World Journal of Engineering Research and Technology
WJERT
www.wjert.org
SJIF Impact Factor: 4.326

DESIGN OF MULTI-LEVEL ALGORITHM FOR DATA STORAGE SECURITY IN CLOUD COMPUTING

Ismail Ahmed[1] and Rashid Husain[2]*

[1]Research Scholar, Department of Mathematics and Computer Science, UMYU, Katsina,
[2]Lecturer, Department of Mathematics and Computer Science, UMYU, Katsina

Article Received on 16/04/2017 Article Revised on 07/05/2017 Article Accepted on 28/05/2017

***Corresponding Author**
Rashid Husain
Lecturer, Department of Mathematics and Computer Science, UMYU, Katsina.

ABSTRACT

Cloud computing is an Internet based technology where a large amount of computing resources are shared as a services. Since data is shared in cloud, security of the data is considered as the major concern in cloud computing. However, privacy and security is the trending topic in cloud computing technology, which leads to the abundant research in that area. In this study the proposed multi-level algorithm that will enhance the data security in cloud computing was designed by combining the multi-prime RSA and MD5 algorithms which will provide integrity and confidentiality of data in cloud computing. In the algorithm, the multi-prime RSA used for encrypting data and the MD5 used to provide a message digest and a digital signature at the same time. The designed algorithm is implemented using a java netbeans environment and we found out that, the algorithm is resistance to attacks such as mathematical attacks, brute force attacks and a hacker cannot separate the signature and the original message.

KEYWORDS: RSA, MD5, encryption, decryption, cloud computing.

INTRODUCTION

Cloud computing, to put it simply, means Internet computing. The Internet is commonly visualized as clouds; hence the term "cloud computing" for computation done through the internet. cloud computing can be considered a new computing paradigm that allows users to temporary utilize computing infrastructure over the network, supplied as a service by the cloud-provider at possibly one or more levels of abstraction. Cloud computing is the most

demanding and emerging technology throughout the world. Some of the major firms like Amazon, Microsoft and goggle have implemented the "CLOUD" and have been using it to speed up their business. With cloud computing users can access database resources via the internet from anywhere, for as long as they need, without worrying about any maintenance or management of actual resources (Rao, *et. al.,* 2009).

RELATED WORK

Many algorithms have been used to provide the security in cloud computing, the proposed algorithm used combination of two algorithms:

MULTI-PRIME RSA ALGORITHM

The Multi-Prime RSA algorithm is found to be the most secure variant when compared to other variants. The Multi-Prime RSA algorithm involves three operations: Key Generation, Encryption and Decryption. The procedure for the practical implementation of each of the operations is discussed below (Zeren, 2011).

Key Generation

The steps which are followed in the implementation of key generation of multi-prime RSA algorithm are described below:

i. Choose more than two primes $p_1, p_2 \ldots\ldots p_r$

ii. Check whether the primes are strong primes or not. If they are strong primes then proceed further else generate another set of primes.

iii. Set n = the product of the chosen prime numbers and $\varphi(n) = (p_1 - 1) \times (p_2 - 1) \times \ldots\ldots \times (p_r - 1)$.

iv. Randomly choose an odd integer e such that $gcd(e, \varphi(n)) = 1$.

v. Then compute $d = e^{-1} \bmod \varphi(n)$.

vi. Therefore the public key is (e, N) and the private key is (d, N)

Encryption

The encryption process is performed in the following way

i. The encryption key, that is, the public key is obtained.

ii. The message which is to be encrypted is divided into blocks.

iii. Once the message is divided into blocks. It is then padded so as to avoid chosen cipher text attack.

iv. These padded blocks are then encrypted to obtain cipher text $c = m^e \bmod n$ where m is the message.

v. Once all the blocks have been encrypted they are converted into *big-endian* format.

vi. The blocks are then added together so as to restore the complete encrypted cipher text.

Decryption

The step by step process of decryption is carried out in the following manner

i. Obtain the cipher text which is to be decrypted.

ii. Read the cipher text and divide them into blocks.

iii. Get the decryption key (private key) from the key generation operation.

v. m, the original message can be obtained as $m = c^d \bmod n$.

vi. The blocks are then unpadded so that we get back the original message.

vii. The blocks are added together in order to retrieve the complete message.

MD5 ALGORITHM

MD5 is a message digest algorithm developed by Ron Rivest at MIT. The algorithm takes as input a message of arbitrary length and produces as output a 128-bit message digest of the input. This is mainly intended for digital signature applications where a large file must be compressed in a secure manner before being encrypted with a private (secret) key under a public key cryptosystem (Michel and Ibrahim, 2014). MD5 processes a variable length message into a fixed output of 128-bits. The input message is broken into chunks of 512-bit blocks (sixteen 32 little endian integers); the message is padded so that length is divisible by 512. The padding works as follows, first a single bit 1, is appended to the end of the message. This is followed by many zeros as are required to bring the length of the message up to 64 bits fewer than a multiple of 512. The remaining bits are filled up with a 64-bit integer representing the length of the original message, in bits. The main MD5 algorithm operates on a 128-bit state, divided into four 32-bit words, denoted A, B, C and D. These are initialized to certain fixed constants. The main algorithm then operates on each512-bit message block in turn, each block modifying the state. The processing of the message block consists of four similar stages, termed *round* each round is composed of 16 similar operations based on the non linear function F, modular addition, and left rotation. The following figure illustrates one operation within around (Michel and Ibrahim, 2014).

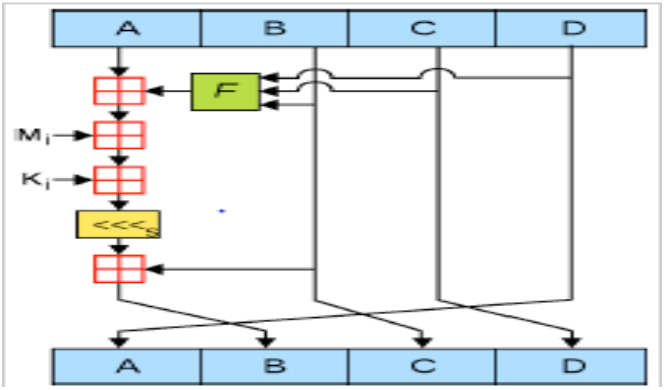

Fig. 1: MD5 algorithm

One MD5 operation, MD5 consists of 64 of these operations grouped in four rounds of 16 operations. F is a non linear function; one function is used in each round. M_i denotes a 32 bit block of the message input and k_i denotes a 32 bit constant, different for each operation. Denotes a left bit rotation by s places; s varies for each operation denotes addition modulo.

There are four possible functions F; a different a different one is used in each round:

$$F(X,YZ) = (X \wedge Y) \vee (-X \wedge Z)$$

$$G(X,Y,Z) = (X \wedge Z) \vee (Y \wedge -Z)$$

$$H(X,Y,Z) = X \oplus Y \oplus Z$$

$$I(X,Y,Z) = Y \oplus (X \vee -Z)$$

Where $\oplus, \wedge, \vee, -$ denote the XOR, AND, OR and NOT operations respectively.

Security issues in multiprime RSA and digital signature

In digital signature, we have one private key to encrypt the message it produce only message digest, it is not producing the fully encrypted data. At the end of the process the sender sends the original data along with this digital signature. There is no security for the original message. The hacker may hack the message and separate the message from the signature. In multi-prime RSA there are many possible attacks such as brute force attack, mathematical attack, timing attack and chosen cipher attacks.

Proposed Multi-Level Algorithm

To enhance the security of data in cloud computing, two algorithms are combined and form a strong algorithm that will encrypt the data and provide a digital signature at the same time using the same public and private key. However, the newly proposed algorithm is as follows:

Step 1: Key Generation

For any integer $r > 2$ (r *is a prime number*)

Let N *be the product of* r, *the randomly chosen distinct prime numbers* $p_1, p_2 \dots p_r$

Compute Euler's totient function of $N, \phi(N)$

$$\phi(N) = (p_1 - 1) * (p_2 - 1) * \dots\dots\dots\dots * (p_r - 1)$$

$$\phi(N) = \prod_{i=1}^{r} (Pi - 1)$$

choose an integer e, $1 < e < \phi(N)$ *such that*

$$gcd(e, \phi(N)) = 1$$

the pair (N, e) *is the public key*

compute the unique d *such that*

$$ed \equiv 1 \bmod \phi(N) \text{ and}$$

i.e $d = e^{-1} \bmod \phi(N)$

private key is the pair (N, d)

Step 2: Digital Signature

Use MD5 algorithm to generate message digest of the document to be send

Let an integer M *reperesent the digest generated*

Generate S (*which is the digital signature*)

$$S = M^d \bmod N$$

Step 3: Encryption

For any message m

The cipher c, *is computed using the public key* (N, e)

$$C = M^e \bmod N$$

Step 4: Decryption

For any cipher text c

The plain text is recovered using private key (N, d)

Compute m

$$m = c^d \bmod N$$

Step 5: Signature Verification

To verify the signature, an integer V is generated using public key (N, e) and the digital signature S

$V = S^e \bmod N$

Use MD5 algorithm to extract the message digest M1 from integer V.

Then compute message digest M2 from the signature S.

Test if M1 and M2 are equal (M1 == M2?) then the signature is valid

Else, invalid.

RESULT AND DISCUSSION

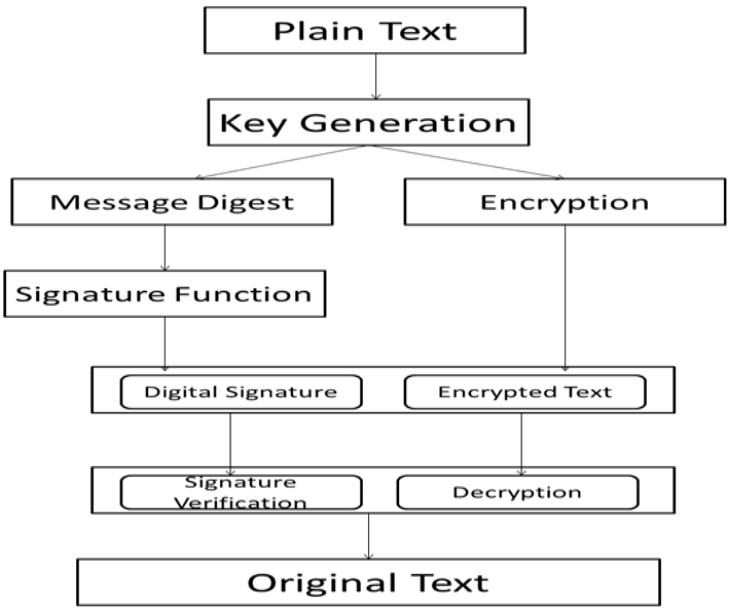

Fig. 2: Framework of the algorithm

The algorithm was implemented using a computer system with Java compiler (integrated development environment 8.0.2) installed on it.

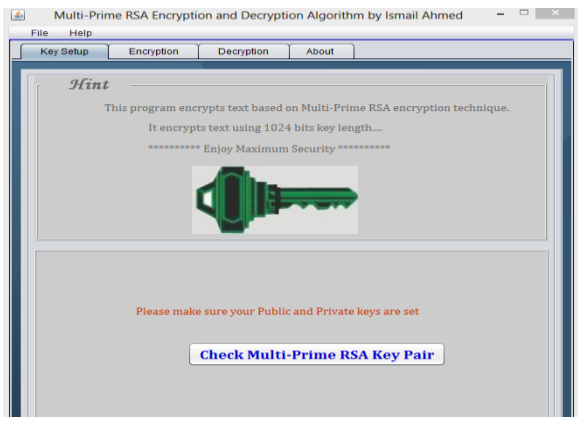

Fig.: i Key Setup

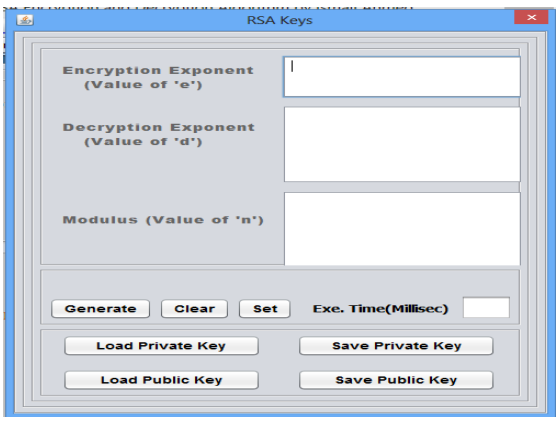

Fig.: ii Key Generation

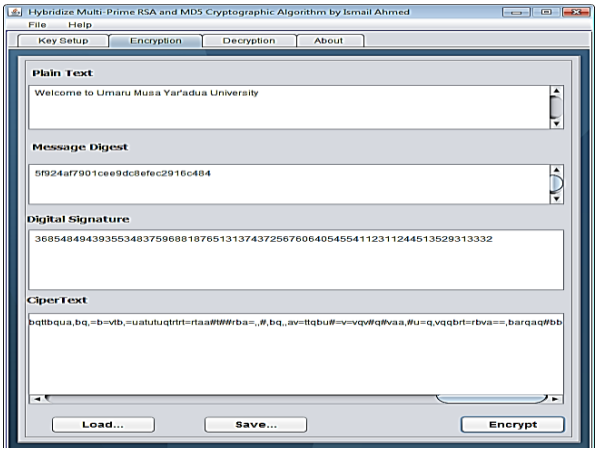

Fig.: iii. Encryption

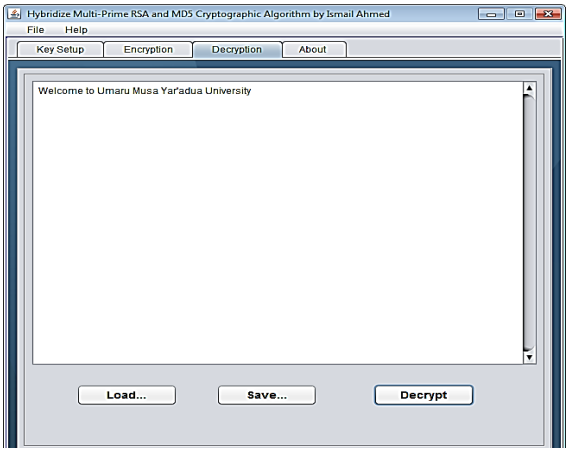

Fig.: iv. Decryption

Multi-Level Algorithm Operation

In the proposed Multi-level algorithm, the user first of all generates a key using a randomly chosen prime numbers. The algorithm will generate a private key and a public key at the same time. These private and public key would be used. The user may also decide to send a plain text before generating the key, and then it will require him to set the public and private key. After generating the key, the algorithm will generate the message digest of the text and a digital signature at the same time using the private key generated. The message digest and the digital signature of the plain text will be displayed by showing a randomly generated numbers as the signature and characters as the digest. Also the algorithm will encrypt the plain text using the public key by generating and displaying a cipher text at the same time. Having generated the digest, signature and a cipher at the same, in order to get the original text back, there is signature verification in the algorithm where by the signature will be verified using the public key and also a comparison of the message digest. The verification is the process of removing the signature provided using the private key and at the same time the decryption of the cipher will be performed using the corresponding private key. After the verification of the signature and the decryption process, then the original text will be obtained as the final output.

The fig.(i) is showing the interface of the implementation (key setup) which requires user to set two keys. In the fig.(ii) is where the user will generate the keys i.e public key and private key. After generating the keys then he set them for a particular operation. In the encryption fig.(iii) shows the encryption and digital signature of a particular operation. The original message is obtained in the decryption fig.(iv) Having combined two strong algorithms to form a single one, the proposed algorithm has a strong ability to resist some kinds of attacks like brute force, who seek probable private keys, mathematical attacks which aim at

factorizing the product of two prime numbers and collision attack. Above all the algorithm can prevent a hacker from separating the original message from the signature. Since the algorithm can encrypt and provide a digital signature to a data at the same time, therefore it provides more integrity and authenticity of a data.

CONCLUSION

This research proposed a new cryptographic algorithm for encryption and decryption of data in cloud environment, the algorithm was developed using Multi-prime RSA and MD5 algorithms. The algorithm makes data very secured by providing integrity and authenticity to it. It also stands for some attacks that break the security of other algorithms. The algorithm was implemented using java programming language.

REFERENCES

1. Apostu, A., Puncan, F., Ularu, G. and Todoran, G. (2008). Study on advantages and disadvantages of Cloud Computing – the advantages of Telemetry Applications in the Cloud. University Politehnica of Bucharest, ROMANIA.
2. Arora, R. and Parashan. Secure data in cloud computing using encryption algorithms. International Journal of Engineering Research and Application, 2013; 3(4).
3. Banu, S. K. and Kausar, A. An approach secret sharing algorithm in cloud computing security over single to multi clouds. International Journal of Scientific and Research Publications, 2013; 3(4).
4. Ganeshkumar, K. and Arivazaghan. Generating a digital signature based on new cryptographic scheme for user authentication and security. Journal of Science and Technology, 2014; 7(6).
5. Hashemi, S. Data storage security challenges in cloud computing. International Journal of Security, Privacy and Trust Management, 2013; 2(4).
6. Sudhansu, R. J. and Nayak, B. Enhancing data security in cloud computing using RSA encryption and MD5 algorithm. International Journal of Computer Science Trends and Technology, 2014; 2(3).
7. Sugunya, M., Boopal, M. E and Naveena, M. Implementing multiprime RSA algorithm to enhance the data security in cloud computing. International Journal of Innovative Research in Scienec, Engineering and Technology, 2015; 4(3).
8. Zareen (2011). Enhancement on implementation of multi-prime and multi-power RSA algorithm. Department of computer Science and Engineering. Thapar University.