Data invulnerability and data integrity verification of multi cloud storage

N.Janardhan[1], Y.Rajasree[2], R .Himaja[3],
[1]*Assistant Professor, K L University, Guntur, Andhra Pradesh, India*
[2] *Computer Science and Engineering, K L University, Guntur, Andhra Pradesh, India*
[3] *Computer Science and Engineering, K L University, Guntur, Andhra Pradesh, India*

***Abstract*— Cloud computing enables highly scalable services to be easily consumed over the Internet on an as-needed basis. A major feature of the cloud services is that users' data are usually processed remotely in unknown machines that users do not own or operate. In this paper, we address the construction of an efficient PDP scheme for distributed cloud storage to support the scalability of service and data migration, in which we consider the existence of multiple cloud service providers to cooperatively store and maintain the clients' data [1]. .Offering strong data protection to cloud users while enabling rich applications is a challenging task. We explore a new cloud platform architecture called Data Protection as a Service, which dramatically reduces the per-application development effort required to offer data protection, while still allowing rapid development and maintenance. We prove the security of our scheme and we also provide distributed auditing mechanisms. We provide extensive experimental studies that demonstrate the efficiency and effectiveness of the proposed approaches.**

I. INTRODUCTION

Cloud storage can be an attractive means of outsourcing the day-to-day management of data, but ultimately the responsibility and liability for that data falls on the company that owns the data, not the hosting provider. With this in mind, it is important to understand some of the causes of data corruption, how much responsibility a cloud service provider holds, some basic best practices for utilizing cloud storage safely, and some methods and standards for monitoring the integrity of data regardless of whether that data resides locally or in the cloud. Integrity monitoring is essential in cloud storage for the same reasons that data integrity is critical for any data centre.

Data corruption can happen at any level of storage and with any type of media. Bit rot (the weakening or loss of bits of data on storage media), controller failures, reduplication metadata corruption, and tape failures are all examples of different media types causing corruption. Metadata corruption can be the result of any of the vulnerabilities listed above, such as bit rot, but are also susceptible to software glitches outside of hardware error rates [2]. Unfortunately, a side effect of reduplication is that a corrupted file, block, or byte affects every associated piece of data tied to that metadata. The truth is that data corruption can happen anywhere within a storage environment.

Data can become corrupted simply by migrating it to a different platform, i.e., sending your data to the cloud. Cloud storage systems are still data centres, with hardware and software, and are still vulnerable to data corruption. One needs to look no further than the recent highly publicized Amazon failure. Not only did many companies suffer from prolonged downtime, but 0.07 percent of their customers actually lost data. It was reported that this data loss was caused by 'recovering an inconsistent data snapshot of … Amazon ESB volumes. What this translates to is that data in Amazon's system became corrupted, and as a result, customers lost data.

Confinement: A secure data capsule (SDC) is an encrypted data unit packaged with its security policy.

For example, an SDC might encompass a sharable document or a photo album along with its ACL. The platform can use confinement and information-flow controls to enforce capsules' ACLs. To avoid unauthorized leakage of user data in the presence of potentially buggy or compromised applications, DPaaS confines the execution of applications to mutually isolated secure execution environments(SEEs).Inter-SEE isolation has different levels, but stronger isolation generally exacts a greater performance cost due to context switching and data marshalling. At one end, a SEE could be a virtual machine with an output channel back to the requesting user. For performance reasons, it's possible to have a pool of VMs or containers in which the data state is reset before being loaded with a new data unit similar to how a thread pool works in a traditional server.

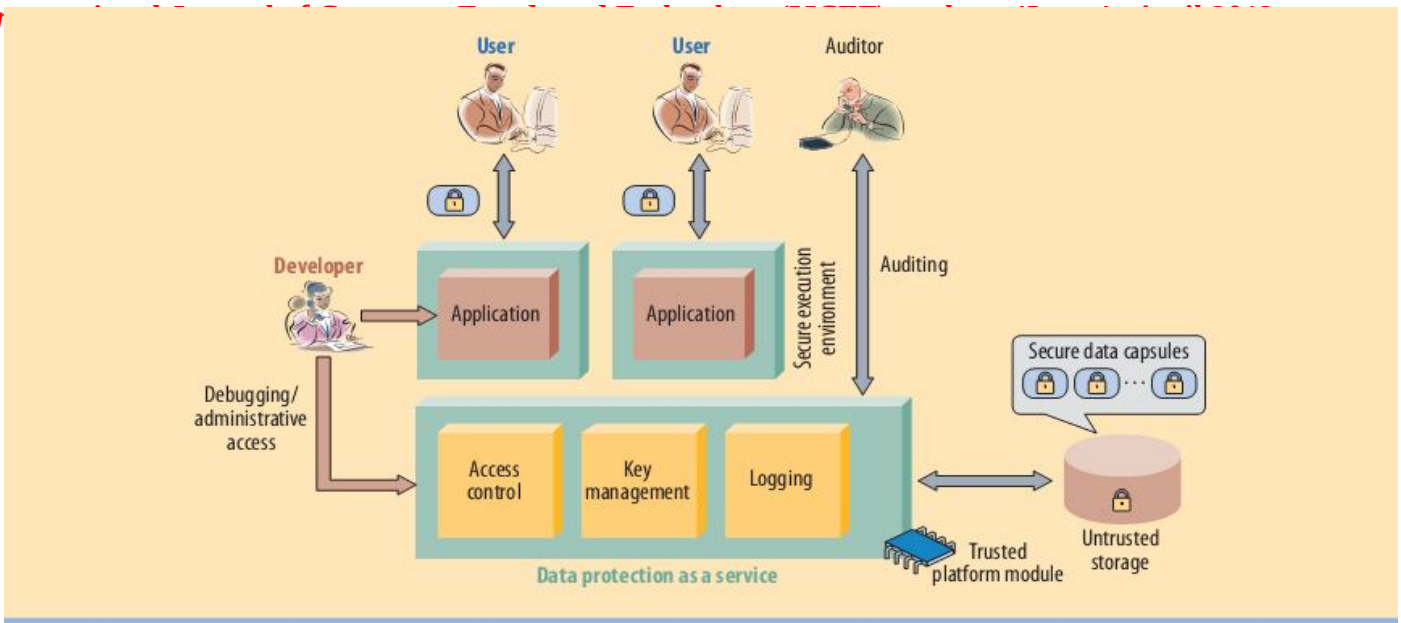

Figure 1. Sample architecture for data protection as a service illustrates how it's possible to integrate various technologies, such as application confinement, encryption, logging, code attestation, and information flow checking to realize DPaaS.

II.RELATED WORK

A more lightweight approach would be to use OS process isolation; an even lighter-weight approach would be to use language-based features such as information-flow controls or capabilities[3].

We can use mechanisms such as C and java for JavaScript to confine user data on the client side as well, al-though we don't include that option as part of the platform. In some cases, applications need to call outside services or APIs provided by third-party websites for example, the Google Maps API. An application might need to export users' data to outside services in this process. Users can explicitly define privacy policies to allow or disallow ex-porting SDCs to such third-party services, and DPaaS can enforce these policies.

Additionally, DPaaS can log all in-stances where data is exported, and an auditor can later inspect these logs. Because our target applications have a basic requirement of sharable data units, DPaaS supports ACLs on SDCs. The key to enforcing those ACLs is to control the I/O channels available to the SEEs. To confine data, the platform decrypts the SDC's data only in a SEE in compliance with the SDC's security policy.

A SEE can funnel the output either directly to the user or to another SEE that provides a service; in either case, the platform mediates the channel. A buggy SEE only exposes a single SDC, an improvement over systems in which malicious input triggers a bug that allows access to all data. The platform also mediates ACL modifications, other-wise known as sharing or not sharing. A simple policy that the platform can enforce without having to know too much about the application is transitive: only currently authorized users can modify the ACL. For example, the creator is the first owner of a data unit, and at any time, any user with the owner status can add or revoke other authorized users. The support of anonymous sharing, in which pos-session of, say, a secret URL grants access to data, is also straightforward.

Whenever data is lost, especially valuable data, there is a propensity to scramble to assign blame. Often in the IT world, this can result in lost jobs, lost company revenue, and, in severe cases, business demise. As such, it is critical to understand how much legal responsibility the cloud service provider, per the service level agreement (SLA), has and to ensure that every possible step has been taken to prevent data loss[4]. As with many legal documents, SLAs are often written to the benefit of the provider, not to the customer. Many cloud service providers offer varying tiers of protection, but as with any storage provider they do not assume liability for the integrity of your data.

Cloud SLA language that contains explicit statements protecting the cloud provider if data is lost or corrupted is common practice. An example of this language is found in the Amazon Customer Web Services agreement, which states, "WE… MAKE NO REPRESENTATIONS OR WARRANTIES OF ANY KIND … THAT THE SERVICE OFFERINGS OR THIRD PARTY CONTENT WILL BE UNINTERRUPTED, ERROR FREE OR FREE OF HARMFUL COMPONENTS, OR THAT ANY CONTENT … WILL BE SECURE OR NOT OTHERWISE LOST OR DAMAGED." In fact this agreement even goes as far as to suggest that a customer make "frequent archives" of their data. As mentioned before, the responsibility for managing the integrity of data, whether in a data centre, private cloud, hybrid cloud or public cloud always falls on the company that owns the data?

There are some common sense best practices that will allow a company to take advantage of the flexibility and accessibility of the cloud, without putting its data at risk. The premise of data protection is to distribute the risk so that the probability of data loss is minimized. Even when storing data in the cloud, it makes sense to keep a primary copy and a backup copy of the data onsite so that access to the data is not dependent upon network performance or connectivity. By adhering to these basic best practices and knowing the details of the cloud provider's SLA, the building blocks are in place to implement a method for proactively monitoring the integrity of data regardless of the storage platform or location.

One method for verifying the integrity of a set of data is based on hash values. A hash value is derived by condensing a set of data into a single unique value by way of a pre-defined algorithm. Since the hash value is derived from the original data itself, if the two hash values are not identical, it is an indicator that at least one of the two copies has been either altered or corrupted.

Make sure that the cloud provider provides the ability to check the hash value of the data and compare it to the hash value of a second copy of data, regardless of where that copy is stored. Undertaking this level of data monitoring manually would be beyond cumbersome. Fortunately, other methods are available, including programmatic checks. Spectra Logic and the other members of the Active Archive Alliance offer tools that will automatically monitor the integrity of the data within their systems.

While an active archive is one method of monitoring data integrity, there remains a critical need for a widely adopted cloud standard protocol that supports integrity monitoring and interoperability. Because not all data centres have homogeneous equipment, nor are they necessarily homogeneous to the cloud hosting infrastructure, interoperability between different storage devices is crucial. The Cloud Data Management Interface (CDMI) standard was put forth in 2010 by the Storage Networking Industry Association (SNIA). A CDMI-compliant system can query another CDMI compliant system for the hash value of an object, thus verifying that the two copies of data are still identical. By monitoring the integrity of the primary copy of data with a backup copy, a company can now verify that the copy of data stored in the cloud has not been corrupted. How frequently these data sets need to be monitored can be determined by the value of the data. Industry standards, such as CDMI, not only ensure interoperability between compliant heterogeneous systems, but also provide a convenient mechanism for data integrity monitoring.

It's hard to dispute that the cloud industry has taken a few punches in the media recently, especially with large vendors like Iron Mountain discontinuing their basic cloud storage services and the previously discussed data loss at Amazon S3. However, the moral of this story isn't that the cloud is an unwise storage platform, but rather that when investigating and implementing cloud strategies, there are more factors to consider than simply cost per gigabyte stored [5].

Cloud storage offers many advantages to companies of any size when properly implemented. What cloud doesn't do is eliminate the need for intelligent data management strategies. Regardless of how or where data is stored, it is absolutely crucial to make certain it will be accessible and restorable when needed. This assurance is at the very heart of data integrity monitoring and verification.

III.SECURITY AND PRIVACY CHALLENGES

It's impossible to develop a single data-protection solution for the cloud because the term means too many different things. Any progress must first occur in a particular domain—accordingly, our work focuses on an important class of widely used applications that includes e-mail, personal financial management, social networks, and business tools such as word processors and spreadsheets [6]. The following criteria define this class of applications: •• provide services to a large number of distinct end users, as opposed to bulk data processing or workflow management for a single entity; •• use a data model consisting mostly of sharable units, where all data objects have access control lists (ACLs) with one or more users; and •• developers could run the applications on a separate computing platform that encompasses the physical infrastructure, job scheduling, user authentication, and the base software environment, rather than implementing the platform themselves. Overly rigid security is as detrimental to cloud service value as inadequate security.

IV REFERENCES

[1]C. Dwork, "The Differential Privacy Frontier Extended Abstract," Proc. 6th Theory of Cryptography Conf.(TCC 09), LNCS 5444, Springer, 2009, pp. 496-502.

[2] C. Gentry, "Fully Homomorphic Encryption Using Ideal Lattices," Proc. 41st Ann. ACM Symp. Theory Computing (STOC 09), ACM, 2009, pp. 169-178.

[3]E. Naone, "The Slow-Motion Internet," Technology Rev., Mar./Apr.2011; www.technologyreview.com/files/54902/ GoogleSpeed_charts.pdf.

[4] A. Greenberg, "IBM's Blindfolded Calculator," Forbes, 13 July 2009; www.forbes.com/forbes/2009/0713/breakthroughs-privacy-super-secret-encryption.html.

[5] P. Maniatis et al., "Do You Know Where Your Data Are? Secure Data Capsules for Deployable Data Protection," Proc. 13th Usenix Conf. Hot Topics in OperatingSystems(HotOS11),Usenix,2011; www.usenix.org/events/hotos11/tech/final_files/ManiatisAkhawe.pdf.

[6] S. McCamant and M.D. Ernst, "Quantitative Information Flow as Network Flow Capacity," Proc. 2008 ACM SIGPLAN Conf. Programming Language Design and Implementation (PLDI 08), ACM, 2008, pp. 193-205.