International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 2 | ISSN : 2456-3307

Efficient Remote Data Integrity Checking With Securely Outsourcing of Key Updates

R. Shanthini, Prof. A. Padma

Muthayammal Engineering College, Rasipuram, Rasipuram, Tamil Nadu, India

ABSTRACT

Key-exposure resistance has always been an important issue for in-depth cyber defense in many security applications. Recently, how to deal with the key exposure problem in the settings of cloud storage auditing has been proposed and studied. To address the challenge, existing solutions all require the client to update his secret keys in every time period, which may inevitably bring in new local burdens to the client, especially those with limited computation resources such as mobile phones. In this paper, we focus on how to make the key updates as transparent as possible for the client and propose a new paradigm called cloud storage auditing with verifiable outsourcing of key updates. In this paradigm, key updates can be safely outsourced to some authorized party, and thus the key-update burden on the client will be kept minimal. Specifically, we leverage the third party auditor (TPA) in many existing public auditing designs, let it play the role of authorized party in our case, and make it in charge of both the storage auditing and the secure key updates for key-exposure resistance. In our design, TPA only needs to hold an encrypted version of the client's secret key, while doing all these burdensome tasks on behalf of the client. The client only needs to download the encrypted secret key from the TPA when uploading new files to cloud. Besides, our design also equips the client with capability to further verify the validity of the encrypted secret keys provided by TPA.

Keywords : TPA, Data Integrity, searchable encryption, Cloud computing, CRM, API, LAN, WAN, MAN

I. INTRODUCTION

1.1 Cloud Computing

Cloud storage has emerged as a promising solution for providing ubiquitous, convenient, and on-demand accesses to large amounts of data shared over the Internet. Today, millions of users are sharing personal data, such as photos and videos, with their friends through social network applications based on cloud storage on a daily basis. Business users are also being attracted by cloud storage due to its numerous benefits, including lower cost, greater agility, and better resource utilization. However, while enjoying the convenience of sharing data via cloud storage, users are also increasingly concerned about inadvertent data leaks in the cloud. Such data leaks, caused by a malicious adversary or a misbehaving cloud operator, can usually lead to serious breaches of personal privacy or business secrets (e.g., the recent high profile incident of celebrity photos being leaked in iCloud). To address users' concerns over potential data leaks in cloud storage, a common approach is for the data owner to encrypt all the data before uploading them to the cloud, such that later the encrypted data may be retrieved and decrypted by those who have the decryption keys. Such cloud storage is often called the cryptographic cloud storage. However, the encryption of data makes it challenging for users to search and then selectively retrieve only the data containing given keywords. A common solution is to employ a searchable encryption (SE) scheme in which the data owner is required to encrypt potential keywords and upload them to the cloud together with encrypted data, such that, for retrieving data matching a keyword, the user will send the corresponding keyword trapdoor to the cloud for performing search over the encrypted data. Although combining a searchable encryption scheme with cryptographic cloud storage

can achieve the basic security requirements of a cloud storage, implementing such a system for large scale applications involving millions of users and billions of files may still be hindered by practical issues involving the efficient management of encryption keys, which, to the best of our knowledge, are largely ignored in the literature. First of all, the need for selectively sharing encrypted data with different users (e.g., sharing a photo with certain friends in a social network application, or sharing a business document with certain colleagues on a cloud drive) usually demands different encryption keys to be used for different files.

Cloud computing is the concept by which multiple customers across different locations share the internet based applications at very minimal or in some cases without incurring any cost. The applications will become utilities for them which can be shared by many. One such application is Google API which can be used to create web based applications. These applications are user friendly and customized for beginners. The applications are easy to create, to configure and can be customized based on need. There are many applications which can run using this concept such as free email, web conference, CRM etc.

The network or internet providing the service to the users is referred as cloud. It consists of private as well as public networks including LAN, WAN, MAN etc. The manipulation, configuration and access of various online applications is referred as cloud computing. It provides data storage using configured email accounts (e.g. drop box or Google drive), infrastructure usage (e.g. use of costly equipments using remote connection based on lease) and applications as mentioned. The beauty of cloud computing is that use need not have to install any software or operating system and can use the service provided by cloud computing from anywhere in very cost effective and convenient manner.

II. LITERATURE SURVEY

2.1. TOWARD PUBLICLY AUDITABLE SECURE CLOUD DATA STORAGE SERVICES

Cloud computing is the long dreamed vision of computing as a utility, where data owners can remotely store their data in the cloud to enjoy on-demand high-quality applications and services from a shared pool of configurable computing resources. While data outsourcing relieves the owners of the burden of local data storage and maintenance, it also eliminates their physical control of storage dependability and security, which traditionally has been expected by both enterprises and individuals with high service-level requirements. In order to facilitate rapid deployment of cloud data storage service and regain security assurances with outsourced data dependability, efficient methods that enable on-demand data correctness verification on behalf of cloud data owners have to be designed.

In this article propose that publicly auditable cloud data storage is able to help this nascent cloud economy become fully established. With public audit ability, a trusted entity with expertise and capabilities data owners do not possess can be delegated as an external audit party to assess the risk of outsourced data when needed. Such an auditing service not only helps save data owners' computation resources but also provides a transparent yet cost-effective method for data owners to gain trust in the cloud. Describe approaches and system requirements that should be brought into consideration, and outline challenges that need to be resolved for such a publicly auditable secure cloud storage service to become a reality.

A high-level architecture description of cloud data storage services illustrated in Fig. 1. At its core, the architecture consists of four different entities: data owner, user, cloud server (CS), and TPA. Here the TPA is the trusted entity that has expertise and capabilities to assess cloud storage security on behalf of a data owner upon request. Under the cloud paradigm, the data owner may represent either the individual or the enterprise customer, who relies on the cloud server for remote data storage and maintenance, and thus is relieved of the burden of building and maintaining local storage infrastructure. In most cases cloud data storage services also provide benefits like availability (being able to access data from anywhere), relative low cost (paying as a function of need), and on demand sharing among a group of trusted users, such as partners in a collaboration team or employees in the enterprise organization. For simplicity, a single writer/many readers scenario here. Only the data owner can dynamically interact with the CS to update her stored data, while users just have the privilege of file reading.

Within the scope of this article, how to ensure publicly auditable secure cloud data storage services. As the data owner no longer possesses physical control of the data, it is of critical importance to allow the data owner to verify that his data is being correctly stored and maintained in the cloud. Considering the possibly large cost in terms of resources and expertise, the data owner may resort to a TPA for the data auditing task to ensure the storage security of her data, while hoping to keep the data private from the TPA. The TPA, who is in the business of auditing, is reliable and independent, and thus has no incentive to collude with either the CS or the owners during the auditing process. The TPA should be able to efficiently audit the cloud data storage without local copy of data and without any additional online burden for data owners. Besides, any possible leakage of an owner's outsourced data toward a TPA through the auditing protocol should be prohibited. Consider both malicious outsiders and a semi-trusted CS as potential adversaries interrupting cloud data storage services. Malicious outsiders can be economically motivated, and have the capability to attack cloud storage servers and subsequently pollute or delete owners' data while remaining undetected. The CS is semi-trusted in the sense that most of the time it behaves properly and does not deviate from the prescribed protocol execution. However, for its own benefit the CS might neglect to keep or deliberately delete rarely accessed data files that belong to ordinary cloud owners. Moreover, the CS may decide to hide the data corruptions caused by server hacks or Byzantine failures to maintain its reputation. Note that in our architecture, Basic security mechanisms such as a preloaded public/private key pair with each entity are already in place to provide basic communication security, which can be achieved in practice with little overhead.

DRAWBACKS

Some suggested requirements for public auditing services and the state of the art that fulfills them. However, this is still not enough for a publicly auditable secure cloud data storage system, and further challenging issues remain to be supported and resolved.

III. SYSTEM INFORMATION

3.1 EXISTING SYSTEM

A cloud storage auditing protocol with key-exposure resilience by updating the user's secret keys periodically. In this way, the damage of key exposure in cloud storage auditing can be reduced. But it also brings in new local burdens for the client because the client has to execute the key update algorithm in each time period to make his secret key move forward. For some clients with limited computation resources, they might not like doing such extra computations by themselves in each time period.

3.1.1 Limitations

1. New local burdens for the client because the client to update his secret keys in every time period
2. Does not support third party verification.
3. End user could be cheated.
4. Using DES encrypts and decrypts process. So it`s provide less security.

3.2 PROPOSED SYSTEM

TPA only needs to hold an encrypted version of the client's secret key, while doing all these burdensome tasks on behalf of the client. The client only needs to download the encrypted secret key from the TPA when uploading new files to cloud. Besides, our design also equips the client with capability to further verify the validity of the encrypted secret keys provided by TPA. Encrypt the file using blowfish algorithm and then store into the cloud.

3.2.1 Advantages:

- In this system hide the user password or encrypt user password and then store into cloud server
- If user enter wrong key for 3 times means account is blocked and send alert message to user.
- More Secure transaction
- TPA not know the user and owner password
- Support third-party verification
- More effective and flexible data sharing and data verification.
- Easy to finding the illegal access

3.3. SYSTEM REQUIREMENTS

3.3.1 Hardware Requirements

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.

- Monitor : 15 inch VGA Color.
- Mouse : Logitech Mouse.
- Ram : 512 MB
- Keyboard : Standard Keyboard

3.3.2 Software Requirements

- Operating System: Windows XP.
- Coding Language : ASP.NET
- Database : SQL SERVER 2005

3.4 SOFTWARE DESCRIPTION

3.4.1 .NET DEFINITION

Many people reckon that it's Microsoft's way of controlling the Internet, which is false. .NET is Microsoft's strategy of software that provides services to people any time, any place, on any device. An accurate definition of .NET is, it's an XML Web Services platform which allows us to build rich .NET applications, which allows users to interact with the Internet using wide range of smart devices (tablet devices, pocket PC's, web phones etc), which allows to build and integrate Web Services and which comes with many rich set of tools like Visual Studio to fully develop and build those applications.

3.4.2 .NET Framework

.NET is a "Software Platform". It is a language-neutral environment for developing rich .NET experiences and building applications that can easily and securely operate within it. When developed applications are deployed, those applications will target .NET and will execute wherever .NET is implemented instead of targeting a particular Hardware/OS combination. The components that make up the .NET platform are collectively called the .NET Framework.

The .NET Framework is a managed, type-safe environment for developing and executing applications. The .NET Framework manages all aspects of program execution, like, allocation of memory for the storage of data and instructions, granting and denying permissions to the application, managing execution of the application and reallocation of memory for resources that are not needed.

The .NET Framework is designed for cross-language compatibility. Cross-language compatibility means, an application written in Visual Basic .NET may reference a DLL file written in C# (C-Sharp). A Visual Basic .NET class might be derived from a C# class or vice versa.

The .NET Framework consists of two main components:

- Common Language Runtime (CLR)
- Class Libraries

IV.CONCLUSION

In this project, how to outsource key updates for cloud storage auditing with key-exposure resilience. The first cloud storage auditing protocol with verifiable outsourcing of key updates. In this protocol, key updates are outsourced to the TPA and are transparent for the client. In addition, the TPA only sees the encrypted version of the client's secret key, while the client can further verify the validity of the encrypted secret keys when downloading them from the TPA. The formal security proof and the performance simulation of the proposed scheme.Considering the practical problem of privacy preserving data sharing system based on public cloud storage which requires a data owner to distribute a large number of keys to users to enable them to access his/her documents, the first time propose the concept of key-aggregate searchable encryption and construct a concrete KASE scheme. Both analysis and evaluation results confirm that our work can provide an effective solution to building practical data sharing system based on public cloud storage. In a KASE scheme, the owner only needs to distribute a single key to a user when sharing lots of documents with the user and the user only needs to submit a single trapdoor when he queries over all documents shared by the same owner. However, if a user wants to query over documents shared by multiple owners, he must generate multiple trapdoors to the cloud.

V. REFERENCES

[1]. M. J. Atallah, K. Pantazopoulos, J. R. Rice, and E. E. Spafford, "Secure outsourcing of scientific computations," Trends in Software Engineering, vol. 54, pp. 215-272 2002.

[2]. D. Benjamin and M. J. Atallah, "Private and cheatingfree outsourcing of algebraic computations," Proc. Sixth Annual Conference on Privacy, Security and Trust, pp. 240-245, 2008.
[3]. C.Wang, K. Ren, and J.Wang, "Secure and practical outsourcing of linear programming in cloud computing," IEEE INFOCOM 2011, pp. 820-828, 2011.
[4]. X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, "New Algorithms for Secure Outsourcing of Modular Exponentiations," Proc. 17th European Symposium on Research in Computer Security, pp. 541-556, 2012.
[5]. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security, pp. 598-609, 2007.
[6]. A. Juels, J. Burton, and S. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security, pp. 584-597, 2007.
[7]. H. Shacham and B. Waters, "Compact Proofs of Retrievability," Advances in Cryptology-Asiacrypt'08, pp. 90-107, 2008.
[8]. G. Ateniese, R.D. Pietro, L. V. Mancini, and G. Tsudik,"Scalable and Efficient Provable Data Possession," Proc. 4th International Conference on Security and Privacy in Communication Networks, 2008
[9]. F. Sebe, J. Domingo-Ferrer, A. Martinez-balleste, Y. Deswarte, and J. Quisquater, "Efficient Remote Data Integrity checking in Critical Information Infrastructures," IEEE Transactions on Knowledge and Data Engineering, vol. 20, no. 8, pp. 1-6, 2008.