Global Scientific JOURNALS

GSJ: Volume 7, Issue 5, May 2019, Online: ISSN 2320-9186
www.globalscientificjournal.com

INTEGRATION OF RSA WITH DELTA ENCODING TECHNIQUES IN CLOUD SECURITY AND PRIVACY

Dr. Thomas Yeboah[1] , Mr. Farouk Umar , Mr. Emmanuel Abaidoo [3] and Mr. Mahendra shrivas[4]

1. Christ Apostolic University College email: thomyebs24@gmail.com
2. Christ Apostolic University College email: thomyebs24@gmail.com
3. Christ Apostolic University College email: thomyebs24@gmail.com
4. BlueCrest University College,mkshrivas@gmail.com

Abstract

With passage of time cloud computing has been on high demand due to its cost and high reliability along with high security and scalability. Cloud Computing enables shared resources, software, and information to be provided to computers and other devices as a utility. However in the area of research it is observed that cloud computing still has some issues in security regarding privacy. In cloud computing data are sent over an insecure channel and they are therefore subjected to so many types of intruders. This means that it has become necessary to protect the secrecy of messages which are sent over an insecure channel so that correct data are received by the receiver at the right time without any intrusion. In order to ensure secrecy of data when sent over insecure channel an encryption is employed. In this research work the encryption algorithm employed is an extension and modification of the RSA cryptosystem. In RSA algorithm the message to be encrypted does not undergo any form of transformation or encoding prior to encryption and the level of encryption is therefore homogeneous. In our work, we extended the level of encryption to two, which makes it heterogeneous. Prior to RSA encryption, the message is subjected to an encoding mechanism using 'Delta Encoding Technique'. Thus the proposed security algorithm preserves the security in cloud computing in two phases that is by RSA algorithm and 'Delta Encoding Technique'. Therefore the researchers objective is to develop an enhancement encryption scheme which is heterogeneous compared with the traditional RSA system that is homogeneous, which brings us toward improved RSA cryptosystem for privacy in terms of the level of transformation.

For this purpose, Key Generation, Encryption and Decryption Time in Original RSA and the proposed algorithm have been compared according to the different size of exponents. Moreover, some of the common attacks against RSA algorithm have been analyzed to detect the resistance of the proposed algorithm against possible attacks. After thoroughly comparison between the original RSA and the proposed algorithm, the results showed that the proposed algorithm was better than the original RSA in terms of security and total execution time.

Keywords—RSA algorithm, Delta Encoding Technique, Cloud Security, encryption, decryption

1. INTRODUCTION

Cloud computing is a general term that delivers hosted services over internet. Cloud computing enables users to remotely run services such as Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) [2]. With the advent of Cloud computing, the issue of security and data privacy has become more prominent because the communications channels are vulnerable and subject to attack by intruders since it involves open communication traffic.

In cloud computing environment, we must safeguard against both obvious and subtle intrusions that can delete or corrupt vital data [1] by using appropriate cryptography technologies. The primarily idea behind cryptography is to send data in a secure manner so that information gets to the right person and at the right time without intrusion. The RSA cryptosystem, named after its Inventors R. Rivest, A. Shamir, and L Adlemanis one of many widely used public-key Cryptosystem schemes which utilizes Integer factorization problem in the encryption/decryption process [9]. RSA is an asymmetric cryptosystem. This means that the algorithm consists of two mathematical transformations: an encryption function E, and a decryption function D. It means that in order to ensure a secure communication between a sender (traditionally called Alice) and the receiver (traditionally called Bob) the sender supposes to apply the encryption function to the original message P (the plaintext), and transmit the resulting ciphertext C = E (P) over the insecure channel. Once C is received by the intended recipient (called Bob), the plaintext is recovered by computing D(C) = P. It is well noted in traditional RSA algorithm that there is always a trade-off between security and efficiency: at some point the, moduli must be large for security; on the other hand, small moduli are preferred for efficiency. How large they have to be, depends on the speed of so-called factorization algorithm [7].

In view of this the main objective of this research is to develop an effective security encryption scheme which is heterogeneous compared with the current RSA system that is homogeneous. In our work we extended the level of encryption to two, which makes it heterogeneous. Prior to RSA encryption, the message is subjected to an encoding mechanism using 'Delta Encoding Technique'.

2. ISSUES IN CLOUD SECURITY

Most security principles in cloud computing can be traced back to the security triad (also called the AIC or CIA triad).

The security triad includes three key security principles that are at the core of all security practices. These three issues of cloud computing security are: confidentiality, integrity and availability; known as the AIC triad [5].

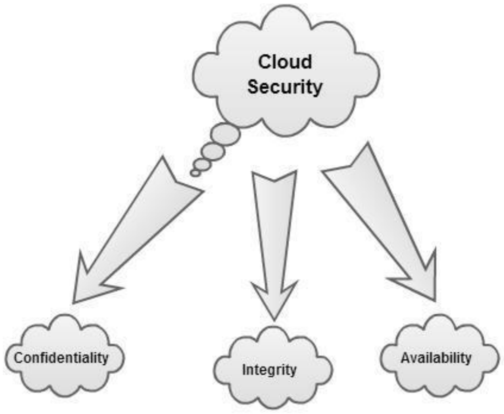


Fig 1-1: AIC Triad

A. Confidentiality

This element ensures that only authorized people are able to access data. It is ensured by: security protocols, authentication services and data encryption services.

B. Integrity

Data integrity prevents the unauthorized modification of data and ensures that unauthorized modification is detected. It is ensured by: Firewalls and intrusion detection.

C. Availability

Availability ensures that systems and data are up and available when needed. It is ensured by: fault tolerance, network security and authentication.

3. RELATED WORKS

RSA Encryption is one of the most popular and simple algorithms used to implement cloud data security and privacy. Due to its simplicity and popularity in nature; there has been lot of research carried out to improve performance of this algorithm and therefore there are so many variants of this algorithm.

Venkatesh et al [10] proposes RSASS system for data security. The scheme uses RSA algorithm for encrypting large files and storing the date. The system can be used for storing large databases. But the use of linear methods compromises with the data retrieval speed.

Hence, this system is good for static data.In [6] De, S. Haldar et al indicated in their research work that in traditional RSA algorithm; there is always a trade-off between security and efficiency: at some point the, moduli must be large for security; on the other hand, small moduli are preferred for efficiency. How large they have to be, depends on the speed of so-called factorization Algorithm.Ashish Agarwal et al. [3] talk about security issues concerned with cloud computing. Theirpaper has talked about some serious security threats that prevails this field.Ashutosh Kumar et al. [4] focused on providing a secure architectural framework for sharing and data gathering. This cynosure of this work is that the authors have made a permission hierarchy at different levels. The authors have focused on security but with view of use hierarchy.Delta encoding is a way of storing or transmitting data in the form of differences (deltas) between sequential data rather than complete files [8]. This Delta encoding techniques include both Newton forward and backward differentials.

4. METHODOLOGY

The methodology adopted in our research is the structured systems analysis method. In this methodology the existed algorithm (RSA algorithm) was critically analyzed and examined to note the possible problems/drawbacks.

4.1 RSA Algorithm

The RSA algorithm involves three steps: Key generation, encryption and decryption.

a. Step 1: Key Generation

1. Choose two distinct prime numbers p and q (e.g p= 3 and q =11).
2. Compute n = p * q = 3 * 11 = 33
3. Compute φ(n) = (p - 1) * (q - 1) = 2 * 10 = 20
4. Choose ***e*** such that 1 < e <φ(n) and e and n are co-prime. Let e = 7
5. Compute a value for ***d*** such that (d * e) % φ (n) = 1. Hence d = 3 ,[(3 * 7) % 20 = 1]
6. Therefore the following are valid
   - ✓ Public key is (e, n) – (7,33)
   - ✓ Private key is (d, n) – (3,33)

b. Step II: Encryption

For a plaintext message m, the encryption function is

c (m) = m^e (mod n)

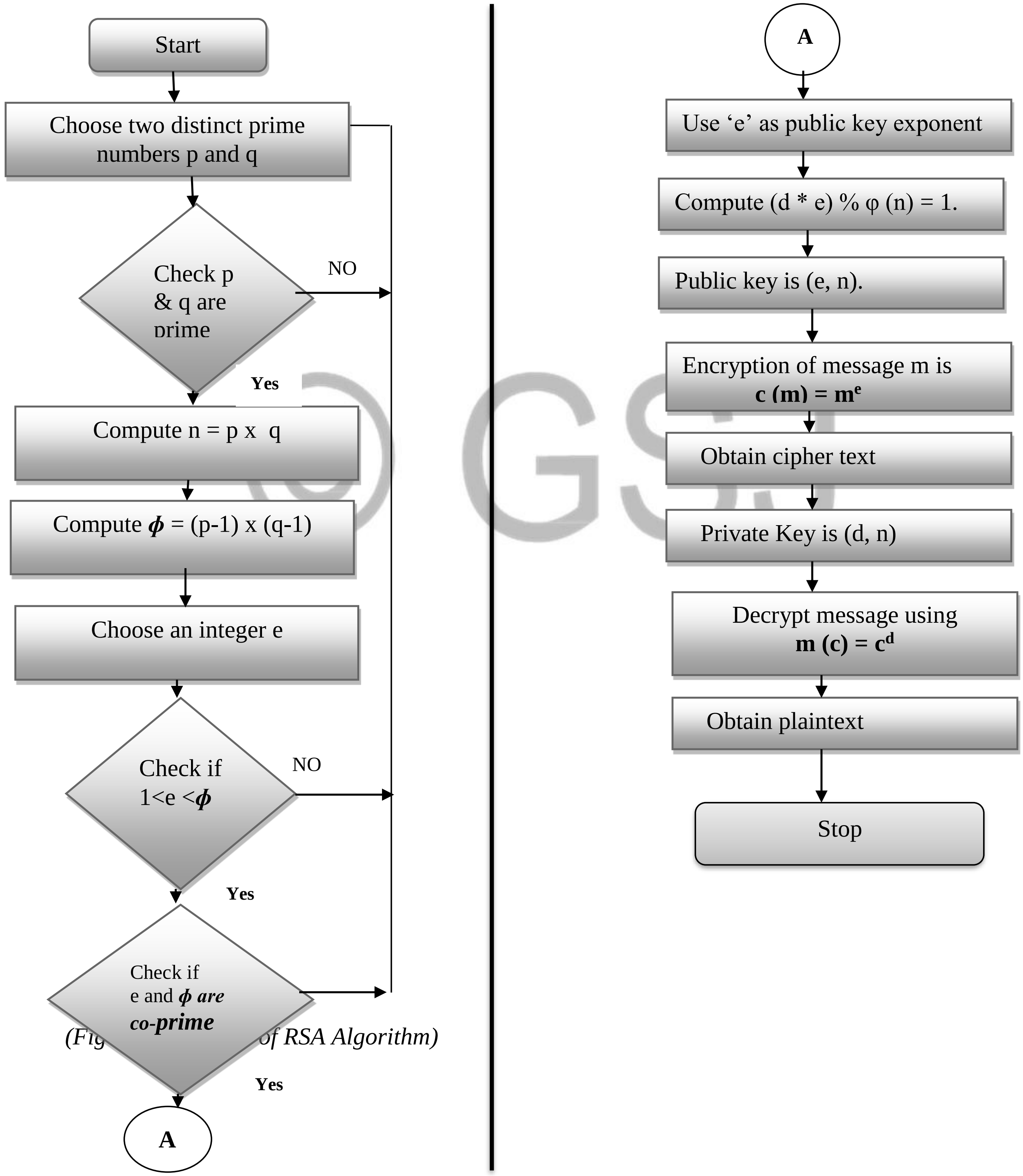
c. Step III: Decryption

Therefore for an encrypted ciphertext c, the decryption function is

m (c) = c^d (mod n)

4.2 RSA Flowchart

The figure 1-2 shows the flowchart for RSA cryptosystem.


Start
Choose two distinct prime numbers p and q
Check p & q are prime
NO
Yes
Compute n = p x q
Compute ϕ = (p-1) x (q-1)
Choose an integer e
Check if 1<e <ϕ
NO
Yes
Check if e and ϕ are co-prime
Yes
A
A
Use ‘e’ as public key exponent
Compute (d * e) % φ (n) = 1.
Public key is (e, n).
Encryption of message m is c (m) = m^e
Obtain cipher text
Private Key is (d, n)
Decrypt message using m (c) = c^d
Obtain plaintext
Stop

(Fig of RSA Algorithm)

4.3 Delta Encoding Technique

In our work we used Newton Forward differential to change the original text into an encoded form by recording difference between successive objects and backward differentials to change the encoded text into the original message.

a. Newton Forward Differential

In Newton Forward Differential data are encoded as follows:

Fd = md (for first value of message m)

and for the subsequent message(s)

Fd+1 = md +1 – md

Where d = 1, 2,3,… and m is the message.

For example, table 1-1 shows a plaintext and its encoded form

Message	10	12	8	6	14	15
Encoded	10	2	-4	-2	8	1

(Table 1-1: Plaintext and Encoded form)

b. Newton Backward Differential

In Newton backward Differential data are retrieved as follows:

md = Fd (First value of original text).

For subsequent values of message we have

md +1 = Fd+1 + md

PROPOSED ALGORITHM FOR SECURITY IN CLOUD ENVIRONMENT

5.1 Algorithm

Our proposed algorithm of cloud data privacy involves Six steps: ASCII Message, Encoded form (Newton Forward), Key generation, encryption, decryption and Newton backward.

a. Step 1: ASCII message

1. Obtain the ASCII code of the message entered.

b. Step 2: Encoding(Newton Forward)

2. Encode message (ASCII Code Values) using Newton Forward differential.

c. Step 3: Key Generation

3. Choose two distinct prime numbers p and q from Encoded message
4. Compute $n = p * q$
5. Compute $\varphi(n) = (p - 1) * (q - 1)$
6. Choose ***e*** such that $1 < e < \varphi(n)$ and e and n are co-prime.
7. Compute a value for ***d*** such that $(d * e) \% \varphi(n) = 1$.
8. Therefore the following are valid
   - ✓ Public key is (e, n)
   - ✓ Private key is (d, n)

d. Step 4: Encryption

9. Encryption: using the encoded data, F.

$$\mathbf{c(F) = F^{e} \pmod{n}}$$

e. Step 5: Decryption

10. Decrypt the encrypted data in (9) using :

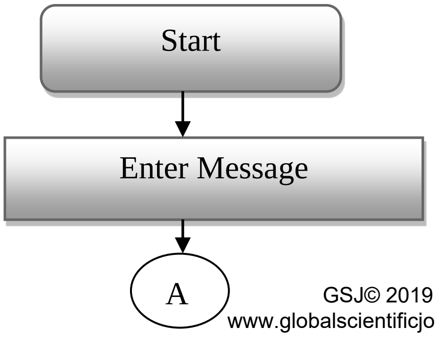
$$\mathbf{F(c) = c^{d} \pmod{n}}$$

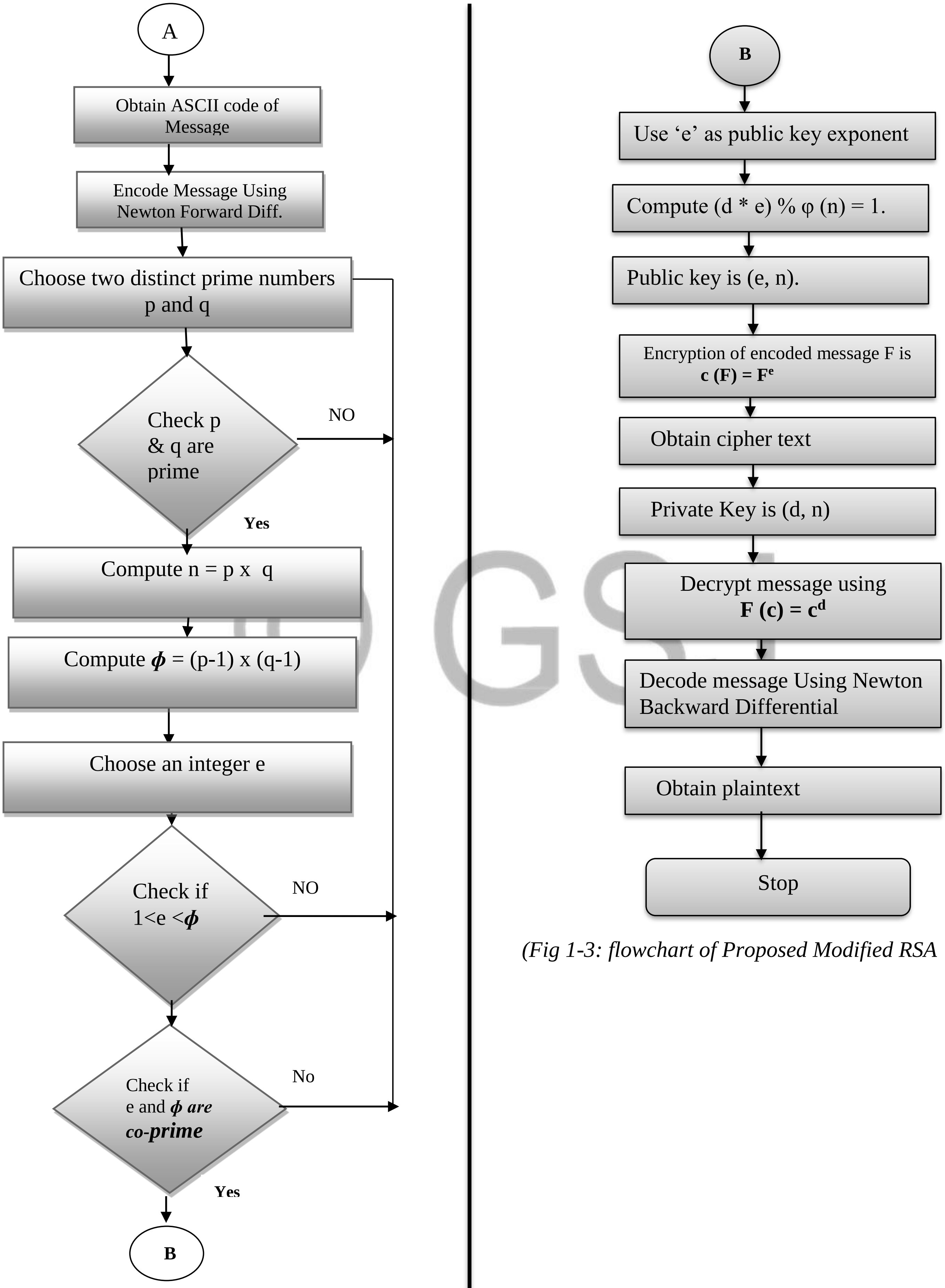
f. Step 6: Encoding (Newton Backward)

11. Encoded message in (10) to obtain original message using Newton Backward differential.

5.2Flowchart for proposed algorithm

Figure 1-3 shows the flowchart of the proposed algorithm




A
Obtain ASCII code of Message
Encode Message Using Newton Forward Diff.
Choose two distinct prime numbers p and q
Check p & q are prime
NO
Yes
Compute n = p x q
Compute ϕ = (p-1) x (q-1)
Choose an integer e
Check if 1<e <ϕ
NO
Check if e and ϕ are co-prime
No
Yes
B
B
Use 'e' as public key exponent
Compute (d * e) % φ (n) = 1.
Public key is (e, n).
Encryption of encoded message F is c (F) = F^e
Obtain cipher text
Private Key is (d, n)
Decrypt message using F (c) = c^d
Decode message Using Newton Backward Differential
Obtain plaintext
Stop

(Fig 1-3: flowchart of Proposed Modified RSA

5.3Illustration of Proposed Algorithm for Security

In this proposed modified RSA Algorithm with Delta Encoding techniques, let us assume that Bob wants to send a message " **I love my wife"** to Alice with the following key attributes: p=3, q=11, n=33,e=7 and d=3. Table 1-2 shows the protected message using the proposed algorithm

ORIGINAL MESSAGE	*ASCII*	*NEWTON FORWARD*	*ENCRYPTED* **c (F) = F^e (mod n)**	*DECRYPTED* **F(c) = c^d (mod n)**	*NEWTON BACKWARD*	*MESSAGE*
I	73	73	28	7	73	*I*
	32	-41	-2	-41	32	
l	108	76	10	76	108	*l*
o	111	3	9	3	111	*o*
v	118	7	28	7	118	*v*
e	101	-17	-8	-7	101	*e*
	32	-69	-9	-69	32	
m	109	77	11	77	109	*m*
y	121	12	12	12	121	*y*
	32	-89	-23	-89	32	
w	119	87	21	87	119	*w*
i	105	-14	-20	-14	105	*i*
f	102	-3	-9	-3	102	*f*
e	101	-1	-1	-1	101	*e*

(Table 1-2: A protected message using proposed algorithm)

6. RESULTS AND DISCUSSION

6.1 **Comparison of Key Generation time and Total Execution**

The simulation result of the proposed algorithm and the original RSA was implemented in java. In the comparison between proposed algorithm and original RSA; different size of exponents (256,512, 1024, and 2048) were considered. Table 1-3 and Table 1-4 show the simulation results of the proposed algorithm and original RSA.

Table 1-3: Key Generation, Encryption and Decryption Times for proposed algorithm.

Size (bits)	*Key generation Time(ms)*	*Encryption Time(ms)*	*Decryption Time(ms)*	*Total execution time(ms)*
256	95	74	71	240
512	101	84	82	267
1024	294	156	148	598
2048	1212	984	964	3160

Table 1-4: Key Generation, Encryption and Decryption Times for original RSA.

Size (bits)	*Key generation Time(ms)*	*Encryption Time(ms)*	*Decryption Time(ms)*	*Total execution time(ms)*
256	91	70	67	228
512	98	79	76	253
1024	289	152	141	582
2048	2542	982	956	4480

According to the results, the encryption, key generation and decryption times for the proposed algorithm were always less than the original RSA algorithm when the exponent sizes were 256, 512 and 1024 bits. The key generation time comparison of the proposed algorithm and original RSA shows drastic increment which resulted in increase in total execution time for the proposed algorithm been larger than the original RSA. The key generation time comparison between proposed algorithm and original RSA is shown in figure 1-4.

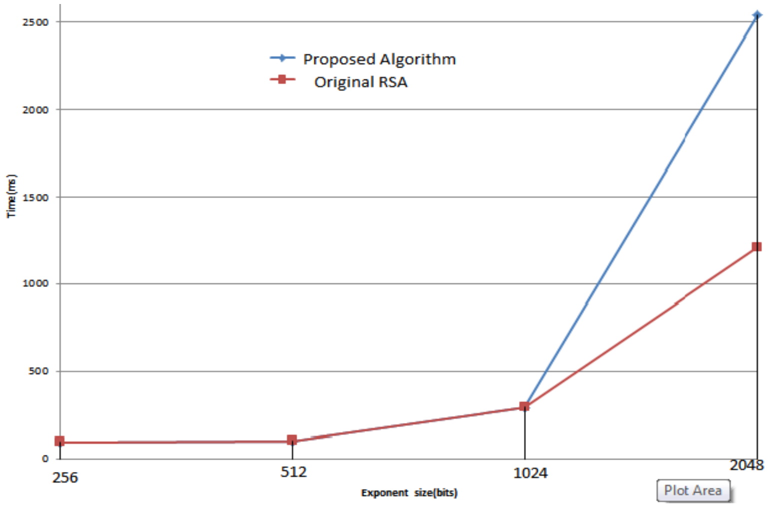


Fig 1-4: Key Generation Times Compared

4.2 **Security Analysis (Compared)**

The security analysis of the proposed algorithm and the original RSA has been investigated according to three attack approaches: theBrute Force, Mathematical Attacks, and Timing attacks.

A. Mathematical Attacks

Mathematical attacks focus on attacking the underlying structure of RSA function. The first intuitive attack is the attempt to factor the modulus N. Because knowing the factorization of N, one may easily obtain M (N), from which d can be determined by d = 1/e mod M (N). In original RSA because it is homogenous in nature it is easily to factor the modulus N whiles in this proposed algorithm it is heterogeneous in nature and therefore make it difficult to factor the modulus.

B. Brute Force attacks

In brute force attacks you need to explore all the possible combinations to guess the private key. In the proposed algorithm the attacker has to guess the encoded private key and the original private key which makes it difficult to guess as compare to the original RSA which requires the attacker to only guess the original private key.

C. Timing Attacks

In cryptography, a timing attack is a side channel attack in which the attacker attempts to compromise a cryptosystem by analyzing the time taken to execute cryptographic algorithms. Timing attack in original RSA may beprevented by including a random delay to the exponentiationalgorithm or multiplying the cipher-text with a randomnumber [10] while heterogeneous encryption algorithm like the proposed algorithm will protect thetransferred message from the timing attack and it is notnecessary for multiplying the cipher-text.

7.CONCLUSION

In this paper, a heterogeneous encryption algorithm has been suggested based on RSA and Delta Encoding Techniques to improve security issues and reduction of execution time in cloud computing environment. Original RSA is based on block cipher. Our work is based on stream cipher. Stream ciphers are generally faster and more appropriate than block ciphers and a dual encryption process has been applied to this proposed algorithm to raise the security level of the algorithm in comparison of original RSA.

8. REFERENCES.

[1]. Almarini, A. and Alsahdi, U. (2012), “Developing a Cryptosystem for XML Documents”, International Journal of Information Science, Vol. 2. No. 5, pp. 65 – 69.

[2]. Amazon Elastic Compute Cloud. http://aws.amazon.com/ec2/ (accessed on January 23, 2015)

[3]. Ashish Al, Aparna A.,(2011), “The Security Risks Associated with Cloud Computing”. International Journal of Computer Applications in Engineering Sciences [VOL I, SPECIAL ISSUE ON CNS, JULY 2011] [ISSN: 2231-4946]

[4]. Ashutosh Kumar D, Animesh Kumar D, Mayank N, Shiv Shakti S,(2012),”Cloud-User Security Based on RSA and MD5 Algorithm for Resource Attestation and Sharing in Java Environment. Software Engineering (CONSEG)”, CSI Sixth International Conference

[5]. Cherdantseva, Y.; Hilton, J.(2013), "A Reference Model of Information Assurance & Security," Availability, Reliability and Security (ARES), Eighth International Conference on , pp.546-555, 2-6

[6]. De, S. Haldar, A., and Biswas, S. (2013), “A Review on Recent Trends in Cryptography”, International Journal of Latest Research in Engineering and Computer (IJLREC), Vol.1, Issue 1.Sept- Oct., pp. 50-55.

[7]. Jajoda , S., Ammana, P. and McCollum , D. C., (1999), “Surviving Information Warfare Attacks, IEEE Computer”, April, pp. 57-63.

[8]. Jeffrey C. Mogul, Fred D., Anja F, and Balachander K.,(1997), “Potential benefits of delta encoding and data compression for HTTP”, Proceedings SIGCOMM '97, Cannes, France

[9]. Rivest, R., Shamir, A., and Adleman, L, (1978). “A Method for Obtaining Digital Signatures and Public Key Cryptosystems”, Communications of the ACM.

[10]. Venkatesh M,.SumalathaR,SelvaKumarC(2012), “Improving Public Auditability, Data Possession in Data Storage Security for Cloud Computing”, ISBN: 978-1-4673-1601-9/12 IEEE