A Proposed Client Based Key Management Framework to Secure Information in Cloud Storage for Public SaaS

Pradeep. K. V, Vijayakumar V

Abstract: ***Distributed computing is the procedure to get different things like servers, stockpiling, programming, and equipment over the internet whenever and anyplace. This component makes for various clients to get to them effectively as a utility with a few potential dangers, which requests the security systems. Subsequently, the requirement for the cryptographic calculation is unavoidable for the exchange of information in a progressively secure way, giving classification and respectability of the information to the clients. Client data put away at Cloud should be secured against potential gatecrashers just as a specialist organization. Because of the various potential assaults, there is a risk to the information at the cloud. The security of information concern is expanded for the majority of the organizations to ensure their touchy information at the cloud. Cryptography is an approach, which involves managing encryption and decryption keys. Customer should be guaranteed that their information is to be sheltered in the cloud. Each shopper needs to have authority over its information and its security. In this paper, we propose management of the key such that the data is secure at the cloud service provider and the key is recoverable in case of losing the key. So, the consumer is assured that even in the loss of the key, its data is recoverable. Here, we propose the Client based Key Management Framework to secure information in Cloud Storage utilizing Split Recoverable and symmetric AES Encryption Algorithm for Public SaaS. The trial examination demonstrates that the Client based KMS diminishes the time unpredictability of key age for both encryption and unscrambling subsequently it is effective when contrasted with existing frameworks***.

Index Terms**:** ***Cloud Computing, Cloud Security, Cryptography, Key Management.***

I. INTRODUCTION

Conveyed registering is the on-demand openness of PC structure resources, especially data storing and handling power, without a direct unique organization by the customer. The term is regularly used to portray server ranches open to various customers over the Internet. Tremendous fogs, predominant today, routinely have limits passed on over various regions from central servers. If the relationship with the customer is commonly close, it may be allowed an edge server Mists may be compelled to a lone affiliation be available to various affiliations (open cloud,) or a mix of both. The greatest open cloud is Microsoft Azure, google cloud, Amazon Web Services. Dispersed registering relies upon the sharing of advantages for achieving insight and economies of scale. Supporters of open and blend fogs note that disseminated registering empowers associations to evade or restrict ahead of time IT system costs. Backers moreover ensure that disseminated processing empowers attempts to get their applications moving faster, with improved reasonableness and less upkeep, and that it engages IT gatherings to even more rapidly alter advantages for satisfying a fluctuating and whimsical need. Cloud providers ordinarily use a "pay-as-you-go" model, which can incite abrupt working expenses if administrators are not familiar with cloud-assessing models.

II. RELATED WORKS

As per NIST [3], Cloud as a model and it includes the accumulation of systems, equipment, programming, servers, and capacity are overseen by suppliers and made them accessible as a support of the end shoppers on interest, which can be provisioned quickly and discharged with insignificant administration exertion. Distributed computing can be utilized to decrease the limitations of conventional figuring, profiting the space, time, power, cost and streamlined business procedures to an association. Cloud storage [4] is an organized online capacity. The distributed storage suppliers store the client's information on virtualized pools. By and large, the distributed storage suppliers introduce information huge server farms where the information is put away. The clients can purchase or rent the capacity limit dependent on their need. In [5], the Storage has given as an administration, which gives amassing that the client uses, including information transmission necessities for the limit. This urges cloud application to scale past their obliged servers. This Service empowers the customers to store their data at remote accumulating circles and access it at whatever point to meet a couple of necessities for keeping up customer's data and information including high availability, steady quality, execution, replication, and data consistency. In the paper [6] the makers familiarize a PKI underwriting plot with secure cloud condition application program of the encryption key, symmetric keys are secured by the open key of the validation. Kumar A and group [7] presents a plan that ties the key with records for individual stockpiling. The plan is anything but difficult to oversee and keep up key update, nonetheless, the program requires the client's login secret phrase consistently refreshed, or if the secret word spills, record security can't be ensured. Kumar scrambles the information put away in the cloud with elliptic bend encryption [8].

Revised Manuscript Received on 30 May 2019.
***** Correspondence Author

Pradeep. K.V, SCSE, VIT-Chennai Campus, Chennai, TN, India.
Vijayakumar .V, SCSE, VIT-Chennai Campus, Chennai, TN, India

Friedrich C [9] advances the plan that isolates the information into serval segments, at that point arbitrarily creates distinctive keys for each bit of information and utilizations these keys to encode information.

In [10] Vimercati S D C D and co., present the over-encryption the executives to ensure redistributed assets. The proposed plan utilizes a two-layer model. Assets proprietor scrambles the assets first before transferring the information to the server. What's more, the server will encode the information the second time before clients seeing the assets.

Amar R. Buchade, Rajesh Ingle [11], tended to the issues like Key Management techniques for different cloud information stockpiling, and furthermore thought about different symmetric Key calculations like AES, TripleDES, Blowfish, and RC4

III. PROPOSED WORK

The Public SaaS administration model keeps running on an open cloud. Such a precedent Gmail. This sort of model is for both business and individual use. The entertainers associated with this model are End Users(Data Owners), Cloud Service Provider (Administration, running, accessibility and security of the Cloud Infrastures), Third parties(supporting usefulness, for example, encryption or key administration for the benefit of customer or supplier). In this model, both equipment and programming will be under the control of the supplier. The information, then again, possessed by the shopper however regulated, put away and handled by the cloud specialist organization which is a reasonable reliance of cloud purchaser on the cloud supplier. On what premise the cloud buyer needs to believe cloud specialist co-op with the goal that the shopper information is secured. Ordinarily, Keys used to encode the information is overseen by the supplier. In any case, here we are proposing the Client based Key Management Framework to deal with the keys, which are utilized to scramble the information before transferring to the cloud. The underneath fig-1 delineates the proposed model design.

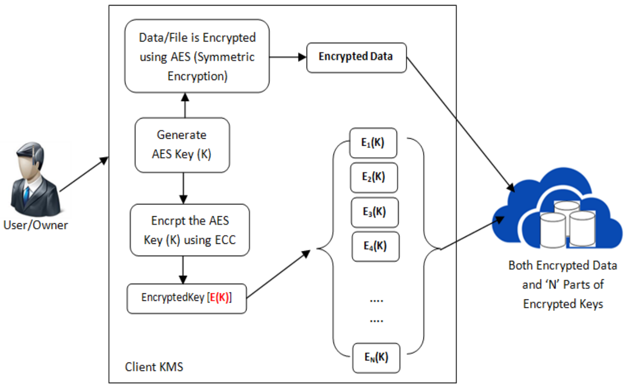
User/Owner
Data/File is Encrypted using AES (Symmetric Encryption)
Encrypted Data
Generate AES Key (K)
Encrpt the AES Key (K) using ECC
EncryptedKey [E(K)]
E₁(K)
E₂(K)
E₃(K)
E₄(K)
....
....
E_N(K)
Client KMS
Both Encrypted Data and 'N' Parts of Encrypted Keys

Fig-1: Client Base Key Management Framework

At first, Before transferring the client information onto a cloud, Key will be created utilizing Standard AES Symmetric calculation and same is accustomed to scrambling the client information before transferring onto the cloud. Later the key is re-scrambled utilizing Elliptic Curve Cryptography and split into 'N' parts utilizing Shamir's mystery key sharing. Presently these 'N' parts will be transferred into either single or multi-cloud. At whatever point the client needs to decode his information, he will pull down just 'K' shares from the cloud and use them to recover the encoded key which in turns unscrambled it utilizing ECC to get the first key. Presently this key can be utilized to unscramble the information utilizing AES.

1. AES Algorithm

In figure-1, Normally the client will encode his information utilizing AES Encryption calculation. The Advanced Encryption Standard (AES) [26] was distributed by the National Institute of Standards and Technology (NIST) in 2001. AES is a symmetric square figure where a solitary key is utilized for both encryption and unscrambling process. The information and yield for the AES calculation each comprise of successions of 128 bits. The key utilized in this calculation comprises of 128, 192, or 256 bits. AES works on 8-bit bytes. These bytes are deciphered as the components of limited field utilizing the accompanying polynomial portrayal:

$$F(x) = b_{n-1}x^{n-1} + b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \ldots.. + b_1x^1 + b_0$$

Where each b_i is having value 0 or 1. The 128-bit input block of AES is arranged in a state matrix of size 4 × 4 as shown in Fig-2. The elements of the matrix are represented by the variable $\mathbf{b_{ij}}$ where $\mathbf{0 \le i,j \le 3}$ and i,j denotes the row and column number, respectively.

b_{00}	b_{01}	b_{02}	b_{03}
b_{10}	b_{11}	b_{12}	b_{13}
b_{20}	b_{21}	b_{22}	b_{23}
b_{30}	b_{31}	b_{32}	b_{33}

Fig-2: State Matrix for Key Generation and Expansion

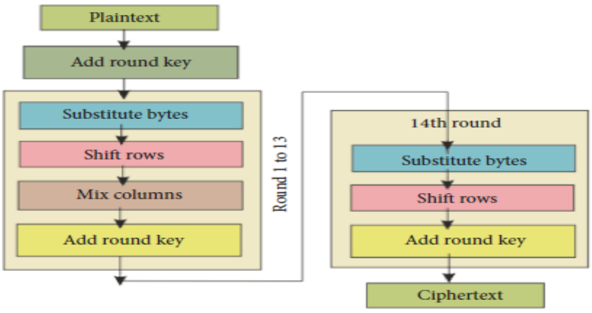

Fig-3: Round function steps in 14-round AES

AES Parameters	AES-128	AES-256	AES-512
Key Size(Bits)	128	256	512
Number of Rounds	10	12	14
Plain Text box Size	128	128	128

Tab-1: AES Number of Round Function Steps

The rounds are taken into consideration for AES based on the key size. For our experimentation, we used a 256 sized key and in this manner, the number of rounds utilized is 14 rounds and spoken to as N_r. The key booking calculation is likewise utilized in this standard is to give keys to every one of the rounds.

The structure of the key booking calculation is with the end goal that the noteworthy any round key reasons the first info key from which the round keys are determined. The information state network is prepared by the different round changes. The state framework advances as it goes through the different strides of figure lastly create the encrypted text. Each round in AES pursues the accompanying advances.

Byte Substitution: The input 16 bytes are substituted by investigating a fixed S-box table and the result is in a grid of four lines and four sections.

Shiftrows: Every one of the four segments of the matrix is moved to the other side. Any entries that 'tumble off' are re-inserted on the right half of the line. No change in the first line i.e., neither shirted nor moved. One byte position is moved in the second line. In the third line, two positions are moved to the other side. Finally, The fourth line is moved three positions to the other side. The outcome is of 16 bytes, which is in other grid, however, moved as for one another.

MixColumns: Each portion of four bytes is right now changed using an unprecedented logical limit. This limit takes as data the four bytes of one fragment and yields four absolutely new bytes, which supersede the principal section. The result is another new structure containing 16 new bytes. It should be seen that this movement isn't performed in the last round.

Addroundkey: The grid or lattice of 16 bytes are legitimately considered as 128 bits and are XORed to the 128 bits of the round key. On the off chance that this is the last round, by then the yield is the ciphertext. Something else, the subsequent 128 bits are deciphered as 16 bytes and we start another close round.

Decryption Process

The Reverse procedure of encryption is the unraveling of an AES ciphertext. Four methodologies are contained by each round involves the four strategies coordinated in the pivot demand – i.e Adding round key, Mixing portions, Shifting sections, and substitution of Bytes. Since sub-frames in each round are in a switch way, not in the slightest degree like for a Feistel Cipher, the encryption and unscrambling counts ought to be autonomously executed, despite the way that they are in all regards solidly related. In the present day, AES is ordinarily gotten and kept up in both equipment and programming. Till date, no helpful study on ciphertext and attacks against this standard has been found. Also, AES has worked in the adaptability of key length, which permits a component of 'future-fixing' against progress in the capacity to perform comprehensive key missions

2. ECC Algorithm

The Data is scrambled and put away in the cloud, Now how to deal with the key, where to keep the key, so what we do here is the key 'K' utilized alongside AES used to encode the information, will be re-encoded utilizing Elliptic Curve Cryptography (ECC). ECC is an uneven cryptography calculation which includes some abnormal state computation utilizing numerical bends to encode and unscramble information. It is like RSA as it's deviated however it utilizes an extremely little length key when contrasted with RSA. Elliptic bend cryptography (ECC) can give a similar dimension and sort of security as RSA however with a lot shorter keys.

Symmetric Encryption Key Size in bits	*RSA and Diffie-Hellman "Key" size in bits*	*ECC "Key" Size in bits*
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

Tab-2: Comparision of Key Sizes with different Methods

Envision we have a lot of focuses (x_i, y_i) in a plane. The set is, huge yet limited. We will indicate this set by E. Next, envision we can characterize a gathering administrator (+) on this set, notwithstanding when the task itself has nothing at all to do with normal number juggling expansion. So given two points P and Q in the set E, the gathering administrator will enable us to compute a third point R, additionally in the set E, with the end goal that P + Q = R. Given a point X ∈E, we will especially be keen on utilizing the gathering administrator to discover X+X, X+X+X, X+X+X+. .+X for a discretionary number of rehashed summons of the gathering administrator. Given a conventional number k, we will utilize the documentation k×P to speak to the rehashed expansion X+X+… .+X in which X shows up, with the administrator '+' being conjured k-1 time.

Presently envision that the set E is mystical as in, after we have determined k × X for a given point X ∈E, it is amazingly hard to recuperate k from k × X. We will expect that the best way to recuperate k from k × X is to attempt each conceivable rehashed summation like X + X, X + X + X, X + X + X + . . . + X until the outcome rises to what we have for k × X.

On the off chance that we could guarantee the above condition, at that point "items" like k×X for X ∈E could be utilized by two gatherings in a Diffie-Hellman like a convention for sharing a mystery session key. The majority of the suppositions we have made above are fulfilled when the set E of focuses (xi, yi) is drawn from an elliptic bend. Now, a shrewd peruser would solicit: If the security of ECC relies upon discovering how often a point X takes an interest in a total like X + X + . . . + X, for what reason would it take an aggressor any more work to make sense of that than it would take for a gathering to ascertain the whole? No doubt all that the aggressor would need to do is continue adding G to itself until the assailant sees the estimation of the aggregate. That is, if some number X_A is your private key, and in the event that you infer your open key by adding the direct G toward itself X_A times, the measure of computational exertion you consume in adding X to itself X_A times ought to be equivalent to what the assailant would need to use on the off chance that he continued adding X to itself until achieving an esteem that is your public key. Elliptic bends are constantly cubic. An elliptic bend in its "standard structure" is portrayed by $\mathbf{y^2 = x^3 + ax + b}$. For some fixed characteristics for the parameters an and b.

This condition is moreover suggested as Weierstrass Equation of trademark 0. Elliptic curves have a rich arithmetical structure that can be put to use for cryptography
The Set of endorsed figurings has been grasped for ECC by NIST in its Suite-B, unequivocally in ECDH and ECDSA, i.e., Elliptic Curve Diffie-Hellman and Digital Signature Algorithms for modernized marks.

3. Shamir's Secret Sharing

A (k,n) Secret Sharing is a computation in cryptography made by Adi Shamir. It is a sort of riddle sharing, where a puzzle is isolated into 'n' parts, giving each part its own one of a kind unique part. To reproduce the primary secret, a base number of parts 'k' is required. In the breaking point scheme, this number isn't actually without a doubt the number of parts. Something different, all individuals are relied upon to duplicate the principal riddle.

The objective is to isolate mystery into 'n' bits of information S_1, S_2, S_3, … Sn so that:

- Any 'K' shares of Information or more than it but less than 'N' makes 'S' easy to enroll.
- Any 'k-1' shares or fewer pieces leaves 'S' absolutely uncertain, i.e., the puzzle 'S' can't be changed with not as much as 'k' pieces.

This plan is called {k,n} limit plot. On the off chance that {k=n}, at that point each bit of the first mystery 'S' is required to recreate the mystery or secret.

The fundamental thought of the course of action is that 2 are attractive to depict a line, 3 are adequate to portray a parabola, 4 to portray a cubic curve, etc. That is, it takes 'k' focuses to depict a polynomial of degree 'k-l'.

Expect we have to use (k,n) limit intend to share our riddle (S, gauge), without loss of accord, thought to be a segment in a restricted field 'F' of size 'P' where 0<k<=n<P; S<P and 'P' is a prime number.

Pick unpredictably 'k-1' positive numbers $a_1, a_2, \ldots \ldots a_{k-1}$ with $a_0 = S$ and $a_i < P$. Gather the polynomial $f(x)=a_0 + a_1x + a_2x^2 + a_3x^3 + \ldots . a_{k-1}x^{k-1}$. Give a chance to develop any 'n' calls attention to of it, for example, set i = 1,2,3,… n to recover {i,f(i)}. Every part is given a point (a non-zero entire number commitment to the polynomial, and the relating entire number yield) close by the prime which portrays the constrained field to use. Given any subset of 'k' of these sets, we can find the coefficients of the polynomial using inclusion. a_0 the secret or mystery is easily predictable.

IV. EXPERIMENTATION AND RESULTS

To test the proposed structure, Amazon S3 can which is a dispersed distributed storage and NetBeans device are used to complete the proposed work. The code is made in java. The AES and ECC encryption are executed with the help of Bounty Castle a Java pack. Wealth Castle is a social occasion of cryptographic API executes the distinctive cryptographic encryption. A key is picked and scramble the report containing the data. We get encoded data in the scratch cushion. Again the Key is re-scrambled utilizing Elliptic bend cryptography, which thus split into 'n' shares utilizing Shamir's Mystery sharing. Presently Both Encrypted Data and 'N' offers of Encrypted keys are set in amazon s3 pail. At later stages for unraveling if the client lost his keys, at that point the putaway document and the encoded key is recuperated by recovering and consolidating 'k' shares from amazon s3 basin and decoded by utilizing ECC. When the first key is recovered by, at that point this key is utilized to encode the archive. The work displayed result diagram of our proposed framework, usage of the AES calculation by taking content, picture, and sound as information. The work utilized 14 rounds method for actualizing AES 256-piece calculation The chart demonstrates the time expected to scramble the contribution as a content, picture and sound information document by the proposed framework

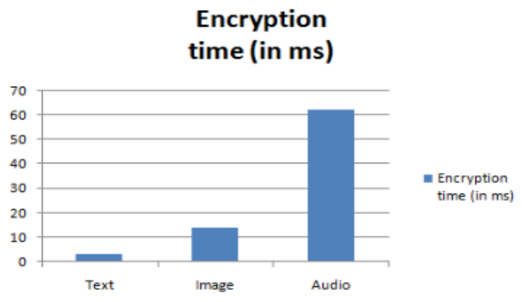

Fig-4: AES Encryption Time

The chart demonstrates the time expected to unscramble the contribution as a content, picture and sound information document by the proposed framework

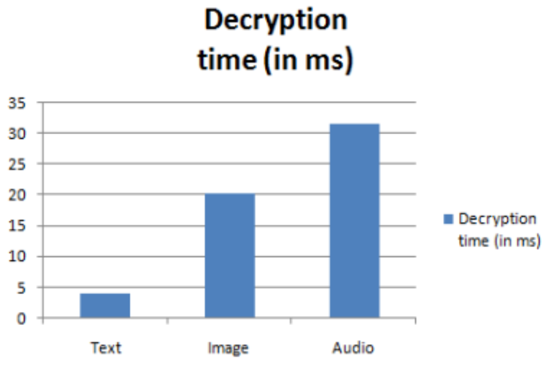

Fig5: AES DecryptionTime

V. COMPLEXITY ANALYSIS

With the key of 'N' bits length, Both RSA and ECC has sufficient energy multifaceted nature of $O(N^3)$. In any case, regardless of this equality in the reliance of the computational exertion on the key size, it takes far less computational overhead to use ECC as a result of the manner in which that you can pull off significantly shorter keys. Because of the parcel more diminutive key sizes included, ECC computations can be executed on smartcards without numerical co-processors. Contactless shrewd cards work just with ECC in light of the fact that different frameworks require a lot of acceptance vitality. ECC is likewise utilized in the calculations for Digital Rights Management (DRM). We are using Shamir' Secret algorithm to split and distribute the encrypted keys in one or more cloud environments and regeneration is easy just to combine only 'k' parts out of 'n' from the cloud.

VI. CONCLUSION

The proposed system is useful for SaaS consumers the individuals who don't generally trust and really rely upon the providers. Here were are utilizing three distinct calculations like AES, ECC and Shamir's Secret sharing for encryption of information, key and furthermore separating the key before transferring all onto a cloud. Indeed, even at one point if the key is lost, it very well may be effectively recovered by the buyer by recovering just 'k' parts out of 'n' parts. It gives a better outcome in the new time, particularly for the sort of dispersed situations. This calculation safeguards the essential three properties of cryptography to be specific classification, honesty, and validation.

ACKNOWLEDGMENT

I might want to thank the commentators and writers, who are all together, perused numerous drafts and changed over their considerations into numerous different drafts and which helped me a ton to think more and utilized their drafts to get ready and propose the above structure.

REFERENCES

1. Sims K. IBM Introduces ready-to-use cloud computing collaboration services get clients started with cloud computing. 2007.
2. Heiser J, Nicolett M. Assessing the security risks of cloud computing. http://www.gartner.om/displaydocument?id=685308, 2008
3. National Institute of Standards and Technology(NIST)
4. http://csrc.nist.gov/publications/drafts/800-146/Draft -NIS T -SP800-146.pdf
5. Lin Weiwei, Liang Chen, and Liu Bo, "A Hadoop-based efficient economic cloud storage system" in Inter. Conf on Circuits, Communication, and System, pp. 1-4, 2011.
6. Jiyi WU, Xiaoping GE, Ya Wang, " Cloud storage as the infrastructure of cloud computing", in Inter. Conf on Intelligent Computing and Cognitive Informatics, pp. 380-383, 2010.
7. Jindi G, Zishan D, Lei S. Research on Key Management Infrastructure in the Cloud Computing Environment.[J]. International Conference on Grid & Cooperative Computing, 2010:404 - 407.
8. Kumar A Lee, BG Lee H Jet Secure storage and access of data in cloud computing [C].Proceedings of 2012 International Conference on ICT Convergence (IC-TC).New York: IEEE, 2012: 336, 339.
9. Nepal S, Friedrich C, Henry L, et al, A secure storage service in the hybrid cloud[C].Proceedings of 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC). New York: IEEE, 2011:334, 335.
10. Chen F H, Liu Y, Qi Y U. Key Management Scheme of Personal Cloud Computing [J]. Modern Computer, 2011.
11. Vimercati S D C D, Forest S, Jajodia S, et al. Overencryption: management of access control evolution on outsourced data[C] Very Large Data Bases2007:123-134.
12. Amar R. Buchade, Rajesh Ingle, Key Management for Cloud Data Storage: Methods and Comparisons, 978-1-4799-4910-6/14 $31.00 © 2014 IEEE
13. Vinayak Bajirao Patil, Prof.Dr.Uttam.L.Bombale, Pallavi Hemant Dixit, “Implementation of AES algorithm on ARM processor for a wireless network, ” in International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, August 2013,pp.3204-3209.2.
14. Xinmiao Zhang and Keshab K. Parhi, “Implementation approach for the advanced encryption standard algorithm,” in IEEE Transactions, 2002.
15. Chih-Pin Su, Tsung-Fu Lin, Chih-Tsun Huang, and Cheng-Wen Wu, “A high throughput low-cost AES processor,” in IEEE Communications Magazine, 2003.
16. “Advanced Encryption Standard (AES)” Federal Information Processing Standards Publication 197, Nov. 2001.5.
17. M.Gnanambika, S.Adilakshmi, Dr.Fazal Noorbasha, “AES-128 Bit Algorithm Using Fully Pipelined Architecture for Secret Communication,” in International Journal of Engineering Research and Applications Vol. 3, Issue 2, March -April 2013, pp.166-169.
18. Rishabh Jain, Rahul Jejurka2, Shrikrishna Chopade, Someshwar Vaidya, Mahesh Sanap, “AES Algorithm Using 512 Bit Key Implementation for Secure Communication,” in International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 3, March 2014,pp.3516-3522.
19. Vinayak Bajirao Patil, Prof.Dr.Uttam.L.Bombale, Pallavi Hemant Dixit, “Implementation of AES algorithm on ARM processor fora wireless network, ” in International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, August 2013,pp.3204-3209
20. Ritu Pahal and Vikas Kumar, “Efficient Implementation of AES,” in International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, Issue 7, July 2013,pp.290-295.
21. K. Soumya, G. Shyam Kishore, “Design and Implementation of Rijndael Encryption Algorithm Based on FPGA,” in International Journal of Computer Science and Mobile Computing, Vol. 2, Issue. 9, September 2013, pp.120 – 127.
22. Sumedha Kaushik and Ankur Singhal, “Network Security Using Cryptographic Techniques,” in International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2, Issue 12, December 2012, pp.105-107.
23. Dr. Prerna Mahajan & Abhishek Sachdeva, “A Study of Encryption Algorithms AES, DES and RSA forSecurity,” in Global Journal of Computer Science and TechnologyNetwork, Web & Security, Vol 13, Issue 15,2013,pp.15-22.
24. H. Kuo and I. Verbauwhede, “Architectural optimization for a 1.82 Gbits/sec VLSI implementation of the AES Rijndael algorithm,” in Proc. CHES 2001, Paris, France, May 2001, pp. 51-64.
25. Navraj Khatri, Rajeev Dhanda, Jagtar Singh,” Comparison of power consumption and strict avalanche criteria at encryption/Decryption side of Different AES standards,’’ International Journal Of Computational Engineering Research, Vol. 2 Issue. 4, August 2012.
26. Das Debasis, Misra Rajiv.“Programmable Cellular Automata Based Efficient Parallel AES Encryption Algorithm”. International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.6, November 2011, pp. 204.15.Hiremath.S. and Suma.M.S., “Advanced Encryption Standard Implemented on FPGA,” in IEEE Inter.Conf.Comp Elec Engin. (IECEE),vol.02,issue.28,pp.656-660,Dec.2009.

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 currently 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 for several years. He has published many articles in national and international level journals/conferences/books. He is a reviewer in IEEE Transactions, Inderscience and Springer Journals.