ISSN : 2231-0711

Charanjeet Kaur et al | IJCSET(www.ijcset.net) | July 2015 | Vol 5, Issue 7,250-254

Hybrid Encryption Scheme for Hadoop Based Cloud Data Security

[1]Charanjeet Kaur
Punjabi University Patiala, Punjab (India)

[2]Er. Gurjit Singh Bhathal
Punjabi University Patiala, Punjab (India)

Abstract: Cloud computing is a concept in which services (Iaas, Paas, and Saas) are leased to the user as per imposition. Cloud computing entire data resides over a set of network resources; this data can be accessed through virtual machines like mobiles PC etc. Cloud computing reduces hardware, maintenance and installation cost. But security is major issue that prevents users for cloud computing. When we relocate data to cloud we use standard encryption technique to secure the data. But when we have to do computations on data hoarded on cloud then we have to decode data i.e. provide private key every time which is not a secure methodTrusted computing and security of services is one of the most challenging topics today and is the cloud computing's core technology that is currently the focus of international IT universe. Hadoop, as an open-source cloud computing and big data framework, is increasingly used in the business world, while the weakness of security mechanism now becomes one of the main problems obstructing its development. This paper first describes the hadoop project and its present security mechanisms, then analyzes the security problems and risks of it, pondering some methods to enhance its trust and security and finally based on previous descriptions, concludes Hadoop's security challenges.

Keywords-Security; Trust; Hadoop; Big Data; Mapreduce;

I. INTRODUCTION

So our main area of concern is "whether the confidential data kept on the central storage on cloud is secure or not".
We use various cryptographic techniques (symmetric or asymmetric) to encrypt our data so that our data reaches to destination safely. In private key cryptography if two parties wants to securely communicate then they should share a secret key .In order to encrypt and then decrypt the data, both parties must have same key. So all the secure communication is limited to those who have a pair of keys. But there is a problem with this technique that how to securely communicate to exchange key, if key has to change from time to time before securely transmitting the message. Public key cryptography solved this problem. In this security relies on hard mathematical problems and each party need a public key for encryption and private key for decryption [1].
Additionally two parties want to be sure about confidentiality and integrity. For each of these constraints appropriate solution is devised and implemented. However ,only a couple of months after the publication of RSA algorithm paper[2], Rivest et al. asked the question whether it is possible to work with encrypted data, without the need of decrypting it first[3].That question started the research for hybrid encryption systems. Why we needed this type of systems e.g. we have some confidential information and we send it to servers for some computation and we do not want to give private key to server .So hybrid encryption can help to preserve this privacy policy.
Nowadays it is very important to design strong encryption algorithms as the power of computers is growing day by day. Thus the hybrid model gives a better non linearity to the plain RSA Hence the possibility of an algebraic attack on the hybrid model is reduced. and as it is merged with AES there is better diffusion. Hybrid mode involves more computations as compared to AES or RSA alone hence; we can say that the encryption time for the hybrid model is much greater than the times for AES or RSA alone. Thus it can be inferred that the hybrid model will take longer time to be broken by cryptanalysis.
HADOOP:
To enhance privacy and data retrieval on Hadoop in cloud computing. Hadoop is an open-source software platform for distributed computing dealing with a parallel processing of large data sets. It has been widely used in the field of cloud computing.

Application Issue	Symmetric Encryption	Asymmetric Encryption
e-Commerce	Less secure	More secure
Ease of key management	Difficult	Easy
Signature non repudiation		Easy
Functionality	Difficult	Easy
High level security	less	More
Secure key transmission	Less secure	More secure
Key update	Difficult	Easy
Future growth	Less	More

II. RSA ALGORITHM

RSA: (Rivest, Shamir, Adleman) RSA is an **algorithm** used by modern computers to encrypt and decrypt messages. It is an asymmetric cryptographic **algorithm**. Asymmetric means that there are two different keys. This is also called public key cryptography, because one of them can be given to everyone. The other key must be kept private
The RSA scheme was developed by three academics Ron Rivest, Adi Shamir, and Leonard Adleman at MIT in 1978. Their proposal is now known as the RSA algorithm, named for the last initials of the researchers. RSA shares many similarities with the Diffie-Hellman algorithm in that RSA is also based on multiplying and factoring large integers. However, RSA is significantly faster than Diffie-Hellman, leading to a split in the asymmetric cryptography field that refers to Diffie-Hellman and similar algorithms as Public Key Distribution Systems (PKDS) and RSA and similar algorithms as Public Key Encryption (PKE).
PKDS systems are used as session-key exchange mechanisms, while PKE systems are generally considered fast enough to encrypt reasonably small messages.

However, PKE systems like RSA are not considered fast enough to encrypt large amounts of data like entire file systems or high-speed communications line**.** RSA, Diffie Hellman and other asymmetric algorithms use much larger keys than their symmetric counterparts. Common key sizes include 1024 bits and 2048-bits, and the keys need to be this large because factoring, while still a difficult operation, is much easier to perform than the exhaustive key search approach used with symmetric algorithms. The relative slowness of public key encryption systems is also due in part to these larger key sizes. Since most computers can only handle 32-bits of precision, different "tricks" are required to emulate the 1024-bit and 2048-bit integers [19].
RSA encryption can also provide authentication services, something that symmetric key encryption cannot do. RSA" stands for Rivest, Shamir, and Adleman, the names of its inventors. RSA is the symmetric key algorithm that is easiest to implement, and it's the best understood. The RSA cryptosystem is a public key cryptosystem that offers both encryption and digital signatures, which provides authentication. The RSA algorithm is based on the difficulty of factoring a number, x, that is the product of two large prime numbers. The two large prime numbers may include up to 200 digits each[2].

RSA public key algorithm RSA is used in the ISAKMP/Oakley framework as one of the possible authentication methods. The principle of the RSA algorithm is as follows:

1. Take two large primes, p and q.
2. 2. Find their product n = pq; n is called the modulus.

3. Choose a number, e, less than n and relatively prime to (p-1)(q-1), which means that e and (p-1)(q-1) have no common factor other than 1.
4. Find its inverse, d mod (p-1)(q-1), which means that ed = 1 mod (p-1)(q-1), e and d are called the public and private exponents, respectively. The public key is the pair (n,e); the private key is d. The factors p and q must be kept secret or destroyed [20].

A simplified example of RSA encryption is:

- Suppose Alice wants to send a private message, m, to Bob. Alice creates the cipher text c by exponentiating:
  c = me mod n

Where e and n are Bob's public key.

- 2Alice sends c to Bob.
- 3 To decrypt, Bob exponentiates:
  m = cd mod n

And recovers the original message; the relationship between e and d ensures that Bob correctly recovers m. Because only Bob knows d, only Bob can decrypt the cipher text.
A simplified example of RSA authentication is:

- Suppose Alice wants to send a signed message, m, to Bob. Alice creates a digital signature s by exponentiating:
  s = md mod n

Where d and n belong to Alice's private key.

- She sends s and m to Bob.
- To verify the signature, Bob exponentiates and checks if the result, compares to m:
  m = se mod n

Where e and n belong to Alice's public key.[12]

The following steps outline RSA key generation:

Generate two RSA primes, p & q, and compute n= pq and U= (p-1)(q-1).

- Select a random integer e from the interval (1,U)such that gcd (Edu)= 1.
- Select an integer d from the interval (1,U)such that ed = 1modU.
- The public key isn, eiand the private key is (n,d).

The following steps outline RSA encryption:

- Let m be the message represented as a number in the interval [0, n-1].
- Compute $c=m^e$modn where c is the cipher text.

We can decrypt RSA with the following:
Compute $m=c^a$ modn where m is the original message [9]

AES ALGORITHM:

The **Advanced Encryption Standard** (**AES**), also referenced as Rijndael (its original name), is specification for the **encryption** of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.
The base for AES is Rijndael cipher which was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen. They submitted a proposal to NIST during the AES selection process. Rijndael belongs to that family of ciphers which have different key and block sizes.
AES is an improvement in the Data Encryption Standard (DES), which was published in 1977. In the algorithm described by Advanced Encryption Standard, the same key is used for both encrypting and decrypting the data. So it means AES is a symmetric-key algorithm.
AES is combination of both substitution and permutation, and is based on a design principle known as a substitution-permutation network. AES is fast in both software and hardware. AES does not use a Feistel network, unlike its predecessor DES. AES which is a variant of Rijndael cipher, has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. The Rijndael specification *per se* is specified with block and key sizes which can be any multiple of 32 bits, both with a maximum of 256 bits and a minimum of 128 bits.
Although some versions of Rijndael have a larger block size and have additional columns in the state, but AES operates on a 4×4 column-major order matrix of bytes, termed the *state*. Most of the AES calculations are done in a special finite field.
The number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text are specified by the key size used for an AES cipher. The number of cycles of repetition is as follows:

- for 128-bit keys - 10 cycles of repetition.
- for 192-bit keys - 12 cycles of repetition.
- for 256-bit keys - 14 cycles of repetition.

Each round has several processing steps, which contain four similar but different stages, which also includes the

one that depends on the encryption key itself. To transform cipher text back into the original plaintext using the same encryption key, a set of reverse rounds are applied.

III. CLOUD COMPUTING, ITS SECURITY ISSUES AND HYBRID ENCRYPTION

Definition [11]:By cloud computing we mean: The information technology (IT) model for computing, which is composed of all the IT components (hardware, software, networking, and services)that are necessary to enable development and delivery of cloud services via internet or a private network.

Cloud computing is basically an idea that data and programs can be stored centrally on cloud and then can be accessed from anywhere in the world through internet using PC's, laptops or phones.

In cloud computing available service models are:

1. Infrastructure as a service
2. Platform-as-a-service
3. Software as a Services

Four deployment models have been identified for cloud architecture solutions, described below:

1. Private cloud
2. Community cloud
3. Public cloud
4. Hybrid cloud

CLOUD COMPUTING SECURITY ISSUES

Cloud computing provides us an advantage that data can be stored centrally in the cloud and can be accessed from anywhere and anytime. This brings many advantages, including data ubiquity, flexibility of access, and resilience. Since cloud computing keeps data outside the control of owner so it introduces security issues also .

This facility raised various security questions like privacy , confidentiality, integrity etc.

Some main issues of cloud computing are:

1. Access control: Cloud based data is usually accessed by many insecure protocols and API's over the public network.
2. Data location: User doesn't know the way user data is stored, where it is stored, data recovery, data encryption and data integrity problem.
3. Authentication: User cannot be classified as no. of users as user changes dynamically as well as user use different resources. The cloud is typically a shared resource, and other sharers (called tenants) may be attackers.
4. Data security: In cloud computing model, a cloud service provider has a major role to play. He has the right to access anything which is stored on cloud.
5. Data recovery: Don't know where data is located so Cloud service provider must tell what will happen to data in case of disaster and how it will be recovered.

Now the question arise "How to keep client private data confidential?."

Therefore, the genuinely unique challenge posed by cloud computing security boils down to just one thing: the data in the cloud can be accessed by the cloud provider. The cloud provider as a whole (or its employees individually) can deliberately or inadvertently disclose customers' data. Moreover, the cloud provider may have subcontractors (typically, a "software-as-a-service" provider will subcontract to an "infrastructure-as-a-service" provider), and the subcontractors may also have access to the data. This paper addresses the question of how a customer could secure its data from malicious or negligent cloud providers.

So we can protect data by encrypting it before sending it to the cloud provider, But to do calculations we have to decrypt it every time. Until now it was impossible to encrypt data and trust a third party to keep them safe and able to perform distant calculations on them, So to allow cloud providers to perform computation so encrypted data without decrypting them requires using the cryptosystems based on Homomorphic encryption.

IV HYBRID ENCRYPTION:

Hybrid Encryption systems are used to perform operations on encrypted data which is kept on the hadoop.in this we combine two algorithm for better data encryption. without knowing the secret key (without decrypted), the client is the only possessor of the secret key. When we decrypt the result of the operation, it is the same as if we had carried out the calculation on the raw data.

A **hybrid encryption** scheme uses public-key **encryption** to **encrypt** a random symmetric key, and then proceeds to **encrypt** the message with that symmetric key. The receiver decrypts the symmetric key using the public-key **encryption** scheme and then uses the recovered symmetric key to decrypt the message

V PRORPOSED SYSTEM

Our work we use VMware machine require collecting evidences about the users who have registered in our cloud, authenticated users who have logged in and unauthorized users who tries log in .All information about these users is maintained in log tables. Log table contain fields as shown in following table 1.

Fields	Description
User name	Username for Hadoop user login
Password	Password for login in hadoop
Ipaddress	It will tell the location and network from where person has logged in. It is also used for authentication purpose
Location	We can get location of person through ipaddress
Login time	time at which person has logged in Hadoop .
Logout time	Time at which person has logged out. VMware Machine
Time spent	Total time spent by person
PSCP	**Pscp** is a program used to swap files (encrypted) between a server and another computer.

Table1: Fields in Log table

So this log table is useful in knowing the nature of users. We can detect intruders if they try to access through new added security layers of Macaddress and IP address. We have applied hybrid encryption Scheme encryption on simple RSA algorithm. We have compared this combine RSA & AES algorithm on basis of some performance measures like encryption time, decryption time. At different CPU frequency.

We have compared both projects on basis Different CPU Frequency, & CPU RAM Encryption time, Decryption time with Hybrid Encryption Scheme Flowchart of proposed work is shown in fig. 1

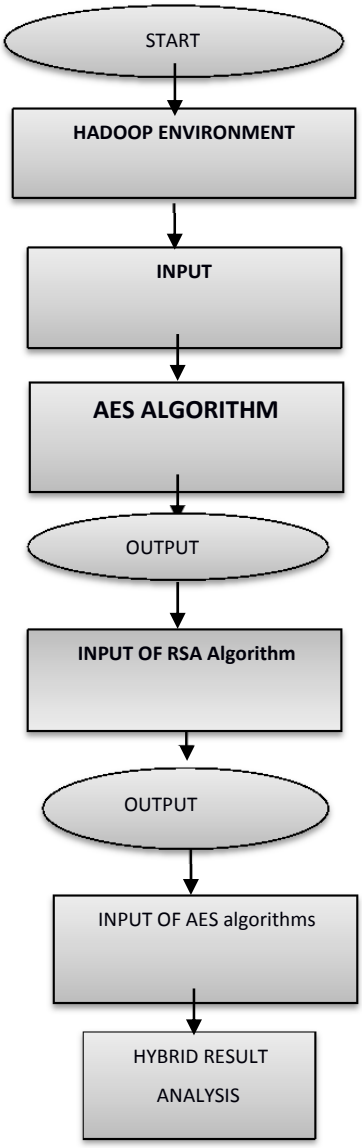


Figure 1: Proposed work flowchart

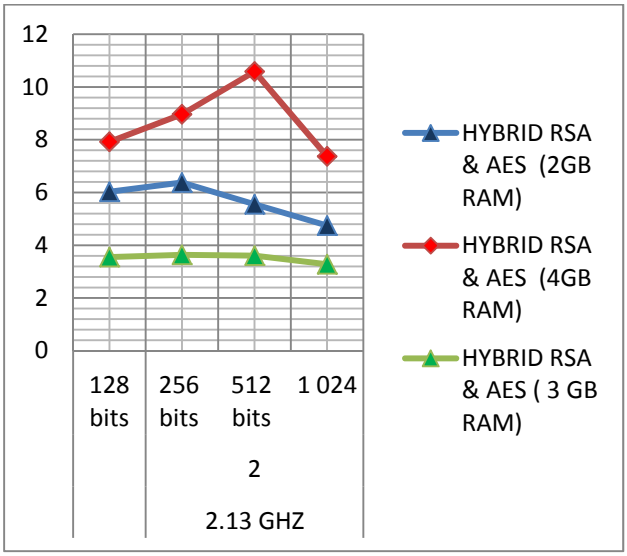
2. Encryption time: Time taken by server to encrypt any file. Its unit is nano seconds.
Encryption time = Encryption end time - Encryption start time.
3. Decryption time: It is time taken by server to decrypt any file. On Hadoop.
.

V. EXPERIMENTAL RESULTS

This system has been designed and implemented in java language to enhance security At Hadoop between client and server. The table below represents experimental results of comparison of both the algorithms. We have Input File at hadoop corresponding factors for both the algorithms in shown in below table 2 and graphs.

Number of Processor	2	**Memory Size**
HYBRID RSA & AES	6.0264	128 bits
(2GB RAM) 2.13 GHZ	6.3782	256 bits
	5.5537	512 bits
	4.7488	1 024 bits
(3GB RAM) 2.30 GHZ	3.554	128 bits
	3.6371	256 bits
	3.609	512 bits
	3.278	1 024 bits
(4GB RAM) 2.40 GHZ	7.9189	128 bits
	8.963	256 bits
	10.5849	512 bits
	7.3656	1 024 bits

Table 2 : Comparison of RSA & AES On different CPU frequency



.

Figure 2: Comparison of Encryption & Decryption time on Different CPU

Graph in fig 2 represents, Different CPU frequency and diif time taken to encrypt and decrypt encryption time using proposed algorithm (RSA & AES) which is lesser than RSA encryption time

NUMBER OF	**PROCESSOR** 1	**MEMORY SIZE**
HYBRID RSA & AES		
3 GB RAM 2.30 GHZ	3.554	128 bits
	3.6371	256 bits
	3.609	512 bits
	3.278	1 024 bits
	4.2214	1096 bits
	PROCESSOR 2	
HYBRID RSA & AES	5.6889	128 bits
4 GB RAM 3GB RAM 2.30 GHz 1.80 GHZ	5.1254	256 bits
	4.9289	512 bits
	5.517	1 024 bits
	5.2199	1096 bits

Table 3: Comparison of RSA & AES On different CPU RAM

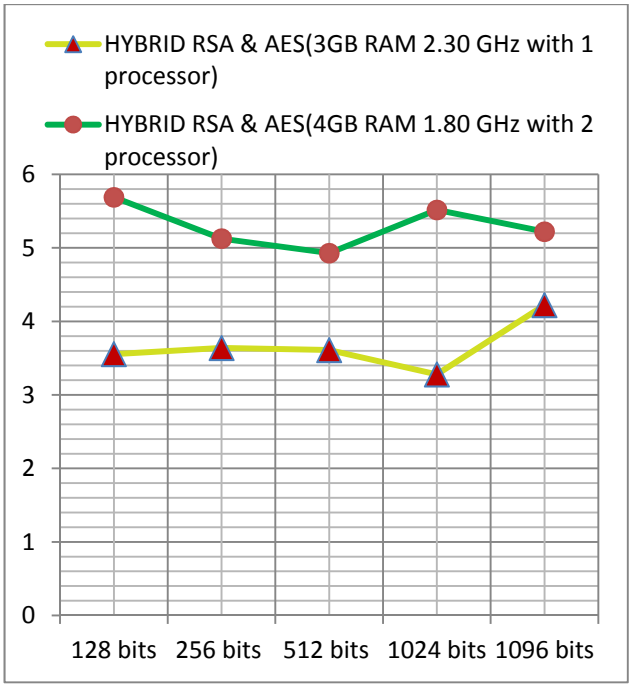


Figure 3: Comparison of Encryption & Decryption time on Different CPU

Comparison Performance of hybrid RSA & AES at different CPU frequency.

Using proposed algorithm (RSA AES) which is more than DES throughput of 2931 bytes/sec.

The criterion in performance of every algorithm is that encryption time, decryption time, be depend on CPU configuration and Memory Size. Large the Memory Size. That encryption time, decryption time should be less and throughput should be more.

During comparison of RSA & AES this criteria is achieved and hybrid OF both Algorithm give better results.

VI. Conclusion And Future Scope

This paper represents “A Framework for secure Hadoop in cloud computing based on hybrid encryption scheme “, security features of the cloud has been enhanced.. Cloud computing is an emerging area within the field of information technology. Security of data is main issue that hamper its growth. . Nowadays it is very important to design strong encryption algorithms as the power of computers is growing day by day. Thus the hybrid model gives a better non linearity to the plain AES and as it is merged with RSA, there is better diffusion. Hence the possibility of an algebraic attack on the hybrid model is reduced. Hybrid mode involves more computations as compared to AES or RSA alone hence; we can say that the encryption time for the hybrid model is much greater than the times for AES or RC4 alone. Thus it can be inferred that the hybrid model will take longer time to be broken by cryptanalysis

5.2 Future Work

Every organization has a huge amount of confidential data. This security of this data is the primary concern for the existence of any organization. No organization can afford the loss of even a small part of its data. It may result in a loss of millions or billions of money. Nearly 80% budget of IT companies is spent on Information security. Data can never be said to be 100% secure. Each time a new security mechanism is developed, its cracks also get developed after some time. So the security mechanisms can be compared to passwords which need to be changed time to time. Security Mechanisms must be upgraded after a short span of time in order to avoid loss of confidential data and harsh consequences thereafter.

References

[1] J. Katz and Y. Lindell (2007), “Introduction to Modern Cryptography (Chapman& Hall/Crc Cryptography and Network Security Series)”, Chapman &Hall/CRC.

[2] R. Rivest, A. Shamir, and L. Adleman (1978), “A method for obtaining digital signatures and public-key cryptosystems,” Communications of the ACM, vol. 21, pp. 120–126.

[3] R. L. Rivest, L. Adleman and M. L. Dertouzos (1978), “On data banks and privacy homomorphism,” Foundations of Secure Computation, Academia Press, pp. 169–179.

[4] M. E. Hellman (1979), "DES will be totally insecure within ten years", lEEE Spectrum.

[5] Alani, M. M. (2010), "A DES96 - improved DES security ", 7th International Multi-Conference on Systems, Signals and Devices.

[6] Seung-Jo Han , Heang-Soo Oh , Jongan Park (1966)," IEEE 4th International Symposium on Spread Spectrum Techniques and Application Proceedings ".

[7] Manikandan. G, Rajendiran.P, Chakarapani.K, Krishnan.G, Sundarganesh.G (2012),"A Modified Crypto Scheme for Enhancing Data Security", Journal of Theoretical and Advanced Information Technology.

[8] Shah Kruti R., Bhavika Gambhava (2012),"New Approach of Data Encryption Standard Algorithm", International Journal of Soft Computing and Engineering (IJSCE).

[9] William Stallings (2005), “Cryptography and Network Security Principles and Practices", Prentice Hall.

[10] Sung-Jo Han, Heang-Soo Oh, Jongan Park(1996), "The improved Data Encryption Standard (DES) Algorithm",Department of Electronic Engineering, Chosun University. South Korea, IEEE.

[11] Vic (J.R.) Winkler (2011), “Securing the Cloud, Cloud Computer Security, Techniques and Tactics”, Elsevier.

[12] Pascal Pailler (1999), “Public-key cryptosystems based on composite degree residuosity classes”. In 18th Annual Eurocrypt Conference (EUROCRYPT'99), Prague, Czech Republic .

[13] Julien Bringe and al.(2007), “An Application of the Goldwasser-Micali Cryptosystem to Biometric Authentication”, Springer-Verlag.

[14] R. Rivest, A. Shamir and L. Adleman (1999), “A method for obtaining digital signatures and public key cryptosystems”. Communications of the ACM, 21(2) :120-126, 1978. Computer Science, pages 223-238, Springer.

[15] Taher ElGamal (1985), “A public key cryptosystem and a signature scheme based on discrete logarithms”. IEEE Transactions on Information Theory.

[16] https://simple.wikipedia.org/wiki/RSA_(algorithm

[17] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

[17] www.cs.umd.edu/~jkatz/gradcrypto2/NOTES/lecture4.pdf