© 2015 IJSRST | Volume 1 | Issue 2 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Engineering and Technology

Data Integrity Proof In Cloud Storage

Bhavesh Prajapati
Department of Information Technology, L. D. College of Engineering, Ahmedabad, Gujarat, India

ABSTRACT

Cloud computing has been envisioned as the de-facto solution to the rising storage costs of IT Enterprises. With the high costs of data storage devices as well as the rapid rate at which data is being generated it proves costly for enterprises or individual users to frequently update their hardware. Apart from reduction in storage costs data outsourcing to the cloud also helps in reducing the maintenance. Cloud storage moves the user's data to large data centres, which are remotely located, on which user does not have any control. However, this unique feature of the cloud poses many new security challenges which need to be clearly understood and resolved. We provide a scheme which gives a proof of data integrity in the cloud which the customer can employ to check the correctness of his data in the cloud. This proof can be agreed upon by both the cloud and the customer and can be incorporated in the Service level agreement (SLA).

Keywords: DNS Hacking, QoS Violation, Denial of Service, Man in the Middle Attack, IP Spoofing, Data Sanitization, Data centre Security, Cloud Computing, Proof of irretrievability, Service level agreement, Cloud storage

I. INTRODUCTION

Data outsourcing to cloud storage servers is raising trend among many firms and users owing to its economic advantages. This essentially means that the owner (client) of the data moves its data to a third party cloud storage server which is supposed to - presumably for a fee - faithfully store the data with it and provide it back to the owner whenever required.

As data generation is far outpacing data storage it proves costly for small firms to frequently update their hardware whenever additional data is created. Also maintaining the storages can be a difficult task. Storage outsourcing of data to cloud storage helps such firms by reducing the costs of storage, maintenance and personnel. It can also assure a reliable storage of important data by keeping multiple copies of the data thereby reducing the chance of losing data by hardware failures.

Storing of user data in the cloud despite its advantages has many interesting security concerns which need to be extensively investigated for making it a reliable solution to the problem of avoiding local storage of data. In this paper we deal with the problem of implementing a protocol for obtaining a proof of data possession in the cloud sometimes referred to as Proof of retrievability (POR).This problem tries to obtain and verify a proof that the data that is stored by a user at a remote data storage in the cloud (called cloud storage archives or simply archives) is Not modified by the archive and thereby the integrity of the data is assured.

Such verification systems prevent the cloud storage archives from misrepresenting or modifying the data stored at it without the consent of the data owner by using frequent checks on the storage archives. Such checks must allow the data owner to efficiently, frequently, quickly and securely verify that the cloud archive is not cheating the owner. Cheating, in this context, means that the storage archive might delete some of the data or may modify some of the data.

Purpose

Purpose of developing proofs for data possession at untrusted cloud storage servers we are often limited by the resources at the cloud server as well as at the client. Given that the data sizes are large and are stored at remote servers, accessing the entire file can be

expensive in I/O costs to the storage server. Also transmitting the file across the network to the client can consume heavy bandwidths. Since growth in storage capacity has far outpaced the growth in data access as well as network bandwidth, accessing and transmitting the entire archive even occasionally greatly limits the scalability of the network resources. Furthermore, the I/O to establish the data proof interferes with the on-demand bandwidth of the server used for normal storage and retrieving purpose.

Scope

Cloud storing its data file F at the client should process it and create suitable metadata which is used in the later stage of verification the data integrity at the cloud storage. When checking for data integrity the client queries the cloud storage for suitable replies based on which it concludes the integrity of its data stored in the client. our data integrity protocol the verifier needs to store only a single cryptographic key - irrespective of the size of the data file F- and two functions which generate a random sequence. The verifier does not store any data with it. The verifier before storing the file at the archive pre-processes the file and appends some metadata to the file and stores at the archive.

Proposed Product Features

Our scheme was developed to reduce the computational and storage overhead of the client as well also minimizes the computational overhead of the cloud storage server. We also minimized the size of the proof of data integrity so as to reduce the network bandwidth consumption. Hence the storage at the client is very much minimal compared to all other schemes that were developed. Hence this scheme proves advantageous to thin clients like PDAs and mobile phones.

The operation of encryption of data generally consumes a large computational power. In our scheme the encrypting process is very much limited to only a fraction of the whole data thereby saving on the computational time of the client. Many of the schemes proposed earlier require the archive to perform tasks that need a lot of computational power to generate the proof of data integrity. But in our scheme the archive just need to fetch and send few bits of data to the client.

II. METHODS AND MATERIAL

Problem Definition

Storing of user data in the cloud despite its advantages has many interesting security concerns which need to be extensively investigated for making it a reliable solution to the problem of avoiding local storage of data. Many problems like data authentication and integrity (i.e., how to efficiently and securely ensure that the cloud storage server returns correct and complete results in response to its clients' queries, outsourcing encrypted data and associated difficult problems dealing with querying over encrypted domain were discussed in research literature.

Existing System

As data generation is far outpacing data storage it proves costly for small firms to frequently update their hardware whenever additional data is created. Also maintaining the storages can be a difficult task. It transmitting the file across the network to the client can consume heavy bandwidths. The problem is further complicated by the fact that the owner of the data may be a small device, like a PDA (personal digital assist) or a mobile phone, which have limited CPU power, battery power and communication bandwidth.

Limitations of Existing

System

- The main drawback of this scheme is the high resource costs it requires for the implementation.
- Also computing hash value for even a moderately large data files can be computationally burdensome for some clients (PDAs, mobile phones, etc.).
- Data encryption is large so the disadvantage is small users with limited computational power (PDAs, mobile phones etc.).

Proposed System

One of the important concerns that need to be addressed is to assure the customer of the integrity i.e. correctness of his data in the cloud. As the data is physically not accessible to the user the cloud should provide a way for the user to check if the integrity of his data is maintained

or is compromised. In this paper we provide a scheme which gives a proof of data integrity in the cloud which the customer can employ to check the correctness of his data in the cloud. This proof can be agreed upon by both the cloud and the customer and can be incorporated in the Service level agreement (SLA). It is important to note that our proof of data integrity protocol just checks the integrity of data i.e. if the data has been illegally modified or deleted.

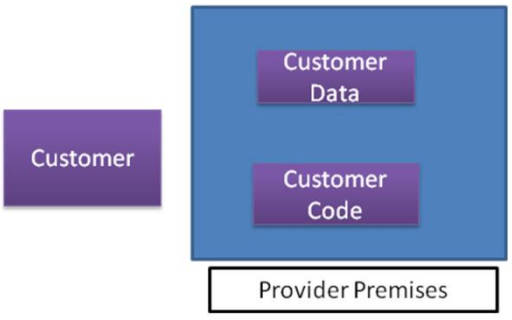
Advantages of Proposed System

- Apart from reduction in storage costs data outsourcing to the cloud also helps in reducing the maintenance.
- Avoiding local storage of data.
- By reducing the costs of storage, maintenance and personnel.
- It reduces the chance of losing data by hardware failures.
- Not cheating the owner.

III. RESULTS AND DISCUSSION

Security a major Concern

- Security concerns arising because both customer data and program are residing in Provider Premises.
- Security is always a major concern in Open System Architectures

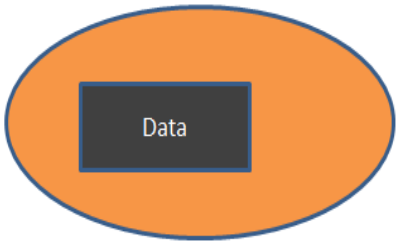


Data centre Security?

- Professional Security staff utilizing video surveillance, state of the art intrusion detection systems, and other electronic means.
- When an employee no longer has a business need to access datacenter his privileges to access datacenter should be immediately revoked.
- All physical and electronic access to data centers by employees should be logged and audited routinely.
- Audit tools so that users can easily determine how their data is stored, protected, used, and verify policy enforcement.

Data Location

- When user uses the cloud, user probably won't know exactly where your data is hosted, what country it will be stored in?
- Data should be stored and processed only in specific jurisdictions as define by user.
- Provider should also make a contractual commitment to obey local privacy requirements on behalf of their customers,
- Data-centered policies that are generated when a user provides personal or sensitive information, that travels with that information throughout its lifetime to ensure that the information is used only in accordance with the policy



Backups of Data

- Data store in database of provider should be redundantly store in multiple physical locations.
- Data that is generated during running of program on instances is all customer data and therefore provider should not perform backups.
- Control of Administrator on Databases.

Data Sanitization

- Sanitization is the process of removing sensitive information from a storage device.
- What happens to data stored in a cloud computing environment once it has passed its user's "use by date"
- What data sanitization practices does the cloud computing service provider propose to implement for redundant and retiring data storage devices as and when these devices are retired or taken out of service.

Network Security

- **Denial of Service**: where servers and networks are brought down by a huge amount of network traffic

and users are denied the access to a certain Internet based service.

- **Like DNS Hacking**, Routing Table "Poisoning", XDoS attacks
- **QoS Violation**: through congestion, delaying or dropping packets, or through resource hacking.
- **Man in the Middle Attack**: To overcome it always use SSL
- **IP Spoofing**: Spoofing is the creation of TCP/IP packets using somebody else's IP address.
- **Solution**: Infrastructure will not permit an instance to send traffic with a source IP or MAC address other than its own.

How secure is encryption Scheme:

- Is it possible for all of my data to be fully encrypted?
- What algorithms are used?
- Who holds, maintains and issues the keys? Problem:
- Encryption accidents can make data totally unusable.
- Encryption can complicate availability Solution
- The cloud provider should provide evidence that encryption schemes were designed and tested by experienced specialists.

Information Security

- Security related to the information exchanged between different hosts or between hosts and users.
- This issues pertaining to secure communication, authentication, and issues concerning single sign on and delegation.
- Secure communication issues include those security concerns that arise during the communication between two entities.
- These include confidentiality and integrity issues. Confidentiality indicates that all data sent by users should be accessible to only "legitimate" receivers, and integrity indicates that all data received should only be sent/modified by "legitimate" senders.

Solution: public key encryption, X.509 certificates, and the Secure Sockets Layer (SSL) enables secure authentication and communication over computer networks.

IV. CONCLUSION

In this paper we have worked to facilitate the client in getting a proof of integrity of the data which he wishes to store in the cloud storage servers with bare minimum costs and efforts. Our scheme was developed to reduce the computational and storage overhead of the client as well as to minimize the computational overhead of the cloud storage server. We also minimized the size of the proof of data integrity so as to reduce the network bandwidth consumption. Many of the schemes proposed earlier require the archive to perform tasks that need a lot of computational power to generate the proof of data integrity. But in our scheme the archive just need to fetch and send few bits of data to the client.

V. REFERENCES

[1] Beginning ASP.NET 4: in C# and VB by Imar Spaanjaars

[2] ASP.NET 4 Unleashed by Stephen Walther

[3] Programming ASP.NET 3.5 by Jesse Liberty, Dan Maharry, Dan Hurwitz

[4] Beginning ASP.NET 3.5 in C# 2008: From Novice to Professional, Second Edition by Matthew MacDonald

[5] Amazon Web Services (AWS), Online at http://aws. amazon.com

[6] Google App Engine, Online at http://code.google.com/ appengine

[7] Microsoft Azure, http://www.microsoft.com/azure/.

[8] A. Agrawal et al. Ws-bpel extension for people (bpel4people), version 1.0., 2007.

[9] M. Amend et al. Web services human task (ws-humantask), version 1.0., 2007.

[10] D. Brabham. Crowdsourcing as a model for problem solving: An introduction and cases.

[11] Data Communications and Networking, by Behrouz A Forouzan.

[12] E. Mykletun, M. Narasimha, and G. Tsudik, "Authentication and integrity in outsourced databases," Trans. Storage, vol. 2, no. 2, pp. 107–138, 2006.

[13] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in SP '00: Proceedings of the 2000 IEEE Symposium on Security and Privacy. Washington, DC, USA: IEEE Computer Society, 2000

[14] A. Juels and B. S. Kaliski, Jr., "Pors: proofs of retrievability for large files," in CCS '07: Proceedings of the 14th ACM conference on Computer and communications security. New York, NY, USA: ACM, 2007, pp. 584–597.

[15] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in CCS '07: Proceedings of the 14th ACM conference on Computer and communications security. New York, NY, USA: ACM, 2007, pp. 598–609.