Variant Security Enforcement for Distributed Mobile Environment with Random Cryptographic Primitives

[1]Viveka Priya N, [2]Sudhakar S, [3]Muruganatham S
[1]Assistant Professor, [2]Student, [3]Student
[1]Department of Information Technology,
[1]University College of Engineering, Trichy, India

Abstract: Development of cloud and its related technologies are fast in recent years, but still the demands from the mobile users are not achieved and there are many challenges related to security, e.g., data security, data integrity. In this paper the proposed method aimed to solve this user needs by adding variant security to the user files and then the file is shared by the user to third party system or untrusted server, whenever the user want the file it can be downloaded from the third party system or server and easily use the files which is uploaded by the user. This system takes the valuable data as the input and then applies the cryptographic algorithms such as AES, Blowfish, RC4 to secure the information and the algorithms are used in a random manner that means every time it changes its pattern for encryption and the pattern is again deployed by the system in reverse manner to recover the information. Compared to other data storage system, this system is a lightweight and provides deployable solution to the consumer in MCC with data integrity, data authentication, data privacy and flexible data sharing with access control. Finally we provide an efficient and easily deployable and reliable solution to the mobile user in mobile cloud computing.

Index Terms- secure data distribution, mobile cloud computing, AES, Blowfish, RC4, data privacy, data integrity, access control, data Security, symmetric key encryption, block ciphers.

---

1. INTRODUCTION

The development of cloud services and cloud related technologies are fast in recent years, but the security between server and client communications are not enough for some highly confidential data. There are many attacks on the cloud may lost its data and other intruders on the cloud stealing the data and other user information which are stored into the cloud storage. For example there is a loophole in iCloud causes many personal photos of iphone user downloaded by hackers. In traditional approach, these issues are solved using single encryption technique, but single encryption is not suitable for the third party system, because the data are vulnerable to attacks.

The proposed method enforces three different cryptographic algorithms to secure a single file of the user. Security is important for all valuable information of the user or an organization. The organization or user who wants to distribute information to his friend using the cloud server, then the information needs security. But traditional approach is not enough for the most valuable data; because traditional methods are use only one encryption to secure the user file.

This method provides security on confidential data before they are stored on the cloud and it is achieved by the cryptographic algorithms, they are AES [1] [15], BLOWFISH [2], RC4 [3]. These are the cryptographic algorithms used to provide security to the confidential information of the user or an organization.

And the algorithms are used in random manner using the user's random number, and the key is generated using the user's password. So, the user can easily store the data into the cloud and access the data using the user's password without compromising security, integrity, privacy and authentication.

In this work, we provide an efficient data sharing system with necessary security and access control. So, this system allows the user to distribute their data flexibly with their friends without compromising security and cost.

Our contributions are listed as follows:

- We search for an efficient, lightweight, fast and deployable encryption and decryption algorithm in software. Finally there are three algorithms are found that is AES, Blowfish, RC4. These are the algorithm easily deployable in software and provide rapid encryption and decryption on data.
- The sequence use of the algorithms not enough to provide efficient security, So, we use a random number, that is used to organize the algorithm as per the user wish, So, the user can easily changes the pattern to secure his/her data.
- We use the algorithms with 256 bit key size. Increasing key size creates more complexity while attacking data but it creates difficulties during encryption and decryption process, this system use only 256 bit key for encryption and decryption, so the process easily deployable in software and provide efficient calculations with reasonable time.
- This approach is mainly focused on the mobile user to distribute the data without compromising security.

The rest of the paper structured as follows. Section 2 provides some related work on this construction. Section 3 introduces some preliminaries involved in the construction. In section 4 we present a formal definition for AES, Blowfish and rc4 which is involved in our construction. And Section 5 present system model. Finally we conclude in section 6.

2. RELATED WORK

There are many remotely accessible devices and data are used. The remotely accessible things need a central control over the network; the control may be a server which is used to access the devices and data from the devices. The lots of work aimed to solve the security issues in cloud and provide an efficient software deployable solution to the issues on the cloud.

However many solution depends on the untrusted server and semi-trusted third parties, in this approach there is no need to depend any untrusted server and third party cloud. And single encryption technique is not enough to solve the problems, so we are using multiple encryptions on single user's data, then the output data is secured one.

The algorithms are handled in random manner using user's random number which selected by the user instructed in the system. So, the user able to change the pattern of the multiple encryptions and make the data more secure.

We implement this solution to the mobile environment, so the user can easily use the solution. This approach needs user's random number and user's password to encrypt the data.

Every user password digest using SHA [4] algorithm, because the password of the user may be small in size, so the SHA algorithm used to correct the key size and helps to encrypt and decrypt the data of the user. And the random number is used to map the algorithms to achieve the randomness into the system. The implemented work is mainly focused on the mobile cloud. So, the proposed work can be used in the mobiles phones.

3. PRILIMINARIES

In this section we provide some meaning for the keyword which is used in this approach. If K is the key and |k| denotes its size, and P denotes the user password. D specifies data of the user and |D| specifies its size and r denotes the random number of the user which is instructed in the system when it is used. → This symbol indicates the mapping of the algorithm and assigning values to functions. rf denotes the random function. P → K which means the user password is changed to key for encryption or decryption. In RC4 algorithm S denotes for state vector and T denotes for permutation. In Blowfish algorithm X denotes input data, XL and XR denotes left and right halves respectively and S denotes for substitution, a, b, c, d are substitution boxes. rf ← {p, r} which means the user password and random number is assigned to rf. And {AES, Blowfish, rc4} → rf, it denotes the algorithms selection by rf. i and j denotes the length of the loop. Ei denotes encrypted data and Di denotes decrypted data.

4. ALGORITHMS

4.1 Advanced Encryption standard

The AES is a symmetric key encryption algorithm [5]. It contain the variable key size of 128, 192, 256 bits, but the block size is limited to 128 bits. It is structured using substitution-permutation network [7]. And it has many features of resistance against all known attacks, design simplicity, speed and code compactness on a wide range of platforms. The input and output to the algorithm is 128 bits. In this method we are using 256 bit key size and 14 rounds.

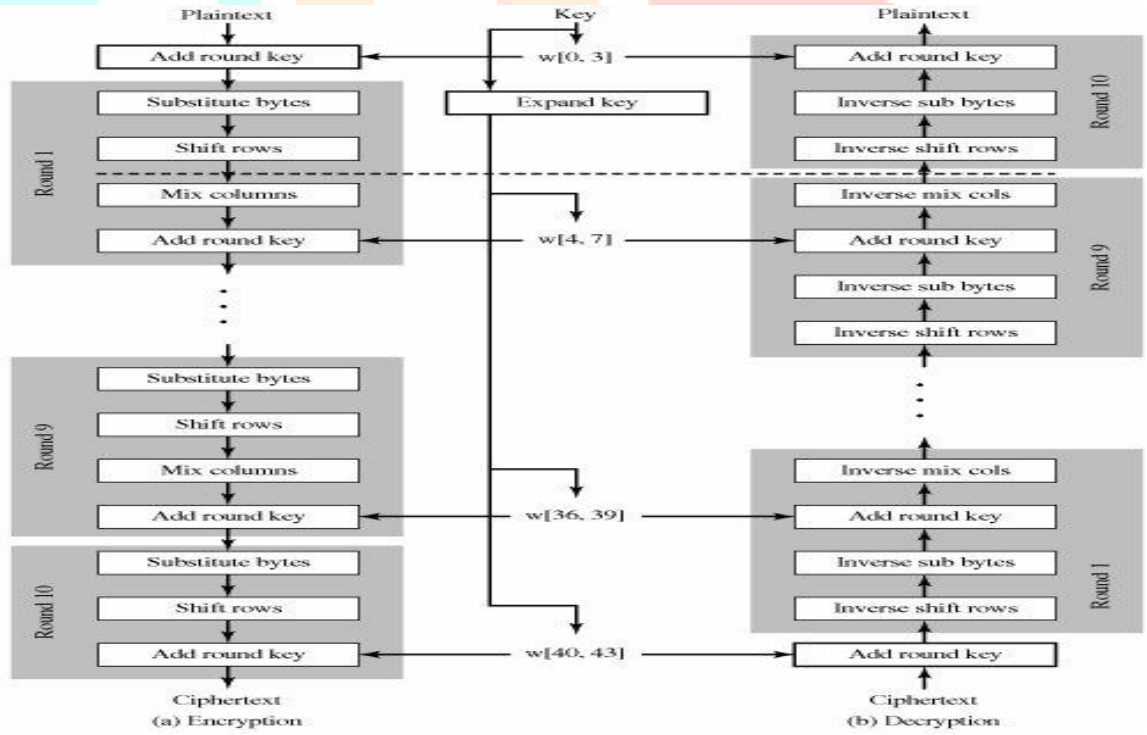

Fig 1 AES data structure

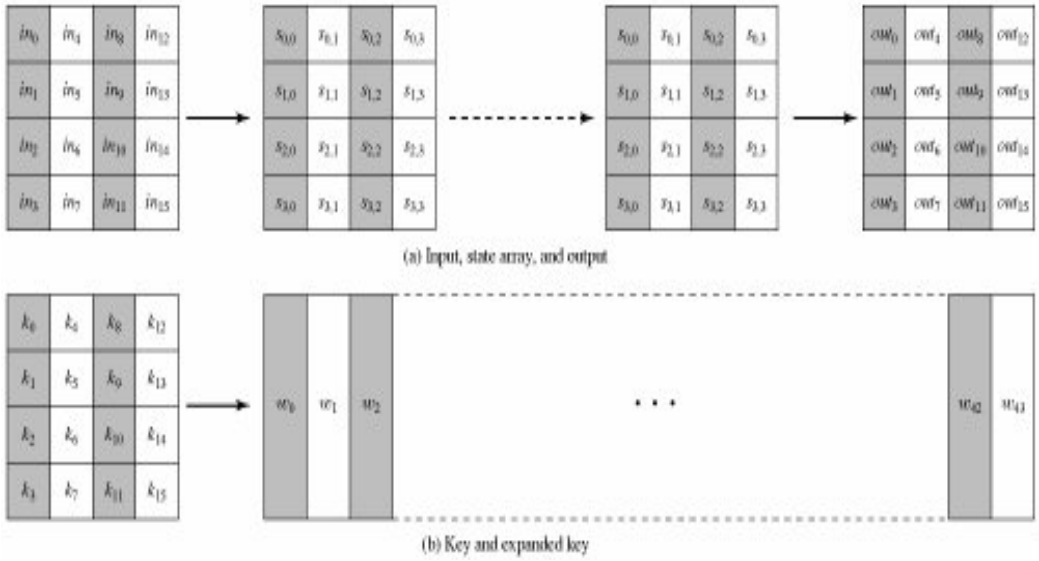

Fig 2 a. Input, state, array and output, b. Key and Expanded key

The key expansion algorithm is shown below

```
KeyExpansion (byte key[32], word [44])
{
word temp
for (i = 0; i < 8; i++) w[i] = (key[8*i],
key[8*i+1],
key[8*i+2],
key[8*i+3]);
for (i = 8; i < 44; i++)
{
temp = w[i 1];
if (i mod 8 = 0) temp = SubWord (RotWord (temp)) ⊕ Rcon[i/8];
w[i] = w[i8] ≈ temp
}
}
```

4.2 RC4

RC4 stands for Rivest Cipher 4. It is a variable key-size stream cipher [6]. This cipher expected can be process very quickly in software, and RC4 used to provide security between web browser and web server. The algorithm contain the key length of 8 bits to 2048 bits and the state vector s need to be initialized to 256 bytes with the element of s(0),s(1),s(2),s(3)…….s(256).

The algorithm for initialing s as follows:

```
/* Initialization */
for i = 0 to 255 do
S[i] = i;
T[i] = K[i mod |K|];
```

The permutation can be initialized as:

```
/* Initial Permutation of S */
j = 0;
for i = 0 to 255 do
j = (j + S[i] + T[i]) mod 256;
Swap (S[i], S[j]);
```

The stream can be generated as:

```
/* Stream Generation */
i, j = 0;
while (true)
i = (i + 1) mod 256;
j = (j + S[i]) mod 256;
Swap (S[i], S[j]);
t = (S[i] + S[j]) mod 256;
k = S[t];
```

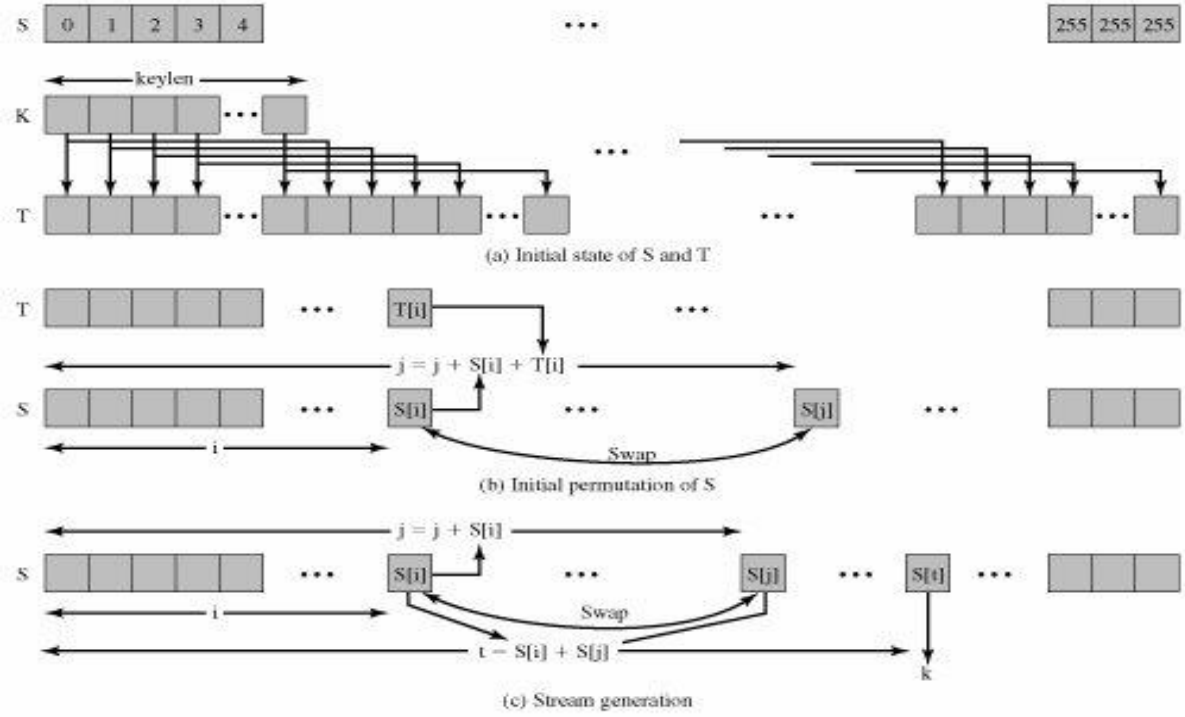

Fig 3 RC4 stream generation

In this approach the RC4 contain the key size of 256 bit and the user's P is converted into K with help of SHA function. Speed of RC4 is compared with others are shown below

1. RC4 Comparison table

Tittle	Key Length	Speed(Mbps)
DES	56 bit	9
3DES	168 bit	3
RC2	Variable	0.9
RC4	Variable	45

4.3 Blowfish Cipher

This is symmetric encryption model, It process the data block by block, This cipher contains the key size of 32 bits to 448 bits and the block size of 64 bits, it has 16 rounds. This is a Feistal network [8] [9] based cipher. Here the algorithm contains the key size of 256 bit and it is mapped using the user's r. Then the data is fed into Blowfish and encrypted output is fed into another algorithm for multiple encryptions.

Calculation for Blowfish is given below
The input data x is divided into two 32 bits halves: XL, XR
For i=1 to 16 → Rounds
XL = XL $\oplus$ Pi
XR = F(XL) $\oplus$ XR
Swap XL and XR
Swap XL and XR (undo the last swapping)
XR = XR $\oplus$ P17
XL = XL $\oplus$ P18
Recombine XL and XR.
Function F:
F(XL) = ((S1,a+S2,b mod 2^{32}) $\oplus$ S3,c)+ S4,d mod 2^{32}).

Here the input is divided into two halves and the input 32 bit is XOR with p arrays and the substitution function is calculated with the input and the output is swapped and fed into the next 16 rounds, after the 16th round the last swapping is terminated and the XL is calculated using XR XOR with p18 array as well as XR is calculated using XR XOR with p18 array. Then both XL and XR are recombined to produce encrypted output.
The input of this cipher is encrypted data or original data from the output of another cipher or user. And the key-size is 256 bit long, so the encryption much robust and easily deployable in software.

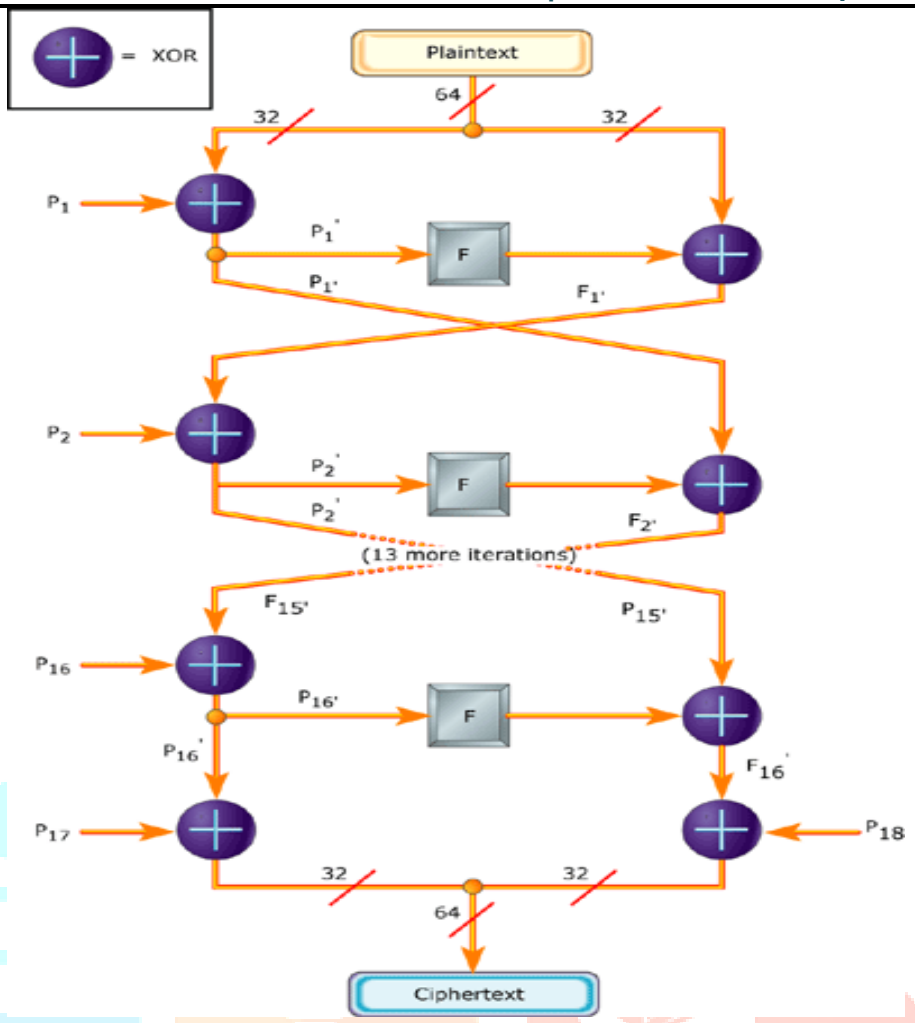

Fig 4 Blowfish Structure

4.4 Random Function

Here the random function is used to secure the valuable data and insecure the data. So, the user needs to provide the random number and password to the system.

```
/* encryption */
Input D, P, r
rf → {r}
rf→{AES,Blowfish,RC4}
/* key generation */
K = digest(P)
if(K==256)
{
Ei←rf(D,K)
}
/* decryption */
Input Ei, P, r
rf→{r}
rf→{AES,Blowfish,RC4}
/* key generation */
K = digest(P)
if(K==256)
{
Di→rf(Ei,K)
}
Di→ Original data
```

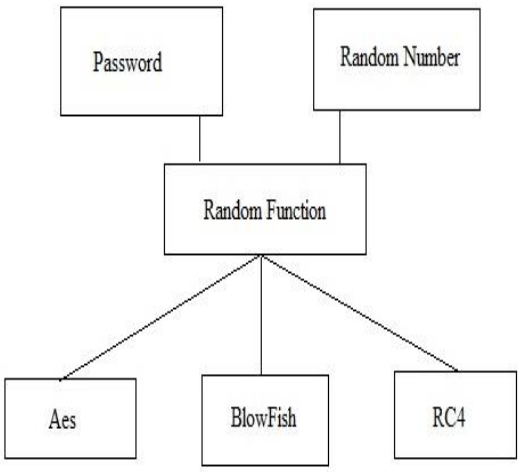

Fig 5 Random Function

Here the diagram shows the work flow of the random function. The two boxes placed at the top of the diagram is the input of the random function and the below three boxes are containing the cryptographic algorithm, that is selected and processed as per the input function and the output of one cipher is fed into the input of another one.

5. SYSTEM MODEL

5.1 Overview

In this model, the data owner need to select the valuable information to secure the data and he/she can easily share the data over the cloud without depends on the trusted servers or any trusted third party system. The user needs to go through the random number selection phase, because that is helpful in encryption and decryption phase. And the system provides security in a cost effective manner and the used cryptographic algorithms provides deployable solution to the user at reasonable time. All the algorithms works rapidly in the software, and the cipher and encrypted data put into some attacks it creates more difficulties, because the random function and key size of the algorithm make the system robust one.

5.2 Encryption

In the encryption process the data owner of this application need to choose the data and enter the random number and password for starting the encryption process. Then the function called in the background and the password of the data owner is converted into key and the random number arranges the encryption algorithm with help of random function.

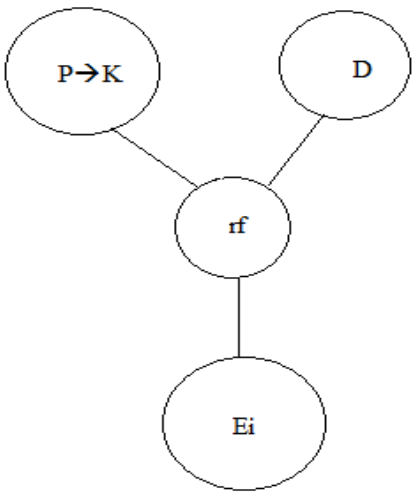

Fig 6 Encryption process

Then it provides the encrypted text as output For example if we provide a input as text data or media data into the application the process just split the data as per the block size and use the key to encrypt the data. And the user can use our method to distribute the data to someone in the world.

5.3 Decryption

The decryption process is started when the user downloads the distributed data or someone sends data to the user at receiving end. If the data is encrypted by this method then the user needs to use this application to decrypt the data which is received by the user. Here the user needs to provide received data as input and provide password and random number which is used for encryption. All these things are fed into the input of the decryption, and then the function started to decrypt the given data

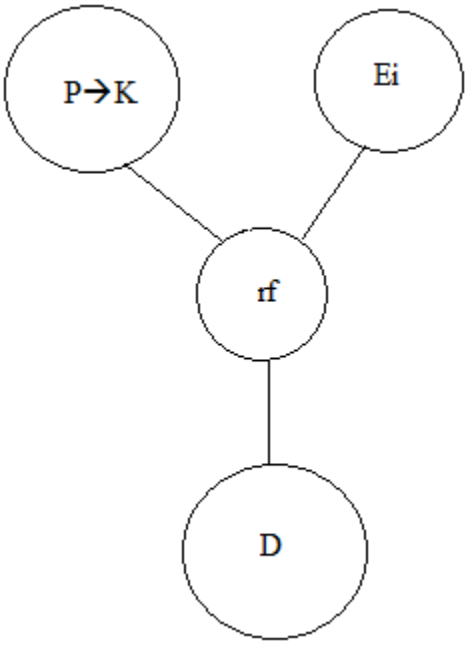

Fig 7 Decryption process

If some error is encountered by the system the data is modified or the random number or password given is wrong.

Then the decrypted file is stored in the memory location the data owner can use the data and modify the and flexibly share the data without compromising the security and cost.

5.4 Design Goals

Data Privacy: The in the server may be accessed by staff of the server maintenance or other authorities may access the data and the data owner lost his privacy, But single encryption is not enough to restrict this access. So, the multiple encryption schemes provide such privacy which is needed by the user.

Data Integrity: The data in the untrusted server or some other third party system can get the data and modify the content of the data which not visible to the sender and receiver, In this method the modification made by any unauthorized person while decrypting the it creates some error. So, the user can easily identify the mistakes in the data.

Data Authentication: The owner of the data can only access the data. No one can even see the content of the data after the encryption done on the data. So, without password and random number, no one can deal with the data.

Access Control: The data owner can use this application to securing the data and stored or shared over the remote server, but the receiver of this data cannot use the data till the sender disclose the password and random number

Light Weight: Both the sender and receiver of the data perform all the operation can performed using small amount of memory, less communication and computation overhead.

Flexible Data Distribution: The data owner can distribute the data to all or someone regarding the owner wishes without compromising security and cost.

Software Deployable: The solution can implement using the software, without need any expensive hardware.

6. CONCLUSION

We propose a real time data distribution system without rely on trusted third party system with achieving the design goals of data integrity, data authentication, data privacy, flexible data sharing with access control. But the user not able to remember the password the data cannot decrypt by the system. Our system is reliable one to use and provide solution for security issues in cloud with efficient and deployable one in real time.

FUTURE WORK

The security compromising areas are identified and provide an efficient approach to enable security on those areas. And implement the system more robust and more users friendly.

ACKNOWLEDGEMENT

We thank the anonymous reviewers for their helpful comments and suggestions.

REFERENCES

[1] Akhil K.M, Praveen Kumar M, Pushpa B.R."Enhanced Cloud Data Security using AES Algorithm". 2017 International Conference on Intelligent Computing and Control (I2C2).

[2] Srinivas Mudepali, Dr.V.Srinivasa Rao, Dr.R Kiran Kumar. "An efficient data retrieval approach using Blowfish encryption on cloud CipherText Retrival in Cloud Computing". International Conferencr on Intellgent Computing and Control System ICICCS 2017.

[3] Allam Mousa and Ahmad Hamad," Evaluation of the RC4 Algorithm
for Data Encryption". International Journel of Computer Science and Applications. Vol 3, No. 2, June 2006.

[4] Pardeep, PushPendra kumar Pateriya."PC-Rc4 Algorithm: An Enchancement over Standard RC4 Algorithm". Intenational journel of Computer Science and Network(IJCSN). Volume 1, Issue 3, June 2012.

[5] Bidisha Mandal, Sourabh Chandra, SK Safikul Alam, Subhendu Patra. "A comparative and Analytical study on Symmetric Key Cryptography". International Conference on Electrnics, Communication and Computational Engineering 2014.

[6] Daniyal M.Alghazzawi, Syed Hamid Hasan, Mohamed Salim Trigui. "Stream ciphers:A Comparitive Study of Attacks and Structures". International journel of computer application. Volume 83-No 1, DEC 2013.
[7] HOWARD M. HEYS, STAFFORD E. TAVARES. "SUBSTITUTION-PERMUTATION NETWORKS RESISTANT TO DIFFERENTIAL AND LINEAR CRYPTANALYSIS". JOURNAL OF CRYPTOGRAPHY. 1994.
[8] KAISA NYBERG. "GENERALIZED FEISTEL NETWORK". INTERNATIONAL CONFERENCE ON THE THEORY AND APPLICATION OF CRYPTOLOGY AND INFORMATION SECURITY. PP 91-104. JUNE 2005.
[9] Feistel, H.: Cryptography and computer privacy. Scientific American 228, 15–23 (1973)
[10] Xuefeng Liu, Yuqing Zhang, BoyangWang, and Jingbo Yan. Mona: "Secure multi-owner data sharing for dynamic groups in the cloud". IEEE Transactions on Parallel and Distributed Systems, 24(6):1182–1191, 2013.
[11] Chris Erway, Alptekin K upc¸ u, Charalampos Papamanthou, and Roberto Tamassia. "Dynamic provable data possession". In Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS '09, pages 213–222, New York, NY, USA, 2009. ACM.
[12] Eman M. Mohamed, Hatem S. Abdelkader. "Enchanced Data Security Model for Cloud Computing". The 8th International Conference on Informatics and Systems (INFOS2012)-14-16 May.
[13] M. Suguna, S. Mercy Shalinie. "Privacy Preserving Data Auditing Protocol for Secure Storage in Mobile Cloud Computing". IEEE WiSPNET 2017 conference.
[14] Adviti Chauhan, Jyoti Gupta. "A Novel Technique of Cloud Security Based on Hybrid Encryption by Blowfish and MD5". 4th International Conference on Signal Processing, Computing and Control(ISPCC 2017),sep 21-23, 2017.
[15] Leonardus Irfan Bayu Mahendra, Yehezkiel Khakham Santoso, Guruh Fajar Shidik. "Enchanced AES using MAC Address for Cloud Services". International Seminar on Application for Technology of information and Communication, 2017.
[16] Jiang Zang, Zhenfeng Zhang, Hui Guo. " Towards secure Data Distribution Systems in Mobile Cloud Computing". IEEE Transactions on Mobile Computing.
[17] Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song. "Provable data possession at untrusted stores". In Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS '07, pages 598–609, New York, NY, USA, 2007. ACM.
[18] Lakshmi N. Bairavasundaram, Garth R. Goodson, Shankar Pasupathy, and Jiri Schindler. "An analysis of latent sector errors in disk drives". In Proceedings of the 2007 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, SIGMETRICS '07, pages 289–300, New York, NY, USA, 2007. ACM.
[19] Dijiang Huang, Tianyi Xing, and Huijun Wu. "Mobile cloud computing service models: a user-centric approach". Network, IEEE, 27(5):6–11, September 2013. [20] Ari Juels and Burton S. Kaliski, Jr. Pors: Proofs of retrievability for large files. In Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS '07, pages 584–597, New York, NY, USA, 2007. ACM.
[20] Nguyen Thanh Hung, Do Hoang Giang, Ng Wee Keong, and Huafei Zhu. "Cloud-enabled data sharing model". In Intelligence and Security Informatics (ISI), 2012 IEEE International Conference on, pages 1–6, 2012.
[21] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. "Attribute-based encryption for fine-grained access control of encrypted data". In Proceedings of the 13th ACM conference on Computer and communications security, CCS '06, pages 89–98, New York, NY, USA, 2006. ACM.
[22] Amos Fiat and Moni Naor. Broadcast encryption. In DouglasR. Stinson, editor, Advances in Cryptology ł CRYPTO 93, volume 773 of Lecture Notes in Computer Science, pages 480–491. Springer Berlin Heidelberg, 1994.
[23] Shivaramakrishnan Narayan, Martin Gagn´e, and Reihaneh Safavi-Naini. "Privacy preserving ehr system using attribute-based infrastructure". In Proceedings of the 2010 ACM Workshop on Cloud Computing Security Workshop, CCSW '10, pages 47–52, New York, NY, USA, 2010. ACM.
[24] M. Nabeel, Ning Shang, and E. Bertino. "Privacy preserving policy-based content sharing in public clouds". Knowledge and Data Engineering, IEEE Transactions on, 25(11):2602–2614, Nov 2013.
[25] Yang Tang, P.P.C. Lee, J.C.S. Lui, and R. Perlman. "Secure overlay cloud storage with access control and assured deletion". Dependable and Secure Computing, IEEE Transactions on, 9(6):903–916, Nov 2012.