SecureCloud: A Secure File System for Cloud

Mukesh Kumar Gupta
Jamia Hamdard,
New Delhi, India
mukeshgupta@dce.ac.in

ABSTRACT

Now a days cloud computing is one of the best innovative technologies having every kind of service to the users. Cloud Computing is an idea of utilizing the shared pool of configurable IT and computing resources as an on-demand service and network access basis. The key concept of cloud computing is based on “pay per usage”. Cloud computing enables us to utilize high end resources so we have to build great application without worrying about infra-structure. Generally a scalable cloud storage services have a hybrid community cloud such as beneficial network storage. The convenience of using these facts, ideas and models are more attractive to the end users. Today the main concerns or issues related to about the privacy, reliability, security, evaluation and other quality factors. In this paper, we introduce some facts and results for next-generation equally distributed cloud storage, cloud computing and processing applications.

General Terms

Legal Aspects, Security Algorithms, Reliability

Keywords

Cloud Computing, Cloud Architecture, Service Models, Deployment Models, Security, File Systems

1. INTRODUCTION

Cloud community networks are an emergent model of infrastructure that aims to satisfy a community’s demand for Internet access and ICT services.

The sharing of other computing resources like storage, which is now common practice in today’s Internet through cloud computing, hardly exists in community networks[3], [4], but it can be made possible through clouds in community networks[5], like hybrid cloud.

The concept of community clouds [1], [2], has been introduced in its generic form before, as a cloud deployment model in which a cloud infrastructure is built and provisioned for an exclusive use by a specific community of consumers with shared concerns and interests.

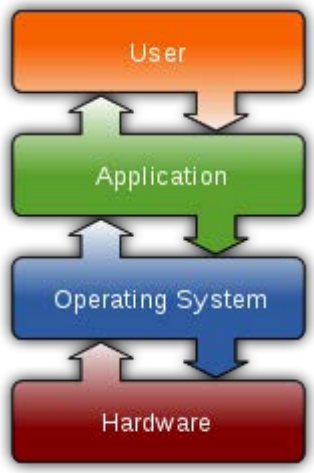

Fig 1: File System in a Computer System

Besides public, private and hybrid models of cloud computing, community cloud, presents another option differing from the others in that it is designed with a

specific community in mind and costs and responsibility are shared among the community members.

Generally private clouds are not always the most cost-effective option but security concerns, government legislation etc. need for sophisticated control [7], or enhanced performance requirements in many scenarios make it difficult to rely on public clouds.

Using the application models of cloud computing, we can refer here to a special kind of a community cloud in which sharing of computing resources is from within community networks, in general.

Realizing a community cloud involves a lot of challenges in both way either technological or socio-economic context, but also promises interesting value in proportion for communities in terms of local services and applications.

In order to conduct the evaluation of applications in a realistic scenario, our concern is to leverage on the infrastructure of cloud, provided by an on-going cloud deployment in the community network and a test bed deployed in the community cloud.

The contribution of this paper is an experimental study on the performance and feasibility of distributed file systems deployed on cloud-based infrastructures within community networks as well as over community networks.

Fig 2: Cloud Computing System Overview

There are a lot of security and trust related questions in the cloud environment i.e. data encryption possibility, data integrity checks, ensuring consumer's privacy, log access data, data failure[8], [9], disaster management, security policy, authentication methods, intrusion detection etc.

In order to ensure that the agreed requirements are continuously met during the period of a contract, the company has to invest in inspecting the service provider's security capabilities and perform ongoing detailed audits.

2. THEORETICAL REVIEW

Basically the concept of cloud computing started since

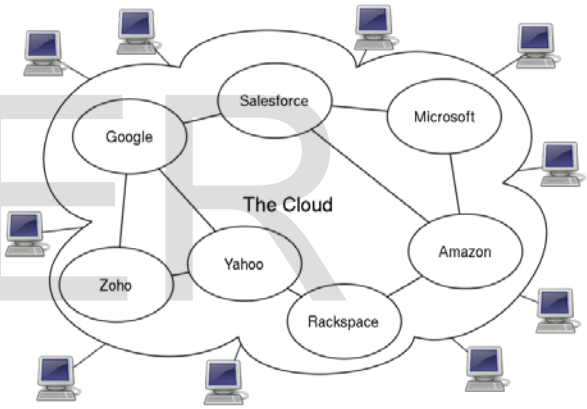

Fig 3: Generalize Cloud Computing System

1951 with the usage of centralized desktop computer, PCs open by means of static customers. Now cloud computing has been advanced from static users to basic ones and from programming to administrations. The term cloud indicates

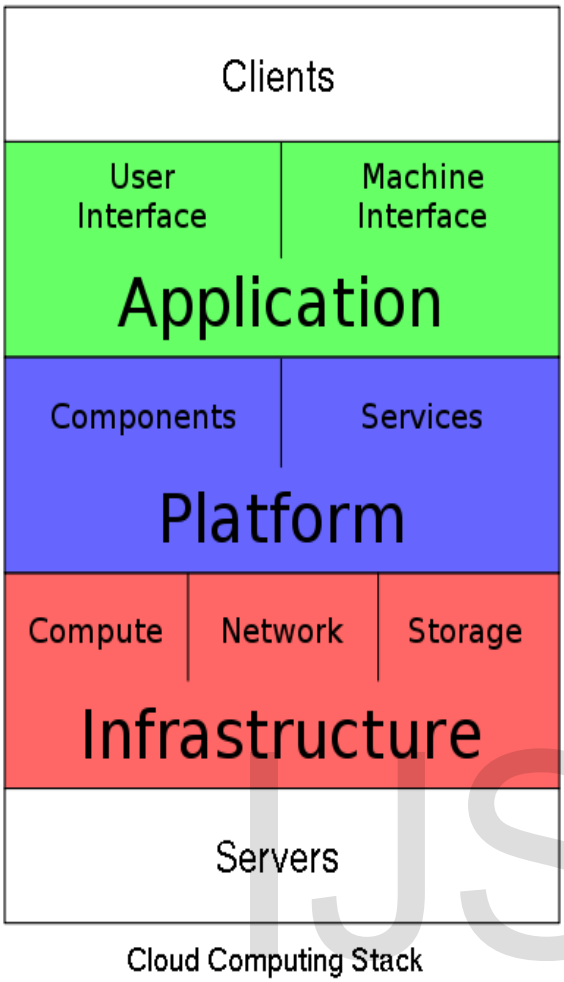

Fig 4: Cloud Computing Stack

to network or internet. Cloud is something that can be used for remote areas. Cloud gives benefits over open and personal systems [6]. Cloud computing is doing our business applications versatile and community networks. Cloud computing gives additionally online information stockpiling and application.

Distributed computing remotely offers controlling, designing, programming assets etc. It is a sort of internet based programming where shared computer hardware, information and data processing are given to computers as well as different gadgets on-request.

Cloud computing consists of two types of working models: Deployment models and service models. Deployment model defines types of access to the cloud that is public, private, hybrid and community network. Service model is divided into three basic services: Software as a Service, Platform as a Service and Infrastructure [10], as a Service that is why it is called SPI model.

3. THE PROBLEM STATEMENT

In a Cloud System, file systems must be secure because they provide critical system functionality. Almost all applications rely on the assumption that the underlying file system conforms to a standard interfaces and is bug-free. File systems are a significant source of security vulnerabilities. More importantly, the file systems defects are semantic bugs [11], [12] i.e. they require an understanding of file system semantics to be detected and fixed.

4. FILE ACCESS SEMANTICS - CONCURRENCY

The semantics of most cloud services like file systems, including HDFS, Google-FS, AmazonS3 and so on are optimized for their anticipated workloads. HDFS does not allow changes to a file once it is created, written and closed. These semantics favour Hadoop or Map-Reduce based application that simply manipulates collection of files etc. The lack of support for concurrent write sharing simplifies data consistency semantics [15] in HDFS. Though, unlike HDFS, Google-FS supports restricted file mutation by appending new data to a file including atomic operations [13] to support concurrent appends to a single file. Typically parallel file systems support a wide variety of operations, especially for highly concurrent file access.

5. BUFFERING AND CONSISTENCY

HDFS is designed to enable high write batch processing. It enables streaming writes through write staging at the client side. Clients send a write to a data server only when they have accumulated a chunk size of data. Initially, clients buffer all writes operations by redirecting them to a temporary file[14] in memory. Once filled the clients flush this buffer to the data server responsible for storing that file chunk. If the file is closed when the buffer is not yet full, the buffer is flushed to the chunk's respective data server.

6. ARCHITECTURE

The ground of our approach is to find a balance between benefiting from the cloud nature of pay-per-use and ensuring the safety of the company's data as well as the legality of the transactions performed over this data. The goal is to achieve such balance by automating the process of selecting a cloud provider and removing the auditing responsibility from the customer's side. Selecting a cloud provider involves: the definition, negotiation, monitoring, and enforcement of mutual expectations and agreements.

Our cloud computing architecture (fig 5) is based on following components:

6.1 Trusted Security Center (TSC):

Handling the legal and security concerns is both expensive and time consuming for companies. Therefore, we propose using a trusted third party in the cloud that is responsible for:

- supervising and attesting compliance of the legal requirements
- studying related security policies and service level agreements
- inspecting the facilities of the service provider

A cloud can have one or more TSCs that can share one or more service registries. As a result, moving a big part of the tasks required to ensure the security of the data to a specialized trusted user. We are planning to realize part of this architecture by expanding security patterns to include public legal patterns as an extension to our work.

6.2 Requirements Formalization Service:

Plenty of potential cloud users are non-IT specialists. Though most of them do have IT departments, it is difficult for them to adjust the way they formalize the company's requirements to fit the generic nature [15] of cloud environment. Especially if it involves new parameters that were not of issue in other environments. The service also transforms service provider's capabilities into a standardized form that is used.

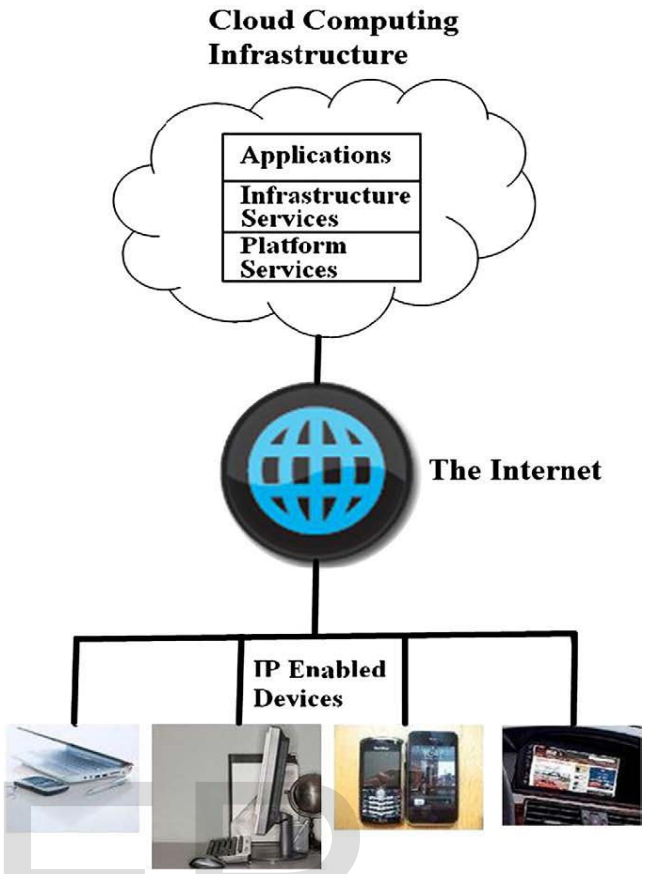

Fig 5: Cloud Computing Architecture

Once the requirements are formalized, the center asks the set of registered cloud providers for their formalized offers and uses the matching service to match formal requirements with provider's capabilities. The resulted list of providers is passed, along with the user's priority list, to the reputation service to be filtered according to providers' reputations. The user then receives the final list of potential providers.

In this section we see one of the architecture's components; the Requirements Formalization Service (RFS). RFS works for both the user side and the service provider side. It formalizes on data retention and security measures into a formal set of requirements and also formalizes service provider's capabilities (a non-formal description of his offers and policies). The main goal of the RFS is to facilitate between user and service provider description and components to enable the process of automatic service provider selection.

The security level classification step defines three levels of data security: high, medium, and low, defined as following:

- *High*: critical data not applicable to be kept beyond organizational boundaries and should be protected by high security measures, e.g ongoing research projects.

- *Medium*: less critical data which still requires protection, but the benefits of keeping it in the cloud outweigh the associated security risks (e.g project management data accessed in multiple locations in large organizations). This way the storage of data in a cloud is a balance between security and convenience.

- *Low*: the remaining data that is of importance to the organization but does not require over-protective security measures as long as it remains accessible to organization's employees or customers such as stock information, business reports, etc.

General security requirements can be specified as extension to the above three levels. Schumacher defined six security levels to specify the security requirements and determine risks. We use three levels only for the consumer's convenience. Nevertheless, it is applicable to map between both classifications as following:

- values *extreme* & *very high* to the value *high* (not suitable for keeping beyond organizational boundaries)

- values *high* to *low* to the value *medium* (potential security risks requires security guarantees from the cloud provider)

- values *negligible* to the value *low* (does not have security requirements), due to the absence of security risks associated with this type of data. However, further requirements can be defined like costs or availability.

Next, the user side should include further description for the data in each level. This description is required so the RFS can determine the geographical and legal restrictions on the data in each level. The consumer needs only to include the categorization like confidential, personal, and neutral.

7. CONCLUSION

The trust model of cloud computing should focus on the pre-failure rather than the post-failure of services. Therefore, it is necessary to offer technological guarantees to the client that the data and processing are not going to be compromised. Hence this paper shows for the development of efficient methods for processing encrypted and hidden data. Such methods will promote the client's trust in the cloud and will allow cloud service providers to provide quality of service assurances.

REFERENCES

[1] M. Armbrust, A. Fox, R. Gri th, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A view of cloud computing. Communications of the ACM, 53:50{58, April 2010.

[2] Y. Aumann and Y. Lindell. Security against covert adversaries: E cient protocols for realistic adversaries. Journal of Cryptology, April 7 2009.

[3] H. Bay, T. Tuytelaars, and L. Gool. Surf: Speeded up robust features. In A. Leonardis, H. Bischof, and A. Pinz, editors, Computer Vision a^AS ECCV 2006, volume 3951 of Lecture Notes in Computer Science, pages 404{417. Springer Berlin Heidelberg, 2006.

[4] I. F. Blake and V. Kolesnikov. One-round secure comparison of integers. Journal of Mathematical Cryptology, 3(1):37{68, May 2009.

[5] M. Blanton and M. Aliasgari. Secure outsourced computation of iris matching. Journal of Computer Security, 20(2):259{305, 2012.

[6] M. Blanton and P. Gasti. Secure and e cient protocols for iris and ngerprint identi cation. In Computer Security{ESORICS 2011, pages 190{209. Springer, 2011.

[7] W. M. B.S. Manjunathi. Texture features for browsing and retrieval of image data. Aug. 1996.

[8] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina. Controlling data in the cloud: outsourcing computation without outsourcing control. In Proceedings of the 2009 ACM workshop on Cloud computing security, CCSW '09, pages 85{90. ACM, 2009.

[9] C. W. C.M. Pun. Fast and robust, color feature extraction for content-based image retrieval. 2005.

[10] I. Damgard, M. Geisler, and M. Kr igaard. E cient and secure comparison for on-line auctions. In Proceedings of the 12th Australasian conference on Information security and privacy, pages 416{430. Springer-Verlag, 2007.

[11] I. Damgard, M. Geisler, and M. Kroigard. Homomorphic encryption and secure comparison. International Journal of Applied Cryptology, 1(1):22{31, 2008.

[12] I. Damgard, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. Smart. Practical covertly secure mpc for dishonest majority ^ASa or: Breaking the spdz limits. In J. Crampton, S. Jajodia, and K. Mayes, editors, Computer Security a^AS ESORICS 2013,

volume 8134 of Lecture Notes in Computer Science, pages 1{18. Springer Berlin Heidelberg, 2013.

[13] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft. Privacy-preserving face recognition. In Privacy Enhancing Technologies, pages 235{253. Springer, 2009.

[14] T. Ermakova, B. Fabian, and R. Zarnekow. Security and privacy system requirements for adopting cloud computing in healthcare data sharing scenarios. 2013.

[15] D. Evans, Y. Huang, J. Katz, and L. Malka. E cient privacy-preserving biometric identi cation. In Proceedings of the 17th conference Network and Distributed System Security Symposium, NDSS, 2011.

[16] J. Garay, B. Schoenmakers, and J. Villegas. Practical and secure solutions for integer comparison. In Proceedings of the 10th international conference on Practice and theory in public-key cryptography, pages 330{342. Springer-Verlag, 2007.

[17] C. Gentry. Fully homomorphic encryption using ideal lattices. In Annual ACM Symposium on Theory of Computing, pages 169{178, Bethesda, MD, USA, 2009.

[18] O. Goldreich. The Foundations of Cryptography, volume 2, chapter Encryption Schemes. Cambridge University Press, 2004.

[19] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. Journal of the ACM, 38:690{728, 1991.

[20] S. Goldwasser, S. Micali, and C. Racko . The knowledge complexity of interactive proof systems. In Proceedings of the 17th Annual ACM Symposium on Theory of Computing, pages 291{304, Providence, Rhode Island, U.S.A., May 6-8 1985.

[21] T. Y. H. Tamura, S. Mori. Texture features corresponding to visual perception. June 1978.

[22] K. Hashizume, D. G. Rosado, E. Fernandez-Medina, and E. B. Fernandez. An analysis of security issues for cloud computing. Journal of Internet Services and Applications, 4(1):1{13, 2013.

[23] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. Image indexing using color correlograms. In Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, pages 762{768. IEEE, 1997.

[24] Y. Huang, D. Evans, J. Katz, and L. Malka. Faster secure two-party computation using garbled circuits. In Proceedings of the 20th USENIX Security Symposium, page 35, August 2011.

[25] Y. Huang, J. Katz, and D. Evans. Quid-pro-quo-tocols: Strengthening semi-honest protocols with dual execution. In IEEE Symposium on Security and Privacy, pages 272{284, May 2012.

[26] M. S. Islam, M. Kuzu, and M. Kantarcioglu. Access pattern disclosure on searchable encryption: Rami cation, attack and mitigation. In Proceedings of the 19th Annual Network & Distributed System Security Symposium (NDSS), February 2012.

[27] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Filterbank-based ngerprint matching. Image Processing, IEEE Transactions on, 9(5):846{859, 2000.

[28] C. K. Milkolajczyk. A performance evaluation of local descriptors. Oct. 2005.

[29] J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In Advances in Cryptology{EUROCRYPT 2008, pages 146{162. Springer, 2008.

[30] K. Kumar and Y.-H. Lu. Cloud computing for mobile users: Can o oading computation save energy? Computer, 43(4):51 {56, april 2010.

[31] G.-H. Liu and J.-Y. Yang. Content-based image retrieval using color di erence histogram. Pattern Recognition, 46(1):188 { 198, 2013.

[32] D. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91{110, 2004.

[33] O. G. Luo Juan. A comparison of sift, pca-sift and surf. 2009.

[34] M. Naor and B. Pinkas. Oblivious transfer and polynomial evaluation. In Proceedings of the Thirty-rst Annual ACM Symposium on Theory of Computing, pages 245{254, Atlanta, Georgia, United States, 1999. ACM Press.

[35] A. E. Nergiz, M. E. Nergiz, T. Pedersen, and C. Clifton. Practical and secure integer comparison and interval check. In Proceedings of the IEEE Second International Conference on Social Computing, pages 791{799, 2010.

[36] V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft. Privacy-preserving ridge regression on hundreds of millions of records. In IEEE Symposium on Security and Privacy, May 2013.

[37] M. Osadchy, B. Pinkas, A. Jarrous, and B. Moskovich. Sci -a system for secure face identi cation. In IEEE Symposium on Security and Privacy, pages 239{254. IEEE, 2010.

[38] P. Paillier. Public key cryptosystems based on composite degree residuosity classes. In Advances in Cryptology - Eurocrypt '99 Proceedings, LNCS 1592, pages 223{238, Prague, Czech Republic, May 2-6 1999. Springer-Verlag.

[39] I. D. R.M. Haralick, K. Shangmugam. Textural feature for image classi cation. Nov. 1973.

[40] J. J. Rodrigues, I. de la Torre, G. FernAandez, and, M. LAspez-Coronado. Analysis of the security and privacy requirements of cloud-based electronic health records systems. Journal of medical Internet research, 15(5), 2013.

[41] M. D. Ryan. Cloud computing security: The scienti c

challenge, and a survey of solutions. Journal of Systems and Software, 2013.

[42] J. P. S. Belongie, J. Malik. Shape matching and object recognition using shape contexts. Apr. 2002.

[43] A.-R. Sadeghi, T. Schneider, and I. Wehrenberg. E cient privacy-preserving face recognition. In Information, Security and Cryptology{ICISC 2009, pages 229{244. Springer, 2010.

[44] A. Shamir. How to share a secret. Communications of the ACM, 22(11):612 { 613, November 1979.

[45] E. Shen, E. Shi, and B. Waters. Predicate privacy in encryption systems. In Theory of Cryptography, pages 457{473. Springer, 2009.

[46] S. W. S.M. Lee, J.H. Xin. Evaluation of image similarity by histogram intersection. Aug. 2005.

[47] M. B. W. Burger. Principles of digital image processing: core algorithms. 2009.

[48] B. Wang, S. S. Chow, M. Li, and H. Li. Storing shared data on the cloud via security-mediator. In International Conference on Distributed Computing Systems-ICDCS 2013, 2013.

[49] R. S. Y. Ke. Pca-sift: a more distinctive representation for local image descriptors.2004.

[50] A. C. Yao. How to generate and exchange secrets. In Proceedings of the 27th IEEE Symposium on Foundations of Computer Science, pages 162{167. IEEE, 1986.

[51] A. C.-C. Yao. How to generate and exchange secrets. In Foundations of Computer Science, 1986., 27th Annual Symposium on, pages 162{167. IEEE, 1986.

[52] H.-J. Yu, H.-S. Lai, K.-H. Chen, H.-C. Chou, J.-M. Wu, S. Dorjgochoo, A. Mendjargal, E. Altangerel, Y.-W. Tien, C.-W. Hsueh, et al. A sharable cloud-based pancreaticoduodenectomy collaborative database for physicians: Emphasis on security and clinical rule supporting. Computer methods and programs in biomedicine, 2013.