Volume 3, Issue 7, July 2013 **ISSN: 2277 128X**

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Comprehensive Approach to Ensure Secure Data Communication in Cloud Environment

V.Nirosha
M.Tech Student
Dept. of CSE
Annamacharya Institute of Technology and Sciences.,
Tirupati,A.P,India

K.Suma Latha
Asst.Professor
Dept. of CSE
Annamacharya Institute of Technology and Sciences.,
Tirupati, A.P., India

Abstract: Cloud computing provides efficient services to its customers via internet. cloud storage system is a collection of storage servers in providing long term services. For secure storage of our data in cloud we use general encryption schemes which limits its functionality. This results in bringing the way an effective encryption scheme, proxy re-encryption and its integrity with decentralized erasure coding. Its main technical operations are encryption, encoding, and forwarding. This paper proposes RSA, AES encryptions with erasure coding for secure data forwarding for secure cloud storage.

Keywords: proxy re-encryption, decentralized erasure code, RSA, AES, erasure codes, secure cloud storage systems.

I. Introduction

Cloud computing provides compelling benefits and cost-effective options for IT hosting and expansion. But new risks and opportunities for security exploits are introduced. [1] Let you know the controls, risks in cloud computing. A cloud storage system, storing data in a third party's cloud system causes serious concern over data confidentiality. General encryption schemes protect data confidentiality, but also limit the functionality. Constructing a secure storage system that supports multiple functions is challenging when the storage system is distributed. In this paper, we focus on designing a cloud storage system for robustness, confidentiality, and functionality by using a different approach. Different systems for data storage in storage servers like Oceanstore [2]. For data robustness the data usually replicated and stored in different storage servers. One more technique is erasure coding, which encodes a message of k symbols into a codeword of n symbols. As an erasure code it can detect and correct combinations of errors and erasures. This finishes the encoding and storing process. The recovery process is the same. At second data confidentiality, So applying some cryptographic techniques before storing in cloud is a better approach. Encryption before encoding is a good work and we can get our data by decoding. This mechanism has problems like 1. user has to perform more computations 2. communication traffic increases 3. user has to manage cryptographic keys. what we are proposing is forwarding the data by storing the cryptographic keys in several servers may be for a thresh hold value limit and these servers must be independent. Now using the threshold proxy re-encryption scheme proposed by the authors Hsiao-Ying Lin and Wen-Guey Tzeng [3] and integrating it with the decentralized erasure code and encryption by using RSA scheme[4]. The operations encoding, encryption and forwarding meet the requirements of robustness, confidentiality in a place of cloud. What exactly we perform is that encrypting the the data before forwading and encoding data at the time of storage.

Our Contribution:

Constructing a cloud storage system for secure data forwarding using encryption and encoding techniques. For encryption we use proxy re-encryption scheme integrating with RSA. Storage servers encode and key servers partially decrypt. The encryption scheme supports encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. A secure distributed storage system is formulated by partitioning the data and performing encryption on each data block and performing re-encryption on each cipher block by demolishing the re-encryption key at user and regenerating it at key servers on demand for partial decryption.

II. Cloud Storage Systems

Cloud storage is a service model in which data is maintained, managed and backed up remotely and made available to user via internet. Cloud storage servers are servers located at different places and provides continuous access. Cloud storage system is a collection of such storage servers.

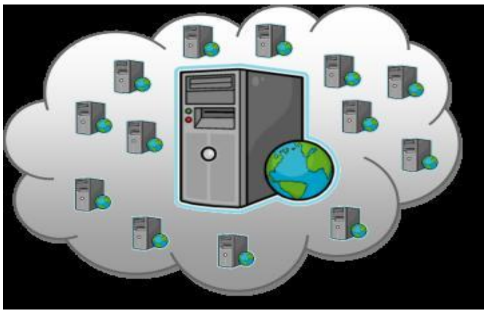


Figure. 2.1. Cloud Storage Server

III. Benefits and Risks

Cloud computing provides compelling benefits fig[3.1] like scalability, 24/7 support, pay per use and lots more. Cloud computing is not without risks or completely secure. Risks like privacy, identity management, authentication, compliance, confidentiality, integrity, availability of data, encryption, network security and physical security. Apart from the security risks [5], other concerns include SLA and third-party (service provider) management, vendor lock-in etc.

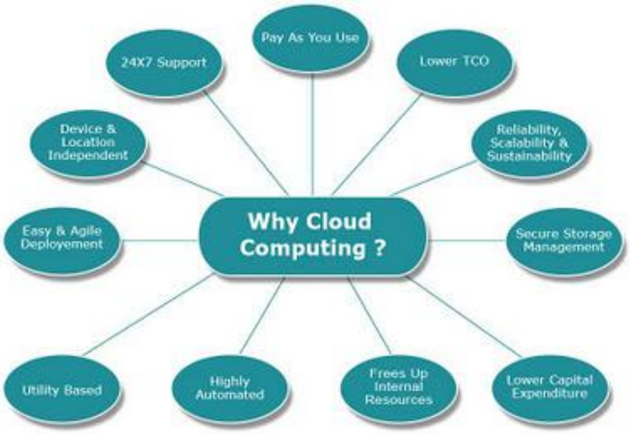

24X7 Support
Pay As You Use
Lower TCO
Device & Location Independent
Reliability, Scalability & Sustainability
Easy & Agile Deployement
Why Cloud Computing ?
Secure Storage Management
Utility Based
Highly Automated
Frees Up Internal Resources
Lower Capital Expenditure

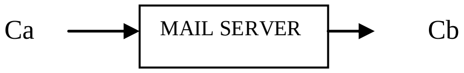
Figure 3.1. Cloud Computing Benefits.

There are various mitigations for network security, physical security, logical access etc. But we go in deep for data security.

IV. Proxy Re-Encryption Scheme

MOTIVATION

Proxy re-encryption is a relatively newly-devised cryptographic primitive. The goal of proxy re-encryption is to securely enable the reencryption of ciphertexts from one secret key to another, without relying on trusted parties.



TECHNIQUE USED

The proxy is entrusted with the delegation key *b/a* mod *q* for the purpose of diverting cipher texts from Alice to Bob via computing (mg^kmod p, $(g^{ak})^{b/a}$mod p). Improved proxy re-encryption [6] scheme is for better granularity. Key-private proxy re-encryption schemes are proposed by Ateniese et al. [7].

In a key-private proxy re-encryption scheme, given a re-encryption key, a proxy server cannot determine the identity of the recipient. Although most proxy re-encryption schemes use pairing operations, there exist proxy re-encryption schemes without pairing [8]. Lin and Tzeng used a threshold proxy re-encryption scheme with multiplicative homomorphic property. This encryption scheme supports the encoding operation over encrypted messages. A secret key is shared to key servers with a threshold value t via the Shamir secret sharing scheme [9], where $t \geq k$. In our system, to decrypt for a set of k message symbols, each key server independently queries 2 storage servers and partially decrypts two encrypted codeword symbols. As long as t key servers are available, k codeword symbols are obtained from the partially decrypted ciphertexts.

V. Decentralised Erasure Code

Assume that there are n storage servers in the networked storage system, and k messages are stored into the storage servers such that one can retrieve the k messages by only querying any k storage servers. Storage servers (ss1,ss2..), key servers (ks1,ks2..). Erasure coding is nothing but, k blocks of source data are encoded to n blocks of encoded data, such that the source data can be reconstructed from any subset of k encoded blocks. Each block is a data item which can be operated on with arithmetic operations.

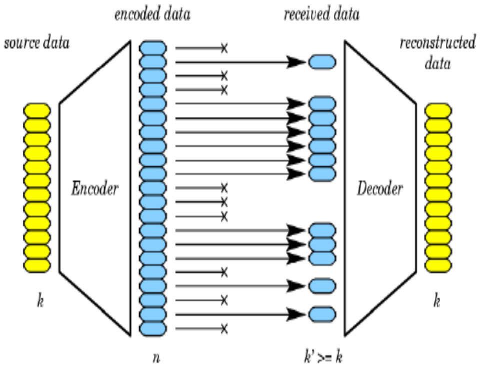


Figure 5.1. An Architecture for Erasure Coding

We can use any of the functions for erasure coding encoder. For example hash function etc,. Alice wants to send her telephone number (555629) to Bob then She breaks her telephone number up into two parts *a* = 555, *b* = 629, and sends 2 messages – "*A* = 555" and "*B* = 629" – to Bob. She constructs a linear function f(i) = a+(b-a)(i-1) in this case f(i)=555+74(i-1) such that f(1)=555 and f(2)=629. She computes the values *f*(3), *f*(4), and *f*(5), and then transmits three redundant messages: "C = 703", "D = 777" and "E = 851". Bob knows that the form of *f*(*k*) is f(i) = a+(b-a)(i-1) , where *a* and *b* are the two parts of the telephone number. Now suppose Bob receives "D = 777" and "E = 851". Bob can reconstruct Alice's phone number by computing the values of *a* and *b* from the values (*f*(4) and *f*(5)) he has received.

VI. General Architecture

Our process can be easily explained through the architecture. This is the general architecture for our system. The data is divided into blocks at user A and encrypts the blocks and then send the cipher blocks to the storage servers. In storage servers the cipher blocks are encoded and then stored. User performs re-encryption when data blocks are need to be forwarded. When the user B wants the data he queries the key servers. Key servers retrieve data from the storage servers after decoding process is performed and the partial decryption is done by key servers after generating the re-encryption key on demand for partial decryption.

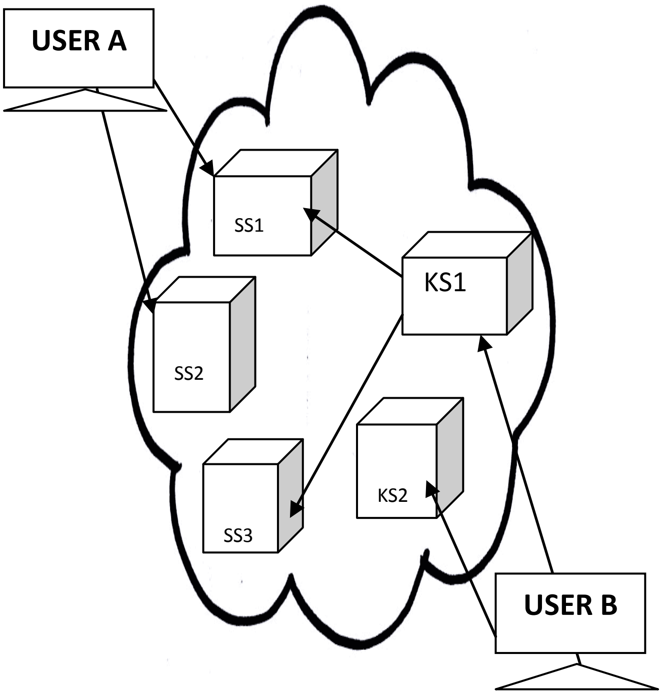


Figure 6.1 A General Architecture For Secure Data Forwarding.

VII. Cryptographic Scheme

There are several cryptographic schemes for easier manipulations we use RSA first time encryption and AES for re-encryption i.e., RSA is used for key generation and for encryption of data blocks and AES is used for re-encryption of cipher blocks.

A. RSA

RSA is a public key algorithm invented by Rivest, Shamir and Adleman. The key used for encryption is different from (but related to) the key used for decryption. The algorithm is based on modular exponentiation. Numbers e, d and N are chosen with the property that if A is a number less than N, then (Ae mod N)d mod N = A. This means that you can encrypt A with e and decrypt using d. Conversely you can encrypt using d and decrypt using e (though doing it this way round is usually referred to as signing and verification). The pair of numbers (e,N) is known as the public key and can be published. The pair of numbers (d,N) is known as the private key and must be kept secret.

Anybody knowing the public key can use it to create encrypted messages, but only the owner of the secret key can decrypt them. Conversely the owner of the secret key can encrypt messages that can be decrypted by anybody with the public key. Anybody successfully decrypting such messages can be sure that only the owner of the secret key could have encrypted them. This fact is the basis of the digital signature technique.

B. AES

These are the steps for AES.

1) KeyExpansion—round keys are derived from the cipher key using Rijndael's key schedule.

2) Initial Round

AddRoundKey—each byte of the state is combined with the round key using bitwise xor.

3) Rounds

a. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.

b. ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.

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

d. AddRoundKey

4) Final Round (no MixColumns)

1. SubBytes
2. ShiftRows
3. AddRoundKey

VIII. Conclusion

In this paper we proposed a technique for forwarding data securely in cloud storage system. We divide the data in to blocks and encrypting those blocks and distributing them to randomly chosen storage servers and encoding those cipher blocks for storage. For secure forwarding the data re- encryption is performed and then sent to storage servers. When asked for retrieval key servers perform generation of re-encryption key on demand for partial decryption. Encoding/decoding operations are performed at storage servers and partial decryption at key servers. As it require huge servers it need to be implemented in real way for real results.

Future Work

As this is a modern world demand for cloud storage system reaches to peak. Implementing such ideas for securing the data in cloud environment is needed badly. We can even use other encryption schemes for best performance.

References

[1] Carroll, M.; van der Merwe, A.; Kotze, P. "*Secure cloud computing: Benefits, risks and controls*", Information Security South Africa (ISSA), pp. 1-9, 2011.

[2] J. 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 Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS), pp. 190-201, 2000.

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

[4] Evgeny Milanov " The RSA Alorithm" 3 june, 2009.

[5] Bernd Grobauer, Tobias Walloschek, and Elmar Stöcker *Siemens, "Understanding Cloud Computing Vulnerabilities*", IEEE Journal On Computer And Reliability Socities, Vol. 9, Issue :2, pp. 50-57, March/April 2011.

[6] 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.

[7] G. Ateniese, K. Benson, and S. Hohenberger, "*Key-Private Proxy Re-Encryption*," Proc. Topics in Cryptology (CT-RSA), pp. 279-294, 2009.

[8] J. Shao and Z. Cao, "*CCA-Secure Proxy Re-Encryption without Pairings*," Proc. 12th Int'l Conf. Practice and Theory in Public Key Cryptography (PKC), pp. 357-376, 2009.

[9] A. Shamir, "*How to Share a Secret*," ACM Comm., vol. 22, pp. 612-613, 1979.

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