International Journal of Advanced Multidisciplinary Research
ISSN: 2393-8870
www.ijarm.com
DOI: 10.22192/ijamr Volume 4, Issue 9 -2017

Research Article

DOI: http://dx.doi.org/10.22192/ijamr.2017.04.09.001

A Secure Data Forwarding for Erasure code Based Cloud Computing Storage

Divya, S* and Karthik, M.
Department of Computer Science, Dhanraj Baid Jain College, Thuraipakkam, Chennai - 97
*Corresponding Author: *sdivi28@yahoo.com*

Keywords

Cloud Computing, Secure Erasure code, Data Forwarding, Distributed storage system

Abstract

Now a days the modern era everything is possible in terms of storage and making use of it as a service. Therefore Cloud computing is used to store, manage, and process data using a network which is hosted rather than a local server or a personal computer. Data robustness is a major requirement for storage systems. There have been many proposals of storing data over storage servers. One way to provide data robustness is to replicate a message such that each storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. Another way is to encode a message of k symbols into a codeword of n symbols by erasure coding. To store a message, each of its codeword symbols is stored in a different storage server. A storage server failure corresponds to an erasure error of the codeword symbol. With this consideration, we propose a new threshold proxy re-encryption scheme and integrate it with a secure decentralized code to form a secure distributed storage system. The encryption scheme supports encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. The tight integration of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data forwarding

Introduction

Cloud computing is a new buzzword in the business industry today. The idea leading to cloud computing paradigm is that the computing resources and software are available to the end user, whether an organisation or an individual, in a virtualized environment (cloud) and the user can access it on demand and using a 'pay as you go' approach. This computing model avoids the need for capital investment. Anyone can use cloud services without investment. Cloud computing is a concept that treats the resources on the Internet as a unified entity, a cloud. Users just use services without being concerned about how computation is done and storage is managed (1). In this paper, we focus on designing a cloud storage system for robustness, confidentiality, and functionality. A cloud storage system is considered as a large scale distributed storage system that consists of many independent storage servers.

Data robustness is a major requirement for storage systems. There have been many proposals of storing data over storage servers. One way to provide data robustness is to replicate a message such that each storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. Another way is to encode a message of k symbols into a codeword of n symbols by erasure

coding. To store a message, each of its codeword symbols is stored in a different storage server. A storage server failure corresponds to an erasure error of the codeword symbol. As long as the number of failure servers is under the tolerance threshold of the erasure code, the message can be recovered from the codeword symbols stored in the available storage servers by the decoding process (2).

With this consideration, we propose a new threshold proxy re-encryption scheme and integrate it with a secure decentralized code to form a secure distributed storage system. The encryption scheme supports encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. The tight integration of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data forwarding. Accomplishing the integration with consideration of a distributed structure is challenging. Our system meets the requirements that storage servers independently perform encoding and re-encryption and key servers independently perform partial decryption. Moreover, we consider the system in a more general setting than previous works. This setting allows more flexible adjustment between the number of storage servers and robustness.

System analysis

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential (3).

Three key considerations involved in the feasibility analysis are economical feasibility, technical feasibility and social feasibility.

Existing system

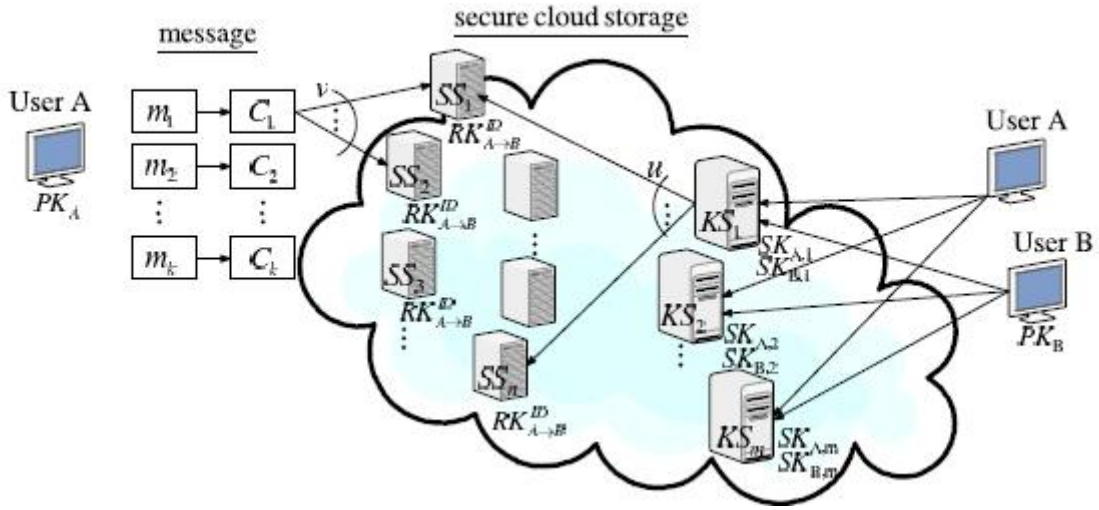
The design of existing is to operate a cloud storage system for robustness, privacy and functionality. There have been many proposals of storing data over storage servers. The encoding process for a message can be split into n parallel responsibilities of generating codeword cipher. A decentralized erasure code is suitable to use in a distributed storage system. After the message cipher are sent to storage servers, each storage server independently computes a codeword sign for the received message cipher and stores it (4).

Proposed system

Our design concentrates on the problem to forward a data to another user by storage servers directly under the authority of the data owner. We consider the system representation that consists of distributed storage servers and key servers. Since storing cryptographic keys in a single device is uncertain, a user distributes his cryptographic key to key servers that shall perform cryptographic functions on behalf of the user. These key servers are highly secluded by security mechanisms. To well fit the distributed structure of systems, we require that servers to independently perform all operations. With this deliberation, we suggest a new threshold proxy re-encryption scheme and combine it with a secure decentralized code to form a secure distributed storage system. The tight combination of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data forwarding. Our system meets the requirements of storage server's independently performing encoding and re-encryption, and key servers independently perform partial decryption. This makes our system to be more effective (4).

Advantage

- A secure cloud storage system implies that an unauthorized user or server cannot get the content of stored messages.
- A storage server cannot generate re-encryption keys by himself.
- If a storage server can generate a re-encryption key from the target user to another user B, the attacker can win the security game by re-encrypting the cipher text to B and decry
- Each storage server independently performs encoding and re-encryption and each key server independently perform partial decryption.



Modules description

Construction of Cloud Data Storage Module

This Module deals with the loading of application data into the server and also deals with the registration of new users to the application. It also provides the login feature to the application where the registered users will be able to login into the application and make use of the cloud storage server application. In cloud login module the user can login his own details. If the user cannot have the account for that cloud system first the user can register his details for using and entering into the cloud system. The Registration process details are Username, E-mail, password, confirm password, date of birth, gender and also the location. After entering the registration process the details can be stored in database of the cloud system. Then the user has to login to give his corrected username and password the code has to be send his/her E-mail. Then the user will go to open his account and view the code that can be generated from the cloud system (5).

Data Encryption Module

In Upload Module the new folder can be create for storing the files. In folder creation process the cloud system may ask one question for that user. The user should answer the question and must remember that answer for further usage. Then enter the folder name for create the folder for that user. In file upload process the user has to choose one file from browsing the system and enter the upload option. Now, the server from the cloud can give the encrypted form of the uploading file. File which was being uploaded into the storage server is firstly encrypted and stores the encrypted file along with the key file in the cloud location. It also stores all the details which was being entered while uploading the file (5).

Data Forwarding Module

In forward module first we can see the storage details for the uploaded files. When click the storage details option we can see the file name, question, answer, folder name, forward value (true or false), forward E-mail. If the forward column display the forwarded value is true the user cannot forward to another person. If the forward column display the forwarded value is false the user can forward the file into another person. In file forward processes contains the selected file name, E-mail address of the forwarder and enter the code to the forwarder. Now, another user can check his account properly and view the code forwarded from the previous user. Then the current user has login to the cloud system and to check the receive details. In receive details the forwarded file is present then the user will go to the download process (5).

Data Retrieval Module

In Download module contains the following details. There are 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 download the file to download the file key. In file key downloading process the fields are username, filename, question, answer and the code. Now clicking the download option the client can view the encrypted key. Then using that key the client can view the file and use that file appropriately. Now enter the code which was sent by application admin while registration. This code is used to download the encrypted key file. Decrypt the encrypted file by giving encrypted key file as input and download the original file (5).

System specification

Hardware requirements

The hardware used for the development of the project is:

PROCESSOR	:	PENTIUM III 866 MHz
RAM	:	128 MD SD RAM
HARD DISK	:	20 GB

Software requirements

The software used for the development of the project is:

OPERATING SYSTEM	:	Windows 2000 Professional
ENVIRONMENT	:	Visual Studio .NET 2003/asp .Net
DOTNET FRAMEWORK	:	Version 1.1
LANGUAGE	:	C#.NET
WEB TECHNOLOGY	:	Active Server Pages.NET
WEB SERVER	:	Internet Information Server 5.0
REPORTS	:	Web Form Data Grid control
BACK END	:	MS-SQL-Server 2005

Software description

ASP.NET is the next version of Active Server Pages (ASP); it is a unified Web development platform that provides the services necessary for developers to build enterprise-class Web applications. While ASP.NET is largely syntax compatible, it also provides a new programming model and infrastructure for more secure, scalable, and stable applications.

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet (6).

The common language runtime manages memory; thread execution, code execution, code safety verification, compilation, and other system services these are all run on CLR. These are based on Security, Robustness, Productivity and Performance (7).

The developing of this applications using ADO.NET, it offers several advantages over previous versions of ADO: Interoperability, Maintainability, Programmability, Performance and Scalability.

XML Web services are applications that can receive the requested data using XML over HTTP. XML Web services are not tied to a particular component technology or object-calling convention but it can be accessed by any language, component model, or operating system. In Visual Studio .NET, you can quickly create and include XML Web services using Visual Basic, Visual C#, JScript, Managed Extensions for C++, or ATL Server (8).

Extensible Markup Language (XML) provides a method for describing structured data. XML is a subset of SGML that is optimized for delivery over the Web. The World Wide Web Consortium (W3C) defines XML standards so that structured data will be uniform and independent of applications. Visual Studio .NET fully supports XML, providing the XML Designer to make it easier to edit XML and create XML schemas.

Visual Basic .NET, the latest version of visual basic, includes many new features. The Visual Basic supports interfaces but not implementation inheritance. Visual basic.net supports implementation inheritance, interfaces and overloading. In addition, Visual Basic .NET supports multithreading concept. Garbage Collection is another new feature in Visual Basic.NET (8).

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services (9).

Conclusion

A cloud storage system, consisting of a collection of storage servers, provides long-term storage services over the Internet. 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 of the storage system because a few operations are supported over encrypted data. Constructing a secure storage system that supports multiple functions is challenging when the storage system is distributed and has no central authority. We propose a threshold proxy re-encryption scheme and integrate it with a decentralized erasure code such that a secure distributed storage system is formulated. The distributed storage system not only supports secure and robust data storage and retrieval, but also lets a user forward his data in the storage servers to another user without retrieving the data back. The main technical contribution is that the proxy re-encryption scheme supports encoding operations over encrypted messages as well as forwarding operations over encoded and encrypted messages. Our method fully integrates encrypting, encoding, and forwarding. We analyze and suggest suitable parameters for the number of copies of a message dispatched to storage servers and the number of storage servers queried by a key server. These parameters allow more flexible adjustment between the number of storage servers and robustness.

References

(1) Hayes, B. (2009). Cloud computing. *Communications of the ACM , 51* (7), 9-11.

(2) H.-Y. Lin and W.-G. Tzeng, "A Secure Decentralized Erasure Code for Distributed Network Storage," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 11, pp. 1586-1594, Nov. 2010.

[3] D.R. Brownbridge, L.F. Marshall, and B. Randell, "The Newcastle Connection or Unixes of the World Unite!," Software Practice and Experience, vol. 12, no. 12, pp. 1147-1162, 1982.

[4] R. Sandberg, D. Goldberg, S. Kleiman, D. Walsh, and B. Lyon, "Design and Implementation of the Sun Network Filesystem," Proc. USENIX Assoc. Conf., 1985.

[5] 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.

[6] S.C. Rhea, P.R. Eaton, D. Geels, H. Weatherspoon, B.Y. Zhao, and J. Kubiatowicz, "Pond: The Oceanstore Prototype," Proc. Second USENIX Conf. File and Storage Technologies (FAST), pp. 1-14, 2003.

[7] R. Bhagwan, K. Tati, Y.-C. Cheng, S. Savage, and G.M. Voelker, "Total Recall: System Support for Automated Availability Management," Proc. First Symp. Networked Systems Design and Implementation (NSDI), pp. 337-350, 2004.

[8] A.G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Ubiquitous Access to Distributed Data in Large-Scale Sensor Networks through Decentralized Erasure Codes," Proc. Fourth Int'l Symp. Information Processing in Sensor Networks (IPSN), pp. 111- 117, 2005.

[9] A.G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized Erasure Codes for Distributed Networked Storage," IEEE Trans. Information Theory, vol. 52, no. 6 pp. 2809-2816, June 2006.

Access this Article in Online	
Quick Response Code	Website: www.ijarm.com Subject: Computer Science
DOI:10.22192/ijamr.2017.04.09.001	

How to cite this article:
Divya, S and Karthick, M. (2017). A Secure Data Forwarding for Erasure code Based Cloud Computing Storage. Int. J. Adv. Multidiscip. Res. 4(9): 1-5.
DOI: http://dx.doi.org/10.22192/ijamr.2017.04.09.001