ISSN:0975-9646

TCP Based Storage Outsourcing with Secure Accessibility in Mobile Cloud Computing

Monika Waghmare, Prof T.A.Chavan

Department of Information technology, Smt.Kashibai Navale College of Engineering, Pune, India.

***Abstract*— Now a day a mobile device like a smart phone is becoming one of the major information processing devices for users. On the other hand, a mobile device is still resource constrained. To overcome this, a mobile device should get resources from an external source. One of such sources present is cloud computing platforms. In Mobile Cloud computing generally front-end consist of users who possess mobile devices and back-end cloud servers. This pattern enables users to access a large volume of storage resources with portable devices in a distributed and cooperative manner. In between the times of uploading and downloading files or data, the privacy and integrity of files need to be guaranteed. As mobile device is battery constrained most important issue associated with this field is battery saving.**

For providing more security in terms of integrity and privacy as well for saving battery life we are introducing one computing platform called as trusted computing platform(TCP) which saves computing power of mobile device. In this system all the computations are done on TCP instead of Mobile device that's why this approach is useful for providing more security with lightweight computations.

Keywords*— *Mobile cloud computing, cloud computing, cloud servers, mobile devices, integrity, privacy

I. INTRODUCTION

Mobile Cloud Computing is a rising computational paradigm these days with fast access of lightweight Mobile Devices to end users and rapid deployment of Cloud Servers. Mobile cloud computing takes the advantages of cloud servers that provide flexible capabilities in terms of computation and storage at back-end, as well as take the advantages of mobile devices that provide persistent accessing and universal computing at front-end. The combination of cloud servers and mobile devices not only balances the exchange between resource requirement and mobility demand for a single mobile user, but also constructs a cooperative connection among multiple mobile users. It broadly enlarges the computing power of individual mobile users and effortlessly links multiple cooperators in a mobile group. The architecture of the *MCC* is shown in Figure 1.

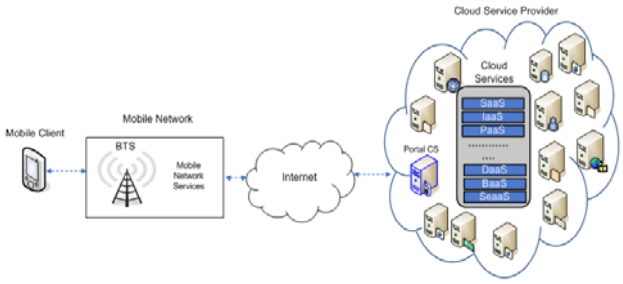


Fig 1: Architecture of MCC [1]

In Mobile Cloud Computing (MCC), Mobile Devices (MDs) essentially need to outsource some computation and storage tasks to Cloud servers as memory of mobile devices is limited. If the cloud servers used are trusted then it would be perfect for the migration or outsourcing; but if servers are distrusted, then there arises a critical problem of security. That is, how to preserve the outsourced computation and storage on the cloud being trusted.

In this paper, focus is on the storage outsourcing in totally distrusted CSs, here computation outsourcing is conducted in trusted CS names as trusted third party server. Mobile device used is also considered as totally distrusted. After a MD generates a file and processes it, it uploads the file into a CS client i.e user or other cooperators may access it in the future as per their need. Obviously, the privacy and integrity of the file must be maintained in the storage of CSs during the period between uploading and accessing file. In this project as computations are offloaded on TCP, energy is saved and battery life of mobile is prolonged.
Though there are some papers present that address similar problems in storage [9-12], subtle design for secure storage outsourcing in MCC has not been broadly explored, especially in MCC environment.

Some solutions assume CSs are trusted, which may be a too strong (unrealistic) assumption to constrain the applicability. Besides, the mobility of MDs and cooperative accessibility of files on CSs are not yet carefully considered. Moreover, as MDs may be lost incidentally, the storage of MDs are vulnerable to exposure, which is inappropriately ignored. In existing system all the computations are done on mobile device that's why more energy is consumed which is disadvantage of that system. In some systems links between mobile and server is not secured which may lead to security violation. Thus, all above situations are tackled in the design of this paper, and our scheme can guarantee the privacy and integrity of outsourcing files or data but maintain lightweight in terms of computation for mobile device.

As MCC platform is based on cloud computing so all the security issues in cloud computing are inherited in MCC with extra limitation of resource constraint mobile devices. Because of this resource limitation, the security algorithms planned for cloud computing environment cannot be directly run on mobile device. There is a requirement of lightweight secure framework that provides security with less communication and processing overhead on mobile devices. This need is the motivation for the paper.

II. RELATED WORK

Liu et al. projected to use hierarchical identity-based encryption algorithm provided that an efficient sharing of the secure storage services in cloud computing. Here the encryption is used just the once and only one copy of the corresponding cipher text needs to be stored. It needs MD having higher computation ability. Wei et al. proposed SecCloud, it is an auditing scheme, used to secure cloud computing based on probabilistic sampling technique [6]. It aims to consider secure data storage, computation and privacy preserving together. As it is an auditing scheme it is time consuming and the result may not be deterministic. Itani et al. proposed an energy-efficient protocol for ensuring the integrity of storage services in mobile cloud computing. The proposed protocol applies incremental cryptography and trusted computing to design secure integrity data structures. Ye et al.[5] developed an access protocol following the requirements to attain exact and efficient data accesses. They utilized regular semantics instead of atomic semantics to advance access efficiency. Park et al.[2] proposed a secure storage BLAST, which is improved by a stream cipher rather than a block cipher with a novel block accessible encryption mechanism based on streaming ciphers. Feng et al.[4] proposed an encryption method D-DOG (Data Division and Out-of-order key stream Generation) to guard data in the distributed storage environments. Itani et al.[8] proposed a PasS (Privacy as a Service), a set of security protocols for achieving the privacy and legal compliance of customer data in cloud architectures. PasS gives a privacy feedback process which informs users of the different privacy operations and makes them aware of any potential risks that may expose the confidentiality of their sensitive information. Xu et al. presented a mobile cloud data processing framework through trust management and private data isolation to protect user's privacy in the cloud.

III. PROBLEM STATEMENT

We can implement cloud security algorithms on mobile cloud also but the problem is that computation power of security algorithms used is high and mobile devices have very less computational power as well as battery life. To tackle this situation development of lightweight security scheme is required.

Hence the aim of this paper is to ***provide Lightweight Storage Outsourcing with Secure Accessibility on distrusted cloud in Mobile Cloud computing that will assure data confidentiality and data integrity.***

IV. SYSTEM DESIGN

In the architecture of this system the main four components are shown in the figure below. The mobile handset will carry mobile applications and server will contain glassfish server and web services that enables communication among the two. Here SOAP/XML is also used to activate communication. Server sends and receives data via XML. XML acts as common language between cloud servers and clients. The links between mobile device and servers are secured with the help of session encryption here Diffie-hellman key exchange algorithm is used.

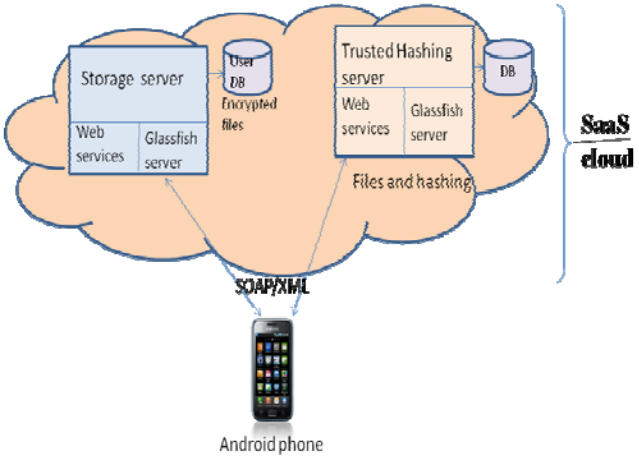


Fig 2: system architecture.

The architecture is mainly divided into three parts:

- Mobile Device
- Cloud servers and database
- User

3.2.1 Mobile device (MD):

It is a device prepared with capabilities such as computing, storage and wireless communication. Here android phone containing multiple mobile applications is used.

3.2.2 Cloud server:

It is a service provider in cloud computing. Server uses glassfish server and web services to communicate with mobile device/applications. Here SOAP/XML is used for connection between client and server.

Two types of servers are used

a) Storage server:

Here outsourced data is stored in the form of encrypted files. It is used for storage purpose only no computation is done.

b) Trusted hashing server:

This is a trusted third party server used for storing log of hash of files for backup. This computing platform performs both computations as well as storage of hash.

Database:

MySQL is used as a database. Here encrypted user files are stored.

c) Mobile User:

It is a person who operates mobile device. Multiple servers may be present who want to access the same file or data in CS. The operated entity is a file or data. Mobile user uses mobile applications through mobile device to connect with the server. Here User sign in/up in the application for the purpose of uploading or downloading file on the server. Here SOAP/XML is used for connectivity between client and server. In this paper cloud service used for deployment is SaaS. Server contains database for storing encrypted user files. And another server called as trusted Hashing server which is used to store hashed data. Mobile client can access both storage cloud server as well as trusted third party server.

V. SYSTEM FLOW

Uploading Process:

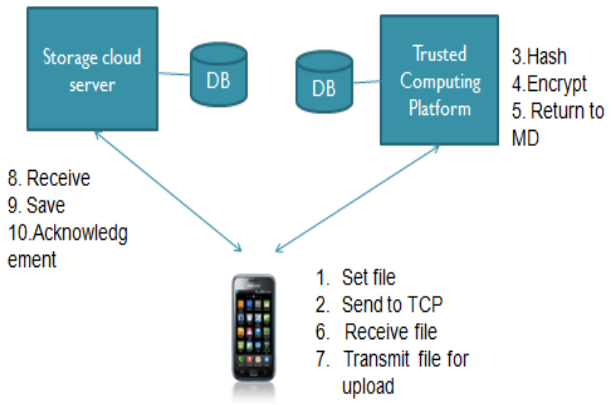


Fig. 3 steps in uploading process

When client wants to upload file on cloud server through mobile device, he has to login into the system with his password. Mobile device then sends this file to the trusted third party server or hashing server. TCP then generates Encryption key (EK) by applying standard hash function on concatenation of file name (FN), original file (FS) and key.

EK= H (key||FN||FS)

TCP encrypts file with encryption key EK by using AES algorithm.

EF=EEK (FILE)

TCP then returns this encrypted file to the mobile device. Mobile device uploads EF on cloud server, and stores H (FN) on trusted third party server i.e hashing server and deletes IK. Storage cloud server only receives files, saves it and sends acknowledgment to the user (MD).

Here we are using symmetric algorithm AES for encryption and decryption because in AES size of file after encryption is same as original size this is advantage over existing algorithm AES

Downloading Process:

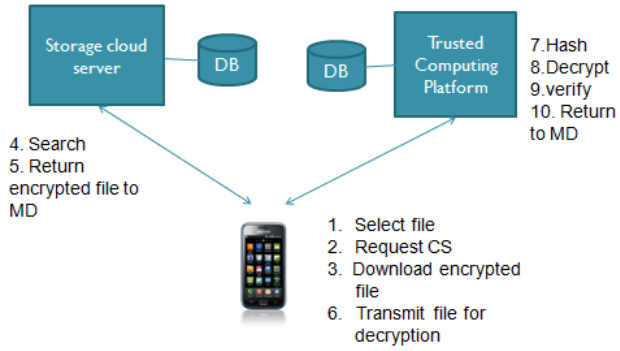


Fig 4. Steps in downloading process

While downloading a file, mobile device selects file name and retrieves encrypted file from storage cloud server. Mobile device then sends this file to trusted file server for decryption and verification. Here again hash is calculated by using sha-1 on file name, key and file size (of encrypted file)

EK=H (key || FN || FS)

Afterword, it decrypts the EF using regenerated EK to get original file.

Decrypted file= DEC (F', EK)

Here system will check integrity of file by comparing hash value of original file and encrypted file. If hash of both the files is same then we can say that file is valid or not modified.

VI. RESULTS

A comparative analysis of AES and RSA is done with the help of decryption and encryption. Effects of several parameters such as number of rounds, block size and size of secret key on the performance evaluation criteria are investigated. In addition, to improve the accuracy of our timing measurement program was executed 10 times for each input file and we report the average of times there by obtained. In this observation key size is 32-bits for AES and RSA. Number of rounds (r) was fixed.

On the basis of execution time, we compare the execution time of each algorithm on all file type for this purpose we mainly used 5 files and recorded their execution (encryption and decryption) times in millisecond for the two algorithms.

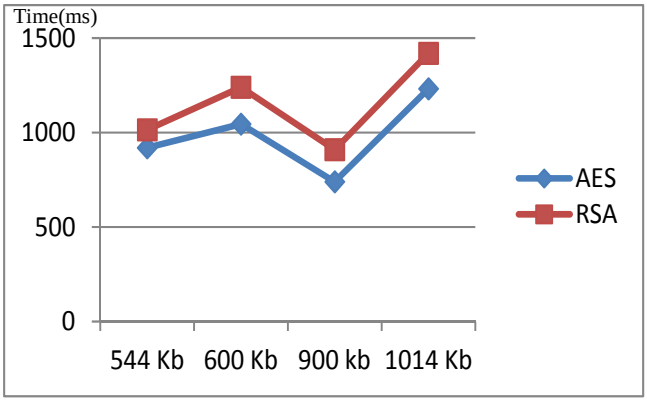
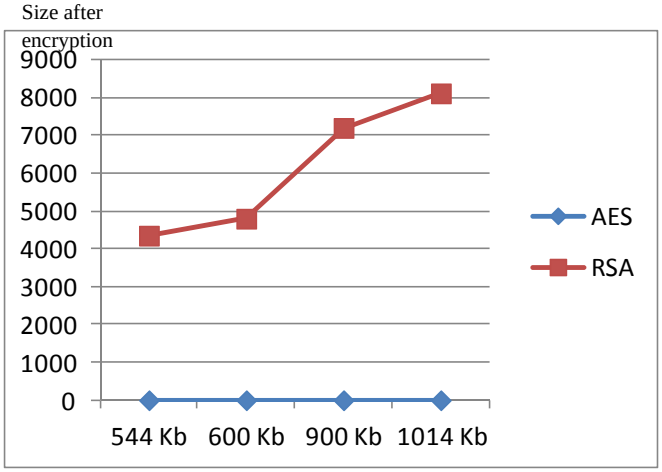


Fig 5: Comparison on the basis of execution time

In above graph X-axis denotes test file size where as y-axis denotes required execution time for uploading file. From this graph we are able to see that execution time of our system is less than existing which uses RSA.



Original file size

Fig 5: Comparison on the basis of file size

From graph we can see that file size after encryption is same in AES which saves storage space

VII. CONCLUSION

In this paper, system is proposed for protecting the confidentiality and integrity of uploading files or data in mobile storage cloud. It is considered that cloud is totally distrusted and mobile device is semi trusted or distrusted.
Here only file names are stored on mobile device and no computation is done on it hence it will save computation power as well as battery life. This system guarantees the security goal as it involves algorithms such as AES, SHA-1, and Diffie-hellman . The AES algorithm is used for local and server encryption,SHA-1 is used for calculating Hash function and session encryption is done for securing links between cloud server and client. This scheme can decrease the computation overhead on mobile. Schemes are flexible to the storage compromise on mobile devices, lightweight, and assume that the cloud servers are distrusted. Thus, they provide a stronger protection for more general and realistic application scenarios comparing with the previous work.AES algorithm is used in foe encryption and decryption which is more secure as well as it does not increases file size after encryption that's why storage space as well battery life is saved as shown in graph.

REFERENCES

[1] Hoang T. Dinh, Chonho Lee, Dusit Niyato, and Ping Wang. A Survey of Mobile Cloud Computing: Architecture Applications, and Approaches,In Wireless Communications and Mobile Computing 2011.

[2] Xu L, Xing T, Zhong Y, et al. Secure data processing framework for mobile cloud computing. In: IEEE INFOCOM 2011 Workshop on Cloud Computing (INFOCOMW11). Shanghai, China, 2011: 711-716.

[3] Priyank Singh Hada, Ranjita Singh, Mukul Manmohan. Security Agents: A Mobile Agent based Trust Model for Cloud Computing.In International Journal of Computer Applications (0975 – 8887) Volume 36– No.12, December 2011

[4] Wei Ren, Linchen Yu, Ren Gao, Feng Xiong.Lightweight and Compromise Resilient Storage Outsourcing with Distributed Secure Accessibility in Mobile Cloud Computing. Tsinghua Science And Technology,ISSN 1007-0214 06/09 pp520 528.Volume 16, Number 5, October 2011.

[6] Manjunatha A, Ranabahu A, Sheth A, et al. Power of clouds in your pocket: An efficient approach for cloud mobile hybrid application development. In: 2010 IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom10). Indianapolis, IN, USA, 2010: 496-503.

[7] Park K W, Kim C, Park K H. Blast: Applying streaming ciphers into outsourced cloud storage. In: 2010 IEEE 16th International Conference on Parallel and Distributed Systems (ICPADS10). Shanghai, China, 2010: 431-437.

[8] Itani W, Kayassi A, Chehab A. Energy-efficient incremental integrity for securing storage in mobile cloud computing. In: 2010 International Conference on Energy Aware Computing (ICEAC10). Cairo, Egypt, 2010: 1-2.

[9] Liu Q, Wang G, Wu J. Efficient sharing of secure cloud storage services. In: 2010 IEEE 10th International Conference on Computer and Information Technology (CIT10). Bradford, West Yorkshire, UK, 2010: 922-929.