Improvement in QoS Based on Data Integrity for Cloud

Shivani Kaushik[1], Anirudh Kumar Tripathi[2], Pankaj Pratap Singh[3], Amit Kishor[4]
[1234]Departement Of Computer Science And Engineering, Swami Vivekanand Subharti University

Abstract- With the increase in usage of internet data security problems are also increasing day by day. The emerging advanced technical needs those technologies which are less time consuming and more secure. To meet these expectations we have worked on hybrid RSA Diffie Hellman algorithm. We have improved the security provided by the original algorithm. This is done by changing the keys sent over the communication channel. Instead of sending the original public key and private key we are sending other public key and private key, which are calculated using original keys. This enhances the security provided by the algorithm. It makes encryption and decryption by unauthorized user difficult. Time consumption which is another parameter to be considered and cannot be ignored as trivial is also taken into consideration in our new algorithm. It is noticed that if we use XNOR gate instead of XOR gate it reduces the time taken by algorithm. Thus we have tried to work on the dual issue in our research paper. The first issue is the security problem and another issue is the time consumption. We have made certain improvements on both the issues.

***Keywords* -** encryption, decryption, public key, prime number, RSA algorithm, diffie Hellman algorithm

I. INTRODUCTION

Today's world involves e-commerce. Internet serves as a source of resources for rising technology. It also provides a global market place. Instead of running data on one’s own device, everything is hosted on cloud storage [13]. We are using cloud computing in our day to day life whether intentionally or unintentionally. Many people don't make out that they are using the cloud while using it. [11].

Cloud uses a network of remote servers hosted on the internet. Cloud stores, administer, and process data, rather than a local server or a personal computer. Cloud computing helps to get rid of the boundaries of the desktop by providing data all around the world. There are several security problems related to this technology. This is a common problem when any technology expands. Storing and transferring data on remote server leads to security problems [7].

Quality-of-service management is one of the challenges faced by cloud. While providing services to the customer, it is necessary to keep away from hacking of data, particularly data of banks and institutions.

When data is stored on a remote server there are variety of threats to data. Data integrity technique ensures no loss in the flow of data. There is a lack of security conservation even after so much enhancement in technology [2].Cryptography provides a way to conceal the data while transferring it to other users. It is art to hide data from unauthorized user. Due to growing technology, the need for data safety has also increased [14]. Since a lot of data is present on the cloud; it includes data which need to be secure.

Authorization of data is checked by the RSA algorithm [1].To provide the security we use cryptography. Cryptography is used to encrypt and decrypt data. Using cryptography we can transfer sensitive data over an insecure network. This helps to avoid unauthorized reader to interpret it. Cryptography involves encryption and decryption of data. Encryption is the process of conversion of plain text into cipher text. Original data is called plain text and unreadable format of data is called ciphertext.

There are two types of cryptography.

1. Symmetric key encryption and
2. Asymmetric key encryption.

Hybrid cryptography uses a combination of both symmetric and asymmetric cryptography.

Symmetric-key encryption uses one key for both encryption and decryption and asymmetric-key uses a different key for encryption and decryption. It helps to successfully exchange secret keys over the public channel [3]. Data security plays an important role in data transmission through the communication channel. Therefore confidentiality, integrity, and availability are considered as the key objectives on the subject of data security [6].

To achieve optimal effectiveness, both the algorithms are combined together and make a hybrid algorithm. The hybrid algorithm helps to achieve optimal efficiency [11].

In this paper, we are using RSA and Diffie Hellman algorithm. RSA is an asymmetric-key algorithm and Diffie Hellman is a symmetric -key algorithm. This combined approach is anticipated to get security advantage of public key and speed advantage of the secret key system.

II. RELATED WORK

Symmetric cryptography -Symmetric-key cryptography is an encryption method in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way). These key represent a shared secret key between parties. This was the only encryption known until June 1976.

The symmetric key is advantageous because: they are reasonably priced to produce a strong key. It provides defense of higher level in comparison to its size of the key. The size of the key is often small and security provided is high. They are relatively economical to process. It is highly efficient because it does not provide any delay in output as a result of encryption and decryption. It also provides authentication because encryption and decryption is to be carried out with the same key i.e. the key which is used for encryption is to used for decryption. Thus as long as the symmetric key is kept undisclosed both the communicating parties can be certain that they are in communication with the authenticated party [8].

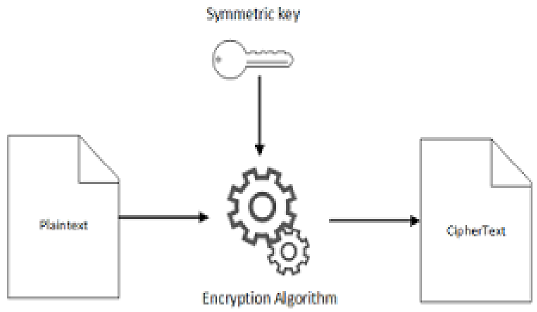


Fig.1: Symmetric key encryption process [11]

In our research paper, we are using Diffie Hellman algorithm. It is a symmetric key algorithm.

Diffie Hellman -

Diffie Hellman algorithm uses symmetric key cryptography. It generates two keys, which can be used for encryption and decryption. This algorithm itself is not used for encryption or decryption. Instead, it produces a set of keys to be used for encryption and encryption purpose. It is used to offer security to a variety of services over the internet. It is one of the most extensively used algorithms. The idea of this algorithm was offered by Ralph Merkle and was named after Whitfield Diffie and Martin Hellman. Diffie Hellman is one of the most primitive practical examples of public key exchange implemented within the field of cryptography [12].

Asymmetric cryptography - It is cryptography which requires two keys. One is a public key and other is a private key. The public key can be distributed. It may be given to trusted or untrusted. But private key needs to be kept undisclosed just like in case of symmetric key cryptography. Neither key will do both the functions.

Asymmetric cryptography has two primary use cases: authentication and confidentiality. Using asymmetric cryptography, messages can be signed with a private key, and then anyone with the public key is able to authenticate that the message was created by someone possessing the corresponding private key. This can be combined with a proof of identity system to know what entity (person or group) actually owns that private key, providing authentication.

The public key is used to encrypt the message while the private key is used to decrypt the message.

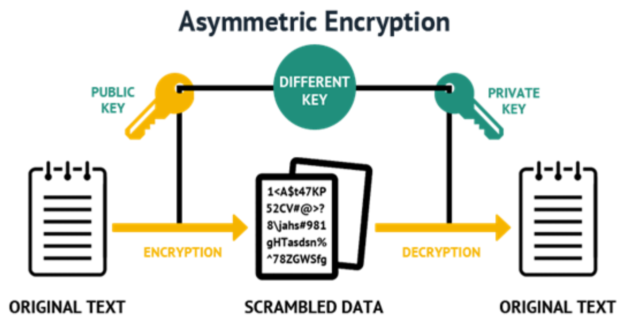


Fig.2: Asymmetric key cryptography

Many protocols like SSH, OpenPGP, S/MIME, and SSL/TLS uses asymmetric cryptography for encryption and digital signature functions. It is also used in software programs, like browsers, which need to establish a secure connection over an insecure network like the internet or need to validate a digital signature. Encryption strength is directly related to key size and doubling key length delivers an exponential increase in strength, although it impairs performance. As computing power enhances and more efficient factoring algorithms are discovered, the ability to factor larger numbers also enhances [4].

For asymmetric encryption to deliver confidentiality, integrity, authenticity, and non-reputability, users and systems need to be certain that a public key is authentic, that it belongs to the person or entity claimed and that it has not been tampered with or replaced by a malicious third party.

In our research work, we are using RSA algorithm.

RSA algorithm:

RSA consists of the public key and private key. The public key is used for encryption and the private key is used for decryption. The security of the RSA algorithm is dependent on the integer factorization problem. So the key selection is crucial in RSA. It takes two prime numbers and multiplies and applies some additional operation on it and generates two sets of keys. If anyone knows the factors after multiplying two prime numbers then encryption can easily break. [5]

Number theory behind RSA:

1. Prime number generation is easy- It easy to assume a random prime number of a given size.
2. Multiplication is easy - given p and q, it's easy to find their product, n=pq.
3. Factoring is hard- given such an n, it appears to be quite hard to recover the prime factors p and q.
4. Modular exponentiation is easy - given n, m, and e, it's easy to compute c= me mod n.
5. Modular root extraction, the reverse of modular exponentiation - is easy given the prime factors p and q, it's easy to recover the value m such that c= me mod n.

6. Modular root extraction is otherwise hard - given only n, e, and c, but not the prime factors, it appears to be quite hard to recover the value m. [9]

Existing rsa diffie- Hellman hybrid algorithm:
Step 1: choose two large prime numbers P and Q and random number A, B, and G, R.
Step 2: set A and B for Diffie Hellman key generation
Step 3: R and G are automatic generated constants.
Step 4: Calculate N=P*Q.
Step 5: find z = (p-1)*(Q-1)
Step 6: choose integer E, which can satisfy GCD (E, z) =1
Step 7: calculate D, where E*D mod z=1.
Step 8: Now calculate following as public number
Calculate C=G^A mod R, Y= G^B mod R
Step 9 : secret key K1 = Y^A mod R,
K2 = X^B mod R.
Step 10:encrypt message using RSA algorithm,
C1= (m^E)mod N.
Step 11: XOR between C1 and key K1,
S= C1⊕KI
Step 12: at receiver side XOR is between and key K2
C1=S⊕K2,
Step 13 : decrypt message using RSA algorithm
M= (C1^D)mod N.

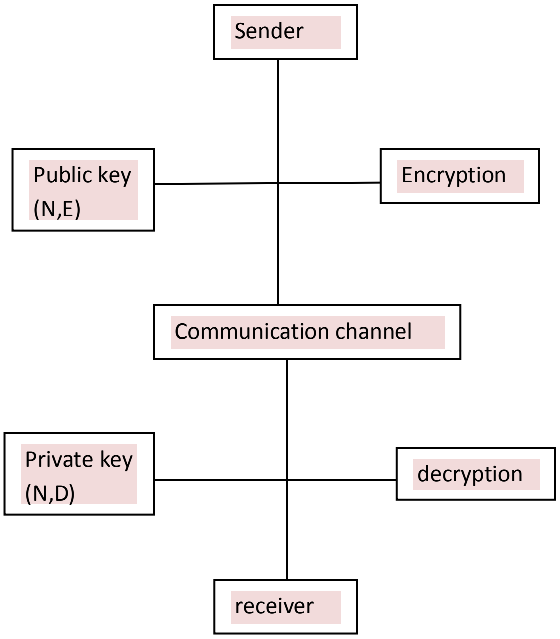


Fig.3: encryption decryption using existing RSA algorithm

III. PROBLEM DOMAIN

Though this hybrid algorithm utilizes benefits of both RSA and diffie Hellman, but still it needs improvement. RSA has certain limitations. One of them is that if any one of the values p, e or d values is known then other values can be calculated. So secrecy is still an issue [11]. Further this algorithm is time taking. We have worked to reduce the time taken by this algorithm.

IV. PROPOSED ALGORITHM

Our proposed algorithm is an improvement over the existing algorithm because it makes calculation more complicated by making calculation a little bit more complex. To make it less time taking , we are using XNOR gate instead or XOR gate.
Step 1: choose two large prime numbers P and Q and random number A, B, and G, R.
Step 2: set A and B for Diffie Hellman key generation
Step 3: R and G are automatic generated constants.
Step 4: Calculate N=P*Q.
Step 5: T=N*2
Step 6: find z = (p-1)*(Q-1)
Step 7: choose integer E, which can satisfy GCD (E, z) =1
Step 8: L= (E*4)-2
Step 9 : calculate D, where E*D mod z=1.
Step 10 : A=D+1.
Step 11: Now calculate following as public number
Calculate C=G^A mod R, Y= G^B mod R
Step 12 : secret key K1 = Y^A mod R,
K2 = X^B mod R.
Step 13:encrypt message using RSA algorithm,
C1= (m^((L+2)/4))mod (T/2).
Step 14: XNOR between C1 and key K1,
S= c1 XNOR KI
Step 15: at receiver side XNOR is between S and K2
C1- S XNOR K2,
Step 13: decrypt message using modified RSA algorithm
M= (C1^ (A-1)) mod (T/2).

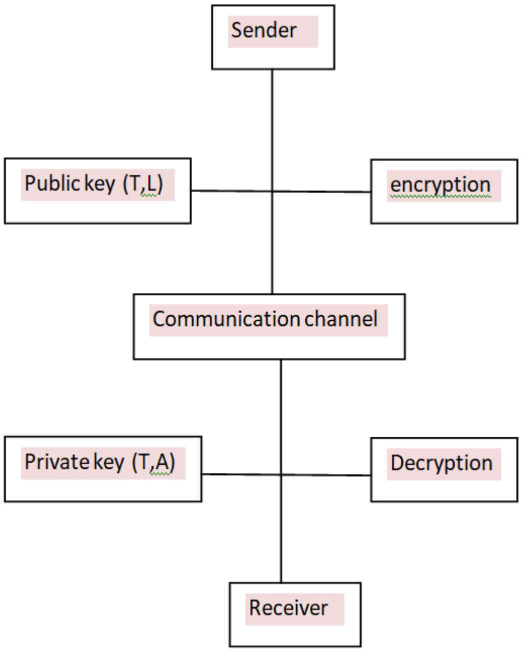


Fig.4: encryption decryption using modified RSA algorithm

For example, let us take an example in which we have to send message =88

Step 1: we have considered two prime number P=11 and Q=17
Step 2: we set a=5 and B=6 for Diffie Hellman key generation
Step 3: R and G are 2 and 997 respectively
Step 4: calculate N=PQ, N=11*17=187
Step 5: calculate T=N*2, T= 874
Step 6: find Z= (P-1) (Q-1), Z=160
Step 7: choose integer E, such that GCD (E, Z) =1, we choose E=7.
Step 8: calculate L= (E*4)-2, L=26
Step 9: calculating the following public keys for Diffie Hellman
X=G^A mod R, Y =G^B mod R,
X=2^5 mod 997=32 and
y=2^6 mod 997=64
Step 10: calculate secret key
K1=64^5 mod 997=740 and
K2 = 32^6 mod 997= 740
Step 11: encrypt message using modified RSA algorithm
C1 = m^ ((L+2)/4)mod (T/2)
C1= 88^ ((L+2)/4 mod (374/2) =11
Step 12: at the sender side XNOR between C1 and K1
S =11 XNOR740 = 751.
Step 13: at the receiver side XNOR between S and K2
C1= 751 XNOR 740=11
Step 14: decrypt the message using the modified RSA algorithm
M=C1^(A-1)mod (T/2)
M=11^23 mod (384/2)=88

V. IMPLEMENTATION

NET framework which is also known as .net is developed by Microsoft. It is a software development platform. It includes a framework class library. It provides language interoperability across several programming languages.

The GUI was developed using Microsoft asp.net framework.

We have used c# as the programming language.

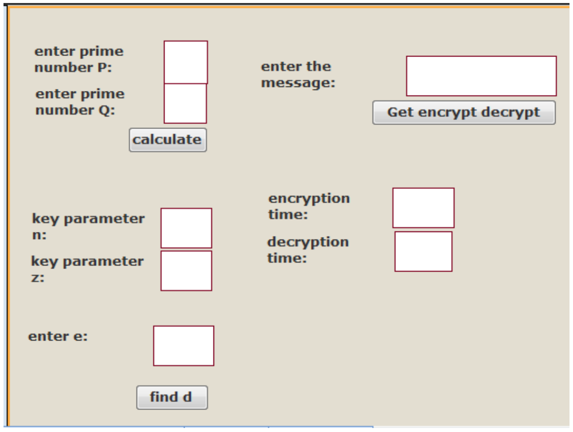
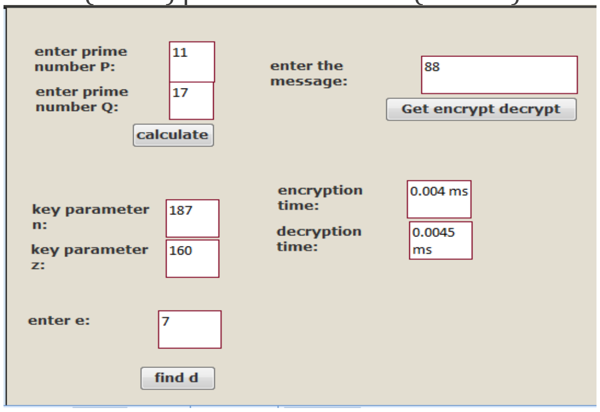


Fig.5: GUI view of proposed algorithm



enter prime number P: 11
enter prime number Q: 17
enter the message: 88
key parameter n: 187
key parameter z: 160
encryption time: 0.004 ms
decryption time: 0.0045 ms
enter e: 7

Fig.6: encryption and decryption time using proposed algorithm

VI. RESULT

Table 1 and Table 2 shows the encryption and decryption time of existing and proposed algorithm.

Exiting algorithm

Exiting algorithm			
P	**Q**	**E[ms]**	**D[ms]**
11	17	0.0036	0.0024
401	277	0.0040	0.0052
1447	941	0.0044	0.0068
5693	5791	0.0044	0.0072
6917	6997	0.0052	0.0072
23899	23909	0.0044	0.0085
36037	36017	0.0056	0.0085
215981	215983	0.0093	0.0401
317263	317267	0.0105	0.0413

Table 1: encryption and decryption time for the existing algorithm

Prososed algorithm			
P	**Q**	**E[ms]**	**D[ms]**
11	17	0.0034	0.0022

401	277	0.0038	0.0050
1447	941	0.0042	0.0067
5693	5791	0.0042	0.0070
6917	6997	0.0050	0.0071
23899	23909	0.0042	0.0084
36037	36017	0.0054	0.0083
215981	215983	0.0091	0.0400
317263	317267	0.0102	0.0412

Table 2: encryption and decryption for the proposed algorithm

VII. FUTURE WORK

Though our proposed algorithm provides dual benefits i.e. security and less time consumption, nevertheless there is the scope of improvement. Time complexity can be further reduced by making more changes to the algorithm. The less time consuming an algorithm is, the more acceptable it is in the emerging technical world.

VIII. CONCLUSION

A hybrid algorithm RSA Diffie Hellman is proposed in this paper. We have tried to make the existing algorithm more promising for this technical world. After going through the available resources of research, we found that security needs to improve. Further, we noticed that the time taken by the algorithm should be minimized. Our hybrid RSA Diffie Hellman algorithm provides more security. The original algorithm sends(N, E) as public and (N, D) as a private key. Once these values are known, the message sent over the communication channel can be easily decrypted. This leads to security issues. We have made the calculation a little bit more complex by introducing T, L, A instead of N, E, D respectively. This helped us to improve the cryptography technique, by enhancing security. Now, instead of sending actual private key and public key over the communication channel, we are sending the new values, T, L, and A . this adds another layer of security. Our algorithm is better than the existing hybrid RSA Diffie-Hellman because it ensures security by sending altered private key and public key, which makes this algorithm more secure by making it complex. It becomes difficult for any unauthorized user to obtain a public key and private key during the transmission over the communication channel Though the calculation is increased, but security is also increased. Security of data is the demand of every emerging technology. Nobody wants to compromise with security. Another improvement we have done in this paper is reducing the time taken by an algorithm to complete its execution. By replacing the XOR gate with XNOR gate, it was noticed that there was some reduction in time taken. This research paper is an approach to optimize the benefits of two well-known algorithm RSA and Diffie-hellman.

IX. REFERENCES

[1]. Mahalakshmi and Suseendran G., (2018). "An analysis of Cloud Computing issues on data Integrity, privacy, and its current solutions".

[2]. K David Raju, L Vijay Kumar, K Anthony Rahul Showry, B Lhoit Krishn ,(2018) ." techniques of providing data integrity in cloud computing".

[3]. Joseph Selvanayagam1, Akash Singh2, Joans Michael3, Jaya Jeswani4 (2018) ."secure file storage on cloud using cryptography".

[4]. https://searchsecurity.techtarget.com/definition/asymmetric-cryptography

[5]. Shreen Nisha, Mohammed Farik (2017), " RSA public key cryptography algorithm - a review", international journal of scientific and technology research volume 6.

[6]. Prabhat Kumar Panda , (2017). " A hybrid security algorithm for RSA cryptosystem".

[7]. Sultan Aldossary , William Allen , (2016). " Data Security, Privacy, Availability, and Integrity in Cloud Computing: Issues and Current Solutions".

[8]. Israt Jahan, Mohammad Asif, Liton Jude Rozario (2015), Improved RSA cryptosystem based on the study of number theory and public key cryptosystems. American Journal of Engineering research.

[9]. https://www.ibm.com/support/knowledgecenter/en/SSB23S_1.1.0.14/gtps7/s7symm.html

[10]. Miss. Renushree Bodkhe, Prof. Vimla Jethani, (2015) "Hybrid encryption algorithm based improved RSA and Diffie - Hellman".

[11]. Gaurav R. Patel, Prof. Krunal Panchal (2014), "Hybrid encryption Algorithm".

[12]. Mahima Joshi, Yudhveer Singh Moudgil, "secure Cloud Storage."

[13]. Sarthak R Patel, Prof. Khushbu Shah, Gaurav R Patel , "Study on Improvements in RSA algorithm"

Authors' Profile

Shivani Kaushik is pursuing M. Tech. from Subharti Institute of Engineering and Technology, Swami Vivekanand Subharti University, Meerut, India. She received her B. Tech Degree in computer science and Engineering from Uttar Pradesh Technical university, Lucknow, India. Her area of interest is cryptography.

Er. Anirudh kumar Tripathi is working as Assistant Professor in the department of Computer Science Engineering and I.T., Subharti Institute of Engineering and Technology, Swami Vivekanand Subharti University, Meerut, India.

Er. Pankaj Pratap Singh received his B. Tech Degree in Computer Science Engineering from Uttar Pradesh Technical University, Lucknow, India, in 2007 and M. Tech degree in Medical Image and Image Processing from Indian Institute of Technology Kharagpur, Kharagpur, India, in 2010. He is currently working as Assistant Professor in the Department of Information technology, Subharti Institute of Engineering and Technology, Swami Vivekanand Subharti University, Meerut, India. His research interests include IOT, Neural Network, Machine Learning, Deep Learning, Image Processing techniques, Cognitive Science, Computer Network and Data Mining techniques.

Er. Amit Kishor is working as Assistants Professor in the department of Computer Science Engineering and I.T., Subharti Institute of Engineering and Technology, Swami Vivekanand Subharti University, Meerut, India. Currently he is pursuing Ph. D. in Computer Engineering from Department of Computer Science and I.T., Sam Higginbottom University of Agriculture, Technology and Sciences, Allahabad.