ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Efficient and Identity Based Distributed Key Management Protocol for Re-Encryption Data in Secure Cloud

B Mohan
Department of Computer Science and Engineering,
Sree Rama Institute of Technology & Science,
Kotha Kuppenkuntla, Telangana 507302, India.

Veeresham
Department of Computer Science and Engineering,
Sree Rama Institute of Technology & Science,
Kotha Kuppenkuntla, Telangana 507302, India.

ABSTRACT:

Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (eg., network, servers, storage) that can be rapidly provisioned and released with minimal management effort or service provider interaction. During last few year , data security and integrity in cloud computing has emerged as a significantly important research area that has attracted increasing attention from both industry and academia. The virtual environment of cloud computing allows users to access computing power that exceeds what it contained within their own physical worlds. Consequently, several Data security and integrity concerns have arisen, including key management, access control, searchable encryption techniques, remote integrity checks and proof of ownership in the cloud.

Keywords: ***Secure Cloud, Re-Encryption Data, Mobile Cloud, Group Key, Security***

Introduction

The recent advancements in technology have changed the way how electronic data is stored and retrieved. Nowadays, individuals and enterprises are increasingly utilising remote services (such as Dropbox [1], Google Cloud Storage [2] and Amazon Simple Storage Service [3]), mainly for economical benefits. These services not only enable information sharing but also ensure availability of data from anywhere at any time. However, the growing use of remote services raises serious privacy issues by putting personal data at risk, particularly when the servers offering such services are untrusted. Unfortunately ,servers get direct access to the data they store and process. For protecting sensitive data from servers in untrusted environments, data could be encrypted before leaving trusted boundaries. Regardless of whether the data is encrypted or not, the server will need to decide who will gain access to it. For regulating access to the data, access control policies could be specified. These are access control policies that will describe who can gain access to the data. State-of-the-art policy-based systems can ensure enforcement of these policies. However, the matter becomes complicated when sensitive policies, which may leak private information, have to be enforced in untrusted environments.

Cloud computing is an emerging paradigm offering outsourced services to enterprises for storing and processing a huge amount of data at very competitive costs. It promises higher availability, scalability and more effective quality of service than in-house solutions. In cloud computing, the outsourced piece of data is within easy reach of cloud service providers. Unfortunately, one of the strong obstacles in widespread adoption of the cloud is to preserve confidentiality of the data [5]. There are several techniques that can guarantee confidentiality of data stored in outsourced environments while supporting basic search capabilities [6–15]. However, they do not support access control policies to regulate access to a particular subset of the stored data. State-of-the-art policy based mechanisms can work only when they are deployed and operated within a trusted domain [16]. In an untrusted environment, access policies may reveal sensitive information about the data

Cite this article as: B Mohan & Veeresham, "Efficient and Identity Based Distributed Key Management Protocol for Re-Encryption Data in Secure Cloud", International Journal of Research in Advanced Computer Science Engineering, Volume 5 Issue 2, 2019, Page 5-13.

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

they aim to protect. To understand how access policies may reveal sensitive information in outsourced environments, let us imagine a scenario where a healthcare provider has outsourced its health record management services to a third party service provider. In this scenario, we do not trust the service provider to preserve data confidentiality. Therefore, we can encrypt health records before storing them in the outsourced environment. Furthermore, health records are associated with an access policy in order to prevent any unintended access. Let us consider the following access policy: only a Cardiologist may access the health record, which is attached to the health record.

Even if the data is encrypted, a curious service provider may still infer private information about the patient's medical conditions. In the example policy, a curious service provider may easily deduce that the patient could have heart problems. A misbehaving service provider may sell this information to banks that could deny the patient a loan given her health conditions.

Related Work

We have Fiat and Naor[14] introduce a k-resistant protocol. Using this safety measures to about k user is provide with O(k log k log n) keys and server communications O(k2 log2 k log n) communication per rekeying. EBS (Exclusion Basis System) proposed by Eltoweissy et al.[13] is a combinational formulation which helps user to switch among number of keys needed to be stored and number of messages to be transmitted. All this is for key updation so that way out to collision is provided.

In the prior days, this assembly generation supervision protocols drawn in the logically generalized Dh key concurrence protocol. Many examples can be quoted like Ingemarsson et al. [18], Steer et al. [28], Burmese and Desmids [9],and Steiner et al. [29] . Later, in 1990s, steiner et al[29] came familiar with extension of DH designation it as DH key trade[29] and in 2001, name was changed to validation services[6].

Later from 2006, there was a sweeping advancement in this Key management generation. In the very year of 2006, Bohli[8] proposed a skeleton for Key management production agreement which is intended to make available security opposing harming participators and active unauthenticated users at every point in the network. In 2007, Katz and Yung [19] proposed the first constant-on all sides and fully scalable group DH protocol which is provably safe and sound in the standard model. Above all, the key feature of group DH is to generate a secret Key management by a standardised group like DKMP other than relying on members inside.

The next expansion in providing defence is identifying the intruders nearby inside the arrangement. For that, Tzeng[31] provided a symposium key union protocol with the assistance of discrete logarithm (DL). Each user in the group require to have nm power polynomials with n signifying number of participants. Later,. in 2008, Cheng and Lain [11] modified Tseng's conference key agreement protocol based on bilinear pairing. In2009, Huang et al. [16] proposed a no interactive protocol based on DL assumption to improve the efficiency of Tseng's protocol.

All the proposals made and developed till now are good. But one main problem is the time constraint. Since this key agreement involves all the communication entities, takes a lot of time for decision. So to trim down this, we have 2 different solutions. (1)All the statement entities high and mighty that there is an offline server active all the time and decides the secret key with this assumption.[4],[14],[25,][3]. (2) All the communication entities high and mighty that an online server is in active state.

Even though they make use of same methodology, there is a slight difference. as a substitute of encrypt in the group sequential key(GTK) by Key encryption key(KEK) and individually axiom the secret key in turn to each user, here in this approach, the in turn of Key management is also said to all user so that they can weigh up their own secret keys. Lain et.al [20] in 1989

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

was the 1st to come up with an algorithm in this approach making use of (t, n) .It consists of (k-1) member. We can also provide some travel document in [2],[21],[25] with the same principle.

Coming to our broadsheet, we are able to make a way out to this problem by provided that confidentiality and authentication. We also came familiar separating the insider and outsider attack.

Lung-Chung Li et al[33] have projected a secure, efficient, and scalable distributed ID-base multiple secrets key supervision scheme (IMKM) for gather-based mobile ad hoc networks. In order to address the highly vibrant topologies and varying link qualities of ad hoc network, the master secret key is generated and circulated by all cluster heads. As a result, not only are central instance avoided, which constitute single points of attack and breakdown, but this also leads to more autonomous and flexible key update method. The proposed SECURE CLOUD is considering mobility of the nodes but the resource heterogeneity is not taking into consideration in to account. Hence, with the motivation of the protocol IMKM, here in this paper we develop a distributed key model that even considering the heterogeneity in node resources.

To achieve all the above, every user should have an account in DKMP to access the Key management transfer service and in turn to achieve a secret key. So, for all these transformation, we need a secret channel for message fleeting to all the statement entities. And also to convey this selected grouping key, to all insiders of network, we need a disconnect and secret channel. This Key management is classified and no mathematical calculations are involved here but it is a in sequence theoretically secure.

Distributed Key management Protocol for Re-encryption data

Having a look at its background, we should be acquainted with: desire two large primes p and q and calculate a public n such that $n = p*q$, which can be referred as predicament of factoring.

Practically resolve the quandary of factoring is complicated. Even though Blakely [1] and Shamir [26] residential a resolution for this, it is not so proficient. According to this scheme, a whole secret key is shared among all the statement entities so that each get a share of t. With more or equal to t shares each can estimate their surreptitious keys. But with less than t, totalling is not possible. This is called (t,n) scheme. It in turn consists of 2 algorithms:

a) Share production algorithm:

Dealer D first picks a polynomial f(x) of degree (t-1) randomly: $f(x) = a_0 + a_1x+,.....a_{t-1}x^{t-1}$, in which the secret $S = a_0 = f(0)$ and all coefficients $a_{0,}a_{1,.....}a_{x-1}$ are in a finite field $IF_p = GF(p)$ with p elements.

D calculates all shares: $S_i = f(i)(\text{mod}_p)$ for $i = 1,.....n$ Then,

D calculates a list of n shares $(S_i, S_2,.....S_n)$ and distributes each share S_i to parallel shareholder P_i privately.

b) Secret modernization algorithm:

This algorithm takes any shares $(S_{i_1},....S_{i_t})$ as input, it can reconstruct the secret s as

$$S = f(0) = \sum_{i \in A} S_i \beta_i$$

$$= \sum_{i \in A} S_i \left(\prod_{j \in A-\{i\}} \frac{x_j}{x_{j-} x_i} \right)(\text{mod } p),$$

$A = \{i_1,....i_t\} \subseteq \{1,2,.....n\}$, β_i for $i \in A$ are Lagrange coefficients. This scheme is able to satisfy all the security

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

related issues like able to calculate the secret key only if t or more than t shares are known.

a) If not more than t shares are known, it is not able to calculate the secret key.
b) Also follows the Shamir's scheme that there is no numerical calculations and all are assume base on the above expressions. After all this a modular inverse is to be calculated for secret reconstruction process. It is discussed in Euclid algorithm [30].

Coming to objectives the proposed protocol is distributed key generation under the consideration of peer resource heterogeneity and security. In proposed protocol model, DKMP undertakes the selection of optimized peers to participate in key generation and authenticates the peer integrity and eligibility to become part of the peer network by receiving group key. At the outset every member should register to the DKMP which intern at registration selects peers with optimal resources to participate in key generation and provides those selected peers a confidential matter by which calculation of secret key is done and genuineness state of the every peer expectant to be part of the network. Then the certain peers generate Key management and for each correct and authorised peer to receive group key, a checksum is appended with cipher text. All around the encryption algorithm provides this security. The confidentiality is achieved by secret contribution scheme proposed. For security, a broad broadcast message is created and sent *tp* all announcement entities where its secrecy is maintained notionally.

Considering heterogeneity of the peer resources in key production and security is the key factor in our paper. So the primary goal is to make available security. Some important goals formulate are:

Selecting peers for key generation: Selecting peers that are optimized in terms of have possessions to take part in key generation.

Fixing the key generation peer group count: The projected protocol selects set of landed gentry such that all other peers can obtain Key management from selected peer in hop level.

Key freshness: That is, the key have to not be used before so that further exertion may not arise.

Key Confidentiality: It is the assurance that the secret in turn is accessed only by sanctioned group members.

Key authentication: Providing guarantee that generation and broadcasting of secret Key management is done by DKMP, a trusted administration but not by any hackers.

In spite of all these Secure and defence issues, we have 2 more threats to be work on

a) Any hacker in person using the valid group user for his works done.
b) Hackers modifying the communication in their way of transfer even before reaching the aim esp. DKMP.

Proposed Protocol Secure Cloud For Encrypted Data

To overcome these, the projected protocol has 3 counteractive measures.

a) Initialization of DKMP.
b) User register
c) Selecting most advantageous peers for Key management generation
d) Key management production and distribution.
e)

Initialization of DKMP: In this primary step, DKMP chooses optimal peers to take part in Key management generation. Then KMP sends all random prime selected as shared checksums of the optimal peers to all peers participate in key generation. Then the peers selected for key generation compute n from shared checksums sent by KMP. This n is made public as stated in the projected theory above in this paper.

User Registration: instantaneously after the DKMP is initialized, it is ready to use and encourage the user registrations. It in turn keeps track of all the registered

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

users and alerts best possible peers about unconstitutional peers.

Optimal peer selection for key production: Since the heterogeneity of the peer computational resources has taken into thoughtfulness, our proposed model selects the optimal peers with eligible computational resource for Key management protocol.

Key management protocol and allotment: As the listing phase ends with the user requirements to the DKMP for validation, it sends the shared checksums of the most advantageous peers to all optimal peers along with the identification of eligible peers to optimal peers selected for Key management creation. Then optimal peers erratically select the secret key t of the hop level requested user and send him the note which is exclusive to him. By this he can access the assemblage key.

All this transformation connecting the DKMP and users is fallows.

Step 1: DKMP receive certificates and about computational resources from assemblage members to initiate the key production.

Step 2: As the confirmation, DKMP response by sending the broadcast messages to selected peers that are most advantageous in property to participate in key production.

Step 3: As a note of agreement, most advantageous peers send a random brave $R_i \in Z_n^*$ to DKMP.

Step 4: DKMP sends all casual challenges as shared checksums of most advantageous nodes to all optimal nodes.

Then finest nodes generates Key management k from these collective checksums received from KM, and generate an interpolated polynomial $f(x)$ with degree t' to pass through $(t+1)$ points, $(0,k)$ and $(x_i, y_i \oplus R_i)$, for $i = 1,2,3,....t'$. Optimal nodes also compute t other points, OP_i for $i = 1,2,3,....t'$, on $f(x)$ and $auth = h(k, OP_1, OP_2, OP_3, ...OP_{t'})$, where h is a one-way hash function and $OP_1, OP_2, OP_3, ...OP_{t'}$ are optimal peers. Then optimal peers send $(auth, OP_i)$, for $i = 1, ..., t'$.

Step 5: Every group member, P_i, after knowing the shared secret, $(x_i, y_i \oplus R_i)$, and other optimal peers OP_i for $i = 1, ..., |OP|$, on $f(x)$ P_i able to compute the polynomial $f(x)$ and recover the Key management and then P_i computes hash value from k and OP_i for $i = 1,2,3,....t'$ then compares with $auth$ for validity.

Experimental Results

The experiment was conducted by embryonic simulation model using java. We manufacture a simulation network with hops count of 80. The recreation parameters described in table 1. substantiation ensures that the buffer is properly allocated to valid packet. The recreation model aimed to compare “above-board Key managementTransfer Protocol Based on Secret Sharing” and projected RE-ENCRYPTION DATA. The concert check of these two protocol carried out against to the threats listed below.

- Rushing attack
- Denial of service
- Tunnelling

The fortification against tunnel attack is the improvement of the RE-ENCRYPTION DATA over SECURE CLOUD[33].

Number of nodes Range	80
Dimensions of space	1500 m × 300 m
Nominal radio range	250 m
Source–destination pairs	20

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Source data pattern (each)	4 packets/second
Application data payload size	512 bytes/packet
Total application data load range	128 to 512 kbps
Raw physical link bandwidth	2 Mbps
Initial ROUTE REQUEST timeout	2 seconds
Maximum ROUTE REQUEST timeout	40 seconds
Cache size	32 routes
Cache replacement policy	FIFO
Hash length	80 bits
certificate life time	2 sec

Table1: Simulation parameters that we considered for experiments

Proposed protocols	Routing strategy	Protects from Rushing attack	Protects from Denial of service	Protects from Routing table modification	Protects from Tunneling
SECURE CLOUD[33]	Mobile	Yes	Yes	No	No
RE-ENCRYPTION DATA	Mobile	Yes	Yes	Yes	Yes

Table 2: Protocols and their ability to handle different attacks

The metrics to verify the presentation of the proposed protocol are

- **Data packet deliverance fraction:** It can be premeditated as the ratio between the quantity of data packets that are sent by the source and the numeral of data packets that are homeward bound by the sink.
- **PACKET DELIVERY FRACTION:** It is the ratio of data packets deliver to the destinations to those generated by the sources. The PDF tells about the presentation of a protocol that how successfully the packets have been delivering. Higher the value gives the better results.

- **AVERAGE END TO END DELAY:** normal end-to-end delay is an average end-to-end delay of data packets. Buffering during route finding latency, queuing at boundary queue, retransmission delays at the MAC and transport times, may cause this delay. Once the time variation between packets sent and received was recorded, separating the total time difference over the total number of CBR packet received gave the average end-to-end delay for the external packets. Lower the end to end delay better is the concert of the protocol.
- Packet Loss: It is defined as the divergence between the number of packets sent by the source and external by the sink. In our results we have considered packet loss at network layer as well as MAC layer. The routing protocol forwards the packet to end if a valid route is known, or else it is buffered until a route is presented. There are two cases when a envelope is dropped: the buffer is full when the envelope needs to be buffered and the time exceed the limit when packet has been buffered. Lower is the packet loss better is the concert of the protocol.

ROUTING OVERHEAD: Routing overhead has been calculated at the MAC layer which is defined as the ratio of total number of routing packets to data packets.
Figure 1(a) shows the envelope Delivery Ratio (PDR) for basic P2P, SECURE CLOUD[33] and RE-ENCRYPTION DATA. Based on these results it is evident that RE-ENCRYPTION DATA recovers most of the PDR loss that experimental in SECURE CLOUD[33] against to basic P2P . The approximate PDR loss well

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

again by RE-ENCRYPTION DATA over SECURE CLOUD[33] is 1.5%, which is an typical of all pauses. The minimum personality recovery observed is 0.18% and maximum is 2.5%. Figure 1(b) indicates SECURE CLOUD[33] advantage over RE-ENCRYPTION DATA in Path optimality. RE-ENCRYPTION DATA used average 0.019 hops longer than in SECURE CLOUD[33] because of the hop level qualifications validation process of the RE-ENCRYPTION DATA that abolish nodes with undo documentation. Here slight advantage of SECURE CLOUD[33] over RE-ENCRYPTION DATA can be observable.

The packet delivery little bit (PDF) can be expressed as:

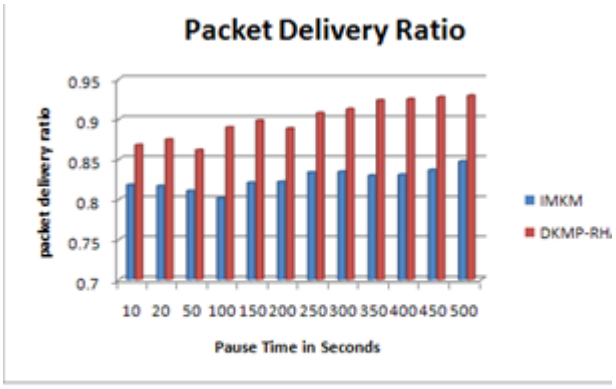
$$P' = \sum_{f=1}^{e} \frac{R_f}{N_f}$$

$$P = \frac{1}{c} * P'$$

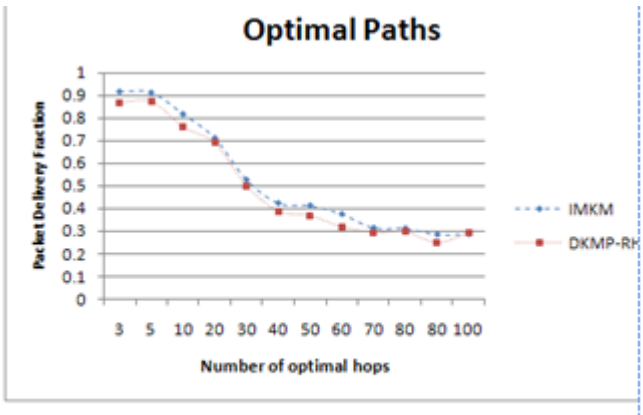
- P is the small part of successfully delivered packets,
- c is the total number of stream or connections,
- f is the unique flow id serving as index,
- R_f is the count of packets received from flow f
- N_f is the count of packets transmitted to flow f.

Figure 1(c) confirms that RE-ENCRYPTION DATA is having fewer packets overhead over SECURE CLOUD[33]. Due to stable paths with no compromised or victimized nodes determined by RE-ENCRYPTION DATA this advantage become possible. The container overhead observed in SECURE CLOUD[33] is average 5.29% more than packet overhead observed in RE-ENCRYPTION DATA. The bare minimum and maximum packet above your head in SECURE CLOUD[33] over RE-ENCRYPTION DATA observed is 3.61% and 7.29% correspondingly.

MAC load visual projection is slightly more in RE-ENCRYPTION DATA over SECURE CLOUD[33]. We can examine this in Figure 1(d), which is because of supplementary control packet exchange in RE-ENCRYPTION DATA for neighbour hop substantiation through certificate exchange. The common MAC load overhead in RE-ENCRYPTION DATA over SECURE CLOUD[33] 1.64%. The least amount and limit MAC load overhead observed is 0.81 and 3.24% correspondingly.

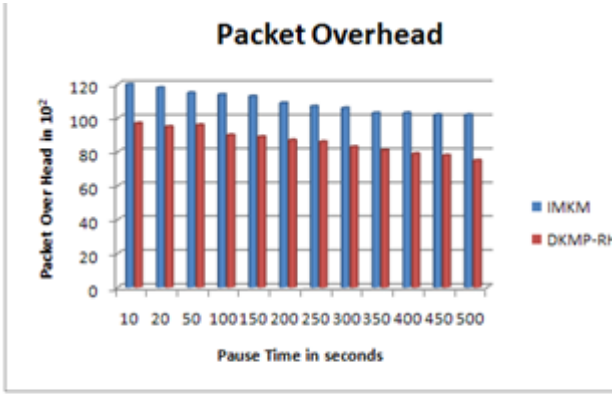


(a) Packet delivery ratio comparison using bar chart

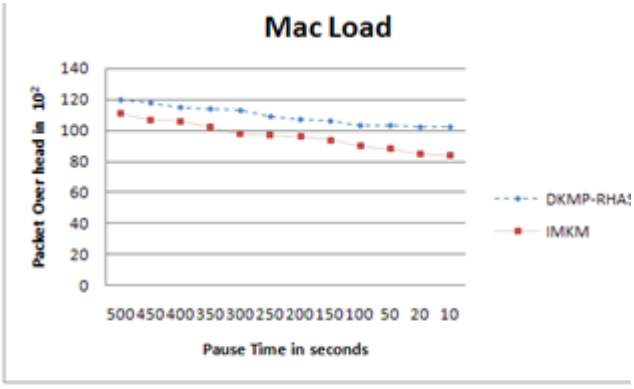


(b) Line chart representation of Path optimality

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com



(c) A bar chart representation of Key management Protocol comparison report



(d) Key management Protocol comparison represented in bar chart format

Conclusion

Tight security mechanism are obligatory to tolerate secure statement among the group member. Thus, a letter session must have safety measures services to provide authentication, integrity, and confidentiality. Key management (KM) is the primary and key part of the safe group communication. The performance of KM generation method, which is required for secure statement, may debase due to less performing members. Thus, the production process must be done is a more precise way but filter less amateur dramatics members. Many changes are taking place in the recent years as enlarge in usage of mobile computer, network cluster statement with model servers. Apart from this, heterogeneity and circulated computer atmosphere became general in the current internet world. Thus, KM supervision system must consider various parameters, differences and environments involved in the communication.

These consideration as the basis, the helpfulness of RE-ENCRYPTION DATA code of behaviour in comparison to SECURE CLOUD[33] is proved. This code of behaviour improves the good organization by taking into account the parameter effecting the performance i.e. computational delay and arrangement latency. Thus, this examine is expected at and thus proved that GKGP is more efficient and maximize the applicability of statement.

7. References

[1] G.R. Blakley, “Safeguarding Cryptographic Keys,” Proc. Am. Federation of Information Processing Soc. (AFIPS ’79) Nat’l Computer Conf., vol. 48, pp. 313-317, 1979.

[2] S. Berkovits, “How to Broadcast a Secret,” Proc. Eurocrypt ’91 Workshop Advances in Cryptology, pp. 536-541, 1991.

[3] R. Blom, “An Optimal Class of Symmetric Key Generation Systems,” Proc. Eurocrypt ’84 Workshop Advances in Cryptology, pp. 335-338, 1984.

[4] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, “Perfectly Secure Key Distribution for Dynamic Conferences,” Information and Computation, vol. 146, no. 1, pp. 1-23, Oct. 1998.

[5] C. Boyd, “On Key Agreement and Conference Key Agreement,” Proc. Second Australasian Conf. Information Security and Privacy (ACISP ’97), pp. 294-302, 1997.

[6] E. Bresson, O. Chevassut, D. Pointcheval, and J.-J. Quisquater, “Provably Authenticated Group Diffie-

ISSN No : 2454-423X (Online)
International Journal of Research in Advanced Computer Science Engineering
A Peer Reviewed Open Access International Journal
www.ijracse.com

Hellman Key Exchange," Proc. ACM Conf. Computer and Comm. Security (CCS '01), pp. 255-264, 2001.

[7] E. Bresson, O. Chevassut, and D. Pointcheval, "Provably-Secure Authenticated Group Diffie-Hellman Key Exchange," ACM Trans. Information and System Security, vol. 10, no. 3, pp. 255-264, Aug. 2007.

[8] J.M. Bohli, "A Framework for Robust Key managementAgreement," Proc. Int'l Conf. Computational Science and Applications (ICCSA '06), pp. 355-364, 2006.

[9] M. Burmester and Y.G. Desmedt, "A Secure and Efficient Conference Key Distribution System," Proc. Eurocrypt '94 Workshop Advances in Cryptology, pp. 275-286, 1994.

[10] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, "Multicast Security: A Taxonomy and Some Efficient Constructions," Proc. IEEE INFOCOM '99, vol. 2, pp. 708-716, 1999.

[11] J.C. Cheng and C.S. Laih, "Conference Key Agreement Protocol with Non Interactive Fault-Tolerance Over Broadcast Network," Int'l J. Information Security, vol. 8, no. 1, pp. 37-48, 2009.

[12] W. Diffie and M.E. Hellman, "New Directions in Cryptography," IEEE Trans. Information Theory, vol. IT-22, no. 6, pp. 644-654, Nov. 1976.

[13] M. Eltoweissy, M.H. Heydari, L. Morales, and I.H. Sudborough, "Combinatorial Optimization of Key managementManagement," J. Network and Systems Management, vol. 12, no. 1, pp. 33-50, 2004.

[14] A. Fiat and M. Naor, "Broadcast Encryption," Proc. 13th Ann. Int'l Cryptology Conf. Advances in Cryptology (Crypto '93), pp. 480-491, 1994.

[15] H. Harney, C. Muckenhirn, and T. Rivers, "Key managementManagement Protocol (GKMP) Architecture," RFC 2094, July 1997.

[16] K.H. Huang, Y.F. Chung, H.H. Lee, F. Lai, and T.S. Chen, "A Conference Key Agreement Protocol with Fault-Tolerant Capability," Computer Standards and Interfaces, vol. 31, pp. 401-405, Jan. 2009.

[17] IEEE 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements, 2004.

[18] I. Ingemarsson, D.T. Tang, and C.K. Wong, "A Conference Key Distribution System," IEEE Trans. Information Theory, vol. IT-28, no. 5, pp. 714-720, Sept. 1982.