Secure Storage & Access of Information in Cloud-Storage with Storage as a Module (SaaM)

William Briguglio & Farhan Mahmood Babar[1]

Abstract— **Recently there has been growing interest in cloud computing as recent technological advancements in networking have made the idea of sending information to a remote server to be processed or stored more attractive. Many still consider cloud computing in its infancy and expect it to mature in the coming decade as large organizations such as Amazon, Apple, and Google begin to offer increasingly powerful cloud computing and storage services. Already, terabytes of sensitive information, whether it be purchasing history, personal documents, medical data, etc., are being stored and processed in the cloud. Cloud computing has enormous potential as companies, consumers, and researchers can leverage the immense processing power and practically endless storage of large remote servers. With so much information currently being stored or processed in "the cloud" and with far more throughput expected in the coming decade, it is especially important that these cloud computing services are made secure. There are also various security challenges in adopting cloud computing. In this paper we identify and provide an overview of cloud computing, as well as take a brief look at some of the security challenges presented with cloud computing, and some proposed architectures which attempt to solve these challenges. Additionally, we proposed an architecture (SaaM) for secure storage and retrieval of information in cloud computing.**

Index Terms— **Cloud Computing, Cloud Storage, Cloud Security, Cloud Architecture**

I. INTRODUCTION

Recently there has been a large increase in networking capabilities that has been making new computing paradigms practical. One such example is Cloud Computing. Cloud computing has been on a steady rise and now plays a large role in both personal and business applications. Various home PC and mobile phone retailers have cloud storage applications preinstalled as a primary store for photographs, videos, and music. More over, several large companies such as Google, Microsoft, and Amazon offer a wide range of cloud computing services for both personal use and use by large enterprises. Cloud computing's wide spread adoption is a testament to its expediency, however with new territory comes new threats, and cloud computing is no different.

Cloud computing is a networking and software architecture and is one of the most promising recent advancements in distributed systems. The term typically refers to data centers which provide on demand access to computing resources and services for numerous clients over the Internet. Private users and enterprises alike are attracted to cloud computing as it offers a minimal cost alternative for numerous problems which traditionally required expensive, large scale hardware solutions. Users of the cloud are not required to manage its necessary infrastructure as cloud computing is typically a third party service "rented" by various users in a multi-tenant model. This is where cloud computing's strength lies, since it allows clients, who would other wise have to spend large amounts of money purchasing hardware that goes underutilized most of the time, to leverage large amounts of computing resources on an as needed basis. Further more this hardware would require around the clock management and maintenance who's cost can now be split between the multiple tenants in the cloud. Numerous ventures, for example banking, social insurance, and education, are moving towards the cloud because of the cost-effective administration facilitated by its per-use design. There are various reasons for organizations to move towards cloud solutions and the flexibility they provide and because of this it is important to ensure this powerful new programming paradigm is secure.

One common use case for cloud computing is cloud storage, an architecture where distributed data centers provide virtualized storage for anywhere from a couple dozen employees of a single organizations, to millions of public users. The storage is abstracted as a single location for clients to store their data and from the cloud user's perspective, can be thought of as rented hard drive that is accessible through the internet. Many small to large enterprises take advantage of third party cloud storage systems to provide a company wide data store, which is accessible over the internet. Cloud storage vendors claim to implement encryption of user data stored on the cloud and various other security features. However, there are numerous situations were a client would like to store highly sensitive data on the cloud, and in accordance with the mutual suspicion security practice, must treat any 3rd party application or service as potentially malicious. In this paper we propose an architecture which allows enterprises to use third party cloud storage platforms while maintaining security given the mutual suspicion assumption. First, we provided a over view of cloud computing and its definition. Next we discuss related works, and point out some of their disadvantages. Then we introduce our proposed architecture, SaaM, and discuss its strengths and improvements over other solutions. Finally we finish with a discussion of future work and a conclusion.

[1]W. Briguglio (briguglw@uwindsor.ca) and F. M. Babar (babar111@uwindsor.ca) are with the School of Computer Science, University of Windsor, ON N9B3P4, Canada.

II. CLOUD COMPUTING MODEL

Cloud computing is defined in [1] by the National Institution of Standards and Technology (NIST) as:

> a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources(e.g., networks, servers, storage, applications, and services)that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models.

The paper goes on to define these characteristics, service models, and deployment models, which we will briefly summarize here.

A. Essential Characteristics of Cloud

1) *On-demand self-service:* User can use computing capabilities as needed without requiring human interaction with the cloud service provider.
2) *Broad network access:* Capabilities are available over the internet and accessed through standard mechanisms, e.g. web browsers, by a variety of devices.
3) *Resource pooling:* The providers computing resources are pooled and dynamically allocated to serve multiple users, who generally have no knowledge of the location of the resources.
4) *Rapid elasticity:* Capabilities can be allocated and reallocated in real-time. To the user, the capabilities appear unlimited and constantly available.
5) *Measured Service:* Cloud systems automatically control and optimize resources by usage monitoring.

B. Cloud Computing Services

1) *Software as a Service (SaaS):* The consumer can use the providers applications running on a cloud infrastructure and has no control over underlying infrastructure. The cloud Service provider (CSP) offers application-level services, available through a thin client, which are tailored to a wide variety of business needs. E.g. Web based email, Salesforce Essentials, iCloud.
2) *Platform as a Service (PaaS):* The consumer can deploy onto the cloud infrastructure their own applications created using programming languages and tools supported by the provider with components such as web servers, database management systems, and software development kits for various programming languages. The consumer has no control over underlying infrastructure but may have control over hosting environment configurations. E.g. SalesForce Lightning Platform, Google App Engine, and Microsoft Azure.
3) *Infrastructure as a Service (IaaS):* The consumer can provision processing, storage, networks, and other fundamental computing resources in order to deploy and run arbitrary software. The consumer has no control over underlying infrastructure but has control over OSs, storage, deployed applications, and some networking components. The service provider owns the equipment and is responsible for housing, running and maintaining it. E.g. Amazon Elastic Cloud Computing (EC2), Amazon S3, GoGrid.

C. Cloud Computing Deployment Models

1) *Private Cloud:* The cloud infrastructure is operated solely for an organization, may or may not be managed by a third party, and may be on or off site. It is increasingly secure but costly when contrasted with public cloud.
2) *Community Cloud:* The cloud infrastructure is shared by several organizations and may be managed by a third party. A community cloud is similar to a private cloud but the main difference is the set of users. While a private cloud is used by a single organization, a community cloud is shared by a few organizations with shared concerns.
3) *Public Cloud:* The cloud is available to the public or large industry group and is owned by an organization selling cloud services. These models are cheaper since they are shared amongst many users, but because of this, are also less secure relative to private clouds.
4) *Hybrid Cloud:* The cloud infrastructure is composed of two or more clouds which are unique entities but bound together by standardized technology which enables data and application portability.

III. LITERATURE REVIEW

Following is a review of related work on the security challenges of Cloud Computing and different architectures proposed as a solution to these challenges.

Chen and Zhao [2] describe the added data security and privacy issues that Cloud Computing brings, from the perspective of the data life cycle. Following is a summary of the key issues put forward in the paper, which are associated with each stage of the life cycle.

1) *Generation:* Consider how to maintain data ownership once it is migrated to the cloud.
2) *Transfer:* Transmission of data to the cloud should use protocols which ensure data integrity and confidentiality.
3) *Use(availability):* Encryption of data is feasible for cloud-storage but for other applications, encryption may cause data to be impractical to use.
4) *Share:* Data owners must ensure the cloud service, and the parties the service shares with, maintain the owners protection standards.
5) *Storage:* Key management and the overhead that comes with encrypting data for a multitude of users should be considered.
6) *Archival:* Ensure storage duration is consistent with archival requirements.
7) *Destruction:* Ensure data is properly destroyed.

Arjun Kumar et al. [3] proposed an architecture for securely storing data within the cloud where the Cloud Service Provider, after the authenticating the users, sends a cryptographic module to the client to perform cryptographic operations. In this proposed architecture, the user provides a pin to the cryptophraphic module to generate a secret key which is used to perform encryption/decryption while sending/receiving data to/from the cloud. There were some disadvantages which we identified in the proposed architecture. One was the use of asymmetric key cryptography, which is computationally expensive when compared to symmetric key encryption and may be too slow for this application. Additionally, the cryptographic keys are generated by pins which must be remembered and managed by the user. Lastly, the cryptopgrahic module is supplied by the CSP. This provides an opportunity for a man in the middle attack which compromises the module while it is being transferred to the client. Also, it violates the principle of mutaul suspicion since the CSP itself may be a malicious actor and can intentionally send a compromised cryptographic module. Lastly, this requires cooperation with the CSP since they must integrate the cryptographic module into their systems.

Sudeepa et al. in [4] proposed a similar method where users upload and download files via a web page in their browser. A key difference in this architecture is the encryption is done via AES which is a well trusted and more efficient algorithm then public key encryption. However, the code for cryptographic functions is still supplied by the CSP and thus the same security issues are present. The key used for encryption is inputted by the user, so key management and distribution is also still a problem.

IV. PROPOSED ARCHITECTURE: SAAM

Our proposed architecture, which we feel improves on the designs discussed in [3] and [4], is referred to as **S**torage **a**s **a** **M**odule, or SaaM. Our architecture draws on trusted software design principles such as modularization, low coupling, and cohesiveness to provide a more intuitively secure and practical cloud storage infrastructure. In our architecture, the cloud storage faculties are integrated as a module, called the **C**loud **S**torage **M**odule or CSM. The CSM is responsible solely for the storage of data received from enterprise endpoints. This makes implementation simple and intuitive. Other duties, such as key management and cryptography, are handled by the **E**nterprise **S**erver. The ES therefore has full knowledge and control over how data sent to the CSM is encrypted, giving the enterprise confidence in their data's security. Furthermore, the low coupling and narrow responsibilities of the CSM mean that practically any CSP could be used to fulfil the role of CSM, allowing small enterprises with fewer resources to safely use cloud storage services. Next we break down the architecture into its four main components and describe their roles and the advantages of SaaM from the perspective of each component.

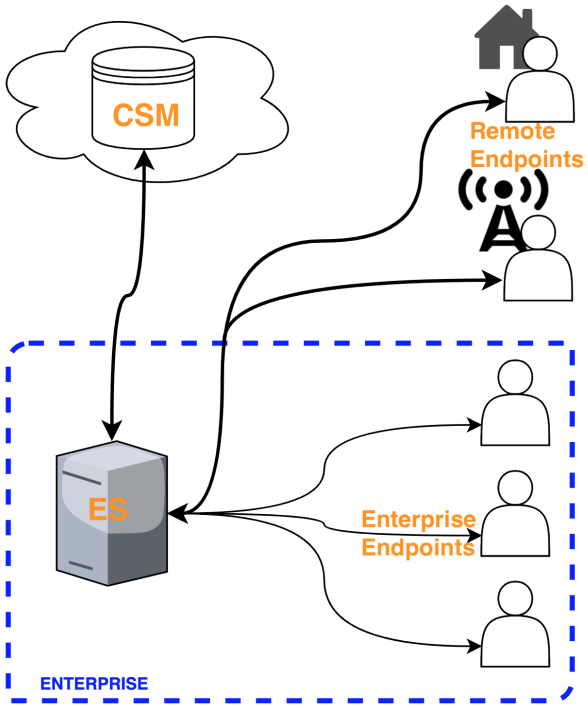

Fig. 1. Storage as a Module Architecture

Enterprise Server (ES)

The enterprise server uploads/retrieves data to/from the CSM. It is responsible for authenticating users who access from either the enterprise endpoints or remote endpoints. This authentication can be done with traditional means such as user name and password. The ES would maintain the credentials for accessing the CSM so the only direct access to the CSM would be through the ES. This means users only need to to be aware of their authentication credentials, no extra responsibility is put on users relative to traditional storage systems. The server could be implemented so that authenticated users simply drag files they wish to be stored on the cloud into a specially marked folder on their desktop, similar to github, dropbox, etc.. Before sending the data to the CSM, the ES would encrypt the data, and after retrieving data from the CSM, the ES would decrypt it before sending it to an endpoint. The data would be sent to these endpoints using traditional secure data transfer protocols such as HTTPS over TLS.

The advantages here are that the enterprise has full control over the encryption/decryption process, and their ES serves as a proxy server between the outside world and their data stored in the cloud. This means the enterprise can decide what cryptographic algorithm they will use, addressing the issue of asymmetric cryptography which was present in [3]. This also means the enterprise can freely create complex custom functionality, even when using a cheap base-line CSP service. For example, the enterprise may choose to reduce stress on the ES by implementing manual or automatic processes for distinguishing between data which is sensitive

and must be encrypted, and data that can be sent straight to the CSM as plain text. Another stress reducing solution could be to configure the ES to leverage enterprise endpoint hardware for the encryption and decryption of data. If the enterprise wants more granular control over who is allowed to access what, they can enforce a tiered security clearance system with different encryption/decryption keys used for each tier.

Keep in mind, all of this functionality can be implemented without needing users to know anything other then their password and username (addressing the issue of key/password management which is present in [3] and [4]), and without the necessity of purchasing expensive PaaS or IaaS services to implement custom functionality on the CSP side. Also, since everything is done by the ES, unless the CSP finds a way to decrypt the data it receives, even an untrusted CSP can be securely utilized as a CSM. This addresses the other issue in [3] and [4] relating to the cryptographic code originating from the CSP.

CSM

The CSM is simply any cloud service which can store and retrieve data while maintaining integrity and availability. Its only responsibility to the enterprise is to store encrypted data received from the ES, and send requested encrypted data to the ES. There are many advantages of treating the cloud storage service as essentially a virtualized off-site hard drive. It means that cheaper cloud storage services can be used. If the CSP is a private cloud, then the modularization makes implementation of the enterprise information and cloud storage infrastructures easier. If it is a public or community cloud, then each enterprise can have separate functionality while all using the same CSM. Finally, since the CSM's role is so simple, it is easy to provide security even with the assumption that the CSP is untrusted. For example, if the CSP is maliciously engaging in corporate espionage, then they cannot steal any data since it is encrypted before they receive it. Thus confidentiality is ensured. Of course the CSP could delete or alter the cipher text, attacking availability and integrity, but this is a moot point. In reality, a CSP would be an organization subject to rules and regulations and it would be too easy to prove that integrity or availability was lost due to their negligence and not the enterprise's.

Enterprise Endpoints

Enterprise endpoints make requests to the ES to either encrypt and send data to be stored in the CSM, or retrieve and decrypt data stored in the CSM. The ES would handle authenticating the user of the endpoint and would also manage keys used for encryption and decryption. It would keep track of what privileges different user accounts have and allow or deny access accordingly. None of the users need to know anything about the CSM access credentials and so users privileges can be revoked instantly, without the fear that the CSP would experience latency in updating privileges, causing a data leak. Furthermore, since the user has no direct access to the CSM, modifications and deletions can be restricted or backups maintained so that attacks on availability or integrity from inside the enterprise become much more difficult.

Remote Endpoints

Remote endpoints are just like enterprise endpoints, in that they make requests to the ES, except that these requests and data transfers take place remotely, over HTTPS, outside of the enterprise network. The same advantages discussed in the enterprise endpoints subsection apply. In addition, the enterprise can make only certain files available for remote access, or disable remote access altogether.

SaaM from the Data Life cycle Perspective

As discussed earlier, [2] outlined the added security concerns associated with cloud services from the perspective of the data life cycle. Here we discuss how our architecture addresses these concerns:

1) Generation: maintain data ownership in the cloud
2) Transfer: transmission of data to the cloud should ensure data integrity and confidentiality
3) Use: encryption of data is feasible for cloud-storage
4) Share: data owners must ensure the cloud service, and the parties the service shares with, maintain the owners protection standards
5) Storage: key management and the overhead that comes with encrypting data for a multitude of users should be considered.
6) Archival: ensure storage duration is consistent with archival requirements
7) Destruction: ensure data is properly destroyed

Point 3 is dealt with since of our architecture is used for cloud-storage and not cloud services as a whole. Point 2 is dealt with since the ES provides confidentiality by encryption and the data would be sent to the CSM by HTTPS with TLS which integrity checks built in. Point 5 is handled by the ES in our architecture and is much simpler in our case since only keys for a single organisation need to be managed. Furthermore, each ES can update their encryption keys according to their own schedule. In regards to point 1,4,6, and 7. These can not be guaranteed but the ES can keep detailed logs of data manipulation commands sent to the CSM so that the CSP may be held accountable for prematurely deleting archives or not properly deleting data. The logs would also help with claiming data ownership. Also, since the data is encrypted, if the CSP does go against their word and share the data without the enterprises knowledge, so long as the encryption isn't broken, there then will be no security concern.

V. FUTURE WORK

For future work, there is one major focus. Since the implementation can be accomplished with mostly tried and tested algorithms, software, and processes, the only not quite tested portion of the architecture specification would be the the encryption and decryption overhead. Therefore, Research should be conducted on the amount of hardware

resources required for timely encryption and decryption of data. Another area for future work could be the definition of efficient algorithms for ensuring data integrity in cloud-storage. Although our architecture can ensure data integrity during transfer from the ES to the CSM, it is not clear how to efficiently ensure integrity of data stored on the cloud.

VI. CONCLUSION

Cloud services are relatively new and increasingly important part of many enterprises today. However, due to their multi-tenant model, they bring a host of security and privacy concerns. Our architecture attempts to address these concerns in cloud storage services while also providing a modularized, practical, and flexible model. ISPs are concerned with the exchange of information in a timely manner between users, CSPs are concerned with the timely storage and retrieval of information for its users. We dont expect ISPs to provide security and integrity, rather we use and implement protocols and software to ensure it ourselves. We believe our architecture takes an analogous approach to cloud storage providers.

REFERENCES

[1] Peter Mell, Tim Grance, "The NIST Definition of Cloud Computing". Version 15, July, 2009, National Institute of Standards and Technology, Information Technology Laboratory.

[2] Dayan Chen, Hong Zhao, "Data Security and Privacy Protection Issues in Cloud Computing". 2012 International Conference on Computer Science and Electronics Engineering.

[3] Arjun Kumar, Byung Gook Lee, HoonJae Lee, Anu Kumari, "Secure Storage and Access of Data in Cloud Computing". 2012 ICT Conference.

[4] Sudeepa R, Dr. H S Guruprasad, "Effective Secure Storage and Retrieve In Cloud Computing". IRACST International Journal of Advanced Computing, Engineering and Application, Vol 3, No 3, June 2014.