International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 02 | Feb-2016 www.irjet.net p-ISSN: 2395-0072

Secure Data Storage and Forwarding in Cloud Using AES and HMAC

Kadwe Yugandhara[1], Jadhav Ashwini[2], Pagar Pooja[3], Patil Suchita[4],Prof.J.S.Pawar[5]

[1] *Student, Computer Engineering, PVG COE Nashik, Maharashtra, India*
[2] *Student, Computer Engineering, PVG COE Nashik, Maharashtra, India*
[3] *Student, Computer Engineering, PVG COE Nashik, Maharashtra, India*
[4] *Student, Computer Engineering, PVG COE Nashik, Maharashtra, India*
[5] *Professor, Computer Engineering, PVGCOE Nashik, Maharashtra, India*

Abstract - ***Cloud computing is a more common term for anything that involves delivering hosted services over the internet and storage capacity as a service to users. Cloud storage, as a subservice of infrastructure as a service (IaaS) in cloud computing, is a model of data storage where digital data is stored in logical pools of storage. As rapid growth and application of cloud storage, users concern more and more about security and privacy issues involved in these techniques. It is required to protect data and applications in cloud from hackers and intruders, cryptography is considered as a key technology to solve security and privacy problems. Here, we mainly focuses on secure cloud storage in which encryption mechanism like AES and HMAC has been used to their designs. Cryptography is an essential that helps to assure our data accuracy and protect the data in cloud environment.***

Key Words: *Cloud Computing, Cloud Storage, AES, HMAC, Cryptography.*

1. INTRODUCTION

Cloud computing plays a very efficient role in the modern era because of it's compelling benefits, security and services. It provides dynamically scalable resources provisioned as a service over the internet. The main idea of cloud computing is to provide bunch of cloud servers for continuous access .The third party, on-demand, self-service, pay-per-use, and seamlessly scalable Cloud computing resources and services provided by the cloud paradigm promise to reduce capital as well as operational expenditures for hardware and software. It will focus on public clouds, because these services demand for the highest security requirements. For security prospects it also includes high potential. For security of user ,the data must be encrypted before sending to the cloud. Various distinct architectures are introduced and discussed according to their security and privacy capabilities and prospects. The use of cloud computing has increased rapidly in number of organizations .Many small and Medium companies use cloud computing services for various reasons, because these services provide fast access to their applications and reduce their infrastructure costs. Cloud providers should address privacy and security issues as a matter of high and urgent priority. Cloud computing is increasing rapidly ,by which lot's of criminals can attempt various way's to use vulnerabilities within the system. Cloud computing provides the software as a service. It is commercial evolution of grid computing. In this, we will provide strong privacy for messages in storage severs, by performing encryption on messages by Advance encryption Standard [AES] algorithm and providing the secret key by Keyed-Hash Message Authentication Code [HMAC] algorithm.

2. LITERATURE REVIEW

We have studied the AES algorithm from 'A Secure Way for Data Storage and Forwarding in Cloud' in this paper we are using AES for encryption purpose. AES algorithm is based on design principle known as substitution permutation network. It is fast as well flexible in both hardware and software. We have studied HMAC algorithm from 'The Keyed- Hash Message Authentication Code' in this paper HMAC is a message authentication code that uses cryptographic key in a conjunction with a hash function. The HMAC function is used by message sender to generate the secret key and message input. We have studied Proxy re-encryption scheme from 'A Secure Erasure Code-Based Cloud Storage System with Secure data Forwarding' in this paper the message are first encrypted by the owner and then stored in a storage server. When user wants to share his messages he sends a re-encryption key to storage server. The storage server re-encrypts the encrypted messages for the authorized user

thus their system has data confidentiality and supports the data forwarding function.

We have get the idea of the partitioning of data from 'Privacy Issues in Knowledge Discovery & Data Mining' in this paper the partitioned data are stored on different storage servers randomly, the encryption and encoding has been performed on partitioned data to achieve good privacy while maintaining the robustness of the system.

3. PROPOSED SYSTEM ARCHITECURE

The idea behind proposed system is to provide privacy and security to cloud users with less computation cost and minimum time. First of all, admin will send file which is encrypted by AES algorithm. After encryption file will get split into four to five sub-parts and then will be uploaded to server and then server will merge the sub-parts into a single file and it will be send to user and at the same time the secret key generated by HMAC will be sent it to the users e-mail id .Only after entering that key the original file will be get download otherwise fake file will get download. If the file gets leaked or hacked by the intruders then the notification will be sent it to the admin. When the file gets downloaded the user can lock it by entering the same or different secret key. Our system is made up of some modules which are discussed below:

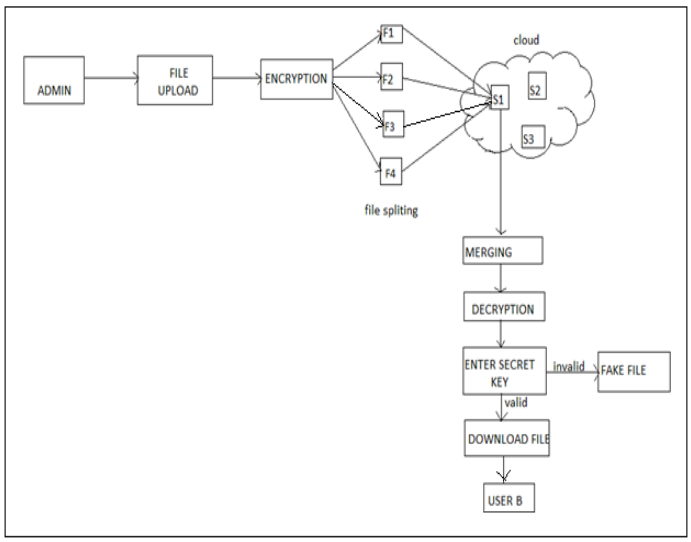

Fig. System Architecture

3.1 Construction Of Cloud Data Storage Module

In Admin Module the admin can login to give his password and username. After that server setup method can be opened. In server setup process for sending Ip-address to the receiver the admin first set the remote servers Ip-address . Then the server can skip the process to activate or Deactivate the process. Then the IP-address can be displayed by the storage server for activating the process. For Deactivating the process the storage server cannot display the Ip-address. By clicking the key server these details can be viewed . In available storage server the activated Ip-addresses are stored. Then we can view the currently available Ip-addresses by clicking the available storage server button.

3.2 Data Encryption Module

In cloud login module the user can login by giving his own details. If the user cannot have the account for that cloud system then first the user can register his details for using and entering into the cloud system. In registration process fields are Username, E-mail, password, confirm password. Then the details can be stored in database of the cloud system after entering the registration process. Then the user has to login by giving his username and password the secret key has to be send to his/her E-mail Id. Then the user will open his account and view the secret key that can be generated from the cloud system. The user has to select one file from browsing the system and enter the upload option. Then ,the encrypted form of the uploading file can be given by the server from the cloud .

3.3 Data Forwarding Module

In forward module we can see the storage details for the uploaded files. When we click the storage details option we can see the file name and forwarded E-mail. This processes contains the selected file name, E-mail address of the forwarder and enter the secret key to the forwarder. Now, another user can check his account properly and view the secret key forwarded from the previous user. Then the current user has login to the cloud system to check the received details. In received details the forwarded file is present then the user will go to the download process.

3.4 Data Retrieval Module

The Download module contains the details such as, username and file name. First, the server process can be run which means the server can be connected with its particular client. Now, the client has to view the secret key to download the file. In file downloading process the fields

are username, filename, and secret key. Now by clicking the download option the client can view the message box of Enter the Secret Key. Then after entering that key the client can view the file and use that file appropriately.

4. ALGORITHM

4.1 Advanced Encryption Standard (AES)

Advanced Encryption Standard, is stated as encryption standard recommended by NIST and it replaced DES. Brute force attack is the only effective attack known against it, in which the attacker tries to test all the characters combinations to unlock the encryption. AES and DES are both block ciphers. It has variable key length of 128, 192, or 256 bits; default 256. It encrypts blocks data of 128 bits in 10, 12 and 14 round which is depend on the key size. AES encryption has advantages as fast and flexible with it can be implemented on various platforms especially in small devices. For many security applications AES has been carefully tested. As per the Rijndael specification it specified with block and key sizes that may be any multiple of 32 bits, both with a minimum of 128 and a maximum of 256 bits. Whereas AES operates on a 44 column-major order matrix of bytes, termed the state, although Rijndeal of some versions have a larger block size with additional columns in the state. In special finite field most of the AES calculation are done.

1. Key Expansions: From the cipher key round keys are derived . Separate 128-bit round key block for each round plus one more required by AES.

2. Initial Round

a) AddRoundKey: Using bitwise xor each byte of state is combined with block of the round key .

3. Rounds-

a) SubBytes: each byte is replaced with another according to a lookup table in non linear substitution step.

b) ShiftRows: the last three rows of the state are shifted cyclically a certain number of steps in transposition step.

c). MixColumns: combining the four bytes in each column, a mixing operation which operates on the columns of the state,

d) AddRoundKey: using bitwise xor each byte of the state is combined with a block of the round key .

4 Final Round (no MixColumns)

5. SubBytes

6. ShiftRows

7. AddRoundKey.

4.2 Keyed-Hash Message Authentication Code (HMAC)

HMAC is a known as message authentication code that uses a cryptographic key in conjunction with a hash function.

HMAC uses following parameters:-

B – Block size (in bytes) of the input to the Approved hash function.

H - An approved hash function.

Ipad - Inner pad; the byte x36 repeated B times.

S - Secret key which is shared between the originator and the intended receiver(s).

S0 - The key after any necessary pre-processing to form a B byte key.

K – Block size(in bytes) of the output to the Approved hash function.

Opad - Outer pad; the byte x5c repeated B times.

t – The data on which HMAC is calculated; text does not include the padded key. The length of text is n bits.

|| - Concatenation

XOR- Exclusive-OR

MAC(t) = HMAC(S,t)= H((S0 XOR opad) || H((S0 XOR ipad) || t))

Step 1 – If the length of S=B; Set S0=S. Go to step 4.

Step 2 – If the length of S>B; hash S to obtain an K byte string then append (B-K)

Zeros to create a B-byte string S0(i.e.S0 = H(S)|| 00..00).Goto step 4.

Step 3 - If the length of S<B; append zeros to the end of S to create a B-byte string S0.

Step 4 – XOR S0 with ipad to produce a B byte string: (S0 XOR ipad)

Step 5 - Append the stream of data text to the string resulting from step 4:(S0 XOR ipad) || t

Step 6 - Apply H to the stream generated in step 5: H((S0 XOR ipad) || t)

Step 7 – Exclusive-OR S0 with opad: (S0 XOR opad)

Step 8 - Append the result from step 6 to step 7:

((S0 XOR opad) || H((S0 XOR ipad) || t)

Step 9 - Apply H to the result from step 8:

H((S0 XOR opad) || H((S0 XOR ipad) || t))

5. MATHEMATICAL MODEL

Let S be the system design for Cloud Architecture.

S ={s, e, x, y, success case, failure case, DD, NDD}

Where,

s = Start State

e = End State

x = Input

y = Output

DD = Deterministic Data

NDD = Non-Deterministic Data

1. Start State: Send by admin / Call by client

2. End State: Download file or Fake file.

3. Input:

Let x be the set such as

x={x 1 , x 2 , x 3 .....x n }

x 1 = Text file.

x 2 = Security Keys

4. Output:

Let y be the set such as

y={y 1 , y 2 , y 3 .....y n }

y 1 = Download file

y 2 =Fake file

5. Deterministic Data: File Database

6. Non-Deterministic Data: Randomly generated keys

6. CONCLUSIONS

Now-a-days there is a major issue of data security in cloud computing. To overcome it we have proposed a secure cloud storage system for data storage and data forwarding functionality. We partition the encrypted data and store them on storage server. It will keep the data secure during transmission and data at rest. It will be helping the user to send the data to cloud without hesitation of data being lost. In future it will be stored on multiple server and multiple clouds also.

REFERENCES

[1] B.Sowmya Sri,Mr.S.Vikramphaneendra,"A Secure Way for Data Storage and Forwarding in Cloud", Volume 3, Issue 9, September 2013.

[2] Hsao-Ying Lin, Member, IEEE, and Wen-Guey Tzeng, Member, IEEE.," A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwardin", IEEE transactions on parallel and distributed systems, vol. 23, no. 6, pp.995-1003 ,june 2012.

[3] Ernesto Damiani, Francesco Pagano,Davide Pagano,"iPrivacy: A Distributed Approach to Privacy on the Cloud", International Journal on Advances in Security, vol 4 no 3 & 4, year 2011,pp.185-197.

[4] Kubiatowicz, D. Bindel, Y. Chen, P. Eaton, D. Geels, R.Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao, "Oceanstore: An Architecture for Global-Scale Persistent Storage," Proc. Ninth Intl Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS), pp. 190- 201, 2000.

[5] Ljiljana Brankovic,Vladimir Estivill-Castro,"Privacy Issues in Knowledge Discovery & Data Mining", Newsletter The University of Newcastle, vol 3.no2, 2008, pp.1-12.

[6] M.Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc. Second USENIX Conf. File and Storage Technologies (FAST), pp. 29-42, 2003.

[7] G.Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re- Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security", vol. 9, no. 1, pp. 1-30, 2006.

[8] National Institute of Standards and Technology,"The Keyed-Hash Message Authentication Code",Federal Information Processing Standards Publication 198-1,July 2008.