Implementation of Information Leakage Avoiding (ILA) Application in Cloud Computing

Poonam Sawdekar
Asst. Prof.
Department of Computer Engineering
DYPIET, Pimpri
Pune-18

Seema Shah
Principal
Department of Computer Science
VIT, Wadala
Mumbai-31

ABSTRACT

Cloud computing provides highly scalable services to be easily consumed over the internet on an as-needed basis. While cloud computing is expanding rapidly and used by many individuals and organizations internationally, data protection issues in the cloud have not been carefully addressed at current stage. Users' fear of confidential data leakage and loss of privacy in the cloud may become a significant barrier to the wide adoption of cloud services. In this paper, we explore a newly emerging problem of information leakage caused by indexing in the cloud. We design three-tier data protection architecture to accommodate various levels of privacy concerns by users.

1. INTRODUCTION

Cloud computing is very hot in this era of computing because of its outstanding abilities in terms of cloud services, computing power, information security and ad hoc nature of network setup / configuration. Cloud computing is a technology which uses internet and one remote server to maintain data and various applications. Cloud computing provides significant cost effective IT resources as cost on demand IT based on the actual usage of the customer.

Information Security is the major aspect of business which needs to be addressed on priority. Through the increasing use of trusted cloud computing platform, business requirement of information security along with ad hoc processing power is also growing. However, the complexity of security is greatly increased when data is distributed over a wider area or greater number of devices and in multi-tenant systems that are being shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible.

Cloud service provider (CSP) can complicate privacy of data because of the extent to which virtualization for cloud processing (virtual machines) and cloud storage are used to implement cloud service. The point is that CSP operations, customer or tenant data may not remain on the same system, or in the same data center. This can lead to Information Leakage.

Information Leakage is an application weakness where an application reveals sensitive data, such as technical details of the web application, environment, or user-specific data. Sensitive data may be used by an attacker to exploit the target web application, its hosting network, or its users. Therefore, leakage of sensitive data should be limited or prevented whenever possible.

2. RELATED WORK

Amazon, Google, Microsoft, Saleforce.com and Sun are considered among the key players in the cloud computing market, but they represent only a small portion of the providers in this space. AWS offers an infrastructure, Google App Engine and Microsoft Azure Services offer platforms as a service for building and hosting web applications on the web infrastructure. Cloud computing raises a range of important privacy issues as acknowledged by a number of recent work, Such issues are due to the fact that, in the cloud, users' data and applications reside – at least for a certain amount of time – on the cloud cluster which is owned and maintained by a third party.

Despite increased awareness of the privacy issues in the cloud, little work has been done in this space. Recently, Pearson et al. has proposed accountability mechanisms to address privacy concerns of end users and then develop a simple solution. Their basic idea is that the user's private data is sent to the cloud in an encrypted form, and the processing is done on the encrypted data. In addition, general outsourcing techniques have been investigated over the past few years and several cryptographic-based approaches for ensuring remote data integrity have been proposed.

3. INFORMATION LEAKAGE AVOIDING (ILA) APPLICATION

3.1. WORKING OF ILA

Figure 3.1 below shows system architecture of Information Leakage Avoiding (ILA) Application. When cloud user login to system through web browser and chooses to upload his data to cloud the service provider creates index file on user's data. If that index file gets leaked then user's sensitive data may get leaked.

An overview of the architecture is given by Figure 1. According to the user's confidentiality requirements, user provide protection to files by giving access specifier to files then according to user's requirement request analyzer will select one of the following three components: (i)strong protection; (ii)medium protection; (iii)low protection. The output to be sent to the service provider will be JAR files which enclose both data and policies.

- Strong protection: The service provider is not allowed to read the sensitive portion of the user's file, so as to negate the risk of indexing being conducted on sensitive portion of the document that could lead to privacy leaks.

- Medium protection: The service provider is prevented from "effective" indexing. In general, the purpose of indexing is to speed up the search of desired data item through random access. Once random access is disallowed, indexes will become useless. Therefore, we propose an approach to disable random access to the data item in the user's file. Our approach does not rely on access control policies. Instead, we prevent random access by enforcing the server to read data in a sequential order. Even if the index is constructed on data copies, its effectiveness will be compromised when data is periodically updated by the user.

• Low protection: The user specifies clearly in the policy the usage of his data file and the usage of indexing. The service provider is assumed trusted and will inform and negotiate with the user the keywords to be used for indexing purposes. We propose a novel and generic technique, called portable data binding, to enforce the strategies adopted by the three components. In particular, we first define the indexing prevention policy to specify the access rights that a service provider will obtain to deal with the user's data. The policies will be tightly coupled with the user's data by physically attaching the two, so that the data will not be left unprotected at any time. Then policies and data will be transported together. Our technique will ensure the policy enforcement data.

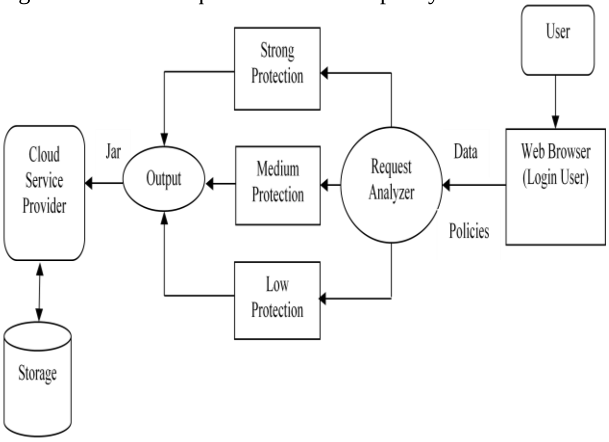

Figure 3.1 Application Architecture

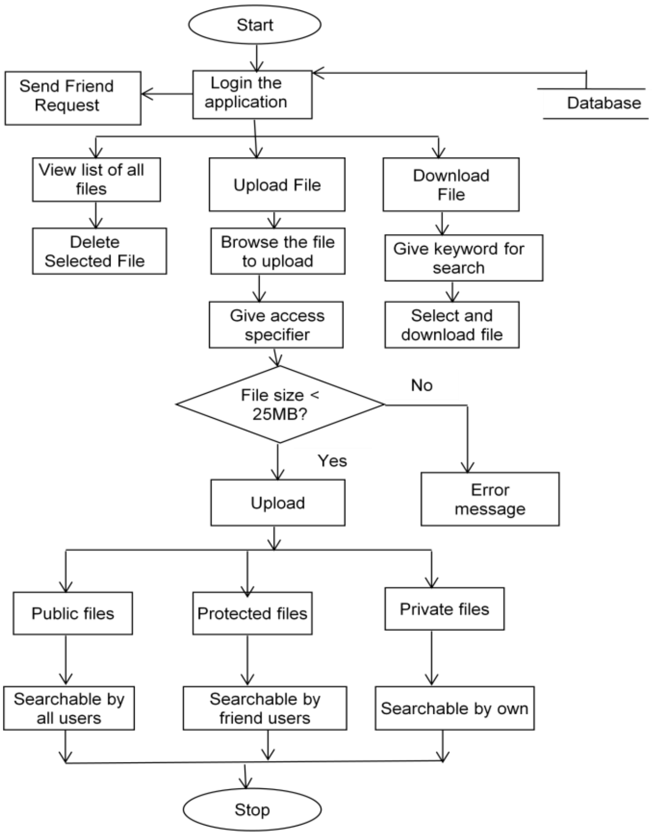

Figure 3.2 Information Leakage Avoiding Algorithm

3.2. ALGORITHM OF ILA

As shown in figure 3.2, when user login to system, it get checked with database. After login user can upload, download

He can send friend request to other user in that cloud network. Also he can check his own list of files uploaded by him. When user selects to upload, he needs to browse the file and select access specifier to file. If uploaded file is public it can be accessible by all users, if it is protected then that is searchable by user's friends only. And uploaded file is private then searchable by user only.

4. RESULTS

In this project we have two results: one is avoiding information leakage by indexing and second is indexing is not depends on types of files.

Result 1:

To avoid information leakage we used a three-tier data protection framework consisting of three protection strategies strong, medium and low, which differ according to the level of confidential requirement of data of the end-users. According to level of privacy requirement the indexing of user's data is done.

Three types of protection levels are provided namely: high, medium and low. User can give low, medium or high level of protection to data files according to importance of his data. User can make friends by using same application to provide access to his medium protected data after friend request is accepted. High protected data are available to user only. Low protected data are available to all users of same application. Index files are access controlled and they are getting stored in different folders according to their level of protection as mentioned in section 4.3, so that others cannot access that index file and less chances of leakage. Following table shows how we are avoiding information leakage in this project.

Table 4-1 Storage of various file types

File type	Private	Protected	Public
Access	Provided Access controlled	Provided Access controlled	No Access controlled
Storage	Stored in separate private folder for each user.	Stored in separate protected folder for each user.	Stored in one common folder

Private files are highly protected file we are saved in private folder for each user and they are access controlled. Protected files are with medium protection level. They are stored in protected folder for each user and access is given to user itself and their friends. Public files are open to all which are low protected. That files are saved in common folder. From above table and screen shot in section 4.3, we can say that index files are not accessible to everyone.

Result 2:
We evaluated the processing time of the strong-protection strategy which is the most demanding and time consuming among the three protection strategies. In the experiments, we varied the file size and tested different file types including text files, pdf, doc/docx, ppt and excel files. Table 5-6 shows the input parameters i.e. original file size and file size for indexing and the results i.e. processing time taken for indexing.
We can observe that the processing time increases with the original file size. In lucene indexing the processing time for 1000KB file for indexing is 1 second. However, we observed that the time taken for processing a 1000KB file is only 1 second, and is a good speed for indexing. For this project, we tested the different cases with varying different file size and file types. We considered 5 file types: text, pdf, doc, ppt and excel.
Following table shows minimum and maximum file sizes of each type of files. The minimum file size is for blank file. The default maximum file size to upload supported by IIS server is 25 MB.

Table 4-2 Minimum and Maximum size of different file types

File Type	Minimum Size (Blank file size)	Maximum Size (By IIS server)
Txt	0 KB	45 KB
Doc/docx	10 KB	25 MB
Excel	8 KB	25 MB
PPT	30 KB	25 MB
Pdf	91 KB	25 MB

As seen in the above section result, Private files are highly protected file we are saved in private folder for each user and they are access controlled. Protected files are with medium protection level. They are stored in protected folder for each user and access is given to user itself and their friends. Public files are open to all which are low protected. That files are saved in common folder. From result 1, we can say that information leakage is avoided by index files which are access controlled.
We can also say that from result 2, indexing is not depends on file type but it varies with file size because indexing is done on the basis of 1000kb/second rate. We combine the results of various file types in the table below 5-13. We are considering maximum file size of each type for indexing and then comparing the processing time for indexing.

Table 4-3 Result for various file types

File type	Size of original file (KB)	File size for indexing (KB)	Processing time for indexing (Seconds)
Txt	45	45	0.045
Doc	25000	25000	25
Excel	25000	25000	25
Ppt	25000	25000	25
Pdf	25000	25000	25

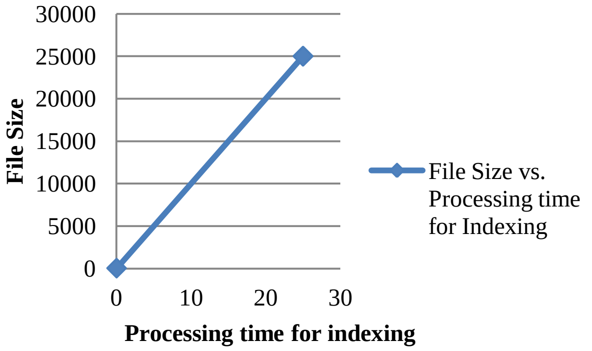

Figure 4.1 Graph of file size vs. processing time for various files

5. CONCLUSION

From above tables and graphs, we can say that indexing is not depends on file type but it increases with increase in file size because indexing is done at the rate of 1000kb/second. So file type is not important as file size increases time taken for indexing is also increases. We took all results for strong protection which is more demanding to avoid information leakage. Strong protection does not make any change in processing time taken for indexing. It only depends on file size, as file size increases time taken for indexing also increase.
The file size for indexing is same as the original file size. We can observe that the processing time increases with the original file size. File type does not matter and as file size increases time taken for indexing are also increases.
From above section comparison of algorithms, these two techniques are completely different to avoid leakage. These two techniques have different approach. In Privacy-Preserving technique provider maintain control in denying access groups over content and index host must apply these policies for each searcher to filter search results appropriately.

5.1. SCOPE FOR FUTURE RESEARCH

The scheme can be extended to support access-controlled search by propagating access policies along with content to the indexing host. Means by giving the access controlled search to user increases the security for data files more.

6. REFERENCES

[1] Anna Squicciarini and Smitha Sundareswaran: “Preventing Information Leakage from Indexing in the Cloud,” in 2010 IEEE 3rd International Conference on Cloud Computing.

[2] Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song.Provable data possession at untrusted stores. In Proc, of ACM conference on Computer and communications security, pages 598–609, 2007.

[3] Mayank Bawa, Roberto J. Bayardo, Rakesh Agrawal, and Jaideep Vaidya. Privacy -preserving indexing of documents on the network. VLDB J., 18(4):837–856, 2009.

[4] Y.-C. Chang and M. Mitzenmacher. Privacy preserving keyword searches on remote encrypted data. February 2004.

[5] Wei-Tek Tsai, Xin Sun, Janaka Balasooriya,” Service-Oriented Cloud Computing Architecture”, Seventh International Conference on Information Technology, 2010.

[6] Chunye Gong, Jie Liu, Qiang Zhang, Haitao Chen and Zhenghu Gong,” The Characteristics of Cloud Computing”, 39th International Conference on parallel Processing Workshops, 2010.

[7] R. Gellman. Privacy in the clouds: Risks to privacy and confidentiality from cloud computing. World Privacy Forum, 2009.

[8] B. R. Kandukuri, R. P. V., and A. Rakshit. Cloud security issues. In IEEE International Conference on Services Computing (SCC), pages 517–520, 2009.

[9] L. M. Kaufman. Data security in the World of Cloud Computing. IEEE Security and Privacy, 7(4):61–64, 2009.

[10] M. Lillibridge, S. Elnikety, A. Birrell, M. Burrows, and M. Isard. A cooperative internet backup scheme. In USENIX Annual Technical Conference, pages 29–41, 2003.

[11] S. Pearson and A. Charlesworth. Accountability as a way forward for privacy protection in the cloud. Hewlett-Packard Development Company (HPL-2009-178), 2009.

[12] B. P. Rimal, E. Choi, and I. Lumb. A taxonomy and survey of Cloud Computing systems. Networked Computing and Advanced Information Management, International Conference on, 0:44–51, 2009.