Data Security on A Cryptographic Approach by Outsourcing Mobile data to Cloud

G Sai Ram[1] P Sony[2] Abdul Raheem[3] L Venkat Girish[4]
[1]*Assistant professor, Dept.of ECE, Siddhartha Institute of Engineering & Technology, Ibrahimpatnam, Hyderabad, Telangana, India.*
[2]*Assistant professor, Dept.of ECE, Siddhartha Institute of Engineering & Technology, Ibrahimpatnam, Hyderabad, Telangana, India.*
[3]*Assistant professor, Dept.of ECE, Siddhartha Institute of Engineering & Technology, Ibrahimpatnam, Hyderabad, Telangana, India.*
[4]*Student, Dept.of ECE, Siddhartha Institute of Engineering & Technology, Ibrahimpatnam, Hyderabad, Telangana, India.*

Abstract - Today data communication mainly depends upon digital data communication, where prior requirement is data security, so that data should reach to the intended user. So for providing data security many cryptography techniques are employed, such as symmetric and asymmetric techniques. Although cryptography and steganography could be used to provide data security, each of them has a problem. Cryptography problem is that, the cipher text looks meaningless, so the attacker will interrupt the transmission or make more careful checks on the data from the sender to the receiver. Steganography problem is that once the presence of hidden information is revealed or even suspected, the message is become known. When computing capacity and storage need of these devices are increasing tremendously, it demands the secure way of storing the data in cost efficient model. This paper describes how securely the mobile data can be stored in the remote cloud using cryptographic techniques with minimal performance degradation

I. INTRODUCTION

Increase in mobile device sophistication also demands high storage sophistication. Mobile Cloud Computing (MCC) service, allows resource constrained mobile users to adaptively adjust processing and storage capabilities by transparently partitioning and offloading the computationally intensive and storage demanding jobs. Also, increase in feature rich mobile applications like mobile wallets, banking apps, and healthcare app etc. exposes sensitive data of the users which are always prone to much vulnerability. The value of data is far more important than the value of device. The main issue in using MCC is securing the user data on mobile cloud since there is a high risk of unauthorized access to the data [1]. So, the main concern of cloud service provider is to provide data security by the same time providing ease of access to the authorised user. New service architectures are necessary to address the security concerns of the mobile data, without any performance issue. Considering all these constraints research has been performed and this paper briefs its result.

Smart phones have certain security mechanism in place to deal with unauthorized access to sensitive and secret data. The most common protection mechanism is password-based protection [2]. User assign password to critical files and data. This password is then used to allow access or deny based upon the correct or incorrect password. This approach has some drawback. The major one is that the password can be seen by others. The unauthorized user attacker can then later use the same password to get an entry into secured data. Pattern used for providing security also have the same problems. Biometric based security is also in place but requires precision and cannot be shared with others in case of emergency. The more advanced security approach is to encrypt the data by the strong encryption algorithm.

II. CRYPTOGRAPHIC APPROACH

The encryption algorithm is most commonly used technique to protect data within cloud environment. The data related to a client can be categorized as public data and private data. The public data is sharable among trusted clients that provide an open environment for collaboration. Private data is client's confidential data that must be transferred in encrypted form for security and privacy [4]. We propose a suitable method that cryptographic algorithms with different key lengths are used in various environments. The number of mobile devices such as smart phones and smart pads grows rapidly recently [5]. End users can access easily to cloud computing environment thought these mobile devices we define that mobile cloud computing is one of specific services of cloud computing and it is a mobile service which is added a cloud computing service [6].

According to key characteristics, modern cryptosystem can be classified into symmetric cryptosystem, asymmetric cryptosystem and digital signature. For a symmetric cryptosystem, the sender and receiver share an encryption key and decryption key. These two keys are the same or easy to deduce each other. The representatives of symmetric cryptosystem are DES (Data Encryption Standard), AES (Advanced Encryption Standard) [7]. For an asymmetric cryptosystem, the receiver possesses public key and private key. The public key can be published but the private key should be kept secret [8-9]. The representatives of asymmetric cryptosystem are RSA (Rivest Shamir Adleman) and ECC (Elliptic Curve Cryptosystem). For Digital signature the representatives is MD5 and SHA1.

III. PROPOSED WORK

The encrypted data is stored in smart phone memory which is then later used for accessing by decrypting the data. All of this method usually processes and computes in the memory of smart phone which is still prone to unauthorized access. In case of lost or theft of smart phone the unauthorized person can have access to the device and a technically proficient person can get an entry into data as encryption techniques are all present on the device itself [3]. In order to overcome these problems, three-tier security using cloud architecture is being proposed as a hybrid approach. In three-tier hybrid approach, as a First tier security, encryption is done using MD5 algorithm with the key (k1) given by the user. As Second-tier security, these encrypted data are again encrypted using AES algorithm. As a third-tier security further encryption of data or key using ECC or RSA algorithm is performed respectively, ensuring more security and the key is shared with the user. All the above methods are performed in both Local and remote environment as shown in the below figure 1.

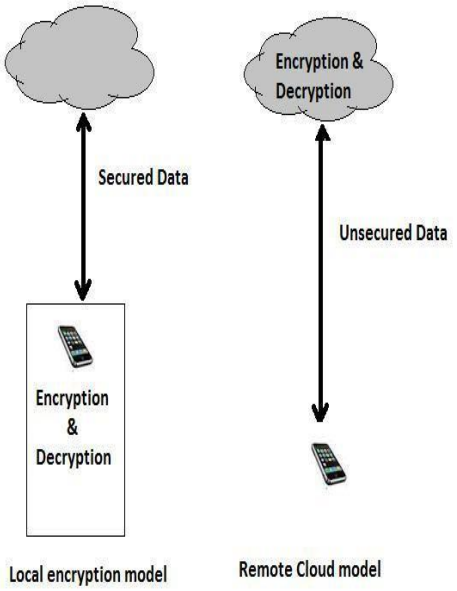

Fig: 1 Practiced Encryption Models

3.1 Implemented Algorithms

The cryptographic algorithms used are Symmetric key algorithms, Asymmetric key algorithms and Combination of these algorithm as a Hybrid Approach. Evaluation metrics for these algorithms are studied based on various previous research work. Encryption techniques will make the data more secure in the local system as well as on the remote cloud. Test has been executed in both the above environment using the following algorithms one at a time.

- AES: In cryptography, the Advanced Encryption Standard (AES) is a symmetric-key encryption standard. Each of these ciphers has a 128-bit block size, with key sizes of 128, 192 and 256 bits, respectively.

- DES: The DES (Data Encryption Standard) algorithm is the most widely used encryption algorithm in the world. For many years, and among many people, "secret code is making" and DES have been synonymous meaning it operates on plaintext blocks of a given size (64-bits) and returns cipher text blocks of the same size.

- RSA: RSA is an algorithm for public-key cryptography, involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. It protected user data include encryption prior to storage, user authentication procedures prior to storage or retrieval, and building secure channels for data transmission

- MD5: a widely used cryptographic hash function with a 128-bit hash value processes a variable-length message into a fixed-length output of 128 bits. The input message is broken up into chunks of 512-bit blocks; the message is padded so that its length is divisible by 512. In this sender use the public key of the receiver to encrypt the message and receiver use its private key to decrypt the message.

- Elliptic Curve Cryptography (ECC) with SHA-512: An elliptic curve is given by an equation in the form of The finite fields those are commonly used over primes (FP) and binary field (F2n). The security of ECC is based on the elliptic curve discrete logarithm problem (ECDCP). This problem is defined as: Given point X, Y on elliptic curve, find z such that X=zY. The following steps describe how ECC works with SHA-512 [9], [10].

  ECC key generation: To generate a public and private key pair for use in ECC communication the steps followed are:

  Find an elliptic curve E(K), where K is a finite field such as Fp or F2n, and a find point Q on E(K). n is the order of Q.

  Select a pseudo random number x such that $1 \leq x \leq (n -$ 1). Compute point P = xQ.

  ECC key pair is (P, x), where P is public key, and x is private key.

- MD5, AES, ECC Hybrid approach: In order to increase the level of security, hybrid of Symmetric and asymmetric key algorithms are used. In this method, actual data is encrypted with MD5 algorithm and the encrypted file is further encrypted with AES and then with ECC ensuring 3 levels of encryption.

- MD5, AES&RSA Hybrid approach: In this technique, actual data is encrypted with MD5 and the encrypted file is further encrypted using AES algorithm. Unlike the previous method, here the generated AES key is encrypted using RSA rather than encrypting the actual data.

Cloud architecture is designed by combining cryptographic algorithms with Mobile device environment. The cryptographic algorithms to be used are selected based on comparative study from previous researches. So, the symmetric, asymmetric and digital signature algorithms AES, DES, RSA, ECC, and MD5 are selected and used for cryptographic application. The cryptographic application is used to encrypt and decrypt data, provides options to application user whether to use asymmetric with digital signature or symmetric algorithm.

Steps Performed:

- Create some input data samples of sizes 25kb, 50kb, 75kb, 100kb, 125kb and 150 kb.
- Run the cryptographic algorithms with all input data sizes in mobile device.

- Make a cloud server instance on application tool and then make a dynamic web project.
- Run the encryption algorithms on cloud server input data sizes and note all observations
- Compare both the results.

Results are compared based on the performance metrics Mean Processing Time and Speed-Up Ratio.

Mean Processing Time:

Mean processing time is the difference between the starting time taken to encrypt the data and the ending time. It is also evaluated both on single system and on cloud network. It is the difference between the time taken to encrypt the data. As the size of input increases the time taken to encrypt the data will increase and with the increase in time speed-up ratio decreases.

Speed-Up Ratio:

It is defined as the difference between the mean processing time of single system and the cloud network. Speed-up ratio will provide tell us how fast the data have been encrypted. It will give us the idea about speed of encryption.

Speedup Ratio = Processing Time in Local − Processing Time in Cloud

IV. RESULTS OBSERVED

Figure 2, 3, 4, 5, 6 are the results observed while performing the above analysis.

CLOUD

	Encryption							Decryption						
Size	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES
25	0.003	0.012	0.043	0.018	68.082	18.736	0.046	0.004	0.020	3.860	0.019	8.192	11.425	3.520
50	0.004	0.015	0.066	0.008	149.105	36.818	0.106	0.006	0.032	3.550	0.008	86.697	22.529	3.801
75	0.005	0.022	0.035	0.028	250.780	55.088	0.132	0.015	0.032	3.547	0.012	145.986	53.528	3.569
100	0.015	0.030	0.049	0.021	330.595	73.243	0.134	0.011	0.030	3.579	0.014	200.769	45.133	3.582
125	0.023	0.038	0.030	0.015	522.671	222.564	0.150	0.012	0.039	3.600	0.025	281.470	294.979	5.287
150	0.012	0.044	0.027	0.017	636.106	286.087	0.123	0.021	0.044	3.571	0.026	263.552	423.973	3.869

Local

	Encryption							Decryption						
Size	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES
25	0.345	0.012	0.109	0.075	740.011	63.704	0.113	0.145	0.020	6.615	0.152	368.992	59.963	7.263
50	0.326	0.015	0.174	0.130	1045.055	161.999	0.101	0.136	0.032	7.355	0.281	522.503	147.419	7.160
75	0.454	0.022	0.100	0.167	1456.445	198.352	0.492	0.203	0.032	7.701	0.427	561.147	184.467	6.363
100	0.632	0.030	0.099	0.231	1668.667	322.271	0.098	0.207	0.030	8.237	0.619	667.449	302.934	6.720
125	0.796	0.038	0.078	0.269	2013.605	791.719	0.196	0.251	0.039	4.847	0.758	806.419	735.368	5.019
150	0.921	0.044	0.324	0.359	2301.879	1658.322	0.212	0.286	0.044	6.383	0.852	886.311	973.040	4.755

Fig: 2 Mean-Processing time table

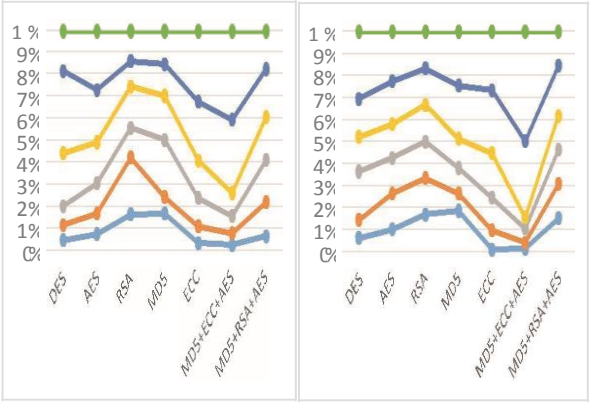

Fig: 3 Encryption and Decryption in Cloud Environment

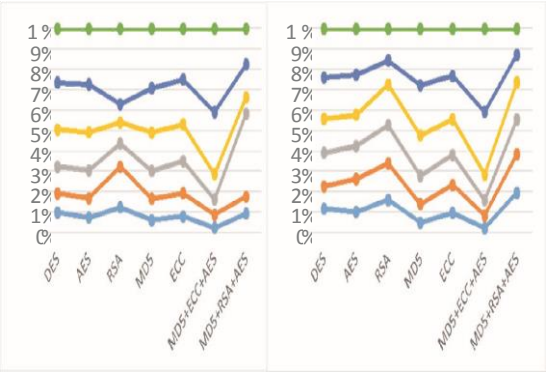

Fig: 4 Encryption and Decryption in Local Environment

Speed_up_ratio

	Encryption						Decryption					
Size	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES	DES	AES	RSA	MD5	ECC	MD5+ECC+AES	MD5+RSA+AES
25	0.342000	0.000000	0.166667	0.667333	671.928333	44.967200	0.067000	0.141000	0.000000	0.016667	0.126667	351.799333	39.637027	3.743000
50	0.321333	0.000000	0.180000	0.422333	955.810000	105.191200	0.004667	0.130333	0.000000	0.005333	0.222333	435.809333	124.898985	3.449667
75	0.449667	0.000000	0.065333	0.130333	1265.894667	143.253833	0.369000	0.107333	0.000000	4.153667	0.415000	411.161021	130.938655	2.794000
100	0.617000	0.000000	0.041667	0.266667	1338.077667	249.027333	0.059667	0.185667	0.000000	4.658000	0.594667	466.879667	257.801427	3.130333
125	0.777667	0.000000	0.048333	0.253333	1490.934000	559.154400	0.049000	0.239000	0.000000	0.246667	0.733000	523.340333	440.286755	-0.269667
150	0.936000	0.000000	0.207667	0.342000	1665.773667	772.434980	0.086667	0.237667	0.000000	2.912333	0.826667	821.758492	649.865491	1.145333

Fig: 5 Speed-Up ratio table

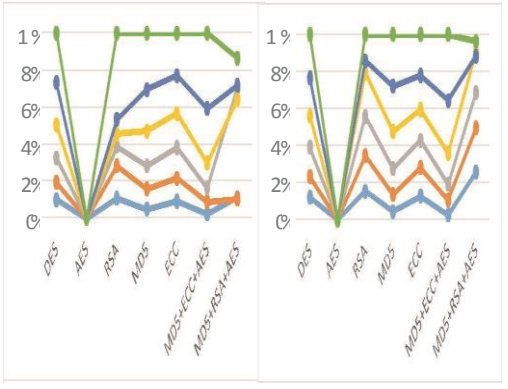

Fig: 6 Speed-up Ratio Chat for Encryption and Decryption

V. CONCLUSION AND FUTURE WORK

In the older techniques these cryptographic algorithms are implemented in the Single system environment. Now due to availability of high-performance computing techniques, similar test has been conducted in the single system environment i.e. local environment and also in the Cloud environment. From the observed results, and based on the considered parameters, storing the mobile data in cloud increases the efficiency. Also, the results reveal that AES algorithm qualifies better than other algorithms in Mean processing time and combination of MD5+ECC+AES algorithm qualifies better than others in Speed-Up ratio. Since it is not wise to take a decision considering only these parameters, other performance measures like Turn-around time, Throughput are planned to be included in the future work.

VI. REFERENCES

[1]. Kumar, K., Lu, Y.-H.: Yung-Hsiang Lu: Cloud Computing for Mobile Users: Can Offloading Computation Save Energy? Computer 43(4), 51– 56 (2010)

[2]. Simoens, P., De Turck, F., Dhoedt, B., Demeester, P.: Remote Display Solutions for Mobile Cloud Computing. Computer 44(8), 46–53 (2011)

[3]. Ayesha Malik, Muhammad Mohsin Nazir, "Security Framework for Cloud Computing Environment: A Review," Journal of Emerging Trends in Computing and Information Sciences, 2012.

[4]. Shashi Mehrotra Seth, Rajan Mishra, "Comparative Analysis Of Encryption Algorithms For Data Communication," IJCST Vol. 2, Iss ue 2, June 20 I I .

[5]. Shahryar Shafique Qureshi1 , Toufeeq Ahmad1, Khalid Rafique2, Shuja-ul-islam3 "Mobile cloud computing as future for mobile applications – implementation methods and challenging issues"-2011.

[6]. Mell P, Grance T (2011) The NIST definition of Cloud Computing. NIST, Special Publication 800–145, Gaithersburg, MD 29.

[7]. Zhang Q, Cheng L, Boutaba R (2010) Cloud Computing: state-of-the-art and research challenges. Journal of Internet Services Applications 1(1):7–18.

[8]. Pearson, S., Y. Shen, and M. Mowbray, "A Privacy Manager for Cloud Computing", in Proceedings of the 1st International Conference on Cloud Computing. 2009, Springer-Verlag: Beijing, China. p. 90-106.

[9]. Wang, Q., et al., "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing", in Computer Security – ESORICS 2009, M. Backes and P. Ning, Editors. 2009, Springer Berlin / Heidelberg. p. 355-370.