Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 9, September 2014, pg.694 – 702

SURVEY ARTICLE

A Survey on a Novel Method to Secure Cloud Computing Through Multicast Key Management

Anand Chafle[1], Ankit Zanwar[2], Puja Deulkar[3]
Department of C.E., B.D.C.E., Sewagram, Wardha
[1] anandchafle@gmail.com
[2] ankitz619@gmail.com
[3] pujadeulkar@gmail.com

ABSTRACT- Cloud computing is the technology that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. This technology allows for much more efficient computing by centralizing storage, memory, processing and bandwidth. In this security is an important issue to provide a security for this cloud we introduce a novel method for securing cloud by providing multicast key for each user. It will be a dynamic session key which will vary in the time of period. Whenever a new user enters into the cloud the new key will be generated .It will withstand for a time period. After that time period the user should renew the key for the further usage of the cloud.

Key words- Cloud computing, Multicast key management, Session key

I. INTRODUCTION

Cloud computing presents an opportunity for pervasive systems to leverage computational and storage resources to accomplish tasks that would not normally be possible on such resource-constrained devices. Cloud computing can enable hardware designers to build lighter systems that last longer and are more mobile. Despite the advantages cloud computing offers to the designers of pervasive systems, there are some limitations of leveraging cloud computing that must be addressed.

Cloud computing, or in simpler shorthand just "the cloud", also focuses on maximizing the effectiveness of the shared resources. Cloud resources are usually not only shared by multiple users but are also dynamically reallocated per demand. This can work for allocating resources to users. For example, a cloud computer facility that serves European users during European business hours with a specific application (e.g., email) may reallocate the same resources to serve North American users during North America's business hours with a different application (e.g., a web server). This approach should maximize the use of computing power thus reducing environmental damage as well since less power, air conditioning, etc. are required for a variety of functions. With cloud computing, multiple users can access a single server to retrieve and update their data without purchasing licenses for different applications.

In this security is an important issue to provide a security for this cloud we introduce a novel method for securing cloud by providing multicast key for each user. It will be a dynamic session key which will vary in the time of period. Whenever a new

user enters into the cloud the new key will be generated .It will withstand for a time period .After that time period the user should renew the key for the further usage of the cloud.

II. LITERATURE SURVEY

Paper [1] A Novel Method to Secure Cloud Computing Through Multicast Key Management International Conference On Information Communication And Embedded Systems Year 2013

In this paper an author proposed that whenever a new user enters into the cloud the new key will be generated .It will withstand for a time period .After that time period the user should renew the key for the further usage of the cloud.

Paper [2] Generation of Shorter Length Keys for Broadcast and Multicast Services Using 2-way Hash Chain Schemes International Journal of Inventive Engineering and Sciences (IJIES) ISSN: 2319–9598, Volume-1, Issue-10, September 2013

This paper proposed our project is focused on new Key Management Scheme, called 2-way Hash Chains Scheme (2HCS), that focus on the reduction of transmission overhead caused above and thus, effectively reduces the number and the size of keying messages.

Paper [3] Hierarchical Identity-based Key Management in Cloud Computing Journal of Convergence Information Technology(JCIT) Volume 7, Number 20, Nov 2012

This paper present a hierarchical identity-based signcryption key management scheme in cloud computing. Their solution adopts identity-based signcryption technology. Identity-based signcryption not only provides privacy protection and unforgeability but also is more efficient manner than a composition of an encryption scheme with a signature scheme. The identity of entities which performs as public key, can simplifies key management in cloud computing. By our hierarchical solution, the scalability in cloud computing also is solved.

Paper [4] Efficient Key Management Scheme for Secure Multicast in MANET IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.11, November 2010

In this paper, an efficient way of multicasting a secure data to a group using a hybrid key management scheme is discussed and from the results it is observed that the storage complexity, communication complexity and computation complexity are very much comparable with the existing method.

Paper [5] Publicly Verifiable Secret Sharing for Cloud-Based Key Management D.J. Bernstein and S. Chatterjee (Eds.): INDOCRYPT 2011, LNCS 7107, pp. 290–309, 2011. Springer-Verlag Berlin Heidelberg 2011

This paper takes such a holistic approach for the case of public-key encryption which is one of the most basic cryptographic tasks. The approach boils down to formalizing the security of public-key encryption in the presence of PVSS. We present such formalization and observe that the PVSS scheme of Stadler can be shown to satisfy our definition, albeit in the Random Oracle Model.

III. PROPOSED WORK

The proposed work is planned to be carried out in the following manner

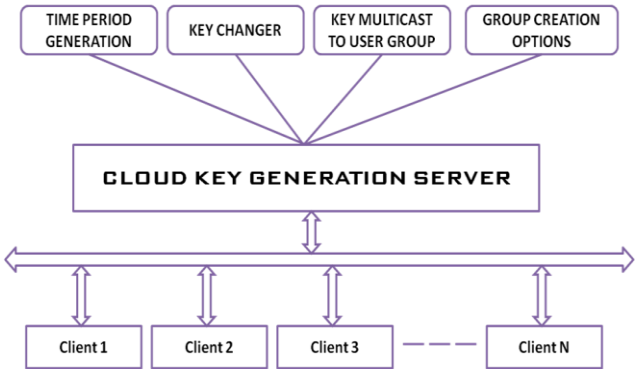


Fig.1: Basic system architecture

A. Keys

The active members of the group receive security featured associations that include encryption keys, authentication/integrity keys, cryptographic policy that describes the keys, and attributes such as an index for referencing the security association (SA) particular objects contained in the SA*GCKSrole.*

In addition to the policy associated with group keys, the group owner or the Group Controller and Key Server (GCKS) may define and enforce group membership, key management, data security, and other policies that may or may not be communicated to the entire membership.

B. Periodic refresh of keys

The determined survival of the keys are periodically refreshed

C. Maintenance protocol during addition and removal of group members

The protocol should facilitate addition and removal of group members. Members who are added may optionally be denied access to the key material used before they joined the group, and removed members should lose access to the key material following their departure.

D. The protocol should support a scalable group rekey operation without unicast exchanges between members and a Group Controller and Key Server (GCKS), to avoid overwhelming a GCKS managing a large group.

E. The key management protocol should offer a framework for replacing or renewing transforms, authorization infrastructure, and authentication systems.

G. *Applying Multicast Key Management for Cloud Computing.*

With this set methodology of secure multicast key management on cloud is been protected .The cloud users are grouped according to their interests for (e.g. business, news, entertainment etc.). For each group a different set of keys been provided for each users .Each group is been structurised in the form of tree [1]

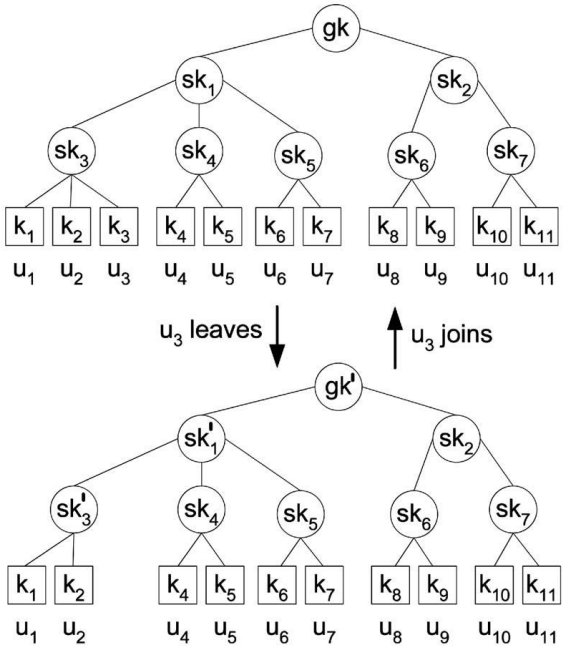


Fig 2: Group Multicasting

Whenever a user enters the group or exist the group the group key will be changed in periodical manner. In this two group operation is been performed one is JOIN and other one is LEAVE.

JOIN:

Whenever a new member joins the group the centralized server or cloud server gives the UID to the new member and calculates the new group key. It is first sent to the new member by unicast. It is then encrypted by the old group key and sent to all the remaining members by one multicast. This can be further enhanced by considering the following three scenarios a) Number of leave request equal to join request

b) Number of leave request is less than join request

c) Number of leave request is greater than join request

LEAVE:

Whenever a present member leaves the group, the group member should send a leaving request to the centralized server or cloud server accepting request the cloud server will relieve the outgoing member and change the key at once to be a security measure by that out went member will not know about new transaction in the cloud. By that the security is established in the cloud. In each cloud the tree structure is maintained and whenever the new member enters or the old member relieve the cloud the new set of multicast key will be generated through centralized cloud and it will be disseminated securely among the group members. By that key the clients can communicate with the centralized cloud for information transaction

IV. AES ALOGRITHM

AES is based on a design principle known as a Substitution permutation network. It is quite fast in case of software as well as hardware. Unlike its predecessor, DES, AES does not use a Feistel network.AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits.

The block size has a maximum of 256 bits, but the key size has no theoretical maximum.AES operates on a 4×4 column-major order matrix of bytes. In AES most of the time calculations are carried out in a special finite field.

Working Of AES:

Advanced Encryption Standard or AES was invented by Joan Daemen and Vincent Rijmen, and accepted by the US federal government in 2001 for top secret approved encryption algorithms. It is also referred to as Rijndael, as it is based off the Rijndael algorithm. Reportedly, this standard has never been cracked.

AES has three approved key length: 128 bits, 192 bits, and 256 bits. To try to explain the process in simple terms, an algorithm starts with a random number, in which the key and data encrypted with it are scrambled though four rounds of mathematical processes. The key that is used to encrypt the number must also be used to decrypt it.

The four rounds are called SubBytes, ShiftRows, MixColumns, and AddRoundKey. During SubBytes, a lookup table is used to determine what each byte is replaced with. The ShiftRows step has a certain number of rows where each row of the state is shifted cyclically by a particular offset, while leaving the first row unchanged. Each byte of the second row is shifted to the left, by an offset of one, each byte in the third row by an offset of two, and the fourth row by an offset of three. This shifting is applied to all three key lengths, though there is a variance for the 256-bit block where the first row is unchanged, the second row offset by one, the third by three, and the fourth by four. The four bytes are taken as input and generated as output.

In the fourth round, the AddRoundKey derives round keys from Rijndael's key schedule, and adds the round key to each byte of the state. Each round key gets added by combining each byte of the state with the corresponding byte from the round key. Lastly, these steps are repeated again for a fifth round, but do not include the MixColumns step. These algorithms essentially take basic data and change it into a code known as cipher text. The larger the key, the greater number of potential patterns that can be created. This makes it extremely difficult to descramble the contents, which is why AES has been Teflon-coated.

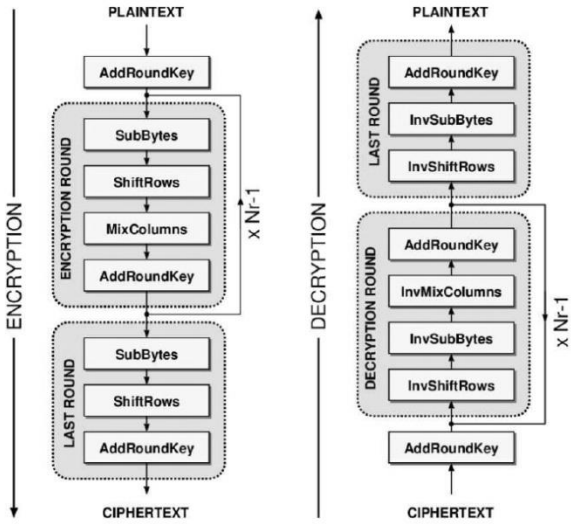


Fig 3: Working of AES algorithm

AES ALGORITHM:

1. **KeyExpansion**—round keys are derived from the cipher key using Rijndael's key schedule

2. Initial Round

1. **AddRoundKey**— every byte of the state is joined together with the round key utilizing bitwise xor.

3. Rounds

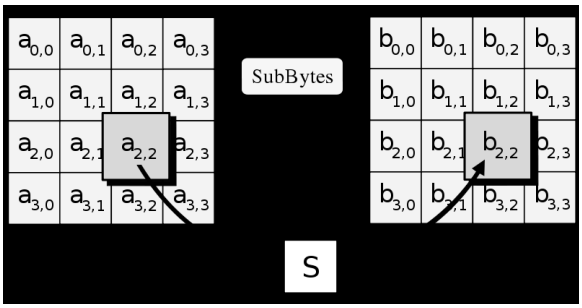
1. **SubBytes**—a non-linear substitution step where each byte is supplanted with an alternate as per a lookup table.
2. **ShiftRows**— a transposition step where each one line of the state is moved cyclically a specific number of steps.
3. **MixColumns**— a blending operation which works on the sections of the state, joining together the four bytes in every segment.
4. **AddRoundKey---** the subkey is combined with the state.

4. Final Round (no MixColumns)

1. SubBytes
2. ShiftRows
3. AddRoundKey

The Shift Rows step:

In the Sub Bytes step, every byte in the network is overhauled utilizing a 8-bit substitution box, the Rijndael S-box.This operation gives the non-linearity in the figure. The S-box used is derived from the multiplicative inverse over GF(28), known to have good non-linearity properties. To avoid attacks based on simple algebraic properties, the S-box is constructed by combining the inverse function with an invertible affine transformation. The S-box is also chosen to avoid any fixed points (and so is a derangement), and also any opposite fixed points

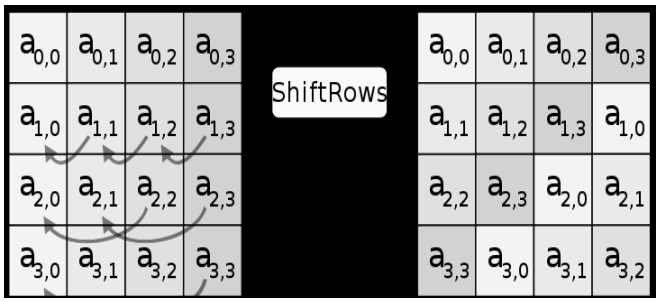


The Shift Rows step

In the Shift Rows step, bytes in each row of the state are shifted cyclically to the left. The number of places each byte is shifted differs for each row. The Shift Rows step operates on the rows of the state; it cyclically shifts the bytes in each row by a certain offset. For AES, the first column is left unaltered. Every byte of the second column is moved one to the left.

Essentially, the third and fourth columns are moved by balances of two and three individually. For the square of size 128 bits and 192 bits the moving example is the same Thusly, each section of the yield state of the Shift Rows step is made

of bytes from every section of the info state (Rijndael variations with a bigger square size have somewhat diverse counterbalances). In the case of the 256-bit block, the first row is unchanged and the shifting for second, third and fourth row is 1 byte, 3 bytes and 4 bytes respectively—this change only applies for the Rijndael cipher when used with a 256-bit block, as AES does not use 256-bit blocks.



The Mix Columns step

In the MixColumns step, each column of the state is multiplied with a fixed polynomial c(x).In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation. The MixColumns function takes four bytes as input and outputs four bytes, where each

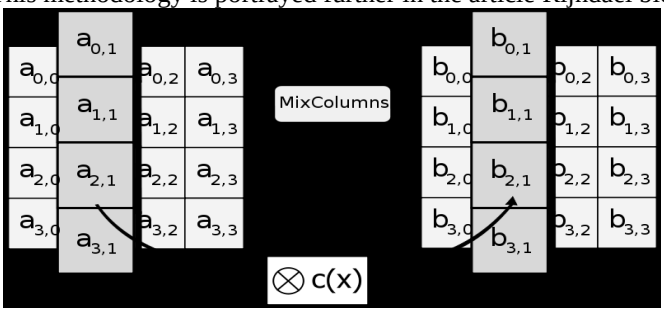
Input byte influences every one of the four output bytes. Together with Shiftrows, Mixcolumns gives dispersion in the figure. During this operation, each column is multiplied by the known matrix that for the 128 bit key is

2	3	1	1
1	2	3	1
1	1	2	3
3	1	1	2

The multiplication operation is defined as: multiplication by 1 means leaving unchanged, multiplication by 2 means shifting byte to the left and multiplication by 3 means shifting to the left and then performing xor with the initial un-shifted value. After shifting a contingent xor with 0x11b ought to be performed if the move worth is bigger than 0xff.

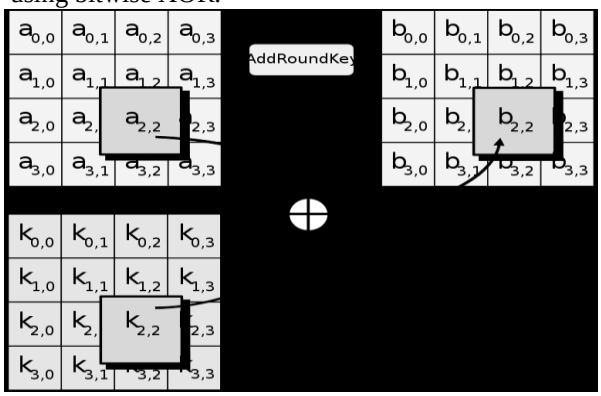
In more general sense, every segment is dealt with as a polynomial over Gf(28) and is then reproduced modulo x4+1 with an altered polynomial c(x) = 0x03 • x3 + x2 + x + 0x02. The coefficients are shown in their hexadecimal likeness the parallel

representation of bit polynomials from Gf(2)[x]. The Mixcolumns step can additionally be seen as an increase by a specific MDS network in a limited field. This methodology is portrayed further in the article Rijndael blend sections.



The AddRoundKey step:

In the AddRoundKey step, each byte of the state is combined with a byte of the round subkey using the XOR operation (ı). In the AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule; each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.



CRITERIA OF A CRYPTOGRAPHIC ALGORITHM

The security of the model has been analysis on the basis of their encryption algorithm and the key management. It has been observed that the encryption algorithm have their own characteristics; one algorithm provides security at the cost of hardware, other is reliable but uses more number of keys, one takes more processing time. This section shows the various parameters which plays an important role while selecting the cryptographic algorithm. The Algorithm found most promising is AES Algorithm with 256 bit key size(256k).

V. CONCLUSION

Security of data in cloud is one of the major issues in cloud computing environment. This paper surveyed the various existing security measures in cloud computing and compare their various security parameters. Multicast key management will provide better security for the cloud for the secure data transaction, through keying and rekeying process. The security of the cloud can extend by applying some batch rekeying methods which will avoid further complexity in rekeying.

REFERENCES

[1] V. Sathana and J. Shanthini,"Automated Security Providence For Dynamic Group In Clode"In International Journal Of Innovative Research In CE",Vol.2,Special Issue 3,July 2014.

[2] B Bhavani Bai,"Ensuring Security At Data Level In Cloud Using Multi Cloud Architecture",In International Journal Of Science And Technology,(ISSN 2321-919X) June 2014.

[3] Anurag Singth Tomar,Gaurav Kumar Tak"Secure Group Key Agreement with Node Authentication"In International Journal Of Advancee Research In Computer Engineering and Technology(IJARCET),Vol.3,Issue 4,April 2014.

[4] Anu Kumari , Krishna Bansal"Secure resource location with the help of phonic coordinate system and host authentication in cloud environment", Vol. 3(2).

[5] K.Sriprasadh,Saicharansrinivasan,and O. Pandithurai" A Novel Method To Secure Cloud Computing Through Multicast Key Management", In International Conference Of Information Communication,2013.

[6] Rabi Prasad Padhy,Manas Rajan Patra and Suresh Chandra Satapathy"Cloud Computing Security Issues And Research Challenges",International Journal Of Computer Seience and IT ,Vol. 1,No.2,2011.

[7] Navai Jose,Chara Knmani A"Data Security Model Enhancement In Cloud Environment", In Journal Of Computer Science And Engineering"(IOSR-JCE),Vol.10,Issue 2,2013.

[8] Gansen Zhao, Chunming Rongy, Jin Liz, Feng Zhangx and Yong Tang, "Trusted Data Sharing over Untrusted Cloud Storage Providers," 2nd IEEE International Conference on Cloud Computing Technology and Science.

[9] (U.S.) Nicholas. Carr, fresh Yan Yu, "IT is no longer important: the Internet great change of the high ground - cloud computing," The Big Switch:Rewining the World,from Edison to Google, , ITIC Publishing House, October 2008 1-1

[10] Ya-Qin Zhang, of computing in the "cloud - Client", The Economic Observer reported, the future http://www.sina.com.cn, 2008 Nian 07 Yue 12 Ri 14:30

[8]Damini E, Di Vermercati SDC, Foresti S, Jajodia S, Paraboschi S, Samarati P, 2008, ''Balancing confidentiality and efficiency in untrusted relational DBMSS'', SIGMOD RIn:Proceedings of the 10th ACM conference on computer and communications security,pp. 93-102.

[9] Atallah MJ, Frikken KB, Blanton M, 2009 "Dynamic and efficient key management for access hierarchies" In:Proceedings of the 12th ACM conference on computer and communications security, pp. 190–202.

[10] Atallah M J, Blanton M, Fazio N, 2009, Frikken KB, "Dynamic and efficient key management for access hierarchies" ACM Transactions on Information and System Security, pp.18:1–43.

[11] Di Vimercati SDC, Foresti S, Jajodia S, Paraboschi S, Samarati P, 2008.
"Over-encryption: management of access control evolution on outsourced data", In:Proceedings of the 33rd international conference on very large databases, pp.123–34.

[12] Di Vimercati SDC, Foresti S, Jajodia S, Paraboschi S, Samarati P, 2007, "A data outsourcing architecture combining cryptography and access control"
In:Proceedings of the 2007 ACM workshop on computer security architecture, pp.63–9.

[13] Germano Caronni , Marcel Waldvogel_ , Dan Sun_ , Bernhard Plattner_, "Efficient Security for Large and Dynamic Multicast Groups" First publ. in: Proceedings / Seventh
IEEE Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises WET ICE '98), June 1998, Stanford, California, USA, pp. 376-383 (Access on dated:13-sep2013)

[14] D. V. Naga Raju, Dr. V. Valli Kumari and Dr. K. V.S.V.N. Raju," Efficient Distribution of Conference Key for Dynamic Groups", International Journal of Computer Theory and Engineering, Vol. 2, No. 4, August, 2010 1793-8201 (Access on dated:13-sep-2013)

[15] Re_k Molva, Alain Pannetrat, "Scalable Multicast Security in Dynamic Groups" (Access on dated:13-sep-2013)

[16] Fu-Kuo Tseng, and Rong-Jaye Chen, " Enabling Searchable Dynamic Data Managementfor Group Collaboration in Cloud Storages" (Access on dated:13-sep-2013)

[17] Boyang Wang †,‡, Sherman S.M. Chow §, Ming Li ‡, and Hui Li † † State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, China ‡ "Storing Shared Data on the Cloud via Security-Mediator", 2013 IEEE 33rd International Conference on Distributed Computing Systems (Access on dated:13-sep-2013)

[18] M. Kavitha Margret, "Secure Policy Based Data Sharing for Dynamic Groups in the Cloud"ISSN: 2278 – 1323 International Journal of Advanced Research in Computer Engineering and Technology (IJARCET) Volume 2, Issue 6, June 2013 (Access on dated:13-sep-2013)

[19] Zhang, J, Varadharajan, V and Mu, Y, "A novel dynamic key generation for secure Multicasting". The 11th IEEE International Conference on Networks, 28 September - 1 October 2003, 391-395. Copyright IEEE 2003.Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au (Access on dated:13-sep-2013)© 2014, IJCSMC

