A SECURING GROUP COMMUNICATION USING PEER TO PEER CLIENT - SERVER NETWORKS

Hema Priya KE [1], Vigneswaran CK [2], Manikandan U [3],Saravana Kumar.M[4], Vasanth Kumar M[5]

Department of Computer Science and Applications

Sri Krishna Arts and Science College

kehemapriya@gmail.com

ABSTRACT

The efficient design of a distributed group key management for a peer to peer (P2P) network with minimal computation complexity in dynamic secure group communication may be a challenging issue. This is often due to the absence of a centralized coordinator. So as to supply this facility, a self-composed distributed group key management framework is proposed for secure P2P communication. During this proposed work, group key computation is performed using the Chinese remainder theorem and secure communication is performed through the RSA encryption algorithm. This self-composed key management may be a one-round protocol during which a shared group key's generated using the general public key of every individual user, and it's derived from the respective private key.

The most advantage of the group key management scheme proposed during this paper is that it reduces the computation complexity of the peer users to O (1). This reduction in computation complexity is achieved by performing one addition and multiplication operation during one member join and one subtraction operation during one member leave operation. The proposed algorithm has been analyzed to reduce the computations with well-known existing distributed group key management protocols.

1. INTRODUCTION

Secure communication is when two entities are communicating and doesn't need a third party to concentrate in. For that they have to speak during a way not vulnerable to eavesdropping or interception.

Secure communication includes means by which individuals can share information with varying degrees of certainty that third parties cannot intercept what was said. aside from spoken face-to-face communication with no possible eavesdropper, it's probably safe to say that no communication is guaranteed secure during this sense, although practical obstacles like legislation, resources, technical issues (interception and encryption), and therefore the sheer volume of communication serve to limit surveillance. With many communications happening over long distance and mediated by technology, and increasing awareness of the importance of interception issues, technology, and its compromise is at the centre of this debate. For this reason, this text focuses on communications mediated or intercepted by technology. can be created to realize secure group communication [1].

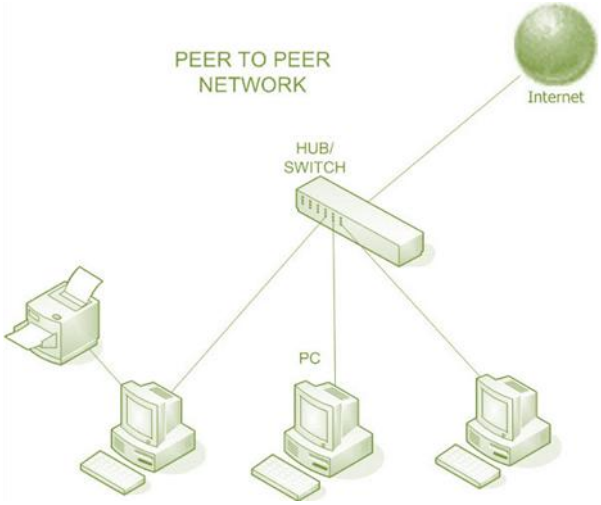
1.1 PEER TO PEER (P2P)

Peer-to-peer (P2P) could also be a decentralized communications model during which each party has the same capabilities and either party can initiate a communication session. Unlike the client/server model, during which the client makes a service request and thus the server fulfils the request, the P2P network model allows each node to function as both a client and server.

P2P systems are often wont to provide anonymized routing of network traffic, massively parallel computing environments, distributed storage, and other functions. Most P2P programs are focused on media sharing and P2P is therefore often related to software piracy and copyright violation. Typically, peer-to-peer applications allow users to regulate many parameters of operation: what percentage member connections to hunt or allow at one time; whose systems to attach to or avoid; what services to offer; and the way many system resources to devote to the network. Some simply hook up with some subset of active nodes within the network with little user control, however. Although uses for the P2P networking topologies are explored since the times of ARPANET, the benefits of the P2P communications model didn't become obvious to the overall public until the late 1990s, when music-sharing P2P applications like Napster appeared. Napster and its successors, including Gnutella and BitTorrent, dig music and film industry profits and altered how people considered acquiring and consuming media. System administrators tend to discourage the utilization of P2P applications. Additionally to docking bandwidth and possibly exposing the administrator's organization legally,

P2P applications are often wont to bypass firewalls and distribute malware. Networks are often found out to stop peer-to-peer "side talk" by PCs [1].

1.2 HOW PEER TO PEER NETWORK WORKS



Typically, when a user downloads a file, the user opens an internet browser, visits the acceptable website and downloads the file. During this case, the web site acts as a server and therefore the user's computer acts as a client that receives the info. This will be compared to a one-way street where the downloaded file is transferred from point A, the web site, to point B, the user's computer. However, if the user downloads an equivalent file through a peer-to-peer network, the download is handled differently. during this case, the user has got to install peer-to-peer software on his computer, which creates a virtual network of peer-to-peer application users. Then when the user downloads a file, it's received in bits that come from various computers within the network that have already got that file. Simultaneously, the info is additionally sent from the user's computer to the computers that invite it. this example is analogous to a street – the file is like numerous small bits of knowledge that come to the user's computer but also leave when requested. In fact, the file transfer load is distributed between the peer computers [2].

2. ARCHITECTURE OF P2P

In a peer-to-peer specification, each computer has an equivalent responsibilities and capabilities. Since there's no server, the computers connect with one another during a workgroup to share files, printers and access to the web. This architecture is practical for workgroups of 12 or fewer computers. As such, it's common in small home or office environments where each PC acts as an independent workstation, storing data on its own disk drive but with the power to share the info with all the opposite PCs on the network [2].

3.ADVANTAGES AND DISADVANTAGES OF P2P

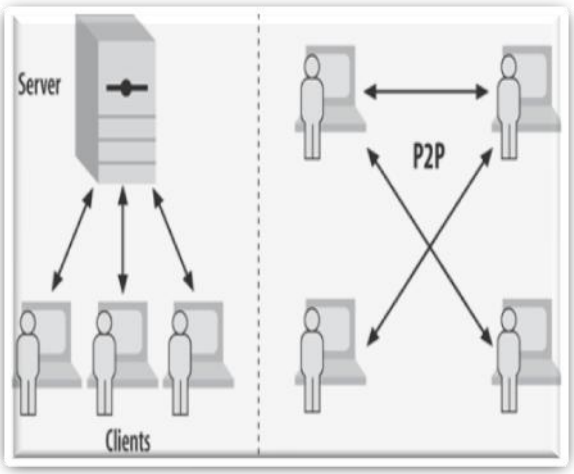
There are varieties of benefits also as disadvantages to peer-to-peer networks.

Advantages include: No got to buy an upscale server. Each user manages his or her own computer, which suggests there's no need for a network manager. Users don't need to have any technical knowledge because the setup is completed via wizards within the software. A P2P networks are often are often found out in homes and little businesses. However, each computer has got to be maintained individually. Less network traffic than a client/server network.

Disadvantages include: Users cannot centrally copy the files and folders. Each computer are often accessed by the opposite computers, which slows down the performance for the user. The files aren't centrally organized into a selected shared area. Rather, they're stored on individual computers. Consequently, it'd be hard to locate the files, if whoever owns the pc doesn't have a logical file system. Each individual user is liable for ensuring that viruses aren't introduced into the network. There is little to no security aside from the permissions [2].

4. P2P VS CLIENT/SERVER

A peer-to-peer network involves two or more computers that share individual resources, including DVD players, printers and disk drives. Each computer within the network acts because the client also because the server, communicating with the opposite computers directly. On a peer-to-peer network, printers on one computer are often employed by any of the opposite computers on the network. All that's necessary may be thanks to connect them, like a Wi-Fi router. A client/server network

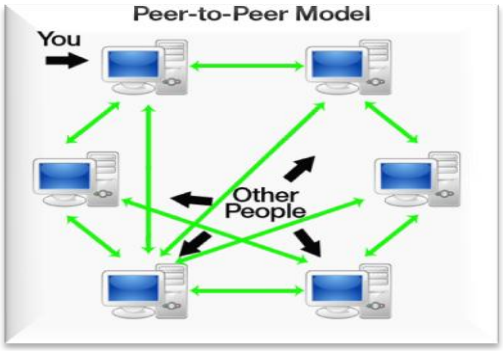


involves numerous clients that hook up with a minimum of one central server where most applications and data are installed. Clients access these resources from the server. Client/server networks typically have faster access speeds due to the massive number of clients they're designed to support.

The clients can function as workstations without sharing resources. Upgrading applications and files is simpler because they're only installed on one computer. during a client/server network, security is handled by the server, not by each individual user [2].

4.1 WHY P2P

An important goal in peer-to-peer networks is that each one client provide resources, including bandwidth, space for storing, and computing power. This is often not case for of a client-server architecture with a hard and fast set of servers, during which adding more clients could mean slower data transfer for all users. Peer to see groups have decentralized control thus they avoid single point failure. Due to decentralized nature they will resist to intentional DoS (Denial -of-Service) attacks. Server-client architecture is often in trouble if servers are down thanks to load caused by sizable amount of requests at same time. When as sizable amount of clients are connecting to same server, we'd like high performance machines at server end, thus is



More costlier as compared to P2P architecture. Peer groups provides us flexibility P2P architecture but we also require that framework be so as that it's adaptable to dynamic number of peers and also does traffic load balancing. So in P2P scenario each peer is communicating to the opposite but requirement is such number of message exchanges should be minimum. Alongside of these feature, for a secure group communication we'll require that re-keying operation should be done when user joins or leaves so as to take care of forward and backward secrecy [3].

5. PROPOSED MODEL

The sponsor user within the P2P network chooses two large prime numbers a and r where a > r. The initialization block generates the mi value for each user such that $a > m_i > r$. The sponsor user is one among the users in the P2P network. Each user within the above mentioned network selects two prime numbers p_i and $q_i \in Z_r^m$. This successively generates the general public key with reference to pi and qi values from which the respective private keys are computed. Each user sends his public key to the group key derivation block located with peer users.

The group key derivation block receives/collects public keys from each user. After receiving the general public keys from the users, the sponsor user distributes the mi values to all other peer users in the network secretly. The public key values are used to compute a common group key H. This computation is performed based on the CRT. The computed group key is used to perform encryption operation to supply secure group communication across the P2P network. The encryption module broadcast the encrypted message ' c ' across the network. Each user uses mi value and therefore the respective private key for performing the decryption [3].

5.1 PROPOSED GROUP KEY MANAGEMENT PROTOCOL

The group key management protocol proposed during this paper works in four steps. Initially, the sponsor user in the P2P network provides mi values to all or any the prevailing group users which are selected from the multiplicative group Zma such Zma > m>i Zmr , where ' a ' may be a 2048 digit prime number. within the proposed key management scheme, we require that each one the mi values selected are prime numbers [19,20,5], [12].

The Initialization block of the sponsor user generates the mi value and provides these values to the opposite corresponding peer users during a secure manner. The sponsor user is one among the users within the P2P network. Each user of the P2P network generates a public key (ei, ∂i) from the multiplicative group Zmr where r is that the prime and also it's a public parameter received from the sponsor user. From this multiplicative group, each user generates his public (ei, ∂i) and computes respective private key (di, ∂i). After computing the private key (di, ∂i), each user sends his public key (ei) to the group key derivation block available within the peer user side. The group key derivation block receives the general public key of all the users through the peer network. the general public key and mi values are wont to estimate the group key 'H' which is liable for data encryption. In computing this group key, we use the prevailing CRT approach. The user who wishes to send the message will encrypt the message using the computed group key. The encryption module broadcast the encrypted message across the network.

Decryption operation is performed by using the respective mi value and personal key which is kept as secret within the user side. the subsequent steps explain the general process involved in our proposed algorithm. This paper analyzes the proposed group key manage- ment protocol for forward and backward secrecy. Some system assumptions for the implemented protocol are made such an adversary has computational power that's present within the class of probabilistic polynomial-time algorithms. Moreover, this adversary has access to the network and is susceptible to eavesdrop, insert, delete, replace and replay messages. Moreover, each group user keeps mi secret and therefore the users shall keep their private key confidential [3].

5.2 BACKWARD SECRECY

This is the technique for preventing a replacement member from accessing the previous communication messages that are sent before to joining the group. So as to access the older communication, an adversary is required to find the old group key. In the proposed protocol, the cipher text is computed such $c \equiv (m \times H) mod\ M.$ If one user joins the group means the group encryption key is updated like $H' = H + \delta_j$ and hence the ciphertext is $c' \equiv (m \times H') mod\ M.$

Therefore, an adversary requires to find the private key and therefore the mi value of any user in the group to decrypt the previously encrypted data. Moreover, user's private key's is computed with respect to the corresponding public key, which is a random value selected from multiplicative group Z_r^m . To find the private key of any peer user, an adversary needs two components public key (e_i) and $\phi(\partial_i)$ which is kept secret by every user. To find the private key di from public key, an adversary can use Wiener's attack.

Theorem 1. Let $\partial_i = p_i \times q_i$ where p_i and q_i are distinct odd primes and let e_i be an integer relatively prime with ∂_i.

In particular, if ei is that the user's public key and di is each user's private key, then the attacker can factor ∂i in polyno- mial time O(ln ∂i). Even after finding the di, finding mi is difficult because it's sent through a secure communication, like Secure Socket Layer. To compute mi, an adversary uses brute force attack. during this work, the dimensions of mi is 1024 bits and any experimental analysis was performed with 1024 bits and 2048 bits. If the time required to per- form one attempt using brute force attack is 1μs, then the entire time required are going to be 2a — 1μs where a is that the size of the multiplicative group.

This property makes the situa- tion infeasible for the adversary to compute the other user's mi value. Therefore, the adversary is inhibited from accessing the communication sent before join operation, which suggests the proposed approach supports the back- ward secrecy [4].

5.3 FORWARD SECRECY

Forward secrecy is that the technique of preventing a mem- ber of the old group from obtaining the presentcommuni- cation after leave operation is performed. When a user leaves the group, he/she may try to find the info, which is shipped by the prevailing users, using any attacks. Within the pro- posed algorithm, it's infeasible for an adversary to decrypt the ciphertext after the leave from the group that was ex- plained for the backward secrecy technique. Because when a user ui leaves from the group, each user subtracts his/her share value like multiplication of Mi, yi and ei, which are stored in δi from H to supply H′. This updated H′ is employed for performing encryption operation to supplysecure group communication

Therefore, a user who had already left from the group cannot decrypt the info during a feasible way because his/her personal keying information isn't included.

Therefore, so as to decrypt the ciphertext the old user requires to find the private key and therefore the mi value. The user who had left from the group may attempt to find the decryption key from the cipher text, which is shipped as a broadcast message from the sender in an infeasible method. Moreover, it'sinfeasi- ble for an attacker to find the di value from the cipher text because the values ∂i and $\varphi(\partial i)$ are kept secret during this method. Thus, an adversary cannot find the info during a feasi- ble method so as to access the present communication, which suggests the forward secrecy is ensured in our pro- posed algorithm [4].

6. CONCLUSION

In this research paper, a replacement and efficient solution to re- duce the computation complexity without increasing high storage complexity for providing secure group communication in P2P network through effective group key man- agement technique is proposed. This solution is predicated on CRT and extended RSA. The proposed algorithm focuses mainly on the minimization of computation complexity of key updating time of a peer user.

The computation time de- creases from 355 ms to 1770 ms within the peer user, just in case if the key size is little (key size = 128 bits). However, when the key size increases (i.e. 1024 bits) the computation time decreases from 3682 ms to 9000 ms for the updation of one key from the dynamic multicast group. With reference to the storage complexity, the amount of keys to be stored by the group members is slightly increased as compared with existing P2P key management protocols. Moreover, in our proposed algorithm, each user sends one broadcast message for informing the members within the group about their public key. This message is shipped so as to compute the group key thereby maintaining an equivalent communication complexity for both the join and leave operations [4].

REFERENCES

1. K.E Hemapriya1 , K. Gomathy2, "A Survey Paper of Cluster based Key Management Techniques for Secured Data Transmission in Manet" IJARCCE,Vol. 5, Issue 10,October 2016, ISSN (Online) 2278-1021.
2. U.Ramya 1 , K.E. HemaPriya 2, "Widespread manner of scrutiny on data pre-processing Techniques" ," International Journal of Computer Science and mobile Application"ISSN Number 2321-8363
3. Hema Priya K E*1, Prathibaa.N.S2 , Anson Sounder3 & Dhinesh Kanna R4," XML Based Approach For Need Of Quality Data For Data Mining" ," GJESR"ISSN 2348 –8034, Vol.6, Issue 3,Marchr 2019.
4. Mrs.KE. Hemapriya*1, Shiva Shamyugtha. M2, Sathya. M3 & Nivetha.G4," a survey on dual-server public-key encryption with keyword search for secure cloud storage" ," GJESR"ISSN 2348 –8034, Vol.6, Issue 3,Marchr 2019.
5. Hemapriya.KE*1, Prabhu.G2, Mudhasir Ahamed Khan M3 & Sathishkumar.N4," A Hybrid Techniques For Traffic Renovation In Wireless Sensor Networks"," GJESR"ISSN 2348 –8034, Vol.6, Issue 3,Marchr 2019.
6. J. Risson, T. Moors. "Survey of Research Towards Robust Peer-to-Peer Networks: Search Methods." Technical Report, University of New South Wales, Sydney, Australia. 2004. http://www.cs.umd.edu/projects/p2prg/p2p-overview.pdf
7. T. Sundsted. "The Practice of Peer-to-Peer Computing: Introduction and History." IBM developers Works. 2001. http://www.ibm.com/developerworks/java/library/j-p2p/

8. M. Balakrishna, M. Kaashoek, D. Karger, R. Morris, I. Stoica. "Looking Up Data in P2P Systems." Communications of the ACM. 2003. http://www.projectiris.net/irisbib/papers/dht:cacm03/paper.pdf
9. D. Schoder, K. Fischbach. "Core Concepts in Peer-to-Peer (P2P) Networking." http://www.ideagroup.com/downloads/excerpts/Subramanian01.pdf
10. N. Naoumov, K. Ross. "Exploiting P2P Systems for DDoS Attacks." International Workshop on Peer-to-Peer Information Management. 2006.