Proposed Secured Framework for Mobile Cloud Computing

Mohammad Arif Ansari[1], Er. Archana Singh[2]

[1] *Student of Department Computer Science & I.T., Sam Higgnbottom Institute of Agriculture Technology &Sciences, Naini, Allahabad, U.P, India*
md.arif.0161@gmail.com

[2]*Department Computer Science & I.T, Sam Higgnbottom Institute of Agriculture Technology & Sciences, Naini, Allahabad, U.P, India*
archana.singh@shiats.edu.in

***Abstract*— Mobile Cloud computing has invoked a new wave of evolution in the field of the mobile world. Although several research works has been done in the field of mobile technology, but cloud computing for mobile world is vastly unexplored. In this dissertation, we introduce the concept of Mobile Cloud Computing (MCC), and its security for data storage into the cloud from mobile device. Recently the market of mobile has been evolving rapidly and as such cloud computing is also growing its pace in mobile technology. That is why mobile cloud computing is a new and fast growing issue today. Cloud computing is the computing that provides virtualized IT resources as a service by using the Internet technology. Cloud computing, functions as such that a user impart IT resources (i.e. software, storage, server, network) as needed, uses them, get a support of real-time scalability according to service load, and pays as he goes. Especially the cloud computing environment distributes IT resources and allocates according to user's request. The security issues in Mobile Cloud Computing can be classified as follows: mobile threats and cloud threats. The main purpose of these threat is that, the person (hacker or thieves) steals personal data (e.g. passwords, contact database, credit card numbers, calendar, location) or to impose on mobile device resources. To overcome these problems we will be proposing a framework for data security into cloud from mobile device. This framework consists of multilevel data security and thus it not only secures but it also works on data saving part and it has security on data retrieving from cloud also this is done in our dissertation by using multilevel of key security for data storage.**

Keywords—Mobile Computing, Cloud Computing, Mobile Cloud Computing, Mobile Cloud Computing security model.

I. INTRODUCTION

Introduction of Mobile Cloud Computing: -Mobile cloud computing (MCC) is the combination of cloud computing and mobile networks to bring benefits for mobile users, network operators, as well as cloud computing provider. Mobile cloud computing refers to the availability of cloud computing services in a mobile environment. It incorporates the elements of mobile networks and cloud computing, thereby providing optimal services for mobile users. In these days mobile devices (e.g., smartphone, tablet pc's, etc) are increasingly becoming an essential part of human life as the most effective and convenient communication tools not bounded by time and place also we know that the market of mobile devices are increasing rapidly. The growth of mobility has changed our lives fundamentally in an unprecedented way. According to Cisco IBSG, close to 80 percent of the world's population has access to the mobile phone and new devices like the iPhone, Android smartphones, palmtops and tablets have brought a host of applications at the palms of people's hands. The rapid progress of mobile computing (MC) becomes a powerful trend in the development of IT technology as well as commerce and industry fields. However, the mobile devices are facing many challenges in their resources (e.g., battery life, storage, and bandwidth) and communications (e.g., mobility and security) (M. Satyanarayanan - 1996)[1]. The limited resources significantly impede the improvement of service qualities. "*Mobile cloud computing at its simplest refers to an infrastructure where both the data storage and the data processing happen outside of the mobile device. Mobile cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing applications and mobile computing to not just smart-phone users but a much broader range of mobile subscribers.* (Forum)" "*Mobile cloud computing is a model for transparent elastic augmentation of mobile device capabilities via ubiquitous wireless access to cloud storage and computing resources, with context-aware dynamic adjusting of offloading in respect to change in operating conditions, while preserving available sensing and interactivity capabilities of mobile devices* (White Paper - 2010)[2]."

So MCC is a combination of cloud computing, mobile devices, ubiquitous Wi-Fi network, mobile web and location based services. Mobile devices have some limitations such as battery power, limited accessing power, low storage, low security, less energy and unpredictable internet connectivity, hence according to one survey, MCC can be defined as:-

''A service that allows resource constrained mobile users to adaptively adjust processing and

storage capabilities by transparently partitioning and offloading the computationally intensive and storage demanding jobs on traditional cloud resources by providing ubiquitous wireless access''.

As Mobile Cloud computing comes with several solutions of our problems like storage of data. It also provides several services which normally run out of mobile device like software, infrastructure, hardware and many more. Now a day's its very helpful in our daily life but on the contrary part it comes with several threats like data theft, communication breach etc. in these day's security in Mobile Cloud Computing is an unsolved problems. Several solutions comes regarding this but it still needs modification.

Introduction of Proposed Framework: Mobile cloud computing (MCC) is simply referred to as an infrastructure where both the data storage and the data processing happen outside of the mobile device. While we are saving or processing our data from our device to cloud we have to ensure that our data is secure. Now a day's security of data is our main concern in Mobile Cloud Computing environment.

So in this dissertation I am proposing a secured framework for Mobile Cloud Computing, in which the user data is being retrieved on one – to – one mapping basis. Also this framework has 4 – level of security for data storage. In this the user registers his data (user data) in the cloud storage provided by the cloud service provider (1^{st} level of security). This is done using a key generation method (using RSA Algorithm) and which leads to the encryption of data at the cloud level. Once the user info or his data is stored in the cloud environment, the user can re-login and choose various services (such as saving data, reading data and many more). If the user chooses to read his data stored in the cloud environment he needs to verify his account details (2^{nd} level of security). Further if he succeeds a Onetime key (OTK) will be generated from the read data module (3^{rd} level of security). If the generated OTK is similar to the OTK which is entered by the user then he can read data stored in the cloud storage otherwise warning message erupts.

Further for decryption of data a private key is required (4^{th} level of security) and that private key should be similar to the private key stored in the cloud database. If the keys are similar then the action succeeds and decryption of data takes place. Hence, the original data shown to the user.

II. METHODOLOGY

Previous Models: - Now a day's data is a very sensitive thing and it needs to be secure. Similarly in Mobile Cloud Computing while saving or retrieving any data on cloud storage, then our main concern is to secure our data because several threats are involved while transferring any data from one place to another place. Main threats on which we are focusing are:- Data Ownership, Privacy of Data, Data loss, Data leakage through poorly written third party application, Insecure network access and unreliable access point etc.

In last few years many solutions are provided by several researchers. Anand Surendra Shimpi (A. S. Shimpi et al. - 2012)[3] proposed a secure framework for processing data in mobile cloud computing. This framework stores data in a secured fashion which helps in protecting the user's privacy. In addition, he has implemented a project named "Focus Drive" which improves the driving safety of teenagers. Jibitesh Mishra (J. Mishra - 2012)[4] proposed a secure architecture for MCC to integrate mobile applications with the various cloud services. This architecture improves the storage and processing of data on mobile devices in a secured manner. It helps in maintaining the integrity and security of data. Itani et al (2012)[5] proposed a framework which was energy efficient for mobile devices to assure mobile user's integrity i.e. using *incremental cryptography and trusted computing*, the data/files of users are stored in the cloud. This framework results in saving 90% of processing energy on the mobile devices when compared to other conventional techniques with more security. Eugene E. Marinelli (2009)[6] developed *Hyrax*, a platform from Hadoop which supports cloud computing on Smartphones. It allows user's applications to utilize data and computing process on networks on Smartphones. It offers a sane performance in data sharing and tolerates node departure. Eugene also implemented a distributed media search and data sharing approach. Jon Oberheide (J. Oberheide et al. - 2008)[7] proposed an architecture which contains three components: **a) *Host Agent:*** It is a lightweight process that runs on each device and inspects the activities of the files on the system. It stores the unique identifier (such as hash) in the cache for files received. If a new file does not hold file identifier, it will be sent to the Network Service. **b) *Network Service:*** This service analyses the files sent by the host agent. There can be multiple instances of Network Services that are running on the cloud using virtualization technique which supports parallel detection of multiple files sent by multiple host agents. **c) *Caching:*** *Local private cache (LPC) and Global shared cache (GSC)* are the two cache agents where LPC can be put into the identifier of inspected files and GSC cache resides on the Network Service which has the identifiers of all inspected files received so far.

Security and privacy are always a key issue when the data are shared between mobile devices and the cloud. Even though WPA2 (Wi-Fi Alliance, 2012) provides layer-2 encryption of the data, layer-6 encryption is still a requirement because it requires

some external applications like bioinformatics or computational chemistry that are executed on mobile devices and remotely on rented/ commercial cloud platforms (such as Google (2012, AWS (2012), Microsoft (2012)) which require an additional layer of security.

Previous model which is proposed by A. N. Khan et al. (2012)[8] shows below which I am taking as a base model.

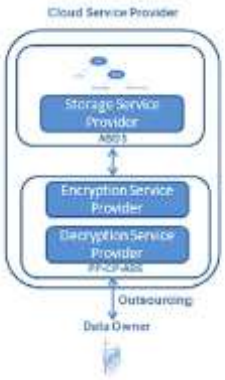
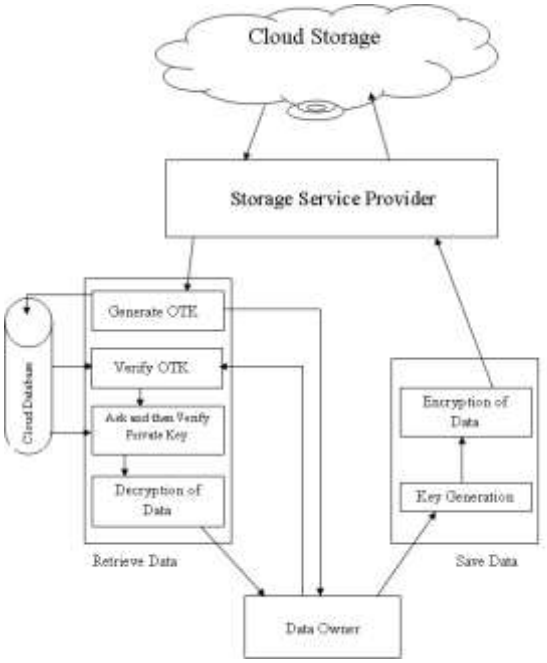


Figure 3.2 – Previous proposed model by A. N. Khan et al (2012)

Proposed Framework: - Mobile cloud computing (MCC) is simply referred to as an infrastructure where both the data storage and the data processing happen outside of the mobile device. While we are saving or processing our data from our device to cloud we have to ensure that our data is secure. Now a day's security of data is our main concern in Mobile Cloud Computing environment.

So in this dissertation I am proposing a secured framework for Mobile Cloud Computing, in which the user data is being retrieved on one – to – one mapping basis. Also this framework has 4 – level of security for data storage. In this the user registers his data (user data) in the cloud storage provided by the cloud service provider (1st level of security). This is done using a key generation method (using RSA Algorithm) and which leads to the encryption of data at the cloud level. Once the user info or his data is stored in the cloud environment, the user can re-login and choose various services (such as saving data, reading data and many more). If the user chooses to read his data stored in the cloud environment he needs to verify his account details (2nd level of security). Further if he succeeds a Onetime key (OTK) will be generated from the read data module (3rd level of security). If the generated OTK is similar to the OTK which is entered by the user then he can read data stored in the cloud storage otherwise warning message erupts.

Further for decryption of data a private key is required (4th level of security) and that private key should be similar to the private key stored in the cloud database. If the keys are similar then the action succeeds and decryption of data takes place. Hence, the original data shown to the user.



Implemented Algorithms: -

- ***For One Time Key (OTK) Generation:-*** For this One Time Key (OTK) generation we are using Random Number Generation Algorithm. In this we can use the *"randperm"* function to create arrays of random integer values that have no repeated values. For example:

R = randperm(15, 5)

R is a 1-by-5 array containing randomly selected integer values on the closed interval, [1, 15]. the array returned by *"randperm"* has no repeated values. This technique is useful when we want to combine results from the same random number commands executed different MATLAB sessions.

All the random number functions, "rand", "randn", "randi", and "randperm", draw values from a shared random number generator. Every time we start MATLAB, the generator resets itself to the same state. Therefore, a command such as randperm(2,2) returns the same result any time we execute it immediately following startup. Also, any script or function that calls the random number functions returns the same result whenever we restart. One way to get different random numbers is to initialize the generator using a different seed every time. Doing so ensures that we don't repeat results from a previous session.

Execute the *"rng('shuffle')"* command once in our MATLAB session before calling any of the random number functions.

rng('shuffle')

We can execute this command in a MATLAB Command Window, or we can add it to our startup

file, which is a special script that MATLAB executes every time we restart. Now, execute a random number command.

R1 = randperm (2, 2)

Each time you call rng('shuffle'), it reseeds the generator using a different seed based on the current time. Alternatively, specify different seeds explicitly. For example,

rng(1);
R1 = randperm(2, 2);
rng(2);
R2 = randperm(2, 2);

Arrays R1 and R2 are different because the generator is initialized with a different seed before each call to the randperm function.

- ***For Security of Saving and Retrieving Data:-*** For security of data while saving and retrieving in cloud from device we are using RSA cryptosystem. RSA is one of the first practicable public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it wasn't declassified until 1997.

In RSA we create and then publish a public key based on the two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message. Breaking RSA encryption is known as the RSA problem. It is an open question whether it is as hard as the factoring problem.

The RSA algorithm involves three steps: **Key generation**, **Encryption** and **Decryption.**

- **Key generation:-**

RSA involves a *public key* and a *private key*. The public key can be known by everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated in the following way:

1. Choose two distinct prime numbers p and q.
- For security purposes, the integers p and q should be chosen at random, and should be of similar bit-length. Prime integers can be efficiently found using a primarily test.
2. Compute $n = pq$.
- n is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.
3. Compute $\varphi(n) = \varphi(p)\varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$, where φ is Euler's totient function.
4. Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$; i.e., e and $\varphi(n)$ are co prime.
- e is released as the public key exponent.
- e having a short bit-length and small Hamming weight results in more efficient encryption – most commonly $2^{16} + 1 = 65{,}537$. However, much smaller values of e (such as 3) have been shown to be less secure in some settings.
5. Determine d as $d \equiv e^{-1} \pmod{\varphi(n)}$; i.e., d is the multiplicative inverse of e (modulo $\varphi(n)$).

- This is more clearly stated as: solve for d given $d \cdot e \equiv 1 \pmod{\varphi(n)}$
- This is often computed using the extended Euclidean algorithm. Using the pseudo code in the *Modular integers* section, inputs a and n correspond to e and $\varphi(n)$, respectively.
- d is kept as the private key exponent.

The *public key* consists of the modulus n and the public (or encryption) exponent e. The *private key* consists of the modulus n and the private (or decryption) exponent d, which must be kept secret. p, q, and $\varphi(n)$ must also be kept secret because they can be used to calculate d.

- **Encryption:-**

Encryption is the process of converting original plain text (data) into cipher text (data). Cloud service provider should give or transmit the Public- Key (n, e) to the user who wants to store the data with him or her. User data is now mapped to an integer by using an agreed upon reversible protocol, known as padding scheme. Data is encrypted and the resultant cipher text (data) C is

$$\mathbf{C = m^e \pmod{n}.}$$

This cipher text or encrypted data is now stored with the Cloud service provider.

- **Decryption:-**

Decryption is the process of converting the cipher text (data) to the original plain text (data). The cloud user requests the Cloud service provider for the data. Cloud service provider verifies the authenticity of the user and gives the encrypted data i.e. C. The Cloud user then decrypts the data by computing,

$$\mathbf{m = C^d \pmod{n}.}$$

Once m is obtained, the user can get back the original data by reversing the padding scheme.

Implementation of Proposed Framework: - In this proposal I am proposing multilevel of security for data while saving and retrieving from cloud and mobile device. This secure framework is worked only for one to one mapping basis. In this framework security is not only done on the part of data saving from mobile device to cloud but it also provide security on the part of data retrieving from cloud to mobile device. In this we done encryption/decryption on data, also use One Time Key (OTK) method for security check. When owner wants to read its encrypted data, he needs to verify OTK also private key to decrypt the data.

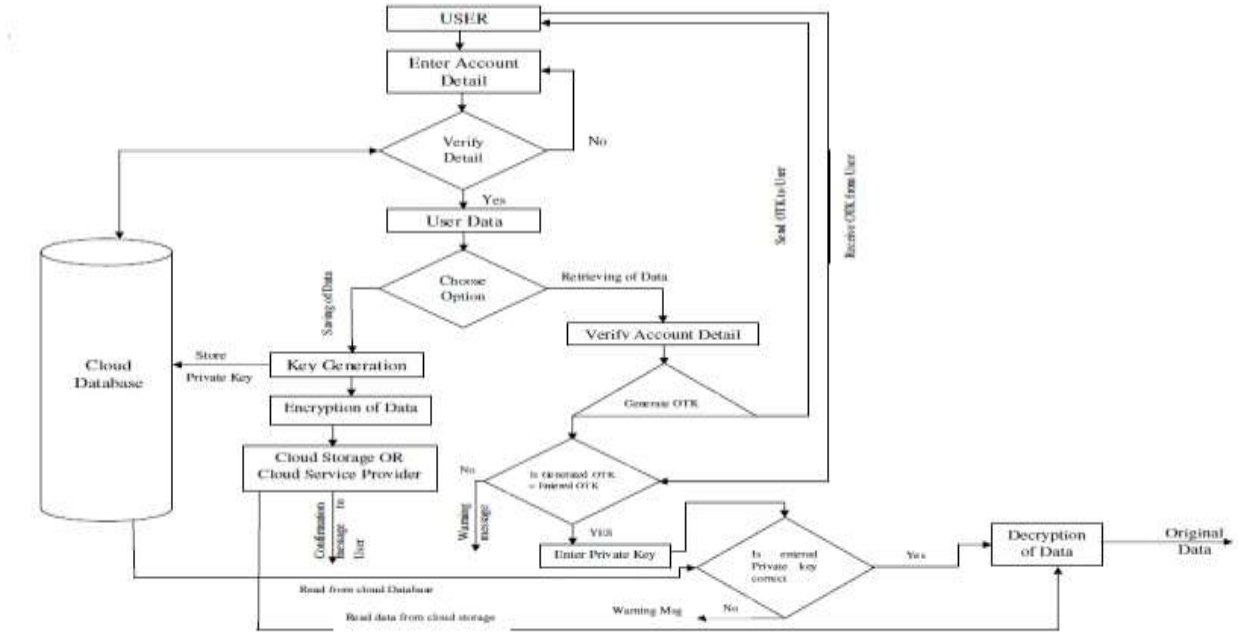


Figure 3.6 – Flow Chart of Proposed Framework

Figure 3.4 shows a flowchart of proposed framework in which whole working is explained. According to this first user register his credentials (creation of account if he is not already registers) or if he has, then he has to enter the details which are verified with the cloud database provided by the cloud database provider. After successfully login, user chooses the option according to his need. If he didn't save his data on the cloud then he have to save data first. For saving of data, first, module will generate public and private key and this is done using a key generation method (RSA Algorithm). When key generation method generates both key then public is forwarded for encryption of data and private key is stored in cloud database. Now encrypted data (which is encrypted with the help of public key) is stored on cloud provided by cloud service provider. When data is successfully stored service provider will send a confirmation message to the user.

Once the user data is stored in the cloud environment, further user can re-login and choose various services (such as save data, read data and many more). If the user chooses to retrieve his, stored in the cloud environment he needs to verify his account details and further if he succeeds an OTK (One Time Key) will be generated from the read data module. This generated OTK is received by the user and also forwarded to verify with the OTK which is entered by the user (same OTK which is received by the user). If generated OTK is similar to the OTK entered by the user then he can proceed to read data otherwise warning message will erupts. Further for decryption of data user have to enter his private key and this private key will match with the same private key which is stored in cloud database. If entered private key is similar to stored private key in database then action succeeds and the decryption of data takes place otherwise warning message will shown. And then user will get its original data to read.

III. RESULTS

Characteristics of Proposed Framework: The proposed framework has some dynamic properties that make it further more powerful, these characteristics are as follows:

- Secure: - It has multilevel security which makes it more secure also it has one time key mechanism which makes this framework more powerful because one time key needs user verification.
- Minimal: - The size of each piece does not exceed the size of the original data.

- Extensible: - When Key (K) is kept fixed, Data (Di) pieces can be dynamically added or deleted without affecting the other pieces.
- Dynamic: - It can be easily enhanced by introducing the communication of data between two separate parties. In this framework we can easily share our data between two parties after extending this framework.
- Flexible: - This proposed framework is flexible because it has more scope to extend and work on it.

Results

- **The data stored in the cloud: -** One of the most important issues related to security risks is data integrity. The data stored in the cloud may suffer from damage during transition operations from or to the cloud storage provider.
- **Address the security issues: -**Multi-level authencity can address the security issues that relate to data integrity, data intrusion, and service availability. In addition, most of the research has focused on providing secure cloud storage. Therefore, providing a cloud database system, instead of normal cloud storage, is a significant goal in order to run queries and deal with databases, in other words, to profit from a database-as-a-service facility in a mobile cloud computing environment.
- **Apply the Encryption algorithm on the data (before storing): -** In relation to data intrusion and data integrity, assume we want to store the data into storage provided by cloud storage providers, and we apply the RSA encryption algorithm on the data which is going to be stored in the storage of service provider. An intruder needs to retrieve at least three values to be able to find out the real value that we want to hide from the intruder.

Key Generation: -

Suppose we have chosen 2 distinct prime number a = 61 and b = 53 at random also it is similar bit length.

Computing n = a * b = 61 * 53 = 3233

Now computing Euler's totient function Ø(n) = (a-1) * (b-1) = (61-1)*(53-1) = 3120

Choosing any integer e, such that 1<e<3120 that is co-prime to 3120. Here we choose e =17.

Computing d, $d = e^{-1} (\text{mod } Ø(n)) = 17^{-1} (\text{mod } 3120) = 2753$

Thus the Public Key (e, n) = (17, 3233) and the Private Key (d, n) = (2753, 3233). This private key is kept secret and it is known only to the user.

Encryption: -

Above Public Key (17, 3233) is used by the user who wishes to store the data.

Let us consider that the user mapped the data to an integer m =65.

Data is encrypted now by using the corresponding Public Key $C = 65^{17} (\text{mod } 3233) = 2790$

The encrypted data i.e. cipher text is now stored by the cloud service provider.

- **Apply Decryption and Random No generation algorithm (to retrieving data): -**

In relation to data intrusion and data integrity, suppose we want to retrieve the data from storage provided by cloud storage providers, we apply Random Number generation and RSA encryption algorithm. In this framework first we match user password with database and then generate a onetime key using random number generation algorithm which verify the authencity of the user, then it uses RSA decryption algorithm to retrieve the original data.

Random Number Generation: -

p = randperm(n,k) returns a row vector containing k unique integers selected randomly from 1 to n inclusive.

Suppose we are taking randperm(6,3)

might be the vector [4 2 5]

or it might be some other permutation of any three integers from 1 to 6 inclusive, depending on the state of the random number generator.

Decryption: -

When user requests the stored data, framework first authenticates the user to ask the password. Now this framework generates one OTK (one time key) for another level of authentication. When user enters generated OTK and it is verified by framework then cloud service provider delivers the encrypted data.

The user then decrypts the data by computing,

$m = C^d (\text{mod } n) = 2790^{2753} (\text{mod } 3233) = 65$

Once the m value is obtained, the user will get back the original data.

Discussion

Any progress must first occur in a particular domain – accordingly, our work focuses on important class of widely used application that include photos,

important document such as word processer and spread sheets, important notes etc.

Some research has been done on the security of data in mobile cloud computing but all are focusing on the storing of data portion. In this framework we are focusing not only security while storing of data but we are focusing on security of data while retrieving.

	Basic theory	Data Protection		D I	Au	Sc
		B S	B R			
Itani et al. [5]	Incremental message authentication code	-	-	Yes	-	Moderate
W. Jia et al. [9]	Proxy re-encryption scheme and identity base encryption scheme	Yes	-	-	-	Highly scalable
S. C. Hsueh et al [10]	Standard cryptography functions	Yes	-	Yes	Yes	Moderate
J. Yang et al [11]	Diffie-Hellman key exchange, bilinear mapping, and merkle hash tree	Yes	-	Yes	Yes	Moderate
Ren et al. [12] (EnS)	Standard cryptography functions	Yes	-	Yes	Yes	Highly scalable
Ren et al. [12] (CoS)	Coding Vector	Yes	-	Yes	Yes	Highly scalable
Ren et al. [12] (ShS)	Exclusive-OR	Yes	-	Yes	Yes	Highly scalable
Z. Zhou and Huang [13]	Bilinear pairing, access policy tree, and secret sharing scheme	Yes	-	-	-	Highly scalable
Proposed Framework	RSA key encryption, decryption and random no generation scheme	Yes	Yes	Yes	Yes	Highly scalable

Table 1 – Comparison of evaluated data security framework

B S – Before Storing, B R – Before Retrieving, D I – Data Integrity, Au – Authentication, Sc – Scalability.

IV. CONCLUSION

The purpose of this dissertation is to provide the data security while saving. We have found that much research has been done to ensure the security of data from mobile to cloud storage whereas every research focuses on the security of data while storing it on mobile cloud storage. In our dissertation we focuses not only on the security of data while storing it on cloud but also we propose security while retrieving the data from cloud to mobile.

At last we are proposing on framework that decreases the risk regarding the data security in mobile cloud computing environment. It helps the user to feel easy regarding its data while using mobile cloud environment.

V. REFERENCES

1. **M. Satyanarayanan,** "Fundamental challenges in mobile computing," in Proceedings of the 5th annual ACM symposium on Principles of distributed computing, pp. 1-7, May 1996.
2. White Paper, "Mobile Cloud Computing Solution Brief", AEPONA, November 2010.
3. Anand Surendra Shimpi and R. Chander, *"Secure Framework in Data Processing for Mobile Cloud Computing"*, International Journal of Computer & Communication Technology, ISSN (Print) 0975- 7449, vol. 3, Iss. 3, 2012.
4. Jibitesh Mishra, Sanjit Kumar Dash and Sweta Dash, "Mobile Cloud Computing: A Secure Framework of Cloud Computing for Mobile Application", Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 2012, pp. 347- 356.
5. Itani et al, "Towards secure mobile cloud: A survey", Proceedings of Analyses paper, 2012.
6. Eugene E. Marinelli, "Hyrax: Cloud Computing on Mobile Devices", Dissertation of Thesis, Carnegie Mellon University, Pittsburgh, 2009.
7. Jon Oberheide and Evan Cooke, "Virtualized in-cloud security services for mobile devices", Proceedings of the First Workshop on Virtualization in Mobile Computing, ACM, New York, USA, 2008, pp 31–35.
8. A. N. Khan, M. L. M. Kiha, S. U. Khan, S. A. Madani (2012), *"Towards secure mobile cloud computing: A survey"*.
9. W. Jia, H. Zhu, Z. Cao, L. Wei, X. Lin, SDSM: a secure data service mechanism in mobile cloud computing, in: Proc. IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, Shanghai, China, Apr. 2011.
10. S.C. Hsueh, J.Y. Lin, M.Y. Lin, Secure cloud storage for conventional data archive of smart phones, in: Proc. 15th IEEE Int. Symposium on Consumer Electronics, ISCE '11, Singapore, June 2011.

11. J. Yang, H. Wang, J. Wang, C. Tan, D. Yu1, Provable data possession of resource constrained mobile devices in cloud computing, Journal of Networks 6 (7) (2011) 1033–1040.

12. W. Ren, L. Yu, R. Gao, F. Xiong, Lightweight and compromise resilient storage outsourcing with distributed secure accessibility in mobile cloud computing, Journal of Tsinghua Science and Technology 16 (5) (2011) 520–528.

13. Z. Zhou, D. Huang, Efficient and secure data storage operations for mobile cloud computing, IACR Cryptology ePrint Archive: 185, 2011.

