International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-5, June 2019

Secure Key Management System in Cloud Environment for Client data

Pradeep. K. V, Vijayakumar. V

Abstract: ***In this evolving technology era, cloud computing has emerged drastically by the means of the Internet and remote server for maintaining applications and data. In actual none of the physical resources are owned by the cloud computing customers, rather they are rented from some third party. The major management approaches involve Users initialization and key generation, expiration and destruction. Significant information is being transferred by the enterprises over the cloud,this leads to the concerning issue of data security. For safeguarding this critical information, the technique of Cryptography is being utilized. The technique of Cryptography incorporates the handling of encryption and decryption keys. With the recommended approach of SKMS (Secure Key Management System), the user's data is stored in the cloud area. By logging with the essential credentials, the user can upload the required files. Files are stored by the authorized user in the cloud domain thereby generating key via key generation. User Data is Encrypted via RSA, which in turn generates both Public and Private Keys. Now the Private Key (Say K_1) is again re-encrypted via ECC to obtain the encrypt key (Say K_2). Thereafter the re-encrypted key K_2 is split into 'N' parts which are located in the cloud environment. For decrypting the file, key K_1 is required, hence 'K' parts are pulled out of 'N' parts from the cloud such that (K<N). Next, by making use of ECC, key K_2 is decrypted to obtain the original key K_1. Now the original K_1 is used to decrypt the file via RSA. The handling stages of the SKMS (SKMS) technique involves user registration and login, file uploading, key-encryption, key-splitting, key-decryption and decryption of the original file. Its depicted form the experimental analysis that SKMS minimizes time complexity of generating encryption and decryption key. This makes it useful in contrast to the already prevailing system***

Index Terms: Cloud Computing, Cryptography, Secure Key Management System (SKMS), Advanced Encryption Standard, Elliptic Curve Cryptography, Shamir algorithm, Key generation, Key encryption, Key splitting, Key decryption.

I. INTRODUCTION

Cloud computing has grown exponentially over time, as a result, massive data is centralized within the cloud for sharing purpose [1].By employing the technique of decryption, the original information can be concealed in a better way. Major threat confronted by cloud data sharing is the risk of security that usually surfaces during online storage of data as there is no protection from various malicious attacks [2]. To guard against such attacks, the technique of encryption and decryption can be enforced which transforms that data and presents it in a scrambled form to the public [3].For maintaining the security and privacy of the owner's data, the sharing of data must be encrypted and then uploaded through proper access control. The cryptographic encryption algorithms can be split into 2 parts based on the key: first, symmetric key encryption and second, asymmetric key encryption [4].

In the symmetric encryption algorithm, only one key is utilized for encryption as well as decryption. On the other hand, the algorithm of asymmetric encryption, two keys are being employed, one key for carrying out encryption and second key for carrying out decryption. From the two keys, one resembles a private key and the other resembles a public key. The encryption technique helps in achieving data security from malicious attacks. By utilizing this encrypted key, the user can store encrypted files over the cloud. For accessing this data, the decryption key is utilized which decrypts the data received.

The existing research proposes the approach of SKMS (Secure Key Management System)minimizes time complexity of generating keys. Using SKMS user's data is stored in the cloud area. By logging with the essential credentials, the user can upload the required files. Files are stored by the authorized user in the cloud domain thereby generating key via key generation. User Data is Encrypted via RSA, which in turn generates both Public and Private Keys.Now the Private key (say K_1), again re-encrypted via ECC (Elliptic Curve Cryptography) to obtain key K_2. Thereafter the re-encrypted key K_2 is split into 'N' parts which are located in the cloud environment. For decrypting the file key K_1 is required, hence K parts are pulled out of from N parts in the cloud such that (K<N). Next, by making use of ECC, key K_2 is decrypted to obtain the original key K_1.

Now the original K_1 is decrypted via RSA. The handling stages of the SKMS (SKMS) technique involves user registration and login, file uploading, key-encryption, key-splitting, key-decryption and decryption of the original file. The proposed approach is highly appropriate for cloud scenario for storage of data/files. It'sespecially beneficial for organizations for managing key for various resources.

Following is the classification of the journal: section 2 elaborates work of the previous author. Section 3 put forth the recommended key management system and aspects of various stages. Section 4 exhibits experimental output. Section 5 presents the conclusion thereby proposing research work for the future.

II. RELATED WORK

GUOFENG LIN et.al, recommends a protocol for collaborative key management in CP-ABE. With the existing infrastructure itself, distributed generation as well as issuing and storing of private keys is taken into consideration.

Manuscript published on 30 June 2019.
***** Correspondence Author (s)
Pradeep. K.V, SCSE, VIT-Chennai Campus, Chennai, TN, India.
Vijayakumar . V, SCSE, VIT-Chennai Campus, Chennai, TN, India.

For updating the keys, a well formulated and immediate attribute revocation is provided for the key update. The collaborative technique resolves key escrow issue as well as a key exposure. On the other hand, it aids in markedly lower client decryption overhead. The proposed scheme, when compared with the rest of the representative CP-ABE approaches, exhibits improvisedperformance with respect to cloud-based outsourceddata sharing on mobile devices. Lastly, security proof is provided for the suggested protocol [5].

Sathishkumar EaswarMoorthy et.al presents that the process of revocation involves evoking the users from their access. EKMI makes use of a DKPABE (Decentralized Key Policy Attribute-based Encryption) along with user revocation within Private Domain and MACPABE (Multi-Authority Cipher-text Policy Attribute-Based Encryption) with attribute revocation within Public Domain. The approach of Lazy revocation is employed for minimizing the cloud server's computation overhead. In the above algorithms, the EKMI system imbibes Tokenization approach and is effective enough against collusion attacks [6].

Lars Keuninckx et.al proposes a novel encryption scheme, which involves generating an encryption key by 2 distant complex non-linear units that are enforced via chaotic driver into synchronization. The proposed scheme being adequately generic for employing on any of the three platforms, that is - photonic, optoelectronic or electronic. The method adopted for generating the key bit-stream from the chaotic signals can be configured again.Theresultant bit series satisfies the randomness conditions (which is obtained from a deterministic process) and as stated by the National Institute of Standards test suite [7].

Jialuo Han et.al put forth a LoRaWAN 1.1 a root key update scheme is recommended for the evaluation of security and improvised LoRaWAN security. The proposed scheme makes cryptanalysis of security keys quiet arduous in LoRaWAN. On the basis of evaluation and simulation its depicted that the recommendedscheme of root key update needs less computing resources in contrast to rest of the key derivation schemes, even the scheme employed in LoRaWAN session key update. It's depicted from the output that the key generated in the scheme reveals an increased degree of randomness, which is a pre-requisite in case of a security key [8].

Hyunsoo Kwon et.al, present a de-duplication scheme which is suggested for useful convergent key management. The scheme has demeritwith respect to the scalability and key management security. For resolving these issues, a novel secure de-duplication scheme is being recommended offering scalable and reliable key management on the basis of paring based cryptography. It requires no additional secure channels for distributing key components at the same time assuring secure key management which was not the case in prior schemes [9].

Kemal Bicakci et.al, have suggestedTwinCloud as a means of client-sidesolution offering the users a secure system with no compromise on cloud sharing usage. By employing the TwinCloud, the issue of complex key exchange can be handled, thereby providing an easy and practical approach for storing and sharing files making the cryptographic and key-distributionoperations hidden from the users., TwinCloud serves as a gateway for storing the encryption keys and files in different clouds thereby making the secure sharing smooth with placing credibility on eitherof the CSPs (cloud service providers)assuming that theydon't collude among each other [10].

Dinkar Sitaram et.al, emphasizes on the need for security when transferring data within or across the clouds, restricting any third party from accessing the transferred information. Encryptionand decryption keys are employed while transferring such data which is controlled by KLMs (Key LifecycleManagers) that utilizes a standardizedprotocol for communication. Hence it's essential that the cloud platforms give a provision for a standardized protocol to enable end-user flexibility, in selecting a trustworthy KLM thus granting integration with external Key Lifecycle Managers [11].

M. Thangavel et.al presents a revised and improvised scheme relying upon RSA public-key cryptosystem. This scheme makes use of 4 large prime numbers for increasing the system's complexity in contrast to the traditional RSA algorithm that relies upon only 2 large prime numbers. ESRKGS (Enhanced and Secured RSA Key Generation Scheme) makes use of a public component 'n' which is obtained by multiplying two large prime numbers, whereas the values for Encryption key- E and Decryptionkey –D is obtained by multiplying four large prime numbers (N) thus resulting in ahighly secure system. The present factorization techniques can just determine the primes p and q. For determining the value of E and D, identifying'n'is not sufficient enough since they rely upon N [12].

Muthi Reddy et.al recommends a novel algorithm DPAK-RE (Dynamic Privacy Aggregate Key Re-Encryption). The focus is laid upon the sharing of secure data with other cloud users. Data Owner is approved to the admittance approach associated with individualinfothat being saved within the jurisdiction. Authorized users can combinevarious groups of top-secret keys, enfolding together a solitary key, thus aggregating an entire group of keys. This aggregate key undergoes Re-encryption thereby generatinga personal key with a constant or fixed key size. By the means of this Personal like, the private information for data sharingis being secured in a cloud environment [13].

Yunpeng Zhang et.al put forth a practical solution for the generation of OTP symmetric key and transmission issuerelated to DNA at the molecular level. Using recombinant DNA technology and by making use of sender-receiver known restriction enzymes, secure key represented by DNA sequence and the T vector is combined thus generating the DNA bio-hiding secure-key and placing the recombinant plasmid within implanted bacteria to achieve secure key transmission [14].

Pranshu Bajpai et .al comprehends and presents key management in ransomware that being a necessity for identifying weaknesses which can be utilized for defensive purposes.Key management evolution has been illustrated as matured ransomware and is assessed in around 25 samples. On the basis of the analysis, a ransomware taxonomy is launched which beingequivalent to hurricane ratings: a Category 5 ransomware is infectious enough from the viewpoint of cryptographic in contrast to Category 3 [15].
Ahmed Elhadad presents a framework that relies upon DNA-proxy re-encryption.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Initially, 3 keys are being generated, one for the owner, second for the proxy and third for the user requiring to access data. The encrypted data is then stored in the cloud by the owner with the help of his key. In case the user needs to access the data, it can be done through the proxy, post-re-encrypting via second generated key designated for the proxy. Eventually, re-encrypted data can be decrypted by the user via third generated key [16].

Rania Baashirah et.al presents a novel cryptographic scheme namely HPAP "Hacker Proof Authentication Protocol". RFIDconfronts the main issue of security and privacy. In order to handle the privacy and security of the system, various RFID authentication protocols have been recommended. Analysis of these protocol depicts that they are not capable of providing security against certain RFID attacks. The protocol for "Multiple Tags" authentication can be improvisedin a one-time process [17].

Hongxiang Gu et.al recommends a group key management scheme relying upon PUF (physically un-clonable function) design: MIPUF (multistage interconnected PUF) are incorporated for securing group communications related to an energy-constrained environment. The MIPUF design has the potential of carrying out key management tasks like key-distribution, key-storage and rekeying safely and effectively. It's elucidated that the design that it can secure against various attacks and the output reveals that the design saves 47.33% of energy at a global levelin contrast to the sophisticated key management scheme based on ECC (Elliptic-curve cryptography) on an average [18].

Saad Fehis et.al, recommends a scheme for providing CKMS (cryptography key management system) as a trusted security service in Cloud Computing, relying upon the trusted platform module (TPM / vTPM). The current scheme makes use of TPM's functions as an option for security and credibility for such type of services. Hence, in this case, TPM's key generation component is being incorporated as a credible option for generating and signing encryption keys via CKMS for concerned customers [19].

Majid R. Alshammari recommends a key distribution protocol that is not only secure and useful but also feasible, simple and practical enough for employing on resource-constrained wireless sensor nodes. For analyzing the work, simulations and hardware implementations are being performed which is compared to the current solutions relying upon various metrics likestorage overhead, energy consumption, key connectivity, man-in-the-middle attack, replay attack and resiliency to node capture attack. The radio wave that is utilized for communications found along with defining the 128-bit AES-128 (advanced encryption standard) to verify and encrypt the data being transmitted [20]. Meigen Huang et.al, recommends the novel scheme of group key distribution for software-defined WSNs (wireless sensor networks)on the basis of PUFs (physical Unclonable Functions. It's quite arduous to clone and predict the PUF and greatly improvise physical security performance of sensor nodes. Moreover, concerning the centralized management of software-defined networking, the challenge of PUF lies in the sensor nodes for reducing communication overhead. Though the algorithm obtains group key delivery using two-way authentication functions via one communication interaction [21]. Rafael Dowsley et.al put forth a novel approach for a distributed cloud key management scheme. Concerning a public cloud application, before the data is stored anywhere else, it is encrypted using a different trusted adapter. The encryption key is not constantly stored at the adapter. Parts of the key is shared by various entities which are evaluated and stored temporarily at the adapter as required [22].

Vaira Prakash Guruswamy et.al presents a feasible technique of providing data security and privacy for the outsourced data within the cloud storage. It incorporates Cryptographic Tree as well as its key management for securing outsourced data. The focus is on the issue of ensuring security to the stored data that is being outsourced in the cloud and accessed and coordinated by multiple legitimate parties. Moreover, with the help of this scheme, multiple legitimate parties can outsource their data to the cloud by making no compromise on security [23]. Farah I. Kandah et.al proposes the approach of CSC (Centralized Stateful Connection) that enables efficient key management for dynamic sensor networks. The proposes schemes well balance security and efficiency that is obtained with the help of public key encryption at the beginning of the node's life in the wireless network which then shifts to symmetric encryption for the remaining communication. Because of the efficiency of Symmetric cryptography, it's highly suitable for sensor networks that need key management for key distribution, at the same time ensuring that same key is granted to both the communicating devices [24]. Adnan Shafiq et.al presents an approach which can be perceived in a way by obtaining pin codes on registered cell phones for carrying out financial transaction resulting in absolute out-of-band authentication. The research recommends an appropriate device using which the user can conduct out-of-band encryption of critical/sensitive data on any PC/Laptop that can execute the AES-256 algorithm via user-controlled keys. Sensitive data is being encrypted by the device in the Laptop/PC, thereby producing encrypted data back to the Laptop/PC. The data is decrypted using the same device. KMS (key management system) is built which takes care of the secure distribution of keys [25].

III. PROPOSED WORK

A. Overview

Security involves safeguarding critical and private information from unauthenticated users. For securing the data, it must remain (confidential) and hidden from unauthenticated access, must be restricted from alterations (integrity) and accessed by only authorized persons as and when required (availability). Key cryptography appears to be a remarkable discovery in the cryptography domain. Both confidentiality and authentication can be achieved by the means of Key cryptography. RSA is one such public-key cryptography. The research proposes the development of a revised and Secure Key Management System of public-key cryptography. By utilizing this method, the user encrypts the data by their own key and then uploads the data on the server. The SKMS(Secure Key Management System) incorporates Encryption (E) and Decryption (D) keys that rely on the key generation for making the system highly secure. cryptanalysis of SKMS utilizes high time in contrast to the traditional key management system.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

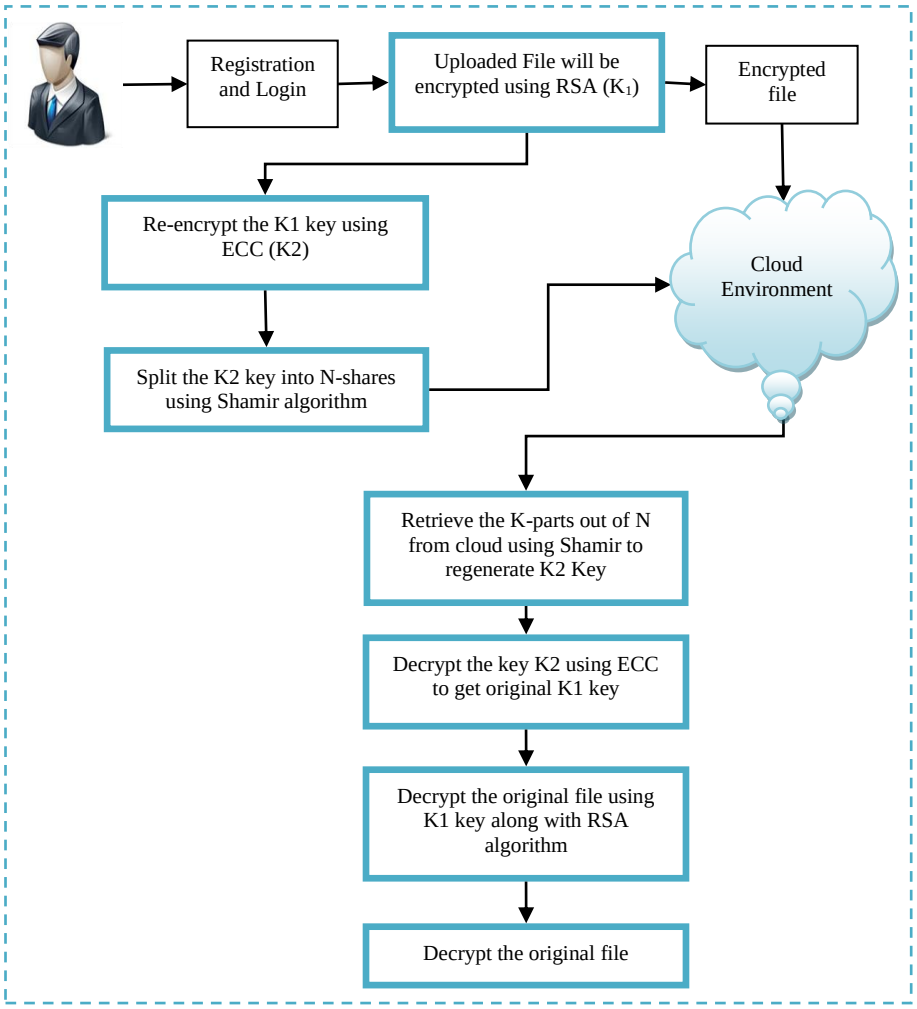

Fig 1: *Overall Architecture*

B. Key Management System

Key plays a vital role in overall security management whether it be the security of a home or for that matter data. Using the Cryptography key, the data can be made confidential from the rest of the users. Cryptography key utilizes Symmetric key and asymmetric key. In case the system has multiple users, key management systemgenerates a key for every user and distributes it to them. Key management involves various techniques such as key generation, distribution, storage, revokingand verifying. Key management is widely implemented across the cloud environment.

C. User operation

At first, users register using their credentials. All the files must be stored in the cloud storage by the users.That is files are uploaded in the cloud and key is delivered to the user for accessing the files.For accessing the files in the cloud, the user'srequests key generation to permit for the same. The key generator provides a key with which the user can download thefiles. This keyis then converted again to the original master key.

D. Key Generation

The key generation decides how many prime numbers to be utilized. It relies upon the value of 'N', which is formed by multiplying the prime numbers. E is also being computed indirectly. That is for computing E, the e1and e2 values are required. This results in more time to attack the system. It's just thenvalue which is kept as a public and private key. As a result, it becomes tough for the attackers to determine value of n since they can't find all the primes that are required to identify the value of N. complexity of the system rises because of the E parameter., The bit length of all the selected primes is of similar length for security reasons which are similarliketraditional cryptography algorithm.

E. Encryption and Decryption

The modern cryptographic systems incorporate publickey algorithms namelyRSA (Rivest–Shamir–Adleman). The file is encrypted by utilizing the public key (K_1) via RSA. The key is re-encrypted again via ECC to obtain K_2. The key K_2 is then divided into K and N part by the means of the Shamir algorithm. For decrypting the file key K_1 is required, hence K parts are pulled out of from N parts in the cloud such that (K<N). Next, by making use of ECC, key K_2 is decrypted to obtain the original key K_1.Now the original K_1 is decrypted via RSA.

F. Rivest–Shamir–Adleman (RSA)

Using the public key, encryption is performed and by utilizing the private key decryption is carried out. The encryption and decryption rely upon the value of n whereas for computing the keys N is utilized.

With such a technique the system becomes more secure and cannot be broken easily.

RSA Encryption and Decryption Algorithm

RSA encryption ( )

Input:
RSA Encrypt = Input File, (M < n)
Public Key parameters {E, n}
Output:
Encrypt key, K_1
Procedure:
Return encrypt key ( )
$K_1 \rightarrow M^E \mid n \mid$

RSA Decryption ( )

Input:
RSA Decrypt = Input key (K_2)
Private Key parameters (D, n)
Output:
Decrypted the key K_2
Procedure:
Return decrypt key ( )
$M \rightarrow K_2^D \mid n \mid$

G. Elliptic Curve Cryptography (ECC)

The equation of an elliptic curve is shown as, the work emphasizes basic knowledge and understanding of cryptography and terms such as encryption and decryption.

The equation of an EC (elliptic curve) is depicted as,

$y^2 = x^3 + ax + b$

Encryption

Consider the message being which is to be sent. This message must be represented on the key. It possesses intense implementation details. Entire research on ECC is carried out.

Consider 'K_1' has the point 'K_2' on the key 'E'. Randomly select 'K_1' from [1 – (n-1)].

Two cipher texts will be generated let it be C1 and C2.

$K_1 = k * P$ …….. (1)

$K_2 = M + k * Q$ …….. (2)

K_1 and K_2 will be Seeded.

Decryption

We have to get back the message being' that was sent to us,

$M = K_2 – d * K_1$ …….. (3)

M is the original message that we have sent.

Proof

$M = K_2 – d * K_1$

'M' can be represented as '$K_2 – d * K_1$'

Substitute eqn (2) and eqn (1) in eqn (3)

$K_2 – d * K_1 = (M + k * Q) – d * (k * P)$

$=> K_1 + k * d * P – d * k * P$ (canceling out k * d * P)

$=> K_1$ …… (4)

H. Shamir

Shamir's algorithm is implemented for securing the data (secret) in a distributed way, especially for securing encryption keys. This secret is divided into multiplesegments, referred to as shares. The shares then combine together to form the original secret. For unlocking the secret throughShamir's secret sharing, minimum no: of shares are required which is referred to as threshold. Using threshold minimum no: of shares can be determined for unlocking the secret.

Initialize split (n, K_1, s, K_2)

For int i← 0 to t-1do
Co-eff [i]←random number from 1 to K_1
End for
For x←1 to n do share ← s
For th←0 to t-1 do
Share ←(share + (coeff[th]*(x^{th} % K_1))% K_1
End for
Shares [x-1][0]=x, shares[x-1][1]=share
End for
Return shares
Merge (tshares [ ], K_1, K_2)
If length (tshares [ ]) < t then
Print merge is not possible
Else
For i←0 to length (tshares [ ]) do
xarray [i]← tshares [i][0]
xarray [i]← tshares [i][1]
End for
Merge (xarray[], yarray [ ] , K_2)
Print Key
End if

I. Mathematical model of KMS

RSA

$ax^2+bx+c=0$ *(Quadratic Equation)*

Assign a=2, b=3, c=4

$x=\frac{-b\pm\sqrt{b2-4ac}}{2a} => \frac{-3\pm\sqrt{9-4x2x4}}{2x2} => \frac{-3\pm\sqrt{3}}{4}$

$2(\frac{-3\pm\sqrt{3}}{4})^2+3(\frac{-3\pm\sqrt{3}}{4}) +4=0$

$\frac{-6\pm\sqrt{3}}{4}+\frac{-9\pm\sqrt{3}}{4} +4=0$

$\frac{-6\pm\sqrt{3}-9\pm\sqrt{3}+16}{4}=0$

$\frac{\pm\sqrt{3}(-6-9+16)}{4}=0$

$\frac{\pm\sqrt{3}(1)}{4}=0$

$\frac{1.732}{4}=0$

0.433

ECC

Cryptographic techniques play an important role in the Elliptic curve.An elliptic curveis improved security with reduced computational requirements.

$y^2 = x^3 + ax + b$ *(Elliptic curve)*

Assign a=-3, b=1, x=1, y=1

$1^2=1^3+ (-3) (1) +1$

1=1-3+1

=1-3+1-1

=-2

Shamir algorithm

Split

S=e3ha0490108963@@a19he844893031

Let N=14, K=8;

K-1 random numbers (a_1, a_2… a_n)

a_i*<P and* a_0*=S;*

Polynomial Function

$F(x) = a_0 + a_0x + a_2x^2 + \ldots + a_{k-1}x^{k-1}$

Then generate random numbers

a_1*=@6hee0he*

a_2*= h804ea39*

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

As a result, Polynomial Function

$F(x) =$ *e3ha0490108963@@a19he844893031+@6hee0hex+ h804ea39x²*

D_{x-1}*= (x, f(x)mod p))*

Where x=1, 2…N

In our case we need 5 points so we calculate then as follows:

D_1*= (1, f (1) mod p) = (1, 30e6aea3988he91@@6hee0heh804ea39)*

D_2*= (2, f (2) mod p) = (2, 30e6aea3988he91@@6hee0heh804ea39)* **.....**

D_{14}*= (14, f (14) mod p) = (14, h804ea39)*

Merge

The formula for the basis polynomial equation

$l_j(x) = \prod_{\substack{0\le m<k \\ m\neq j}} \frac{x-xm}{xj-xm} = \frac{(x-x0)}{(xj-x0)} \cdots \frac{(x-xj-1)}{(xj-xj-1)} \frac{(x-xj+1)}{(xj-xj+1)} \cdots \frac{(x-xk)}{(xj-xk)}$

(x_0, y_0)*= (1, 30e6aea3)*

(x_1, y_1)*= (2, 988he91@)*

(x_2, y_2)*= (3, @6hee0he)…*

(x_7, y_7)*= (7, h804ea39)*

Then the basic polynomial become

$l_0 = \frac{x-x1}{x0-x1}.\frac{x-x2}{x0-x2} = \frac{1}{3}$*(x-2) (x-4),* $l_1 = \frac{x-x0}{x1-x0}.\frac{x-x2}{x1-x2} = \frac{1}{2}$*(x-1) (x-4),* $l_2 = \frac{x-x0}{x2-x0}.\frac{x-x1}{x2-x1} = \frac{1}{6}$*(x-1) (x-2)……*

$l_7 = \frac{x-x0}{x7-x0}.\frac{x-x1}{x7-x1} = \frac{1}{8}$*(x-2) (x-4)*

Interpolated polynomial is derived from

$F(x) = \sum_{j=0}^{2} yj.lj(x)$

$F(x) = \frac{30e6aea3}{3}$*(x-2) (x-4)-* $\frac{988he91@}{2}$*(x-1) (x-4)* $+\frac{@6hee0he}{6}$*(x-1) (x-2) -* $\frac{@6hee0he}{8}$*(x-2) (x-4)*

$F(x) =$*@6hee0hex²+h804ea39 x+5481390490034x-988he91@ (mod p)*

$F(x) =$*@6hee0hex²+h804ea39 x+ e3ha0490108963@@a19he844893031 (mod p)*

This is secret key e3ha0490108963@@a19he844893031

J. File Decryption

Upon successful authentication, the user can access and download the concerned file or documents. As soon as the file is downloaded, it is decrypted automatically. This ensures that with the available key management system, uploadingand downloading of data can be performed in a secured manner. The research ascertains that with the implementation of mentioned techniques cyber-attacks and unauthorized users can be strictly prohibited. On receiving the matching filesfrom the key generatorwith respect to the concerned search query, the authorized user makes use of the private key for decrypting the file and finally get the required file.

IV. RESULT AND DISCUSSION

A. Experimental Result

The mentioned techniques prove to be immensely beneficial in Cloud computing key management system. These keys have played a vital role in regard to the security domain.The existing review elaborates privacy and security relying upon the cloud computing methods.Experiments are carried over the test system. The algorithm's performance is assessed taking into consideration the factors such as execution time andsecurity.

Table 1 presents the files which are utilized for performing the tests. Assessment of3separate algorithms have been carried out namely, ESRKGS (Enhanced and Secured RSA Key Generation Scheme), DNA (DNA proxy re-encryption), DPAK-RE (Dynamic Privacy Aggregate Key Re-Encryption) and SKMS (Secure Key Management System)

Table 1: *Comparative of Performance*

S.No	No of Techniques	Execution Time (ms)	Security	
			Encryption (bits)	Decryption (bits)
1	Enhanced and Secured RSA Key Generation Scheme (ESRKGS)	113.7	9000	9500
2	DNA proxy re-encryption (DNA)	112.56	5000	4500
3	Dynamic Privacy Aggregate Key Re-Encryption (DPAK-RE)	350	8000	8000
4	Secure Key Management System (SKMS)	95.12	9500	9500

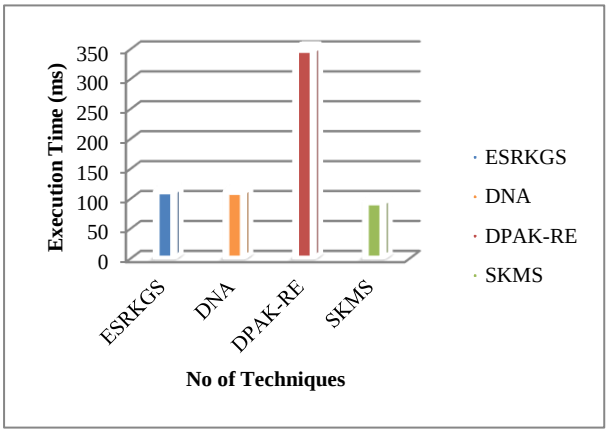

Fig 2: *Comparison of Execution time*

Figure 2 describes and compares the key management system based cryptography models performance with the approach of SKMS (Secure Key Management System) and ESRKGS, DNA, and DPAK-RE. The proposedSKMS is highly effective and delivers greatexecution timein contrast to the rest of the existing techniques.

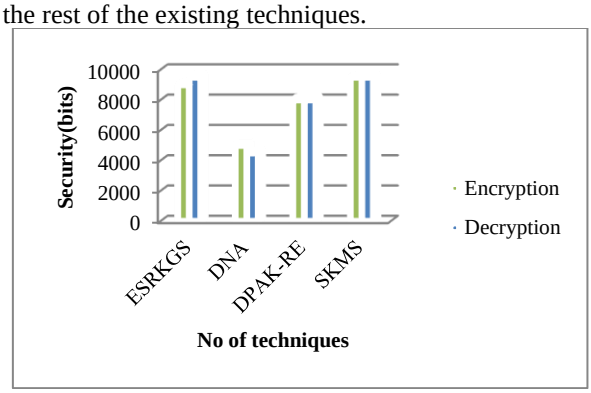

Fig 3: Comparison of Security

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Figure 3 describes and compares the key management system based cryptography models performance with the approach of SKMS (Secure Key Management System) and ESRKGS, DNA, and DPAK-RE. The proposed SKMS is delivering greater security in terms of key encryption and decryption in contrast to the rest of the existing techniques.

V. CONCLUSION

Safeguarding private and confidential information is of paramount significance and functionality of cloud storage. A novelapproach of the secure key management system is for accessing the file in the cloud storage system. A separate secret key is generated dynamically specifically for file encryption and decryptionand the aggregate key gets stored in the user system. As a result, this minimizes the cost, memory and access time of keys that have to be stored in the cloud. A novel SKMS (Secure Key Management System) is recommended which offers high security to the data being shared in the cloud. It's revealed from the experimental output and the safety proof that the recommended scheme is practical enough and has the potential of achieving a useful, secure data sharing in the distributed computing.

REFERENCES

1. Mathias Bjorkqvist, Christian Cachin, Felix Engelmann and Alessandro Sorniotti "Scalable Key Management for Distributed Cloud Storage", © IEEE, International Conference on Cloud Engineering, 2018, p.p.250-256.
2. Sarang Kahvazadeh, Xavi Masip-Bruin, Rodrigo Diaz, Eva Marín-Tordera, Alejandro Jurnet, Jordi Garcia "Towards An Efficient Key Management and Authentication Strategy for Combined Fog-to-Cloud Continuum Systems", © IEEE, Efficient key management and authentication strategy for the combined fog to cloud continuum systems, 2018.
3. Shiyu Luo, Zhichao Hua, and Yubin Xia "TZ-KMS: A Secure Key Management Service for Joint Cloud Computing with ARM Trust Zone", © IEEE, Symposium on service-oriented system engineering, 2018, p.p. 180-185.
4. Zakarya DRIAS, Ahmed SERHROUCHNI, Olivier Vogel "Identity-Based Cryptography (IBC) Based Key Management System (KMS) for Industrial Control Systems (ICS)", © cybersecurity in networking conference, 2017, p.p. 1-10.
5. GUOFENG LIN, HANSHU HONG, AND ZHIXIN SUN "A Collaborative Key Management Protocol in Ciphertext-Policy Attribute-Based Encryption for Cloud Data Sharing", © IEEE access, Digital Object Identifier, 2017, p.p. 9464-9475.
6. Sathishkumar Easwarmoorthy, Sophia F, Aravind Karrothu "An Efficient Key Management Infrastructure for Personal Health Records in Cloud", © IEEE, WiSPNET, 2016, p.p.1651-1657.
7. Lars Keuninckx, Miguel C. Soriano, Ingo Fischer, Claudio R. Mirasso, Romain M. Nguimdo & Guy Van der Sande "Encryption key distribution via chaos synchronization", scientific reports, 2017, p.p. 1-14.
8. Jialuo Han and Jidong Wang "An Enhanced Key Management Scheme for LoRaWAN", © MDPI, cryptography, 2018, p.p.1-12.
9. Hyunsoo Kwon, Changhee Hahn, Dongyoung Koo, and Junbeom Hur "Scalable and Reliable Key Management for Secure De-duplication in Cloud Storage", © IEEE, International Conference on Cloud Computing, 2017, p.p.391-398.
10. Kemal Bicakci, Davut Deniz Yavuzy, Sezin Gurkan "Twin Cloud: Secure Cloud Sharing Without Explicit Key Management", © IEEE, Security and Privacy in the Cloud, 2016.
11. Dinkar Sitaram, Sudheendra Harwalkar, Utkarsh Simha, Sreekanth Iyer, Shiv Jha Standards-based Integration of Advanced Key Management Capabilities with Open stack", © IEEE, International Conference on Cloud Computing in Emerging Markets, 2016, p.p. 98-103.
12. M. Thangavel, P. Varalakshmi, Mukund Murrali, K. Nithya "An Enhanced and Secured RSA Key Generation Scheme (ESRKGS)", © Elsevier, Journal of information security and applications, 2014, p.p. 1-8.
13. Muthi Reddy P, S. H. Manjula andVenugopal K. R. "Secured Privacy Data using Multi Key Encryption in Cloud Storage", © IEEE, Emerging Applications of Information Technology, 2018.
14. Yunpeng Zhang, Xin Liu, Manhui Sun "DNA based Random Key Generation and Management for OTP Encryption", © BioSystems, 2017, p.p. 1-12.
15. Pranshu Bajpai, Aditya K Sood and Richard Enbody "A Key-Management-Based Taxonomy for Ransomware", © IEEE, APWG Symposium on electronic crime research, 2018.
16. Ahmed Elhadad "Data sharing using proxy re-encryption based on DNA computing", © Springer, Soft Computing, 2019.
17. Venkata and Abuzneid "An Improved Novel Key Management Protocol for RFID Systems", © IEEE, Long island systems, applications, and technology conference, 2018.
18. Gu and Potkonjak "Efficient and Secure Group Key Management in IoT using Multistage Interconnected PUF", © IEEE, 2018.
19. Saad, Nouali, "A Trusted Way for Encryption Key Management in Cloud Computing", © Springer, International Conference on Advanced Information Technology, 2017.
20. Majid R. Alshammari and Khaled M. Elleithy "Efficient and Secure Key Distribution Protocol for Wireless Sensor Networks", © MDPI, Sensors, 2018, p.p.1-25.
21. Huang and Li "PUF-Assisted Group Key Distribution Scheme for Software-Defined Wireless Sensor Networks", IEEE COMMUNICATIONS LETTERS, VOL. 22, NO. 2, February 2018.
22. Antonia, "A Distributed Key Management Approach", © IEEE, 2016, p.p. 509 – 514.
23. Vairaprakash Gurusamy, S. Kannan, T. Maria Mahajan "Cryptographic Tree and Its Key Management for Securing Outsourced Data in the Cloud", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 7, Issue 5, May 2017, p.p. 255-259.
24. Farah I. Kandah, Oliver Nichols, Li Yang "Efficient Key Management for Big Data Gathering in Dynamic Sensor Networks", © IEEE, Workshop on Computing, Networking and Communications (CNC), 2017, p.p. 1-5.
25. Adnan Shafiq, Safiullah Khan and Hamza Zuberi "FPGA based out-of-band encryption module with key management system", © IEEE, Computing and digital systems, 2019, p.p. 270-275.

AUTHORS PROFILE

Pradeep K V, is an assistant professor, in SCSE, VIT Chennai Campus, Chennai. He has more than 10 years of teaching experience. His area of interest in research is Image processing, Cloud Computing, Security in Cloud, Parallel programming. Currently pursuing Ph.D. in cloud security under the guidance of Vijayakumar Varadarajan at VIT University, Chennai.

Vijayakumar Varadarajan is a Professor in SCSE at VIT University, Chennai, India. He has more than 18 years of experience including industrial and institutional. He also served as a Team Lead in industries like Satyam, Mahindra Satya, and Tech Mahindra. He has published many articles in national and international level journals/conferences/books. He is a reviewer in IEEE Transactions, Inderscience and Springer Journals.