
K	-	5/27	=	0.185
S	-	2/27	=	0.074

The above probabilities of symbols are *descending* order as,

P(R) = 0.223, P(K) = 0.185, P(W) = 0.148, P(E) = 0.112
P(N) = 0.074, P(O) = 0.074, P(S) = 0.074, P(A) = 0.038,
P(T) = 0.038.

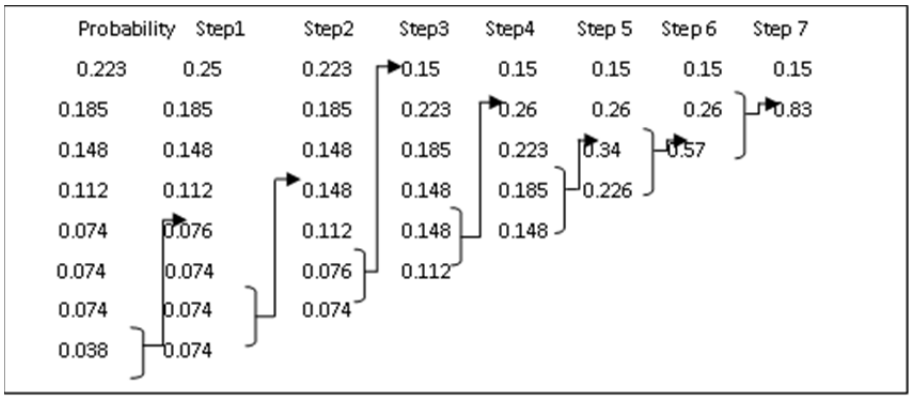


Fig 4 The Procedure To Combine The Symbols

Huffman code with Seven Symbols

Symbols	values	ASCII	Frequ -ency	Huff man	Code Word Li	Pi	$\sum^{9}_{i=1}$ PiLi
A	65	01000000	1	00	2	0.038	0.076
N	78	01001110	2	01	2	0.074	0.148
E	69	01000101	3	01	2	0.112	0.224
T	84	01010100	1	11	2	0.38	0.76
W	87	01010111	4	100	3	0.148	0.444
O	79	01001111	3	111	3	0.074	0.222
R	82	01010010	6	110	3	0.223	0.669
K	75	01001011	5	101	3	0.185	0.555
S	83	01010011	2	1110	4	0.074	0.296

Average Length (P_iL_i) = (0.079*1) + (0.148*2) + (0.224*3) + (0.76*1) + (0.444*4) + (0.222*3) + (0.669*6) + (0.555*5) + (0.296*2)

$$\sum^{9}_{i=1} \mathbf{P_i L_i} = \mathbf{3.994}$$

= 3.394 bits/symbols

However, the source considers 1000 symbols in data; the average length of an encoded message is 3,994 bits. Since, compressed data consists 3,994, which is $\leq 2^{16}$-block. Size

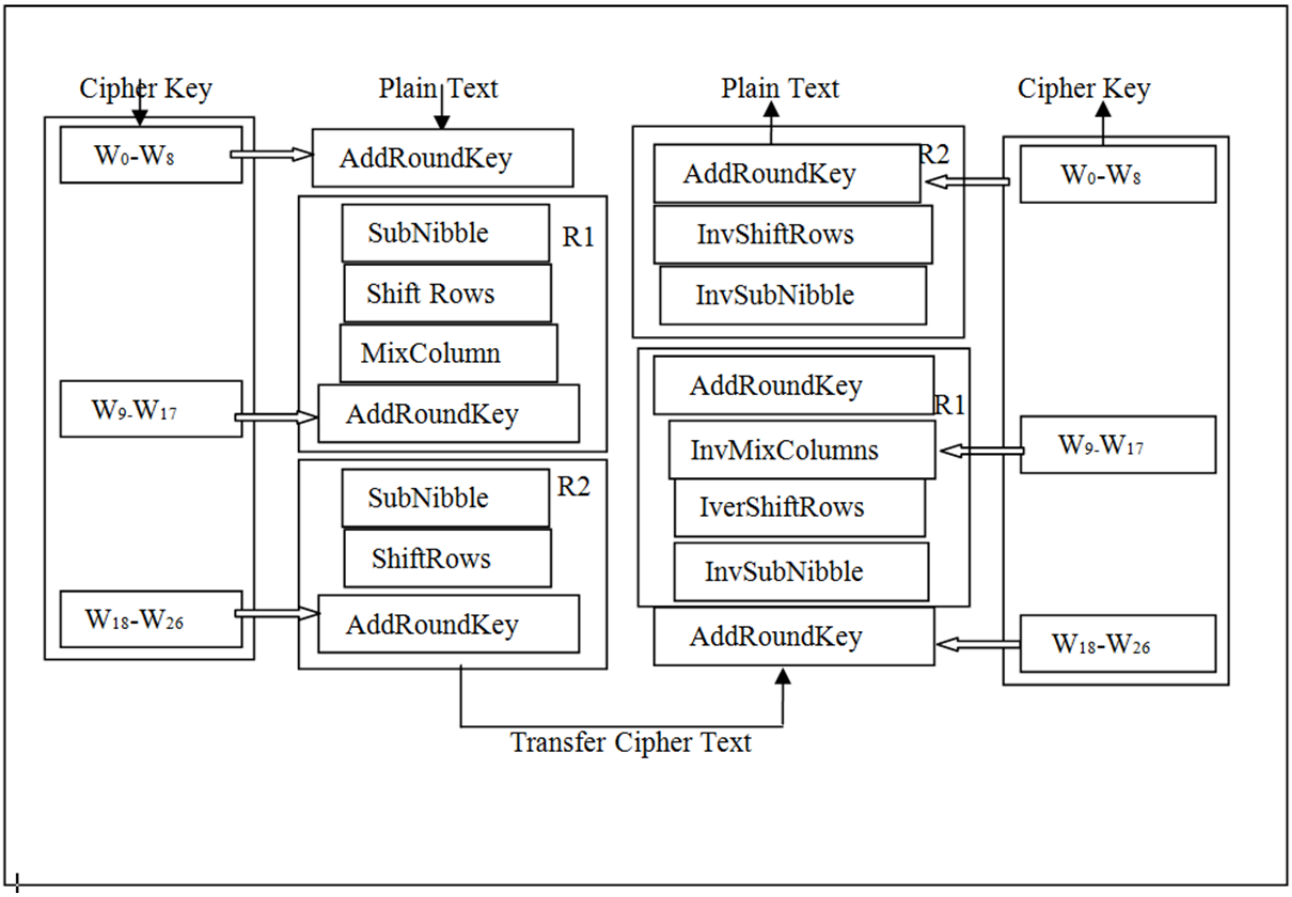


Fig 5 Encryption and Decryption of SES

4.3 SES Algorithm specification

Simplified Encryption Standard (SES) developed by professor Edward Schaefer of Santa Clara University. SES is based on the Simplified Encryption Standard (SES) algorithm, which is symmetric –key block cipher published by the **National Institute of Standard and Technology (NIST)** in December 2001. In this algorithm required a block size is 128 bits and three different key sizes of 128, 192, and 256 bits, so it defines AES-128, AES-192, and AES-256. It uses 10, 12 or 14 rounds. Whereas SES key size 128 bits, and performed 3 rounds.

Key length here is not fixed vary between 1 byte to 16 bytes. Initially key is expanded to 16 bytes expanded key. This same key is applied in each round during encryption and also during decryption. Key is basically XORed with data blocks during encryption as well as decryption.

Nibble: A nibble is group of n –bits that can be treated as single entity, a row and column wise n*n matrix. When it can treat as row matrix, the bits are inserted into left to right. We use uppercase letter N to refer the nibble.

State: In SES data a block is also refer red to as a state, define the uppercase letter S. In this case each element of a state described as $S_{r,c}$ nibbles are put together in the matrix form known as state.

Words: The words are derived from the nibbles for encryption of the compressed data. Words W_0, W_1 are initially made from nibbles. There are 26 words are created in key-expansion phase. We use uppercase letter W to refer the words and length is W_{i-1}.

4.4 Key expansion

The key expansion routines create three $2^{'6}$-bits round keys from one single $2^{'6}$-bits cipher key. The first key round key is used for pre-round transformation (AddRoundKey); the remaining rounds as middle round, bottom-up round. The key expansion routine creates round keys word by word, where a word is an array of N nibbles. The routine create wv (values of words) as 26 words, which are called w_0, w_1, w_2 …w_{26}.

KEY EXPANSION ALGORITHM

Key Expansion (K)

External words: K, t_{wr}, R_w, S_w, W_i, Nn, WPR, WMR , WBR, ARK_{ij}

t_{wr} ← (R_w - S_w)

for t_{wr} = tw_1, tw_2

do

// Pre-Round

WPR ← (N_0 ⊕Nn)

ARK_0 ← (W_0 to W_8)

// Middle-Round

W_9 ← (W_0 ⊕w_1)

WMR ← (WMR ⊕WPR)

ARK_1 ← (W_9 to W_{17})

// Bottom-up -Round

W_{18} ← (W9 ⊖w_2)

WBR ← (WBR ⊖WPR)

ARK_2 ← (W_{18} to W_{26})

ARKj (ARK_0, ARK_1, ARK_2)

ARKj (ARK_0, ARK_1, ARK_2) The process is as follows:(here the symbols ⊕ is addition, and ⊖ is subtraction)

The first pre-round words (W_0, W_8) are made from the cipher key. The cipher key is arrays of 19 nibbles (N_0 to N_1) become W_0; the next two nibbles (N_2 to N_3) become W_1 still W_{17} creation.

The rest of the words (twr = tw_1, tw_2) are made as follows:

Wi = twr Wi-v. Here, t_i a temporary word, is result of applying to routine, SubWord(Sw) and RotWord(Rw) on Wi-v and Adding the results in the Middle Round although Subtracting the results in Bottom-up round, RC[Nr], where Nr is the round number in word creation

4.5 Encryption and Decryption Phases

The Encryption is applying to the original text-data intended for hiding the secret information from the eavesdropper. There are two types encryption method as Symmetric key,

Asymmetric key. Symmetric key means same key is used to encrypt and decrypt the message. This differs from asymmetric (or public-key) encryption, which uses one key to encrypt a message and another to decrypt the message. Here, use SESis a symmetric block cipher method used for encryption of compressed data using private key.

Rests of Words ARK_0 = 030407110404050607

$t_1 \leftarrow 75\text{-}57 = 18$ ARK_1 = 212532434751566269

$t_2 \leftarrow 21\text{-}12 = 9$ ARK_2 = 121319242328283435

The AddRoundKey Created using cipher-key and above words. There are 3 AddRoundKeys are K_0, K_1, K_2. The algorithm requires initial set of W_i words and each of the R_j Rounds requires AddRoundKeys.

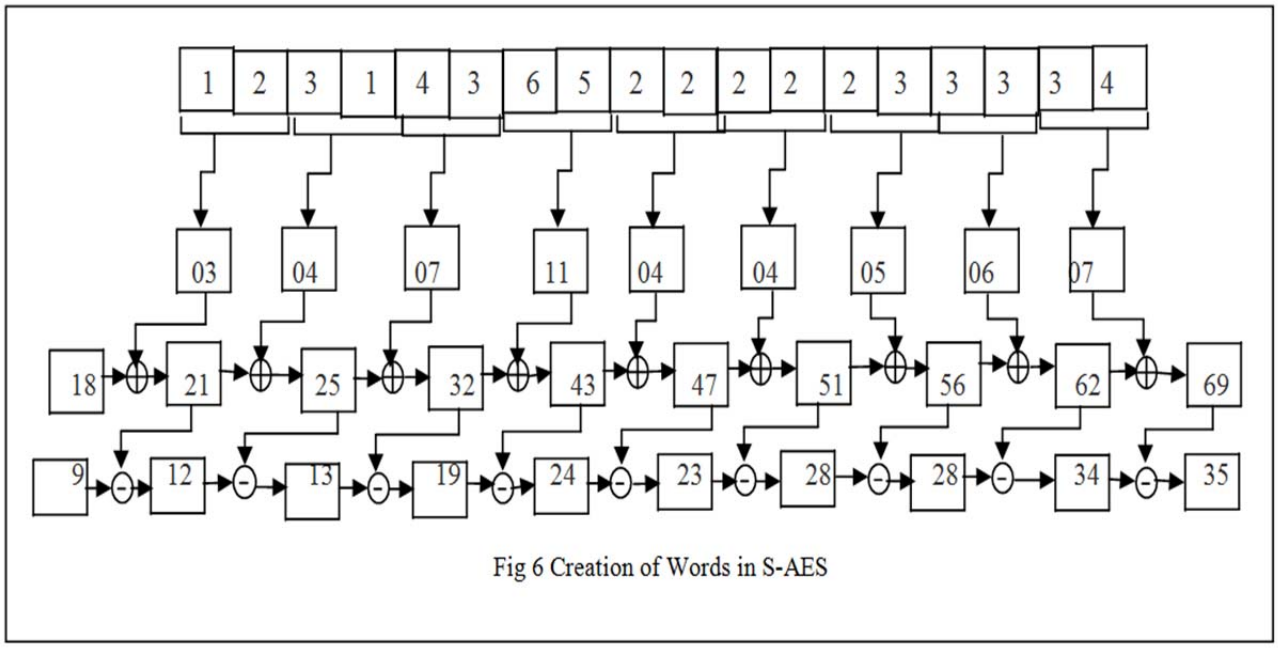


Fig 6 Creation of Words in S-AES

Add Round Key (K) as,

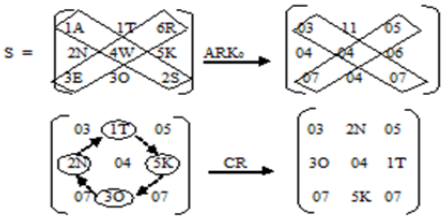
Rounds (R_i)	AddRoundKeys (ARK_j)
R_1	$K_0 \leftarrow (W_0, W_8)$
R_2	$K_1 \leftarrow (W_9, W_{17})$ such as $(W_9 (t_1 \oplus W_0), WMR (WMR \oplus WPR))$
R_3	$K_2 \leftarrow (W_{18}, W_{26})$ such as $(W_{18} (t_2 \ominus W_9), WBR (WBR \ominus WPR))$

Fig 7 Process of S-AES

STEP 1: Select the compressed data for encryption, "1A2N3E1T4W3O6R5K2S" which is briefly described below.

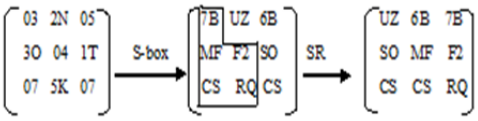
ENCRYPTION

STEP 2: In S-AES, 1st process is hiding the data using AddRoundKey value. Now the marked states values should be changed. Later than remaining state position rotated clock-wise. *(ARK0 = 03040711040405067).*

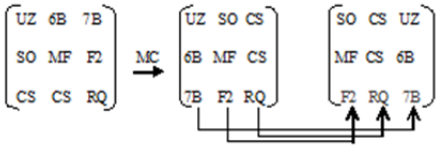


[Note: After that, sender send key to decryption process as following rules ARK(0,0)→CT(0,0), K(0,2)→CT(0,2), ARK(1,1)→CT(1,1), ARK(2,0)→CT(2,0), ARK(22) →CT(2,2)]

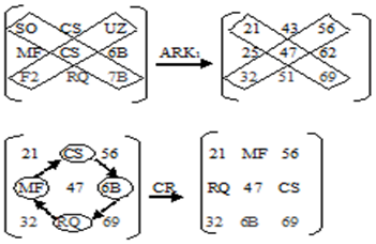
STEP 3: Sub nibbles are transformed using "S-box" in R-1 which is to substitute the each state. After sub-byte transformation followed by rows are shifted.(SR - Shift-Row).



STEP 4: Then perform the mixing - column operation. That means, columns in matrix form are distorted the positions. (MC - Mixing-Column).



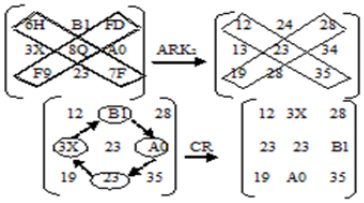
STEP 5: Subsequently established the R-2. Again execute R-1 procedure without mixing-column function. *(ARK1 = 212532434751566269)*



[Note: After that, sender send key to decryption process as following rules ARK(0,0)→CT(0,0),K(0,2)→CT(0,2), ARK(1,1)→CT(1,1),ARK(2,0)→CT(2,0), ARK(22) →CT(2,2)]

STEP 6: Same way, the clock-wise rotation followed by using S-box to substitute the each state. After sub-byte transformation followed by rows are shifted.(SR - Shift-Row)

STEP 7: When rows are shifted, using ending keys of the ARK2 to execute the absolute encryption data.



(Where, CT – compressed text, ARK – AddRoundKey, ST-State Transmission, CR – Clock Wise Rotation, Sb - S-box, SR-Shift row, MC – mixing column).

Now encryption of compressed data is "1223193X23A028B135" is a cipher text. It should be defined as below. in that case will be transfer through open network to destination.

En = CT(ARK(ST,CR)(Sb(SR(MC))))

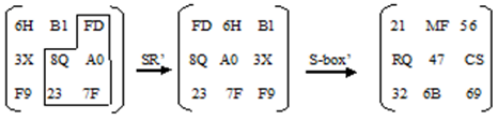
DECRYPTION

Decryption process is reverse of the encryption process. Though, user need know the secret key as "ARKeys" of 3 values. However, these keys are changed; It means some of the state positions are modified. Likewise, perform the each operations are correctly. Afterward get the encryption of compressed plaintext.

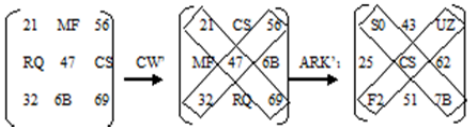
STEP 8: In these procedure 1st construct the R-2 using ARK'2. But keys values are tainted, such as *(ARK'2 = 6H13F9248Q28WY347F)*

Round 2:

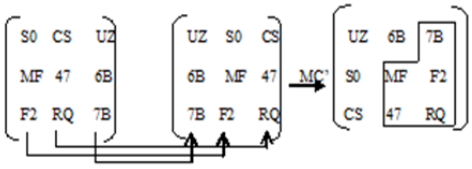
STEP 9: Then, using shift rows as opposed direction of encryption progression. Followed by inverse sub nibble with inverse S-box.



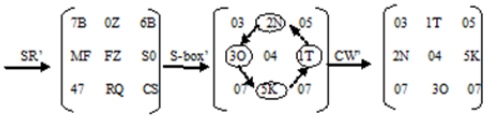
Round 1: (ARK'1 = S025F243CS51UZ6269)



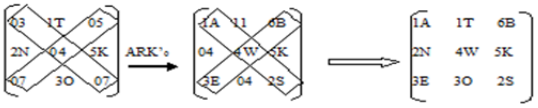
STEP 10: In R-1 later than ARK'1 applied, do mixing-column, same way of inverse step – 4.



STEP 11: In that case, later than ARK'1 applied, do shift rows, alike inverse step – 3, then utilize the S-box' table for substitution.



STEP 12: At last using *ARK'0 = 1A043E114W046B062S* to cipher text like inverse step - 2.



STEP 13: At the moment, carryout the compression techniques of "HUFFMAN ENCODING", such as Decompression, can obtain the original text.

5. Conclusion And Future Scope

Encryption techniques are often used to protect the multimedia content from the unauthorized users. In this paper, simplified encryption techniques are applied on when data compression is fulfilled to the reduced file size. The developments of combine the both processes is very speed and more secure through the open network. AddRoundKeys are derived from the cipher-key using temporary words. These approaches of key scheduling are more protected to the data on transmission time. Even SES key length is maximum 128 bits, and no.of rounds are 3, Key expansion is simplification from the SES algorithm.

Here experimentally compression along with encryption techniques manual calculations is presented. We conclude that the time, cost and bandwidth consumption for selective SESencryption on compressed Data is less than DES encryption techniques. So, the selective encryption technique is better than DES encryption techniques as it takes less time with that is inaudible to the unauthorized users. In future, the security of the method can be use to multimedia data like images or audio or video to the system.

Refrrences

1. Behrouz A. Forouzan, Debdeep Mukhopadhayay, "Cryptography and Network Security", 2nd *Ed*t
2. William stalling, "High Speed Network And Works And Internets"
3. S.Annadurai, R. Shanmugalakshmi, "Fundamentals Digital Image Processing".
4. Dr. Kanak Saxena and Mohini Chaudhiri, "Fast and Data Transmission using Symmetric Encryption and Lossess Compression ", *IJSCMC, Vol. 2, Issues. 2, Feb 2013.*
5. M.Pitchaiah, Philemon Daniel Praveen, "Implementation of Advanced Encryption Standard Algorithm", International Journal Of scientific & Engineering Vol 3, Issues 3, Mar 2012.
6. T. SubbamastanRao, M. Soujanya, T.Hemalatha, T.Revathi, "Simultanous Data and Compression Encryption", T. Subhamastan Rao at al. / (IJCSIT) *Vol* 2. 2011.
7. Nigam Sangwan, "Text Encryption and Huffman Compression", International Journal Of Computer Application. *Vol* 54-No.6 *Sep* 2012.
8. Paul A.J , Mythili .P, Paulose Jacob K., "Matrix Based Key generation to enhance key avalanche in advanced encryption standard", *ICVCI*- 2011, proceeding published by *(IJCA).*
9. Suhas J., Mahendra Manangi, Parul Chaurasia Pratap Singh "Simplified SESfor Low Memory Embedded Processor", Global Journal of computer application. *Vol.*10 *Issues* 14(Ver,10) *Nov* 2010.