ISSN 2047-3338

Design of an Efficient Publicly Verifiable Data Management Scheme Based on Provable Data Possession: A Survey

Abubakar Usman Othman[1], Muhammad Bashir Abdullahi[1], Ogwueleka Nonyelum Francisca[2], Mohammed Babatunde Ibrahim[1] and Umar Musa[1]

[1]Department of Computer Science, Federal University of Technology Minna, Nigeria
[2]Department of Computer Science, Federal University Wukari, Nigeria

***Abstract*– The development of wireless access technologies such as 3G/4G, mobile devices and sensors are now used as information collection nodes for the cloud which offers immense benefit for mobile users. Federating cloud service providers and caching frequently accessed data drive an effective way to seamlessly access and managed data in mobile cloud computing environment thereby overcoming server failover, frequent disconnection of database server, lock-in by single service provider and data location management issues. However, privacy and the security of mobile users' data remain the challenges that outweigh the colossal success factor of a federated data Management in Mobile Cloud Computing. This is owing to the fact that data are hosted on public clouds owned by commercial service providers. Thus, new data management techniques are required. This review: (a) highlight the current works proposed for data management techniques in mobile cloud computing, (b) security of data stored in the cloud, and (c) data verifiability and the limitation of the proposed frameworks.**

***Index Terms*– Data Management, Mobile Cloud Computing, Surrogate Object, Data Possession and Federated Cloud**

I. INTRODUCTION

VARIOUS layered architecture are available to provide services as a utility for cloud computing. This layered architecture include the cloud backbone layer which consist of servers and switches, the supervisor layer which hosts software that manage the cloud infrastructure resources, software infrastructure layer which is responsible for handing over network resources to upper layer that provided a building block for an emerging computing paradigm which delivers IT as a service, the platform infrastructure layer provides application development platform and some set of application programming interfaces (API), the application layer provides platform for users to access different applications contained in the cloud provider's files centres using web [1].

Resource constraint mobile devices can depend on cloud for computation, and information sharing and upload to perform computationally intensive operations such as data storage. The availability of such enabling environment for ubiquitous wireless infrastructure and resource constraint mobile devices to perform computation operations provide a platform for emerging computing paradigm known as mobile cloud computing (MCC) [2]. Mobile devices cannot perform computationally intensive and storage demanding operations because mobile devices being battery powered have minimal processing power, less energy, limited storage capacity and with less security. Moreover, if a cloud service provider lock-in or the database disconnect, mobile cloud users cannot access data which brings the need for federating multiple clouds. So federating with multiple clouds provides an effective way to seamlessly access data, but there is concern of mobile user's data being exposed to commercial cloud service providers due to lack of confidentiality and integrity [3]. This challenges outweigh the colossal success factor of a federated data management in distributed mobile cloud computing. Federated data needs to be efficient and secured to ascertain the integrity of mobile user's data files uploaded in cloud storages to avoid been exposed to the commercial service providers which is the biggest issue the current research area provable data possession (PDP) is being used as a scheme for providing solutions to proof of possession. [4], proposed a Public Provable Data Possession technique for resource constraint mobile devices that ensures the privacy, confidentiality of mobile user's data files stored in cloud's storage. In their scheme, a trusted third party does the encryption/decryption, encoding/decoding, signature generation, and verification of data files for mobile users. Though offloading saves energy, the addition in the number of users brings degradation in output.

However, to solve the problem of securing the federated data files and the resource constraint of mobile devices, many schemes aim at providing solutions to the arising problem are proposed. For instance, it was proposed that storage demanding operations should be moved to the cloud [5]. In another direction, it was suggested that to offload storage operation on cloud storage servers, the internet condition and the communication overhead should be taken into consideration to make the offloading beneficial for the mobile users [6]. Mobile users communicate with the cloud service provider using embedded browser applications which are developed with web development languages such as HTML and Java Script. Mobile users get connected with the base transceiver station and access the mobile web services.

Mobile users utilises mobile web services to communicate with cloud using the web. There are two cloud servers: portal cloud and back-end cloud servers. The *portal cloud server* is responsible for processing mobile users request for cloud services usages while the *back-end cloud server* provide a platform for portal cloud server to provide different services to mobile users.

A security frame-work for efficient and secured data storage services in mobile cloud computing (MCC) was proposed by [7]. Their proposed scheme, Privacy Preserving Ciphertext Policy Attribute-Based Encryption scheme ensures the privacy of mobile user's data in the cloud. Since the intensive operation of encryption and decryption are outsourced to a third party, the resource constraint mobile devices are relieved of the burden. The problem with this technique is that the cipher text increase linearly with the number of cipher text attributes.

Provable Data Possession (PDP) is a method employ to verify integrity of data stored in clouds. Remote integrity checking was proposed by [8]. In their work, they used RSA-based functions for hashing entire data files in cloud storages for every verification challenge send by the verifier. Their work failed to handle large data files that require longer time to efficiently computes and move hash values.

Federation in its simplest form means two cloud service providers sharing some data. The data could be files, resources, users' identity, resource information or any other useful information that may demand constant use in the future. Surrogate objects enable seamless data access for users. Surrogate objects are software entities that are hosted on some mobile support station (MSS) and act on behalf of mobile devices as well the middleware component between the mobile user and cloud service provider (CSP) in the cloud. Federation increase data availability and avoid vendor lock-in as data are replicated to more surrogate objects to provide seamless data access and efficient data management in mobile cloud system.

II. REVIEW

A) Federated Cloud

Ravimaran et al. [3] define federation as two service providers sharing user's identity, resources, files, resource information or any other information. The author presented a federation scheme in MCC environment that collects data from various database servers located in different clouds and makes it available in multiple surrogate objects. For each data access request, there is different access mechanism to acquire the customised and personalized data for executing the particular task initiated by a user. The authors used surrogate object over the cloud to act on behalf of each mobile user and can automatically perform local caching for faster, less expensive, safer and seamless data access for mobile operations. The aim of their work is to have a new system for better response time, better bandwidth utilization and also to provide support for mobility and disconnection. The authors did not include the security of uploaded mobile users' sensitive information on the cloud in their studies. When mobile user's data are outsourced for storage in cloud federation, the content of data file is exposed to the commercial service provider. Also, a malicious attacker can impersonate mobile user and gain access to the stored data files.

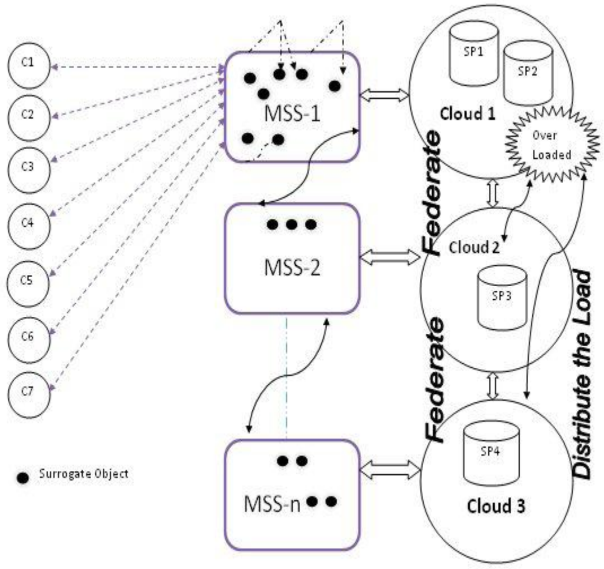

Fig. 1: Data Access in Federated Clouds. Source: Ravimaran et al. [3]

The scheme above aimed at addressing the problem of vendor lock-in, fast data access by data owner. The scheme failed to look at the privacy, confidentiality and integrity of mobile users data stored on cloud federation. A security system is needed to address the privacy issue of mobile user's data stored on cloud federation. Data files should be stored without the third party and cloud storage knowing the content of the data file.

Mobile Device: Mobile device is a communication device that can depend on cloud computing and storages to perform operations that are computationally intensive including searching, multimedia processing and data mining. Mobile devices can be used as information collecting nodes for the cloud Data are sent to the cloud through the mobile device. A mobile user uploads his or her data to the cloud. Mobile devices need to have a remote identification with the surrogate object.

Mobile Support Station: The Mobile Support Station is mobile stations that host Surrogate objects. When a mobile user joins the cloud, it registered itself with the Mobile Support Station that host the Surrogate object. The Mobile Support Station is responsible for assigning a unique identification for the mobile device or mobile user and also passed the identity information to the underlying middleware running on its cloud. The registration information is then passed on the other clouds about the new entry of the mobile user or device into the cloud system. It is a mobile station that fits into the mobile cloud computing environment which aids the actualisation of cloud federation.

The Cloud Storage Service Provider: It is a storage resources provided by a cloud as Data storage as a Service (DaaS). The cloud provides storage resources platform for users to store their information. The storage system provides an easy and efficient access of data in the federated clouds

The Surrogate Object: The surrogate object is a software entity that acts on behalf of mobile user. The object is created and resided in the static network which is the mobile support station. It also acts as a middleware entity between mobile users and cloud service providers. They are hosted by mobile support station. The surrogate object is assigned an identifier, SOID corresponding to its mobile user. The surrogate object performs in place of the mobile device. The SO also acts as a middleware components between the storage service provider and the mobile user in the cloud and can adequately store local caching for faster data access.

B) Distributed Storage Systems

The provisioning of Web-based storage devices by Network-Attached Storage (NAS) and Network File System (NFS) enable users to access these devices through a web connection. When storages are distributed scalability is increased because a storage provider can decide to join or leave the cloud without being control from a central authority. Replication of messages through erasure code for storage in different storage servers provides robustness against failure of servers as well reduce the expansion rate of messages. When a message is sent to cloud for storage, it is encrypted and then encoded as a codeword, also a vector of symbols, and the storage server stores codeword symbols each. A storage server failure is addressed as an erasure error of stored codeword symbol. To store a message that has k blocks, individual storage servers linearly add the blocks together with randomly selected coefficients and stores the codeword symbols and coefficients. To get the message, a user queries k storage servers for the codeword symbols and coefficients and rectify the linear system.

C) Provable Data Possession (PDP)

Jian et al. [4] proposed a Provable Data Possession (PDP) scheme for resource constraint mobile devices that ensures the protection, confidentiality, and integrity of mobile users' data stored on cloud. The scheme offloads mobile users' jobs of encrypting, decrypting, encoding, decoding, and signature generation on trusted third party (TTP) to save energy of the mobile device and to also reduce the processing burden to the barest minimum. The system model is composed by three entities: The mobile-end-user who utilises the cloud storage. The TTP handles the encryption, decryption, decoding, encoding and signature generation for mobile users to reduce to the minimum processing and computation overhead.

The cloud storage service (CSS) that contains huge storage for mobile users and also perform and make available the proof of data possession if requested by either TTP or the data owner. In their work, they used Diffie-Hellman key exchange to securely exchange and distribute key to two users and not involving any third party, bilinear mapping and Merkle Hash Tree (MHT) which is built as a binary tree where leaf nodes have hash value of data blocks. The mobile user encipher data using shared symmetric key and delivered the enciphered data to TTP to generate pair of asymmetric keys for encryption and decryption. The TTP use encryption key *(ek)* to encrypt the data and encodes the data after dividing the data into small chunks and then the TTP construct MHT. Their scheme ensures the privacy, confidentiality and integrity of mobile user's data stored on cloud but not on federated mobile cloud. Although the offloading of encryption, encoding, decoding and decryption of operation, signature generation and verification to the trusted third party, which saves energy. The growth in mobile users leads to degradation in output.

Vrushaliet al. [9] proposed a provable data possession (PDP) scheme in multi clouds is proposed to support service scalability and data migration. In their proposed system, Integrity Verification and Availability using PDP in multi cloud, is based on multi-prover, zero knowledge proof system, which helps to satisfy knowledge soundness, completeness and zero knowledge properties. PDP is a secure technique which enables a storage service provider to prove data integrity even if data is not downloaded. The ability to prove the data integrity without downloading the data makes it very crucial and important for large-size data to be proved and verify that the data have been tempered with or deleted without retrieving the latest data stored. The authors used a verification mechanism together with hash index hierarchy (HIH) and homomorphic verifiable response (HVR) to proof the integrity of users' data without downloading the data from storage. In the HIH, three layers are used to illustrate the relationship between and among the blocks for stored resources. The three layers are; the express layer which shows the representation of stored data, the service layer that offers and manages cloud storage and services, and the storage layer that realises data storage on physical devices.

Furthermore, The HVR uses a map between two groups which reduces the communication overhead and also conceal the location of stored data in a distributed cloud storage system environment. The authors failed to consider the privacy and confidentiality of users' data stored on cloud storage. The result in the proposed work shows that the users' communication overhead remains constant and the interaction between cloud service providers needs c-1 times constant size communication overheads where c is the number of cloud service providers. In their work, they evaluate the performance of the system in terms of computational overhead and the results shows that if values of sectors s, are increased, the overhead is reduced when a fixed-size data is used.

D) Erasure Code

Priyadharshini et al. [10] designed a cloud storage system that is robust, provides confidentiality as well as functionality. It becomes a serious problem for users, concerning the confidentiality of their data. A user need to divide his data into small chunks, encrypts the data and then store them in various storage servers. In authors proposed system, the storage servers encode the user's data using erasure codeword symbols. A user can share his data by sending a re-encryption key to storage server that is responsible for performing re-encryption operation. The storage server re-encrypts the already encrypted encoded symbol into a re-encrypted encoded symbol. The authors integrate the re-encryption scheme with decentralized erasure code. Their scheme supports the encoding operation over encrypted data and forwarding operation over encoded and encrypted data. The authors failed to address the problem of dynamic data operation that allowed for data modification, data insertion and data deletion.

Furthermore, they explained that a random linear code with a sparse generator matrix is a decentralized erasure code. An encoder construct an erasure code as following: firstly, for each row, the encoder randomly specify an entry as I and repeats this process for an $i^n\ k^k$ times with replacement. Secondly, the encoder randomly sets a value from iF to each marked entry. This brings to an end the encoding processes. A decoding is successful if $k * k$ sub matrix formed by the k-chosen columns are invertible. The probability of successful decoding is equally the probability chosen sub matrixes' inverse.

$$\begin{matrix}
64 & -16 & 0 & -16 & 0 & 0 & 0 & 0 & 0 \\
-16 & 64 & -16 & 0 & -16 & 0 & 0 & 0 & 0 \\
0 & -16 & 64 & 0 & 0 & -16 & 0 & 0 & 0 \\
-16 & 0 & 0 & 64 & -16 & 0 & -16 & 0 & 0 \\
0 & -16 & 0 & -16 & 64 & -16 & 0 & -16 & 0 \\
0 & 0 & -16 & 0 & -16 & 64 & 0 & 0 & -16 \\
0 & 0 & 0 & -16 & 0 & 0 & 64 & -16 & 0 \\
0 & 0 & 0 & 0 & -16 & 0 & -16 & 64 & -16 \\
0 & 0 & 0 & 0 & 0 & -16 & 0 & -16 & 64
\end{matrix}$$

A sample sparse matrix generated from Matlab spdiags command (>>A)

In addition, the data owner choses v number of servers with replacement and forward a copy of M_i to all. All servers choses coefficient for every received encrypted message and performs a linear combination of all. The coefficient selected by servers from columns of matrix and output of the linear combination is a codeword element. Individual servers equally do the computation on its own given rise to decentralized codes.

The encoding procedure of the message could be divided into sub n tasks of outputting codeword symbols. Decentralised erasure codes are utilised adequately in a distributed storage system for integrating n block messages hosted. The authors adopt this method to ensure the confidentiality of mobile user's data file stored on federated cloud storages.

Wang et al. [11] described regenerating codes were as providing optimal recovery bandwidth among storage nodes. The authors proposed a rebuilding for array codes in distributed storage systems were regenerating codes were designed for distributed systems where wide-area bandwidth hinders or reduce recovery performance. The proposed work is based on analytical evaluation which provides that minimising recovery data directly translate into an improved input and output performance for cloud file systems.

E) Mobile Cloud Computing

Sending and collecting data such as bank accounts details and passwords, security numbers and health critical information is been made possible using mobile phones. A security model that does not allowed data leakage in distributed mobile system using surrogate objects was proposed by Anitha et al. [12]. To secure the data on the object during transaction execution and reconciliation, the users need to be authenticated. There is efficiency in the propose system since the surrogate objects are made to act on behalf of the server which drastically reduce the communication overhead. It also ensures confidentiality of transaction due to the presence of an encrypted tunnel between database server and the objects. The result of simulation shows that the proposed system is more secured and also provides minimal communication overhead compared with the existing systems. In this study, the author used unique identification assigned to mobile users and it respective surrogate objects assigned to it when it enters or registered in the cloud to enable it access the cloud. Non-registered mobile user will be denied access into the cloud if it tries to access the cloud.

Parveen et al. [13] proposed a secured data management paradigm on traditional clouds but not on mobile cloud. Their proposed system secured data management paradigm for mobile grid environment using surrogate object, consists of collection of surrogate objects which acts as a place holder to solve the location management problem to handle the mobility of mobile hosts. This model support and handles concurrency and recovery as mobile host shift from cell to cell. The authors built an authentication protocol on surrogate objects to deal with the insecurity challenges through surrogate objects. The authentication protocol is also built to provide cryptographic secure mechanism so that the identity of users and resources could be verified. A certificate standard, X.509 is also used for authentication. A secure communication is made possible among mobile hosts in the authors work using SSL and or TLS.

Peng et al. [14] presented a Cloud-based Storage achieve for Mobile computing (CS-Mobile) that addresses the challenges of data storage in MCC. The framework provides an easy-to-use file navigation service and also provides a mechanism for users to verify their data with minimal burden. The aim of the author's work was to provide a lightweight storage system to resource constraint mobile users. The framework composed four entities including mobile user, federated cloud storages, independent security server and CS-Mobile. While the scheme considers federated cloud storages, it does not support dynamic data operation in mobile cloud computing.

Itani et al [15] presented an Energy efficient framework for limited energy mobile devices to ascertain the correctness data stored on cloud through incremental cryptography and trusted computing. The system is composed by three entities including mobile user, service provider and TTP. They discussed data upload, data insertion, data deletion, and verification for files in MCC. The authors described the use of incremental cryptography with respect to integrity verification. When uploading data on the cloud storage server, the mobile user generates an Incremental Message Authentication code using secrete key. The problem with their system is that, the privacy of mobile user's data is overlooked. The trusted coprocessors can handle a particular number of mobile users which did not give room for scalability. An increase in the number of mobile users results to degradation in performance.

Hsueh et al. [16] proposed a scheme for smartphones to provide security and verify data uploaded on cloud servers. The introduced a mechanism to authenticate users who uploaded data on cloud storage. The proposed system consists of the mobile device that utilises the services provided by CSP and the certification authority authenticates each device; the telecommunication module which is responsible for generating passwords and also keeps record of information about mobile device to use the services of the cloud. In the work, the authors failed to address the processing and storage capability of the mobile device and, the encryption, decryption and hash computations performed by the mobile user through mobile devices consumed considerable amount of energy. Again, the advisory can impersonate the mobile user by utilising the user's credentials since the certification authority is left with whole responsibility having access to users' credentials.

Jia et al. [17] developed a network model which is composed by data owner (DO), Data sharer, and Cloud service provider. Information and security management are outsourced on cloud and the content of data is hidden to the cloud by using proxy re-encryption and identity base encryption schemes. The authors used a semi trusted proxy to transform information enciphered with user A's key into another information enciphered with user B's public key and the identity based encryption technique is used on bilinear mapping. The system failed to address the computationally intensive operations performed by mobile users. Also, performing cryptographic operations consume a lot of energy.

III. CONCLUSION

In this survey, various data management techniques, security frameworks and data verifiability techniques were studied in detail with much emphasis on the limitations of the proposed systems, and with respect to mobile cloud computing.

Though most of the frameworks and techniques provided good solutions, they are not free of challenges. To achieve an efficient public verifiable data management in the MCC, the limitations contained in this proposed solutions as outlined in this paper need to be studied in detailed and addressed.

REFERENCES

[1] Sotomayor, B., Montero, R. S., Foster, I. (2009). Virtual infrastructure management in private and hybrid clouds. *IEEE Internet Computing,* 13 (5), 14-22.

[2] Abdul N. K., Mat, K. M. L., Samee U. K., & Sajjad A. M. (2012). Towards secure mobile cloud computing: A survey. *Future Generation Computer Systems,* doi:10.1016/j.future.2012.08.003. Available at www.elsevier.com/locate/fgcs. Retrieved on June 21st, 2014.

[3] Ravimaran, S., Muhammed, M., & Sharief, A. (2012). Federated data management in distributed mobile cloud. *European Journal of scientific Research,* 86 (4), 482-492.

[4] Jian, Y., Haihang, W., Jian, W., Chengxiang, T., & Dingguo, Y. (2011). Provable data possession of resource constraint mobile devices in cloud computing. *Journal of Networks,* 6 (7), 1033-1040.

[5] Canepa, H., & Lee, D. (2010). A virtual cloud computing provider for mobile devices, *ACM* workshop on Mobile Cloud Computing & services Social Networks and Beyond, MCS '10, San Francisco, USA.

[6] Kumar, K., & Lu, Y.H. (2010). Cloud computing for mobile users: Can offloading computation save energy? *IEEE Journal of Computer,* 43 (4), 51–56.

[7] Zhibin, Z., & Huang, D. (2011). Efficient and secure data storage operations for mobile cloud computing, *IEE Journal of Computer,* 46 (7), 49-55.

[8] Deswarte, Y., Quisquarter., J. J., & Saidane., A. (2003). Remote Integrity Checking. *Proceedings of Conference on Integrity and Internal Control in Information Systems.* IICIS' 03, San Francisco, USA.

[9] Vrushali K. G., (2014). Integrity verification and availability using PDP in multi-cloud. *International Journal of Computer Science and Information Technologies,* 5(6), 8103-8105

[10] Priyadharshini, B., Carmel, M., B., & Ramesh, M., K. (2013). A secure code base cloud storage system using proxy re-encryption scheme in cloud computing. *Journal of Computer Engineering,* 9 (2), 22-27.

[11] Wang, Z., Dimakis, A. G., & Bruck, J. (2010). Rebuilding for array codes in distributed storage systems. *IEEE Transactions on Computing,* 34 (8), 203-212.

[12] Anitha, R., Ravimaran, S., & Valarmathi, P. (2012). Security model that prevents data leakage in distributed mobile systems using surrogate objects. *Special issue of International Journal of Computer Application,* 8(6), 0975-8887.

[13] Parveen, H., & Maluk, M. (2012). Secured data management paradigm for mobile grid environment using surrogate objects. *13th international conference on mobile data management, IEEE.*

[14] Peng, X., & Yanping Z. (2013). CS-Mobile: A cloud-based distributed storage middleware for mobile devices. *International Journal of Smart Home,* 7(1), 87-98.

[15] Itani, W., Kayssi, A., & Chehab, A. (2010). Energy-efficient incremental integrity for securing storage in mobile cloud computing, in: *Proceedings of Internal Conference on Energy Aware Computing, ICEAC* '10, Cairo, Egypt.

[16] Hsueh, S. C., Lin, J. Y., & Lin, M. Y. (2011). Secure cloud storage for conventional data archive of smart phones, in: *Proceedings of 15th IEEE International Symposium on Consumer Electronics, ISCE* '11, Singapore.

[17] Jia, W., Zhu, H., Cao, Z., Wei, L., & Lin, X. (2011). SDSM: a secure data service mechanism in mobile cloud computing, in: Proceedings of *IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS,* Shanghai, China.