Cryptographic Based Security Algorithms for Cloud Computing

Mayank Patwal and Ms. Tanushri Mittal

Abstract— we know that Cloud Computing provide a broad range of features but the safety and confidentiality of data are major challenges here. It is required that data comes through a valid sender, and valid receiver might access it without any interruption and it is also required that if any intruder makes changes in data then receiver may identify it. For this we are using combine features of DES Algorithm and RSA algorithm to encrypt and decrypt data in this paper.

Index Terms— DES Algorithm,RSA Algorithm,Digital Signature.

1 INTRODUCTION

Data Encryption Standard (DES) also known as the Data Encryption Algorithm. DES algorithm provides improvement over the RSA algorithm. The speeds of DES encryption can be several M per second, It can be well suited for encrypted numerous message. RSA algorithm will be based upon the issue of factoring, and it is computing velocity is slower than DES,RSA algorithm is merely well suited for encrypting a tiny bit of data, The RSA encrypt the data essentially 117 bytes of once.

DES is really a block cipher. It encrypts the data in block height and width of 64 bits each. That's 64 bits are plain text goes as the input to DES, which produces 64 items of cipher text. Same key and algorithm can be used as encryption and decryption.DES uses 56 bit key but initial key is made up of 64 bits. Key is 56 items of 8, 16,24,32,40,48,56,64 are discarded (these bits may be used for parity checking to make certain the true secret doesn't contain any errors).Two fundamental features of cryptography Diffusion (Substitution) and Confusion (Permutation) rounds. In each round key and data bits are shifted, permuted, XORed and sent through, 8 round 64 bit plain-text is handed to initial permutation (IP).Then IP generates two halves left plain-text (LPT)and right plain-text (RPT).Each LPT and RPT undergoes 16 rounds. With the last LPT and RPT are rejoined. Decryption is same process perform rounds in reverse order.On 15 May 1973, the National Bureau of Standards (now called National Institute of Standards and Technology (NIST)) appealed in the Federal Register demanding an encryption algorithm that would match the criteria given below:

- Maximum security level correlated to a small key used for an encryption and decryption.
- Easily understand by everyone.
- Not depend on the algorithm's confidentiality.
- Reasonable and adaptable.

At the end of 1974, IBM proposed "Lucifer", which was enhanced on 23 November 1976 to become the DES (Data Encryption Standard). In 1978 DES was approved by the NBS. ANSI (American National Standard Institute) standardizes the DES in the name of ANSI X3.92, also known as DEA (Data Encryption Algorithm). Developed in 1974 by IBM in cooperation while using the National Securities Agency (NSA), DES have been the worldwide encryption standard in excess of twenty years. For these twenty years it has held up against cryptanalysis remarkably well and is also still secure against all but likely the most powerful of adversaries. To its prevalence over the encryption market, DES is a superb interoperability standard between different encryption equipment. The predominant weakness of DES is its 56-bit key which, in excess of sufficient to the time frame through which it had been developed, is becoming insufficient to guard against brute-force attack by modern computers. Caused by the requirement of greater encryption strength, DES evolved into triple-DES. Triple-DES encrypts using three 56-bit keys, for encryption strength equal to a 168-bit key. This implementation, however, requires 3 x as numerous rounds for encryption and decryption and highlights another weakness of DES – speed. DES was developed for implementation on hardware, and DES implementations in software are sometimes less capable than other standards which have been developed with software performance in your mind.

Unquestionably the first and the most important modern symmetric encryption algorithm is always that included in the Data Encryption Standard (DES).The DES was published by the Us' National Bureau of Standards in January 1977 for algorithm to be played with for unclassified data (information not focused on national security).

The 16-round Feistel network, which constitutes the cryptographic core of DES, splits the 64- bit data blocks into two

- *Mayank Patwal is currently pursuing masters degree program in Computer Science & Engineering from Graphic Era University, India. E-mail:mayank.patwal@gmail.com*
- *Tanushri Mittal is an Assistant Professor at the Department of Computer Science & Engineering at Graphic Era University, India.*

32-bit words, L Block and R Block (denoted by L0 and R0). In each iteration (or round), the second word Ri is fed to some function f and the outcome is put into the very first word Li. Then both tests are swapped as well as the algorithm proceeds to iteration. The function f of DES algorithm is key dependent and is made up of 4 stages.

1. Expansion (E): The 32-bit input word is first expanded to 48 bits by duplicating and reordering half of the bits.

2. Key mixing: The expanded word is XORed with a round key constructed by selecting 48 bits from the 56-bit secret key, a different selection is used in each round.

3. Substitution: The 48-bit result is split into eight 6-bit words which are substituted in eight parallel 6×4 bit S-boxes. All eight S-boxes are different but have the same special structure.

4. Permutation (P): The resulting 32 bits are reordered according to a fixed permutation before being sent to the output.

1.1 Principle of the DES

DES is a symmetric encryption algorithm that uses 64-bit blocks and 8 bits (one octet) which are used for parity checks (that verify the key's integrity). Every parity bit of the key (1 every 8 bits) is used to verify one of the odd parity of key's octet, that is every parity bit is adjusted to have an odd number of '1's in the octet it refers to. The key length is of 56 bits that means only 56 bits are actually used in the DES algorithm.

Steps involved in DES algorithm are combinations, permutations and substitutions between the text message (to be encrypted) and the key, while making definite that the operations can be done in both the directions (for decryption). The combination of substitutions and permutations is known as product cipher.

The key of ciphered contains 64 bits which are of 16 blocks of 4 bits. Provided that "only" 56 bits are really used for encryption, these can be 256 (or 7.2*1016) different keys!

1.2 The DES algorithm

The main parts of the algorithm are as follows:

- partitioning of the text into 64-bit (8 octet) blocks;
- primary permutation of blocks;
- The blocks are divided into two parts: left and right, named L and R;
- Permutation and substitution steps are repeated 16 times (called rounds);

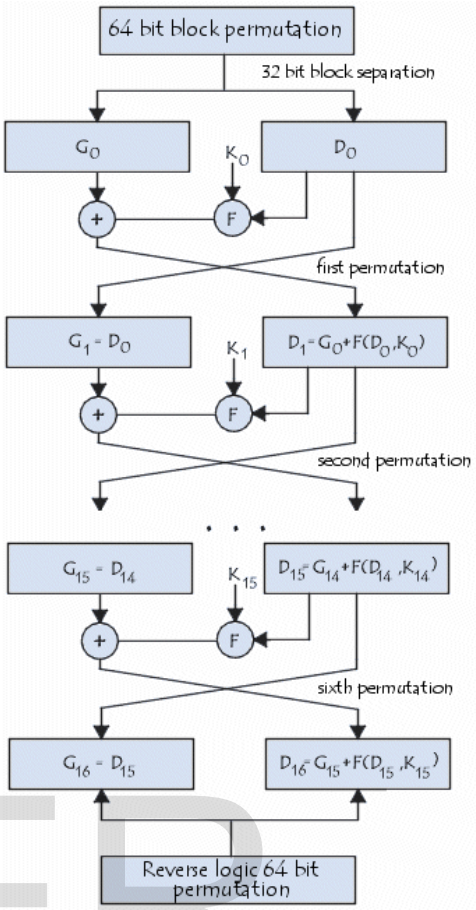

Fig 1: DES Process

2 Implementation of Proposed Work

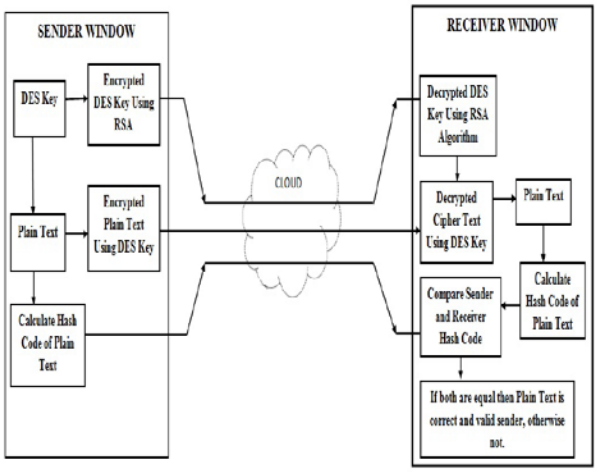

Fig 2: Implementing DES with RSA Algorithm for Security of data in Cloud Computing

Here we are securing data through DES and RSA with Digital Signature (hash function). To achieve this we have to follow given steps:

Step 1: At Sender Window

1. In first stage we generate a DES key for Plain Text and then encrypt the given Plain Text using this DES Key.
2. In second stage we encrypt the DES Key through RSA Algorithm.
3. We calculate the hash function for given original Plain Text.
4. We combined encrypted DES key, Cipher text and hash function and send to all data to Receiver through cloud.

Step 2: At Receiver Window

1. In first stage we separate all data (Encrypted DES key, Cipher Text and Sender hash function).
2. We decrypt available DES key through RSA Algorithm.
3. We decrypt the Cipher Text through DES key and get the Plain Text.
4. And finally we calculate the hash function for available Plain Text, and compare it with sender hash function. After this we have two possibilities:-

   - If both are same then we have right Plain Text and valid sender.
   - Other was not.

3 Implementation of DES Algorithm

DES is based upon the encryption techniques of confusion and diffusion. Confusion works through the substitution of specially chosen chapters of data that happen to be substituted for corresponding sections on the original data the location where the range of substituted information is in relation to the important thing along with the original plain text. Diffusion works through permutation in which the data is permitted by the rearranging the order of various sections. These permutations also are based upon the key and the original plain text like substitutions. The substitutions and permutations are specified by DES algorithm. Chosen section of the key and the data are manipulated mathematically and then used as the input to look up table. These tables are known as S-boxes for substitution table where P-boxes, for permutation tables, in software these lookup tables are known as arrays and key data input is used because index to array. Usually the S-boxes and P –boxes are combined so that the substitutions and permutation for round can be completed with a single look up.

In order to calculate the inputs to the S-box and P-box arrays, some portion of the data are XORed with some portion of the key. One of the 32-bit halves from the 64-bit data plus the 56-bit key is employed. Considering that the secret's longer versus the data half, the 32-bit data half is sent via an expansion permutation which rearranges its bits, repeating certain bits, to form a 48-bit product. Similarly the 56-bit key undergoes a compression permutation which rearranges its bits, discarding certain bits, to create a 48-bit product.

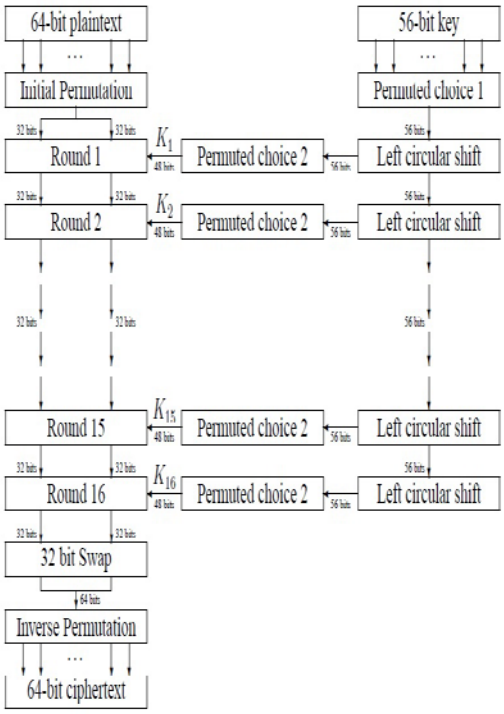

Fig 3: Flow Diagram of DES Algorithm Process

This same process of S-box and P-box substitute and permutation is recurring sixteen times, revising the sixteen rounds of the DES algorithm. There are also primary and final permutations which happen before and after the sixteen rounds.

4 Implementation of RSA Algorithm

RSA algorithm is public key encryption. This algorithm is brought to life by Ron Rivest, Adi Shamir and Len Adelman in 1977. It is hottest asymmetric key cryptographic algorithm. It may well used to provide secrecy. Therein algorithm uses the top number to come up with people key and key depending on mathematical fact and multiplying huge numbers together. It uses the block size data during which plain-text and cipher text are integers between 0and n for a lot of n values. Size n is known as 1024 bits. The real challenge in the case of RSA algorithm would be the selection and generation of the public and private key. Within this two different keys can be used encryption and decryption. As sender knows about the encryption key and receiver knows about the decryption key, the way we can generate encryption and decryption get into RSA. The whole processes are made in below:

Choose large prime numbers p and q such that p~=q.
Compute n=p*q
Compute φ (n) = (p-1)*(q-1)
Choose the public key e such that
gcd (φ (n), e) =1; 1<e< φ (n)
Select the private key d such that
d*e mod φ (n) =1
So in RSA algorithm encryption and decryption are

Performed as-
Encryption
Calculate cipher text C from plaintext message M such that
C=M ^e mod n
Decryption
M=C^d mod n

4.1 Proposed RSA Algorithm with Digital Signature

Step 1:

Choose two any large prime number such that p & q.

Calculate n=p*q.

Phi (n) = (p-1) (q-1).

Choose any integer e such that 1<e<phi (n).

Here e and phi (n) should be co-prime to each other.

e is used as public key exponent.

Find the suitable value for d so that it can satisfy congruent relation d*e=1modules phi (n) d is used as private key exponent.

Now we have public key (n, e) a private key (n, d). Keep all value of p, q, e, d and phi private.

Step 2: Digital Signature

Sender A side:-

Creates a message digist of given information which is to be send by Hash function

Hash fuction-

Using private key (n, d) compute signature s= (msg) ^d mod n

Here msg is used as a plaintext.

Send this signature s to receiver B.

Step 3: Encryption

Sender A does following-

Get receive B public key (n, e).

Compute the cipher text

cip=(msg)^cip mod n.

Send cipher text (cip) to B.

Step 4: Decryption

Receiver B does the following-

Using private key (n, d) compute

Msg= (cip) ^d mod n.

Extract plain text (msg).

Step 5: Varifying Signature

Receiver B does following-

Using Sender A's public key (n, e) compute

z=s^cip mod n.

Extract message digest, if both message digest are exactly same then signature is valid.

5 Results and Screen Shots

Step1: To accomplish over goal first we generate a 64-bits DES key using window given in figure 4. To do this first select the 'DES Key' tab in above window then click on the 'Key Generation' button. A 17 digits long number will be display on the 'key text box'. At last click on 'save key', which saves this key for further processing.

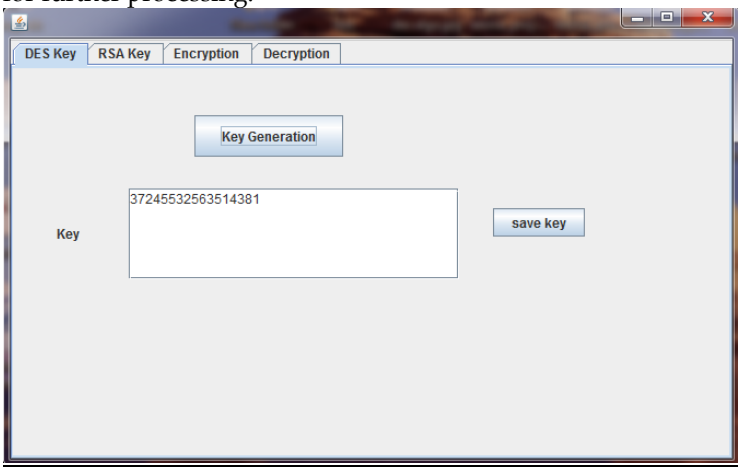

Fig 4: Automatic DES Key Generator

Step 2: After generating AES key select the 'RSA key' tab, a window given in figure 5 will appear. First click on 'Generate RSA Key' button, it will automatically generate two prime no p and q, and give their product in 'N=P*Q' text box. And also generate encryption key and decryption key, which are 256 bits long. After this save both keys using given buttons.

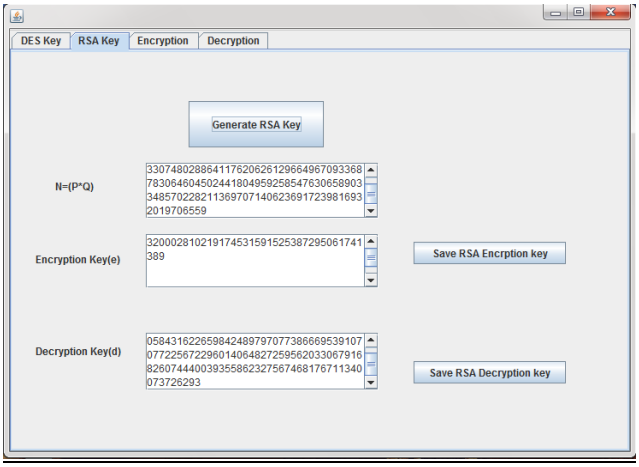

Fig 5: Automatic RSA Key Generator

Step 3: Then select 'Encryption' tab shown in figure 6. Select a file using 'browse' button. Then click on 'Generate Hash Code' button. It will generate the 'hash code' of selected file. Save this code for further processing. Chose saved DES and RSA encryption key using browse buttons. Then click on 'Save Encrypted file' button which automatically encrypt the selected file using DES key. After this click on 'Save Encrypted DES key' which encrypts the DES key through RSA Encryption key.

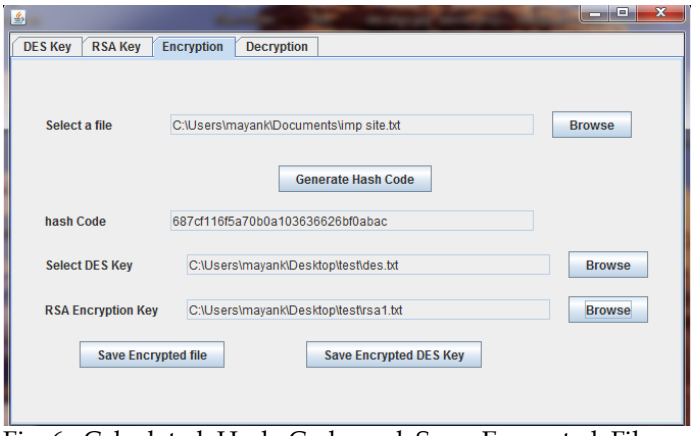

Fig 6: Calculated Hash Code and Save Encrypted File and Save Encrypted DES key

Step 4: Then go to 'Decryption' tab. Select encrypted DES key using browse button.After selecting RSA decryption key. Then click on 'Save Decrypted DES key' which automatically decrypt (using RSA decryption key) and save the DES key. Select the encrypted file using browse button and click on 'Save Decrypted File', it will decrypt the given file through DES key. At last generate the hash code of decrypted file and compare both hash code using appropriate buttons. If both hash code are same then it will denote that we have correct data and also a valid sender otherwise there is an intrusion attack.

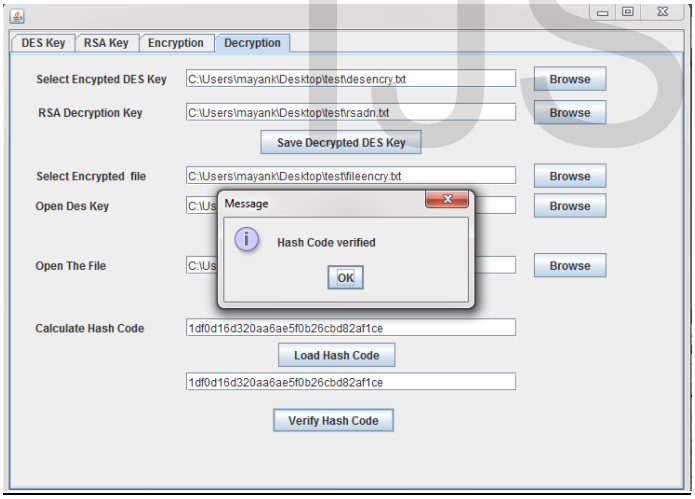

Fig 7: Save Original file using Decrypted DES key through the RSA Key

6 Conclusion

Internet use and network size is growing speedily day by day and a huge amount of users sharing information through internet and we also know that Cloud Computing provides a wide range of features but the security and privacy of data are major challenges here. It is necessary that data comes through a valid sender, and valid receiver might access it without any interruption and it is also necessary that if any intruder makes changes in data then receiver may know it. So the privacy and security of this shared information becomes an important issue. Cryptography plays an important role in information protection. There are a variety of algorithms which are used to protect information from outsiders through encryption and decryption. For this we are using combine features of DES and RSA algorithms to encrypt and decrypt data in this paper. DES algorithms are used to encrypt and decrypt data, where RSA algorithm is used to secure DES keys so that the unauthorized user can't access the private data. For more security we use Digital Signature here which verifies the integrity, confidentiality and authenticity of data. If there is an intrusion attack then receiver may know it through Digital Signature. Here we provide a greater security in data transition by using RSA, DES and Digital Signature. I have done best by my side to develop effective algorithms but we know any things want improvement with time and this improvement is limitless.

References

[1] R. Stephen Preissig, Data Encryption Standard (DES) Implementation on the TMS320C6000, Texas Instruments, Application Report, SPRA702 - November 2000.

[2] Shah Kruti R., Bhavika Gambhava, New Approach of Data Encryption Standard Algorithm, International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[3] Shah Kruti Rakeshkumar, Performance Analysis of Data Encryption Standard Algorithm & Proposed Data Encryption Standard Algorithm, International Journal of Engineering Research and Development, e-ISSN: 2278-067X, p-ISSN: 2278-800X, Volume 7, Issue 10 (July 2013), PP. 11-20.

[4] Prashanti. G, Deepthi. S, Sandhya Rani. K, A Novel Approach for Data Encryption Standard Algorithm, International Journal of Engineering and Advanced Technology (IJEAT), ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013.

[5] Dr. Rehab F. Hassan, New Approach for Modifying DES Algorithm Using 4-States Multi-keys, Eng. & Tech. Journal, Vol.28, No.20, 2010.

[6] Eman M.Mohamed and Hatem S. Abdelkader,"Enhanced Data Security Model for Cloud Computing",in Informatics and System (INFOS2012),2012 8th International Conference on,2012.

[7] Nentawe Y. Goshwe."Data Encryption and Decryption Using RSA Algorithm in a Network Environment", IJCSNS International Journal of Computer Science and Network Security, VOL.13 No.7, July 2013

[8] Shah Kruti R., Bhavika Gambhava."New Approach of Data Encryption Standard Algorithm", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012

[9] Jen-Sheng Wang, Che-Hung Liu, Grace TR Lin, How to Manage Information Security in Cloud Computing, 978-1-4577-0653-0/11, IEEE, 2011.

[10] P. Syam Kumar, R. Subramanian and D. Thamizh Selvam, Ensuring Data Storage Security in Cloud Computing using Sobol Sequence, 978-1-4244-7674-9/10., IEEE, 2010.

[11] Mr. Prashant Rewagad, Ms.Yogita Pawar, Use of Digital Signature with Diffie Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing, 978-0-7695-4958-3/13, IEEE, 2013.

[12] Mayank Patwal and Tanushri Mittal, A Survey of Cryptographic based Security Algorithms for Cloud Computing, HCTL Open International Journal of Technology Innovations and Research, Volume 8, March 2014, ISSN: 2321-1814, ISBN: 978-1-62951-499-4.

[13] T.Zhongbin, W.Xiaoling, J.Li, Z.Xin, and M. Wenhui, "Study on Data Security of Cloud Computing,"in Engineering and Technology (S-CET), 2012 Spring Congress on, 2012, pp. 1-3.

[14] Zhifeng Xiao and Yang Xiao, Senoir Member, "Security and Privacy in Cloud Computing," IEEE communication surveys & tutorials,2012.