Chapter 2

State Of The Art

2.1 Introduction

Information Security is one of the key issues that slows down the development of cloud computing and its feature of complications with data privacy and protection continue to hinder taking the market [1]. Although the cloud storage services potentially offer several security advantages with their standardized accessing interfaces and benefits of scale (i.e., the same investment of security infrastructures on centralized server sets would provide better capability with regard to issues like data filtering, update management and deployment of information security policies) and agility of reactions to hacking events or other security-threatening behaviors. However, in some other ways, due to its essence of multi-tenancy model [2] which indicates problems caused by sharing physical storage resources with other users, also its nature of Cloud Software as a Service implies merely the accessing and auditing interfaces are open to users but nothing else which implies the hidden internal mechanism is untrusted to users, cloud computing will increase some other security risks to the clients. Additionally, security approaches about cloud storage, which is essentially a web application, must consider interactions and mutual effects among multiple network objects (i.e., other applications or users) [3] which lead to the invalidation or reduction of traditional local secure storage methods. Modern security technologies, and some mainstream cloud storage security solutions in particular, are seen as referable potential solutions to this problem. Their features and impacts also reasons for some of them not being inappropriate in certain scenario would be discussed in this chapter.

2.2 Cryptography

"Cryptography provided secrecy for information sent over channels where eavesdropping and message interception was possible" [4]. In order for concealing sensitive contents, cryptography is always used on plaintext in the storage media not only intelligible contents themself but also the metadata that might be utilized for deducing the corresponding contents.

2.1.1 User-level Cryptography

A typical way for encrypting the file or data is through some encryption software as tools, such as enigma under FreeBSD, crypt in Unix or applications with encryption modules embedded. The basic goal of information concealing achieved in this way if used properly but none of them is entirely satisfying with regard to security, generality and availability [5]. Manual errors like forgetting to delete the plaintext file after encryption or inappropriate key management usually make results far from expectation. As a conclusion, if encryption is too close to the user level, the high frequency of human interaction caused errors is not worth the cumber for practical and everyday usage [6].

2.1.2 System-level Cryptography

Generally to avoid the manual flaws of user level encryption, encryption modules serves as infrastructures of the underlying system. The key of designing such a system is about identifying what should be potentially trusted and what is the component to be cryptographically protected in accordance with priority level. Storage and communication risks are usually most urgent parts to be concerned about.

Physical media like hard drives could be well protected by customized hardware design or additional firmware functionalities. Disk controllers could be used to encrypt entire disks or individually cipher file blocks with a specified key and the procedure of encryption is completely transparent as long as the key is determined to the hardware [6]. Seems an ideal alternative of user level cryptography but problem happens when performing data management like sharing or backup. Data sharing would not be done until the encryption key exchange has been made which might be done via unreliable medium. To backup a disk without encrypting raw contents means exposing a decrypted version of ciphers but, however, to back up such disk with same encryption mechanism giving rise to not only extra expense of backup procedure but another potential unreliability concern. The very nature of backup is to ensure the availability as a redundancy but the extra cryptography issues leads to the fact that to guarantee the availability of cryptography modules is a necessary condition prior to assure the availability of content backup itself, which changes the essence of backup drastically.

Additionally, such mechanism would not protect data commuting with the disk so it is not sufficient to secure data in remote physical storage entities. Encrypted network connections between storage entities could be a solution of securing the data exchange procedure. The cryptography could be utilized in the form of end-to-end encryption communication protocol and cryptographic authentication [6]. However, some specialized hardware might be required and these extra cryptographic operations could cause a significant penalty of network performance.

2.3 Secure Storage On The Cloud

File system on the cloud, because of its web application circumstance, has more concerns in comparison with local file system. Besides enhancement of its service availability, generally features like constant monitoring and auditing mechanism would be well redesigned and integral to the system to achieve better security performance [7]. Most service providers offer access control mechanism as a basic security issue. Some service carriers (i.e., Dropbox) make use of a combination of identity based access control mechanism and encrypted data storage while others (i.e., Google Drive) just stick to concentrate on better access control and auditing performance and revolutionarily redesign the traditional file system, Google File System for instance. Also, besides the traditional identity based authentication, modern mechanism like two-step verification which basically refers to additional authentication step via text, voice call or other similar approaches, adds an extra layer of secure storage based on carriers' multi-platform feature.

2.3.1 Security in Dropbox

Dropbox uses Amazon's Simple Storage Service (S3) as storage infrastructure. All files stored online by Dropbox are encrypted during transmission and before storage with Secure Sockets Layer (SSL) and AES-256 bit encryption [8]. Amazon's Simple Storage Service does not encrypt data before storage but a service named Server Side Encryption (SSE) is provided for users to encrypt file and perform key management on cloud as well.

2.3.2 Security in Google Drive

Without using any local encryption on the physical storage infrastructure, Google improve their file system dramatically in terms of the access control and auditing mechanism. The Google File System has some innovative specific fields for multi-user management and data sharing purpose. For example, exportLinks allow data sharing without generating physical replicas or file relocation but the URL generated simultaneously and ready to be published. Another field writersCanShare indicates the writer's permission of sharing the file with others which potentially controls the information leakage.

Google also offers another access control mechanism called authorization scope. Authentication scopes imply the permissions users are required to authorize for the following operations. For example, when requiring for authorization with URL:
https://www.googleapis.com/auth/drive.readonly
means a successful authentication result would allow read-only access to the list of Drive apps a user has installed while with URL:
https://www.googleapis.com/auth/drive.readonly
indicates the positive authentication result would allow read-only access to file metadata and file content.

2.3.3 Storage Security Risks on the Cloud

Privacy concerns for some internal reasons are always there as long as the content is stored in plaintext on the cloud or encrypted with key that accessible by internal people. Dropbox has been criticized for storing authentication information on disk in plain-text which indicates that Dropbox's terms of service is contradict to its privacy policy although the company claim as employees of Dropbox aren't able to access user files [9]. Employees can effortlessly hack the system to access the data and profit unconsciously [10]. This kind of attacks cannot be protected by classical security technologies as long as it is technically possible [11].

External attackers would also try to hack the storage service provider's system to access stored data and such hacking could be prevented by traditional approaches [12]. It is undoubtedly that it would raise the data security risk if stored plaintexts are exposed nakedly after hacking behaviors breaking into the system. Given the fact that these day and night service especially those which are profit potential are more attractive to attackers, they are essentially more information secure risky. Some services claim that the stored data are all well encrypted but tragedy happens when key information are stored in the same bucket and easy to be discover by hackers.

Anyway, storage on the cloud in plaintext, no matter how unbeatable the security design and implementation is, are highly risky whether it is an internal or external attack. Customer data is uncontrolled and could be leaked to untrusted parties potentially in this way [12].

2.4 Searchable Symmetric Encryption based Security Storage Architecture

Encrypted data storage, assuming the private key could be stored properly and confidentially without any possibility of no matter internal or external hacking risk, its security could be assured theoretically. However, there could be availability performance penalty: the regular private-key encryptions prevent searching over encrypted data, client would lose the ability to selectively locate fragment of expected data at the same time [13, 15]. Searchable symmetric encryption (SSE) allows user to use third part storage service of its data, while keeping the capacity of selective searching over data pools [14].

The system prototype is described as a possible architecture for a cryptographic storage service in the paper by Seny Kamara and Kristin Lauter from Microsoft. The system is composed of three parts: a data processor (DP), a data verifier (DV) and a token generator (TG). Data processor handles data prior to being sent to the cloud; Data verifier verifies if the data in the cloud has been modified unauthorizedly based on auditing information; Token generator generates tokens that grant the permissions to the cloud storage provider and enable it to get segments of expected data of customer. Additionally, a credential generator who works based on certain access control mechanism is implemented. It works by issuing credentials to the various parties in the system which will enable them to decrypt encrypted files according to the policy [14].

The key features of a cryptography based storage service let customers take control of their data and its security properties are derived from cryptography, a trusted issue but not those untrusted human factors like legal mechanisms or physical security. The advantage in terms of security make the data is always encrypted and data integrity can be audited any time that security hacking poses little, or say, no risk for the customer [13]. What is more, the utilization of symmetric searchable, which theoretically leaves the indexes available but sensitive contents encrypted, improves the availability of encryption based security storage.

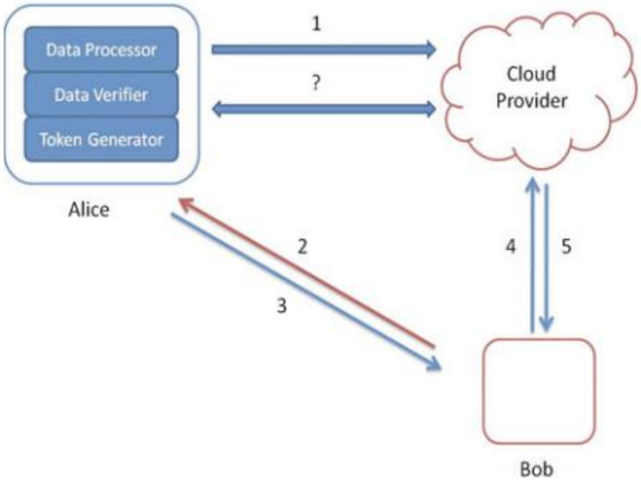

Fig1. A Customer Architecture
Kamara, S., & Lauter, K. (2010). Cryptographic cloud storage. In *Financial Cryptography and Data Security* (pp. 136-149). Springer Berlin Heidelberg.

The whole process illustrated in the paper [13] is expressed as follows:
“(1) Alice’s data processor prepares the data (encrypted with symmetric searchable encryption) before sending it to the cloud;
(2) Bob asks Alice for permission to search for a keyword;
(3) Alice’s token and credential generators send a token for the keyword and a credential back to Bob;
(4) Bob sends the token to the cloud;
(5) the cloud uses the token to find the appropriate encrypted documents and returns them to Bob.
(?) At any point in time, Alice’s data verifier can verify the integrity of the data. ”

2.5 FileMap

“FileMap is a file-based map-reduce system for data-parallel computation” [16]. It is designed and implemented around several themes like file-based and data replication. Its idea of data replication mechanism which is similar to RAID4 or RAID5, which means the file is split in the physical separated media and stored, could be referred to in the design of secure cloud storage. Files could be split into segments and stored in different entities which are logically or physically separated, or a combination of them. For example, to do secure cloud storage in logically separated entities, certain file could be split into n segments and stored in one provider’s cloud storage service with n identities (i.e., n segments stored in n accounts of Dropbox respectively). To do it in physically separated entities, as same as the first situation, file has to be split into n segments but the difference is storing them with n different cloud storage provider's service (i.e., n segments stored in Google Drive, Dropbox, SkyDrive… respectively). Given the fact that some providers providing encrypted storage service like Dropbox, a pseudo searchable encrypted cloud storage prototype with could be made by:

(1) Regenerating the file and add index information in the metadata. Splitting the file to index and contents segments.
(2) Storing the index segments in plaintext stored cloud storage service but who has better access control mechanism deployed like Google Drive.
(3) Storing the n content segments in n accounts of encrypted cloud storage service like Dropbox.

[1] Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, *34*(1), 1-11.

[2] Dillon, T., Wu, C., & Chang, E. (2010, April). Cloud computing: issues and challenges. In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on* (pp. 27-33). IEEE.

[3] Ou, X., Govindavajhala, S., & Appel, A. W. (2005, July). MulVAL: A logic-based network security analyzer. In *14th USENIX Security Symposium* (pp. 1-16).

[4] Robling Denning, D. E. (1982). *Cryptography and data security*. Addison-Wesley Longman Publishing Co., Inc...

[5] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, *1*(1), 7-18.

[6] Blaze, M. (1993, December). A cryptographic file system for UNIX. In *Proceedings of the 1st ACM conference on Computer and communications security* (pp. 9-16). ACM.

[7] Ghemawat, S., Gobioff, H., & Leung, S. T. (2003, October). The Google file system. In *ACM SIGOPS Operating Systems Review* (Vol. 37, No. 5, pp. 29-43). ACM.

[8] Dropbox Security and Privacy. (May, 2013) How secure is Dropbox? https://www.dropbox.com/help/27/en

[9] de Icaza, Miguel (April 19, 2011). "Dropbox Lack of Security". Retrieved July 19, 2011.

[10]Dhillon, G., & Moores, S. (2001). Computer crimes: theorizing about the enemy within. *Computers & Security*, *20*(8), 715-723.

[11]Yao, J., Chen, S., Nepal, S., Levy, D., & Zic, J. (2010, May). Truststore: making amazon s3 trustworthy with services composition. In *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing* (pp. 600-605). IEEE Computer Society.

[12] Wang, F., Uppalli, R., & Killian, C. (2003, October). Analysis of techniques for building intrusion tolerant server systems. In *Military Communications Conference, 2003. MILCOM'03. 2003 IEEE* (Vol. 2, pp. 729-734). IEEE.

[13] Kamara, S., & Lauter, K. (2010). Cryptographic cloud storage. In *Financial Cryptography and Data Security* (pp. 136-149). Springer Berlin Heidelberg.

[14] Curtmola, R., Garay, J., Kamara, S., & Ostrovsky, R. (2006, October). Searchable symmetric encryption: improved definitions and efficient constructions. In *Proceedings of the 13th ACM conference on Computer and communications security* (pp. 79-88). ACM.

[15] Song, Dawn Xiaoding, David Wagner, and Adrian Perrig. "Practical techniques for searches on encrypted data." *Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on*. IEEE, 2000.

[16]mfisk,FileMap. https://github.com/mfisk/filemap/wiki (June 2013)