Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering (Security & Cloud)

Elixir Comp. Engg. 112 (2017) 48975-48979

ISSN: 2229-712X

# Secure Data Sharing Model for Cloud Systems

Rushdi A. Hamamreh
Computer Engineering Department, Faculty of Engineering, Al-Quds University.

**ARTICLE INFO**

**Article history:**
Received: 5 September 2017;
Received in revised form:
4 November 2017;
Accepted: 14 November 2017;

**Keywords**
Cryptosystem,
Cloud Computing,
Distributed System,
Hill cipher,
Matrices,
SSCC.

**ABSTRACT**
Secure Data Sharing Model for Cloud Computing (SSCC) is a new cryptosystem model that aims to store user's data files in a secure manner to keep it away from the unauthorized access. We have developed a new cryptosystem model that based on the Original Hill cipher algorithm using matrices manipulation and symmetric key cryptography. It gave a good impression by comparing it with previous ways where the comparing results between SSCC and Hill Cipher 6.86% improvement in the encryption time. While when we compare SSCC and Advance Encryption Standard (AES) the difference was 3.94% for AES which means that the proposed model has a chance to compete with AES model. Another comparison is done to cover the two cases: SSCC1 and SSCC2, the SSCC1 with and without compression decreasing the encryption time in a percentage of 96.287%.



## I. Introduction

Cloud Computing technologies are emerging as a common way of provisioning infrastructure services, applications and general computing and storage resources on-demand. This cloud model is composed of eight essential characteristics, four service models [2], and four deployment models [3][16]. There are many benefits for cloud environment that include: reduces costs, scalability,and flexibility and provides large-scale infrastructure resources for high-performance computing [4].

However, research results show that security concerns especially data security and privacy protection issues, will remain the major reason of objection for decision makers to adopt cloud computing. As shown in figure 1, a survey report that collected information from 263 IT professionals by asking different questions related to the cloud represent security as a first rank according to IT executive.

Application software databases are moved now towards cloud computing storage where users may not feel that it is trustworthy enough according to these factors: Trust management,Security provider, Privacy protection, Ownership Data location and Relocation,Data integrity,Data recovery, Performance and availability, Data Backup,Data portability and conversion [6]. Securing the data stored in the cloud is very important to companies and corporations before moving their crucial data from their in-premises hosting to the cloud.

Our research is about a new technique for encryption data and storing them into a cloud storage .The contribution is about making a more secure data storage into the cloud represented by the following points:Encoding, Padding, Compression and Encryption.

The paper is organized as follows. Section II gives summary of the related works. Section III illustrates more detailsaboutCryptosystem models. Section IV describes the proposed model and how it executed.

Section V is about simulation and testing results analysis. Finally, In Section VI, we give the conclusion and future work for this paper.

## II. Related work

Many researchers proposed a lot of solutions to the security issue in cloud computing environment, either using the Original Hill Cipher principle or within the modified versions for that model which based mainly on the matrix manipulation principle, see figure 1.

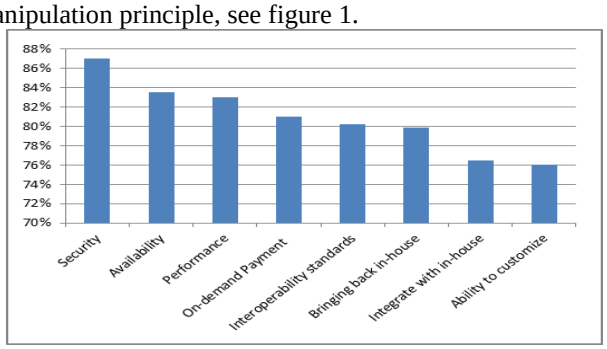



**Figure 1. Cloud computing issues [5].**

Modified Hill Cipher gave a solution for the disadvantages points on the original Hill cipher by supporting iterations and interlacing. Despite that iteration will increase the complexity when work with images as an input data and interlacing leads to confusion on the output of the cipher text, it provides a more secure results for the plaintext rather than images using binary conversion principle which is suitable for integers, so the key matrix can be generated to be always in a non-singular form.

Also, Biometric traits can be solved in a secure manner using a cryptosystem model [7] based on modified Hill Cipher within a key matrix and gain more efficient results.

Information hiding can increase the security degree of the cryptosystem model, so that a combination between this

Tele:
E-mail address: rhamamreh@eng.alquds.edu

model and a steganography method will secure the transmit plaintext message using the LSB insertion technique. In [8] Two random techniques are used subsequently, one for XOR the Cipher text and the other for the LSB insertion process into any image. This will provide a robust cryptosystem model using both encryption and steganography processing which work against cryptanalytic attacks.

Another technique support the one way hash algorithm can be used to clarify the principle of a singular matrix in GF(2) using the Hill cipher multiplication method .This matrix is filled by only one entry for each row and 0 otherwise , so that some of matrices can be represented only using m entries . In this model [12], Hill cipher is used symmetric encryption and multiplied by a key matrix and authors propose a class of matrices in GF (2) based on singular matrices [9] where it is easy to prepare and used.

A new technique based on the original Hill cipher was proposed to solve the flexibility of the key matrix and enhance the security of this model against the brute force attach by using public ideas that gain more powerful complexity than linear algebra steps which can be done using a specific options, Authors proposed a robust cryptosystem algorithm for singular matrices based on the original Hill cipher [10].

Finally, using matrices as a formula for the input text and the key used by the encryption process will improve the results gain from this process, many algorithms use that principle like Row Column Diagonal (RCD) [11] and output a more powerful results than the traditional ways.

## III. Cryptosystem models

Cryptosystem model is referred to a model that applied the cryptography methodology and key exchange techniques at the same time to generate a more secure model .In the field of Cryptography there are two several techniques for encryption/decryption processes , i.e. Symmetric and Asymmetric key Cryptography.

### *A. Symmetric or single key cryptography:*

In this model the same key called a secret key is used for encryption and decryption, figure 2(a) demonstrates the simplified model for symmetric key encryption technique, the plain text is encrypted using that key and the output differ when we use a different key also the decryption process done using the same key.

One of the symmetric key cryptography that use the matrix manipulation is the Hill cipher [12] that depend on key matrix which must be invertible to be used in the both sides of encryption/decryption process. In the encryption side; a Plaintext with a fixed-length Block size m and number of elements q, the values of $K_{nxn}$ matrix entries are vary between (0, q – 1) included, and K must be an invertible matrix ,Each block of the Plain text matrix also contains entries between (0, q-1) included and represent a vector of n dimension .This process is done using equation 1:

$$c = m\ k\ mod\ N \quad (1)$$

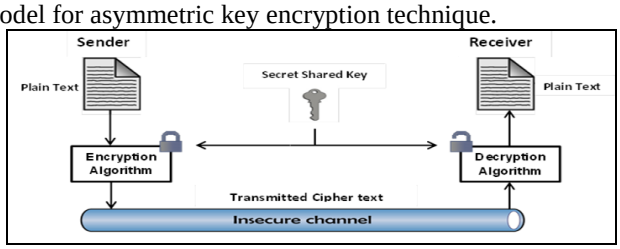
In the decryption side; to introduce a cipher text of vector c, we need first to find the inverse matrix $k^{-1}$ to $k$, where that matrix must be nonsingular. Then we can decrypt the incoming cipher text message using equation 2:

$$m = c\ k^{-1}\ \ N \quad (2)$$

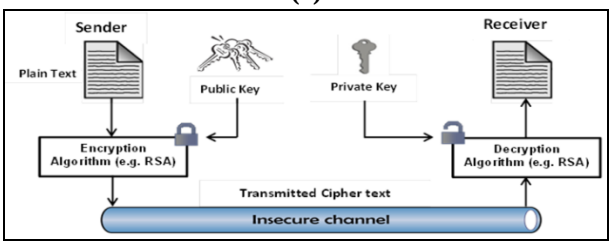
Moreover, not all matrices are invertible so a singular matrix must be converted to a nonsingular one to be chosen as a key matrix [13]. Hill cipher defines its simplicity by using letters frequencies to represent the plaintext message and the results of multiplication are given by a high speed and also high throughput. As the matrix size increase, more letters and its frequencies are hidden. Despite Hill-cipher is robust against cipher text it is easy to be broken using plaintext attack.

### *B. Asymmetric or public key cryptography*

In this model user must have two types of keys to complete the encryption/decryption process; a public key for encryption and a private key for decryption between two parties like Alice and Pop. Alice encrypt her plaintext using the public key of Pop before sending it to him then decrypt it using its own private key , so that no one can decrypt the cipher text without the private key of Alice to keep the his privacy . There are different methods to implement this type of cryptography model .These are RSA,ECC, Diffie-Hellman and Digital Signature, figure 2(b) demonstrate the simplified model for asymmetric key encryption technique.




**(a)**




**(b)**

**Figure 2. Cryptosystem models, (a) Symmetric (Single key) cryptography, (b) Asymmetric (public key) cryptography.**

This paper proposes a symmetric encryption algorithm to protect the user's data stored in the cloud storage from the unauthorized access depending on the original Hill cipher principle.

## IV. Proposed Model

In this model, data needs to be encrypted before uploaded to the cloud storage, so that cloud providers can't extract the original plaintext of the file to find its content, also user can download his data files from that storage then decrypt it using his own secret key.

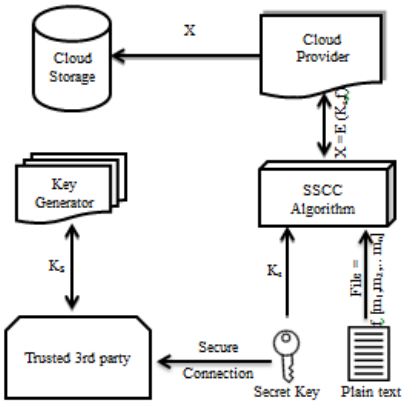



**Figure 3. Abstract view of SSCC.**

SSCC uses matrices manipulation principle to produce cipher text .This model uses two matrices: plaintext (P) and key (K) as a type of triangular matrices to gain benefit from their properties. A secure communication channels and a group manager Certificate Authorities provide private keys for dynamic groups in the cloud, when a new member wants to join or revoked from this group, the private keys of the other members will not be changed .And the revoked member can't access to any data files if it come from untrusted third party [14]. Figure 3 provides an abstract view of our model.

Our model has two techniques to be accomplish that varies by the plaintext matrix form, the first technique is called SSCC1 has an upper triangular matrix with zero elements above the diagonal, the second technique is called SSCC2 has non-zero elements for all entries in this matrix.

The phases of the SSCC algorithm are processed as the following:

1. Input**:** User input a plaintext (M).
2. Encode: the plaintext processed and encoded I, then E is filled into matrix with $E_{mxn}$ matrix[15].
3. Split: plaintext is split into n numbers of sub blocks ($p_1$,$p_2$, ….. $p_n$) , depending on a block Size value (B) .
4. Compress: Each sub-block is converted into its ternary value then a composition of this stream of values is converted to one decimal value that represents compressed block. These set of blocks are inserted into an upper triangular matrix, figure 4 illustrates the compression process completely :

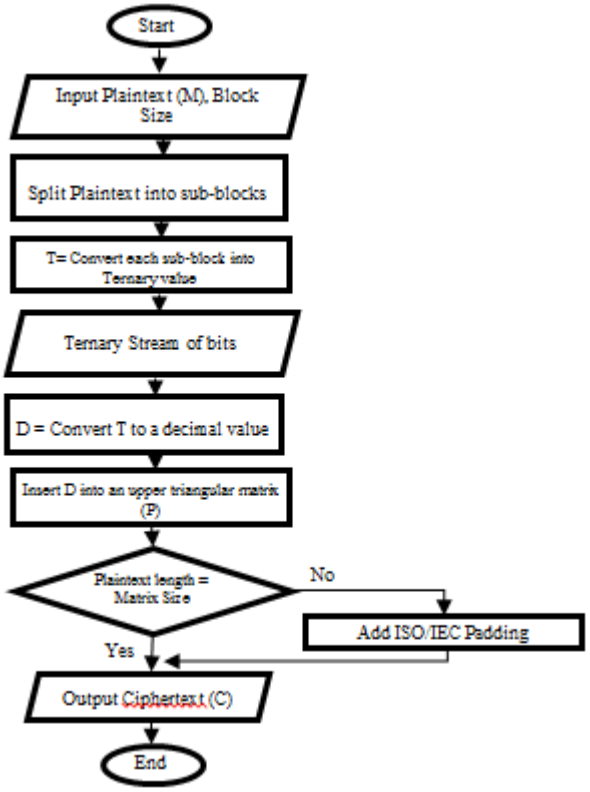



**Figure 4. Flow chart of compression process.**

1. Pad: The zeros elements in this matrix are replaced with padding values using ISO/IEC padding technique to complete the shape of this matrix , i.e. all elements above the diagonal are nonzero elements.
2. Generate and Result: The key used for the encryption is generated using several algorithms, a specific key generator that used Diffie-hellman key exchange principle then stored into a trusted third party where the value is unique for each user, and then the key matrix is prepared to be used with the plaintext matrix (K).
3. Decode: The process of encryption is done through the multiplication between two matrices :Plaintext(P) and Key (K) to generate a third matrix called Cipher text I ,this phase is done by reducing the complexity of matrices multiplication by multiply only the non-zeros entries, this will reduce the time needed and do it correctly.
4. Output: cipher text I is uploaded to the cloud storage.

### *A. Mathmatical model*

In this section we take care about the model that used for two different phases; UPLOAD and DOWNLOAD.

**UPLOAD to the Cloud storage (Encryption)**

User needs to follow the previous steps in the forward direction to complete the process of uploading his data to the cloud storage, see equation 3 and the compression process is done using the equation 4 for more details see Algorithm 1.

Let Key matrix (K), $K=\begin{bmatrix} k_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ k_{1n} & \cdots & k_{nn} \end{bmatrix}$ ; and

Plain text matrix (P), $P=\begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & p_{nn} \end{bmatrix}$ ;

Then;

$$\begin{bmatrix} k_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ k_{1n} & \cdots & k_{nn} \end{bmatrix}\begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & p_{nn} \end{bmatrix}=\begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix} \quad (3)$$

Where,

K: nxn lower triangular matrix that contains the secret key for one user. $K_s = [k_{s1}\ \ k_{s2}\ \ \ldots\ldots\ K_{sm}]$; where $k_{si}$ is an integer value represent a subpart of that key.

P: nxn upper triangular matrix that contains encoded and compressed data, so that each element represents a block of data, using the following formula:

$$p_{ij} = compression((encode(pcb)) \quad (4)$$

Where, pcb is a padded compression block.

$1 \le I \le n$ and $1 \le j \le n$

C: nxn Cipher matrix.

```
M = readFile("Plaintext.txt")
E=encode(M) % a=0 , b=1 , …… , z=26 and fill them in Emxn matrix
T= dec2terI % convert decimal values to values of base 3values
B= split(T, BlockSize) % split into equal Sized sub-blocks
compBlocks = ter2dec(B , n )
K= upperTriang(key) %insert key vector into upper triangular matrix
pcb = padCompBlocks(compBlocks) % add padding if needed
P = lowerTriang(pcb ) % insert pcb vector into lower triangular matrix
R=K  P
C = decode ( R )
uploadToCloud(C,"ciphertext.txt" )
```

Algorithm1: Pseudo code for encryption

**Download from the Cloud storage (Decryption)**

If $C = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & c_{nn} \end{bmatrix}$ ; Cipher text matrix

User has the secret key and the incoming cipher matrix; so the plaintext matrix (P) can be easily found by solving a system of linear equations as mentioned in equation 5, for more details see Algorithm 2.

$$C = KP \quad (5)$$

Where:

$$p_{ij} = removePadding(decom(decode(c_{ij}))$$

$$1 \le I \le n \quad \text{and} \quad 1 \le j \le n$$

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & p_{nn} \end{bmatrix}$$ ; Plain text matrix

```
C= downloadFromCloud("ciphertext.txt")
E=endcode( C )
padCompBlocks=lineSolve(E,K) %matrices L U
decomposition
pcb = extractVector(padCompBlocks)
comBlocks=removePadding(pcb)
T=dec2ter(comBlocks)
B=split(T,blockSize)
E=ter2Dec(B)
P=decodeI
SaveTo(P,"paintext.txt")
```

Algorithm 2: Pseudo code for decryption

## V. Simulation and Results

A different file sizes are fed to our algorithm that generate a cipher text for the original plaintext and also calculate the encryption time for each file size. First, we start with small file sizes to calculate the capacity for each matrix varies from 1x1 to 10x10 and calculate the results .These matrices can be either an upper triangular matrix used in the 1st technique (SSCC1) or a fulfilled matrix used in the 2nd technique (SSCC2) within a compression process represented by a compression value (CV) in both techniques, these results are according to Table 1.

**Table 1. Encryption time for SSCC using different matrix size, Compression Value = 9.**

| Matrix size nColumns x nRows) | File Size(bytes) | | Encryption time (sec) |
|---|---|---|---|
| | 1st technique (SSCC1) | 2nd technique (SSCC2) | |
| 1x1 | 9 | 9 | 0.001 |
| 2x2 | 27 | 36 | 0.004 |
| 3x3 | 54 | 81 | 0.007 |
| 4x4 | 90 | 144 | 0.012 |
| 5x5 | 135 | 225 | 0.018 |
| 6x6 | 189 | 324 | 0.024 |
| 7x7 | 252 | 441 | 0.033 |
| 8x8 | 324 | 576 | 0.041 |
| 9x9 | 405 | 729 | 0.049 |
| 10x10 | 495 | 900 | 0.063 |

Compression value (CV) can be increased to gain better results where the time needed for encryption process decrease because each block from the plain text matrix contains more data so matrix with small n value can contain larger file size. In Table2, a comparison is done based on the same matrix size with different compression value to find the file size in bytes that can contained within each matrix, we notice that using CV=135 byte will leads to more bytes within 9x9 matrix that represent the optimal case when we also care of on the encryption time.

In this simulation a different block sizes are used in the process for splitting the input text into sub blocks then composite them into several compressed blocks. We test our algorithm using two techniques and compare between them and others see figure 4; these techniques are: full or triangular plain
text matrix .We started from 32 Kb and end with 160 kb with fixed matrix size =9x9, we choose this size of matrix to be a reference because we gain the optimal results using it when compared with other sizes that varies from 1x1 to 10x10.

**Table 2. Encryption time for SSCC1 and SSCC2 using different matrix size, and different compression value.**

| Matrix size (nColumns x nRows) | File Size(Bytes) | | | | | |
|---|---|---|---|---|---|---|
| | CV = 18 Byte | | CV = 27 Byte | | CV = 135 Byte | |
| | SSCC 1 | SSCC 2 | SSCC 1 | SSCC 2 | SSCC 1 | SSCC 2 |
| 1x1 | 18 | 18 | 27 | 27 | 135 | 135 |
| 2x2 | 54 | 72 | 81 | 108 | 405 | 540 |
| 3x3 | 108 | 162 | 162 | 243 | 810 | 1215 |
| 4x4 | 180 | 288 | 270 | 432 | 1350 | 2160 |
| 5x5 | 270 | 450 | 405 | 675 | 2025 | 3375 |
| 6x6 | 378 | 648 | 567 | 972 | 2835 | 4860 |
| 7x7 | 504 | 882 | 756 | 1323 | 3780 | 6615 |
| 8x8 | 648 | 1152 | 972 | 1728 | 4860 | 8640 |
| 9x9 | 810 | 1458 | 1215 | 2187 | 6075 | 10935 |
| 10x10 | 990 | 1800 | 1485 | 2700 | 7425 | 13500 |

**Table 3. Average time for different algorithms, matrix size 9x9.**

| Algorithm | Mean of the algorithm time (ms) | Time for 1 KB in (µs) |
|---|---|---|
| SSCC1 | 1519 | 8.065 |
| SSCC2 | 789 | 4.48 |
| MRHC [10] | 846 | 4.81 |
| AES [10] | 758 | 4.31 |

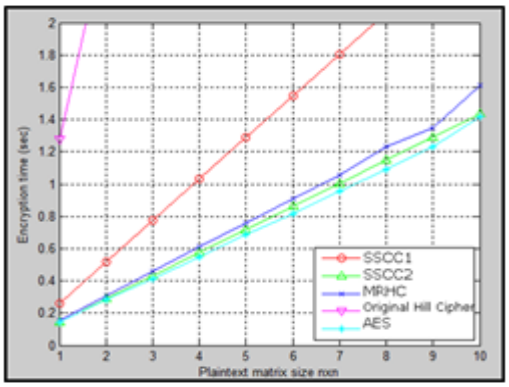



**Figure 5. Encryption time for different algorithms, matrix size 9x9, compression value 135 byte.**

Also, comparisons are made on the encryption time between files with and without compression process for 9x9 matrixes, see Figure 6.

Using our techniques the needed time for completely encrypt the plaintext is decreased by 96.287% such that when we apply the first technique (SSCC1) either before or after compression process using the 9x9 matrix and file size 320 kb using equation 6:

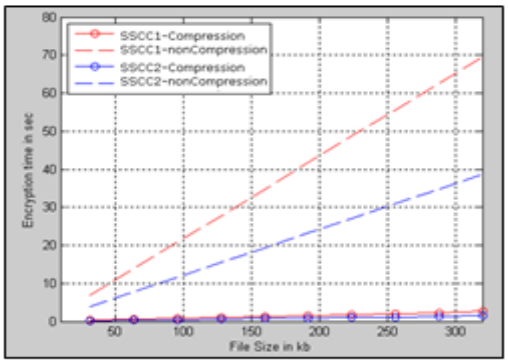



**Figure 6. Comparison between compression and non-Compression techniques.**

$$ET\% = \frac{CET - NCET}{CET} \times 100\% \qquad (6)$$

*Where,*

ET: The percentage of decrease in the Encryption Time.

CET: Compression Encryption Time.

NCET: Non-Compression Encryption Time.

Moreover, when we follow the second technique (SSCC2) either before or after compression the percentage of decreasing time is 67.146%

We implement our Cryptosystem model in an IntelASUSP6100 laptop, 2.00GHz 2 core(s), 4GB RAM/ Microsoft windows7/MATLAB 7.10.0 simulator.

## VI. Conclusion and Future Work

This study offers a Secure Data Sharing Model for Cloud Computing (SSCC). It can be applied on Distributed Systems and aims to protect data from unauthorized access. We developed two different techniques to satisfy large file size using the same matrix by increasing number of blocks exists on it or increasing the compression value so one block can contain more bytes.

SSCC also used to test the process with and without compression steps to gain ensure the big role of the compression phase in the proposed model .SSCC aims to contribute the existing Hill Cipher algorithm by reducing time needed for the process of encryption and also access a robust and secure mode by providing a set of modification like: Encoding, compression, plain text padding and encryption, so attacker has no direct choices to check because the algorithm is complex and composite of a set of complicated steps.In the Future we will added features for data that is either be text or numbers and a key exchange scheme to cover both the generation and distribution of the secret key and make it hard to guess and stronger than the Brute force attack.

## References

[1]M. Nazir, P. Tiwari, S. Tiwari,"Cloud Computing: An Overview", available at: http://ebooks.hctl.org/cloud-computing/chapter-1.pdf, 2015, accessed on 14/01/2017.

[2]P. Mell, T. Grance. "The NIST Definition of Cloud", National Institute of Standards and Technology, 2011.

[3]Y. Sun , J. Zhang,, Y. Xiong,, G. Zhu,"Data Security and Privacy in Cloud Computing",2014.

[4]K. Hashizume, D. Rosado, E. Fernández-Medina, E. Fernandez. "An analysis of security issues for cloud computing", Journal of Internet Services and Applications, 2013.

[5]I. Khalil, A. Khreishah, M. Azeem. "Cloud Computing Security: A Survey", available at: http://www.mdpi.com/,2014, accessed on 14/01/2017.

[6]D. Dinadayalan, S. Jegadeeswari, D. Gnanambigai. "Data Security Issues in Cloud Environment and Solutions", World Congress on Computing and Communication Technologies, 2014.

[7]B. Acharya, M. Sharma, , S. Tiwari,V. Minz. "Privacy protection of biometric traits using modified hill cipher with involutory key and robust cryptosystem", Procedia Computer Science,2010.

[8]B. Acharya, H. Agrawal, A. Modi, ,U. K Agrawal."Combined Implementation of Robust Cryptosystem for Non-invertible Matrices based on Hill Cipher and Steganography", Proc. Of Int. Conf. on Advances in Computer Science, 2010.

[9]A. Berisha, B. Baxhaku, A. Alidema."A Class of Non Invertible Matrices in GF (2) for Practical One Way Hash Algorithm", International Journal of Computer Applications, 2012.

[10]R. Hamamreh, M. Farajallah, "Design of a Robust Cryptosystem Algorithm for Non-Invertible Matrices Based on Hill Cipher", International Journal of Computer Science and Network Security, 2009.

[11]S. K. Agrawal, S. Joshi, B. M. Purohit,"Secure Data Communication In A Cloud Environment using Row Column Diagonal (RCD)",2015.

[12] W. Stallings," Cryptography and Network Security ".Pearson Education India, 2008.

[13]S.Tang, F. Liu , "A one-time pad encryption algorithm based ononeway hash and conventional block cipher" , Consumer Electronics, Communications and Networks (CECNet), 2012 .

[14]Z. Zhu,R. Jiang,"A Secure Anti-Collusion Data Sharing Schemefor Dynamic Groups in the Cloud",2016.

[15]Paixao CA, Coelho FC. ,"Matrix compression methods", PeerJ PrePrints3, available at: https://doi.org/10.7287/peerj.preprints.849v1,2015,accessed on 15/01/2017.

[16]Yuri Demchenko, Jeroen, Van der Ham. "On-Demand Provisioning of Cloud and Grid Based Infrastructure Services for Collaborative Projects And Groups".International Conference on Collaboration Technologies and Systems (CTS 2011). Philadelphia, USA, 2011.