Enhancing Data Security Using AES Encryption Algorithm in Cloud Computing

Snehal Rajput, J.S. Dhobi and Lata J. Gadhavi

Abstract Cloud Computing offers computation as per utility of the customer and hence it is known as utility computing. This model is attractive mainly for business oriented people because it reduces total cost of operation, maintainance cost, increases return of investment. But the only thing that is impeding popularity of cloud computing is security issues. This paper discuss about AES encryption algorithm (RIJNDAEL) that secure data stored on cloud. This method is more efficient than DES which is symmetry based algorithm and offers 56 bits key size whereas RIJNDAEL algorithm is asymmetry and offers 128 bits key and 128 bits blocks. RIJNDAEL is block cipher algorithm and is secure against cryptanalytic attacks. It is versatile, which means it can be implemented on different working environment efficiently, its key agility is good which means setup time of key is less and this algorithm is easy to be understood and using it. Rijndael requires less memory and hence make it is well suited for environments which have less space such as 8 bit micro-processor, also it shows marvellous performance in terms of software and hardware implementation.

Keywords AES encryption · RSA encyption · Blowfish · DES encryption · ISO · IETF · NIST · IEEE

S. Rajput (✉) · J.S. Dhobi
Government Engineering College, Gandhinagar, India
e-mail: snehalrajput89@gmail.com

J.S. Dhobi
e-mail: jsdhobi@gecg28.ac.in

L.J. Gadhavi
Saffrony Institute of Technology, Mehsana, India
e-mail: lata.gadhvi@saffrony.ac.in

S.C. Satapathy and S. Das (eds.), *Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems: Volume 2*, Smart Innovation, Systems and Technologies 51, DOI 10.1007/978-3-319-30927-9_14

1 Introduction

Clouds are large pool of virtualized resources which comprises of hardware, platforms, and software. These resources are dynamically allocated among user to balance the load and to utilise resources optimally [1]. It offers remarkable features such as an illusion of infinite computing resources, pay- as-much-use model, elimination of any frontal commitment, self service interface, and elastic capacity, resources are abstracted and virtualised [2]. Cloud computing services are of three types:

Infrastructure as a Services (IaaS), Platform as a Services (PaaS), Software as a Service (SaaS) [3]. Infrastructure as a services provides various infrastructure to the user such as data server, storage, firewall, network and hardware. Various example are Amazon EC2, Simple Queue Service, Simple Storage Service, VPC Service etc. Platform as Services provides framework, platforms to develop our own applications such as Microsoft Azure, GoogleApp Engine, Force.com. Software as a services provides application as service to user such as video conferencing, office suits, social networking. Various examples are Salesforce, SQLAzure, GoogleApp etc. Cloud computing deployment models are: public cloud, private cloud, and hybrid cloud and community cloud.

Public Cloud: It is available to all end users developed by Cloud service provider. Services are charged as per usage basis.
Private Cloud: It is developed by particular Organisation for their personal use.
Hybrid Cloud: It is use to make certain organisation scalable. Here any private cloud can be extend to public cloud.
Community Cloud: Here numerous organisation combine to form their private cloud called community Cloud.

The biggest advantage of cloud is cost saving which pulled business people toward cloud computing and the biggest disadvantage of cloud is security. Cloud is booming technology because it contain more positive aspects compare to negative aspects. Some of the advantages are: User whenever needed can extend the resources as per their requirement, the maintenance cost is almost negligible, it is scalable. Cloud offers auto scaling and load balancing of workload. Cloud service provider guarantees QoS to the user through Service Legal Agreement(SLA). It is fault tolerance as replica of servers are installed at various site. Some of the challenges in cloud computing is to secure data stored on cloud, network latency, statelessness. The worrisome challenges is security of data stored on cloud, which need to be protected against various attack using some algorithm. Once stored data user himself do not know where his data is stored. Some cloud service provider uses one time authentication method i.e. username, password to authenticate user; some uses two way authentication or multiway authentication and some uses encryption techniques for server security. Using cryptographic technique, user can protect their

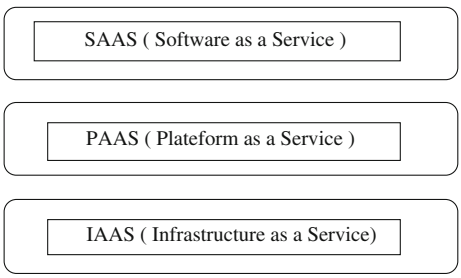

Fig. 1 Cloud deployment model

information from unauthorized access. Rijndael algorithm when implemented on cloud computing will make data stored on cloud more secure and speedy to access.

In 1997 several international organisations (ISO, IEEE and IETF) called for the implementation of AES algorithm. Later on, in 1999 NIST published five finalist: RC6, Rijndael, Serpent MARS, and Twofish. After second round Rijndael came out best among them when compare in terms on security, performance [2], cost, simplicity, versatility, key agility and hardware implementation. NIST concluded Rijndael with the following statement: Rijndael consistently showed remarkable performance both in terms of hardware and software on various computing environments. Its key setup time and key agility is excellent. As it requires very low memory, it is very much suitable for restricted space environments,without affecting its performance (Fig. 1).

2 Related Work

Cloud computing focuses on utility, and deployes computing resources on demand. Cloud provider must ensure that the all the resources provided by them i.e. infrastructure, platform, software, as well as client information is secure and customer must ensure that cloud provider has implemented proper algorithm which guarantees data security on cloud. This paper focus on securing data from threats without hindering performance. Also this paper shows the comparison of various well known algorithms.

Encryption is the technique to secure sensible data from the threats while transferring data onto cloud. Encryption technique can be asymmetry or symmetry.

In Asymmetry algorithm, different key are required during encryption and decryption whereas in symmetry algorithm same key is used during encryption and decryption time. Due to generation of different key at encryption and decryption time, asymmetric type of algorithm are much slower in compare to symmetric type of algorithms. Again, symmetric algorithm can be of block cipher or stream cipher.

In block cipher, ciphering of data occur blockwise while in stream cipher, ciphering of data occur bit wise.

2.1 Digital Signature Along with RSA Encryption

Generally *digital signature along with RSA encryption* is used to secure financial transactions, documents. RSA is most widely used asymmetric algorithm proposed by Ron Rivest, Adi Shamir, and Leonard Adleman. Here digital signature will summarize the document into message digest. This algorithm goes into two steps, first key generation through RSA algorithm and in second step preparing message digest. During encryption sender will encrypt it using its private key and receiver will decrypt it with its public key and get the message digest and again using hashing function plaintext can be generated. The main drawback of RSA algorithm is that, it is too slow.

2.2 DES Algorithm

It is a symmetry key Block cipher algorithm. At encryption site, it use 56 key size with plaintext of 64 bit and generates 64 bit ciphertext, at the decryption side it uses 64 bit ciphertext and generate 64 bit plaintext using same 56 bit of key size. The encryption process consists of two permutation viz. initial permutation and final permutation and 16 Fiestel cipher round. Here while encryption each block is XORed with previous block. The first block XORed with 64 bit vector called as *Initialisation vector.* This method is not viable now-a-days and is not secure due to 56 bit key size. It uses iterative round key having 16 round but since the same round transformation is used in each and every round, we can conclude that the DES algorithm has only one round transformation, which is easy to crack. Hence we can say that DES algorithm is not secure [4].

2.3 Single Sign-on Algorithm

Single sign-on algorithm provides a single interface to the users to access a group of software or resources. Here SSL or SAML can be used [5].

(i) Password Manager Agent (PMA): is installed on browsers of client as an extension. The main function is to communicating with the single interface onto cloud server for simultaneously sign on various SaaS.
(ii) Single Sign On SaaS Application (SSOSA): It is an application that manages usernames and passwords of users, encrypts those data and stores into PCS (password cloud server), stores keys into key cloud server, decrypts usernames and passwords, connects to various cloud computing application and serve request of users.
(iii) SSL: Secure socket layer to data transmission between PMA and SSOSA

(iv) Advanced Encryption Standard (AES) is used by SSOSA for encrypting information and storing it onto password cloud server.

2.4 Elliptical Curve Cryptography

It is a public key cryptography Algorithm, first proposed by Neal Koblitz and Victor Miller. Here each user has its own pair of private and public key. In any plane, an Elliptic curve on a field say 'f' consists of set of points (Xi, Yi).

The Standard equation of ECC is given by

$Y^2 = x^3 + ax + b$ where a and b are parameter [6].

The crux of ECC Algorithm is the of computation of new points on the curve and then its encryption to be share among users as information. Group Operator is used to find P which is one of the point on the curve. The computation goes as P+P, P+P+P, ……This method can be use for authentication of users using key agreements.

Public key of (Y_a) User A: X_a + P (where X_a is private key of A and P is some point) X_a = k1

Public key of (Y_b) User B: X_b + P (where X_b = k2 is private key of B and P is some point). Now both will exchange their keys. A calculates the session key by $K_a = X_a \times Y_b = k1 \times k2 \times P$.

B calculates the session key by $K_b = X_b \times Y_a = K2 \times kl \times P$ which means both KA and KB are same.

Other numeral encryption techniques are double DES, triple DES, MD, MD-5, blowfish, SHA, Elliptic Curve Cryptography (ECC) system.

3 Proposed Algorithm

AES algorithm can be classified into two type: First,128-bit plain text block paired with 128-bits key block and secondly,128-bit plain text block paired with 256-bit key block. Mostly 128-bit plain text block paired with 128-bits key block are widely used we will examine such case; The minimum number of round and the maximum round are 10, 14 respectively. This algorithm make uses of key alternative block cipher. Rijndael is similar to DES but only thing differs is that latter involves only bits in operation against entire block. The only difference between AES and Rijndael is that AES uses fix 128 bit block sizes, and the key lengths differs to 128, 192 or 256 bits whereas in Rijndael algorithm the block size in bits and the key size can be any multiple of 32 bits, with a minimum length of 128 bits and a maximum length of 256 bits.

3.1 Procedure [7]

The round transformation consists of four different transformations. The C code is as follow:

```
Round(State,RoundKey)
{
ByteSub(State);
ShiftRow(State);
MixColumn(State);
AddRoundKey(State,RoundKey);
}
```

The final round differs slightly as,

```
FinalRound (State,RoundKey)
{
ByteSub(State);
ShiftRow(State);
AddRoundKey(State,RoundKey); }
```

In Rijndael algorithm, *State* is defined as any intermediate round result. The State can be defined as a array of bytes which have four rows and number of columns (Nc) is equal to the (block length divide by 32). The Cipher Key is similarly defined as a array consists of four rows and number of columns of the Cipher Key (Nk) is equal to the (key length divide by 32).

The *ByteSub Transformation* functionoperates on each state independently and is a non-linear byte substitution where multiplicative inverse is applied first and affine function is done on it. Here each state is mapped to S-box and required array is obtained.

The *ShiftRow transformation* function shifts each row by some offset. First row is not shifted, second row is shifted by s1 byte and third row by s2 byte…

For example:

State array	New state array
W 1 Q 4	W 1 Q 4
A B C D	B C D A
0 P Q R	Q R 0 P

The MixColumn transformation: here state as considered as polynomial and multiplied with certain fix polynomial. Let C(x) = B(x) XOR A(x), where B(x) if fix matrix (Table 1).

Table 1 Time required in an AMD K7-700 processor (Per round, Using 100 k rounds) [10]

	DES (64, 64)	DES (64, 128)	Rijndael (128, 128)
Ciphering	3.4 µs	6.9 µs	35.8 µs
De-ciphering	3.5 µs	7.0 µs	36.0 µs

$$\begin{pmatrix} C0 \\ C1 \\ C2 \\ C3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 2 & 1 \\ 4 & 3 & 1 & 2 \end{pmatrix} \text{XOR} \begin{pmatrix} A1 \\ A2 \\ A3 \\ A4 \end{pmatrix}$$

The Round Key addition: Here XOR operation is performed between state and key blocks.

4 Limitation

Speed is less due to s-box generation [8]. Also there is few limitations of the ciphering technique with its inverse such as:

- When comparing Encryption with the decryption. Decryption is not implemented much on a smart cards because it requires more code and more CPU cycles for execution (Still it is faster than other algorithms.)
- While implementing this algorithm in software, the encryption and its decryption implementation codes are different.
- While implementing this algorithm on hardware, the decryption technique partially re-use the encryption circuitry, hence this will add further cost (Table 2).

Table 2 Time required in 8051 Micro-controller (Per round, Using 100 rounds) [10]

	DES (64, 64)	DES (64, 128)	Rijndael (128, 128)
Ciphering	2.8 ms	6.1 ms	28.8 ms
De-ciphering	2.7 ms	6.0 ms	28.0 ms

5 Comparison Between AES, DES and RSA [9]

Factors	AES (Rinjdael)	DES	RSA
Key size	128, 192, 256 bits	56 bits	1024–4096 bits
Block size	Any multiple of 32 bits	64 bits	512 bits or larger
Ciphering and deciphering key	Same key	Same key	Same key
Encryption and decryption	Faster	Moderate	Slower
Power consumption	Less	Less	High
Security	Excellent	Moderately secure	Least secure
Hardware and software algorithm	Faster compare to DES, RSA	Faster	Slow
Attacks vulnerabilities	Brute force attack	Brute force attack, Linear and differential cryptanalysis attack	Brute force attack
No. of rounds	10 or 12 or 14	16 Rounds	1 Rounds

6 Conclusion

We have seen numerous ciphering algorithm and have compared Rijndael algorithm performance with them and have found that Rijndael algorithm can be implemented in small space environment with remarkable performance, also it is secure against Numerous attack.

References

1. IBMCloudArchitectureforCrete052011b.pdf
2. The Rijndael's Algorithm. http://www.esat.kuleuven.ac.be/~rijmen/rijndael/
3. A secure data access control method using AES for P2P storage cloud. IEEE Sponsored 2nd International Conference on Innovations in Information, Embedded and Communication Systems (ICJJECS) 2015
4. http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf
5. Shimbre, N., Deshpande, P.: Enhancing distributed data storage security for cloud computing using TPA and AES algorithm
6. Daernen, J., Rijrnen, V.: The design of Rijndael algorithm

7. Biham, E.: A note on comparing the AES candidates. In: Proceedings of the 2nd AES Candidate Conference, Rome, pp. 85–92, 22–23 March 1999
8. Web & security. Glob. J. Comput. Sci. Technol. Netw. Online ISSN: 0975–4172 & Print ISSN: 0975-4350
9. Penchalaiah, N. et al.: Int. J. Comput. Sci. Eng. (IJCSE) **02**(05), 1641–1645 (2010)
10. Daemen, J., Rijmen, V.: AES submission document on Rijndael. Version 2, http://csrc.nist.gov/CryptoToolkit/aes/rijndael/Rijndael.pdf. Sept 1999