International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Survey on Peer to Peer Network Security

M.L.Soundarya[#1], Sethu Bindu P[#2]
SCOPE, VIT University

Abstract: ***Recently, peer-to-peer (P2P) networks gained popularity in the area of file sharing applications. Along with this popularity there are many security problems and vulnerabilities. In this paper, we inspect the base on which most P2P networks are built, and from this, we see how attacks on P2P networks leverage the very essence of the networks itself: decentralization of resources and of control. Additionally, we look at the privacy and usage attacks that arise in P2P networks as well as approaches that can be used address some of these issues.[1] There would also be a discussion about multicasting, which is used to enhance the security of the network.***

I. INTRODUCTION

Peer to peer networks have grown in a significant way to attract the internet users. The installation process of this peer to peer networks and the computer configuration on these networks is very simple. It is very different from the traditional client server application where the server shares all contents and resources. In these type of networks there exists no central dependency. Therefore if one peer fails it doesn't affect others unlike in client server whole network gets effected. Resources can be shared efficiently and user can control the shared resources. The cost is comparatively less for building and maintaining this type of network. Not only can nodes within a P2P network both receive and send data, they can also share resources such as processing power, storage, peripherals, and network capacity. There are many issues that can cause a problem to P2P network and most of them are caused by the decentralized and anonymous characteristics inherent to P2P networks. The P2P environment is particularly challenging to work in because of the security issues faced by the network and untrustworthy nature of peers characterizing most P2P systems today. In this paper we mainly see about the security issues related to peer to peer networks and what are the ways to resolve these attacks.

II. BACKGROUND OF PEER TO PEER NETWORKS

This kind of architecture is in use for more than 30 years. The first kind of P2P systems was released in late 1960s which was called the ARPANE. It connected UCLA, Stanford Research Institute, UC Santa Barbara and the University of Utah, where every host on the internet could FTP or telnet into every other host. Two students who completed their graduation from Duke University and one student other from University of North Carolina in 1979 developed Usenet. It is based on UUCP (Unix-to-Unix-copy protocol), in which each UNIX machine could connect with another machine and it can exchange files with it and disconnect. Usenet is used in such a way it can exchange data between the two schools. Napster was emerged in May 1999 which is developed by a freshman at North Eastern University and because of which p2p networks popularity got increased. It is a centralized server in which all members of Napster could search the local hard disks of other members for desired mp3 files.
Some protocols:

A. Napster

Napster was the first large P2P network scheme and application that the public used. [2] Users can use this application in such a way that they can share MP3 files among themselves. It was officially operational from 1999 to 2001. Napster utilized a hybrid centralized P2P model where a server that is centralized made the pairing of two users possible based on files that are desired and owned.
The major drawback which was considered by many is that Napster's protocol only allowed for MP3's to be indexed and searched. A work-around was developed that enclosed other file types into which is similar to MP3, deceiving Napster's servers into being searchable. The most common tool used to enclose files was aptly known as "Wrapster."
The state of the user is relayed, when requesting file locations from the Napster server,
Making anonymity unattainable. The users have to create a TCP connection among themselves and pass files here and there so that they can share files. The TCP/IP protocol has no security measures, making it file quite unsafe to prying eyes.

Architecture is another drawback related to this kind of P2P networks. The centralized architecture is liable to attack and can easily

Volume 4 Issue IV, April 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

be taken out of order. If the central server is accommodated, either by DDoS attacks or physical hardware attacks, the whole network will stop to function because the main node is not being operated. [4] Napster led to creation of many clone applications with the intention of sharing files and not just MP3's.

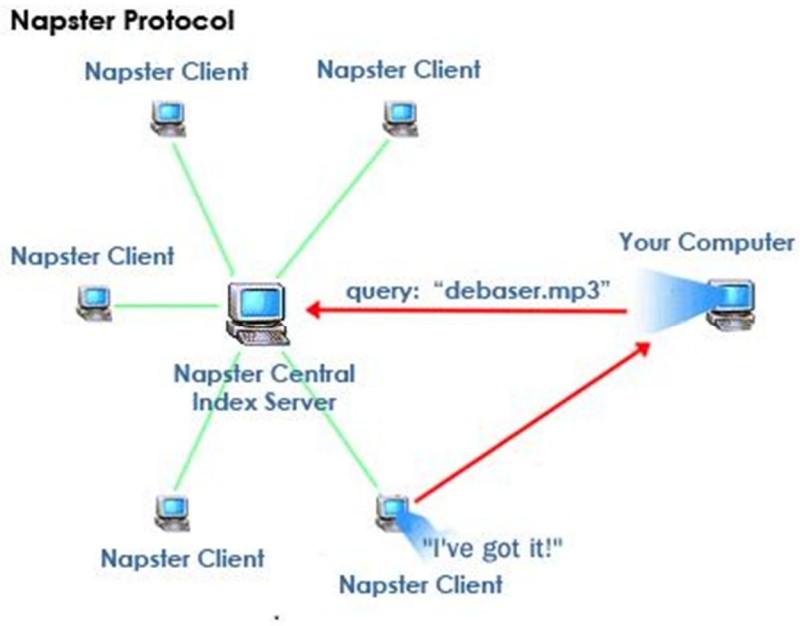

Figure 1

B. Gnutella

Gnutella began at a subsidiary of America Online in early 2000.This protocol does not use a central server instead it uses a "flat ad-hoc topology." [3] In this each and every node or user act as both server as well as client that is it is able to issue and respond to queries as well. This network is a decentralized P2P network in which every node is connected to many other nodes. Therefore here this architecture insures the networks survivability, if one node goes offline i.e. if does not work properly, the whole network does not suffer.

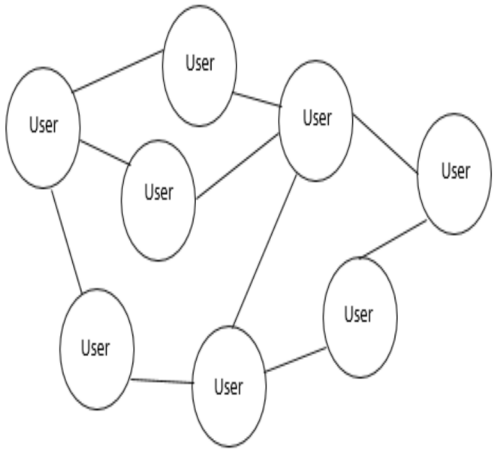

Figure 2

If the users want to find files that they want to download, then they ask their neighbors if they have that particular file. If they do not have the files, they in turn ask their own neighbors. This process continues till the file that is required is discovered or the Time-To-

Live (TTL) counter has decreased to zero from a certain starting number. If the desired file is found, using http files are downloaded. The file sharing happens only between users.

Drawbacks of the Gnutella protocol include users who have very low speed to transmit data, high bandwidth from messages that are elevated, and users who do not wish to share files. If users are within the network which have a low transfer speed, then the network is fragmented. Requests from neighbors can swamp them and due to their low time to respond, these nodes can essentially be damaged by normal traffic. The nodes cannot respond to queries if they are damaged which means they will act like dead nodes i.e. They will not contribute to the network as a whole but they will still be connected to their neighbors. In order to determine the existence of each node in Gnutella network, it needs to periodically "ping" their neighbors and update the connected nodes. It sends back a "pong" whenever a node receives a ping along with information like which port is being used, the IP address of the node, number of files shared, and number of bytes shared. These messages are piled up within the network and they contributed to roughly 50% of the network bandwidth.

Most of the Gnutella users will not share files leading to 25% of the user Base bearing 99% of the networks load. The protocol does not allow users to participate, so it's not possible to wipe them out. Now users are not able to locate the files that they are looking for and creating a network that is very inefficient.

Users can stay unspecified because of the "daisy-chaining" of passing requests all through the network. If a user receives a request from a neighbor, it cannot be determined if it came from the neighbor or from others who are in the network, leading itself to an unspecified action. Nevertheless, every ping message generally consists of the original user's Descriptor ID, which is a unique identifier for that user on the network. If another user can decode the Descriptor ID, then the original user's identity can be determined.

C. Bit Torrent

This protocol was developed by Bram Cohen, a University of Buffalo student in 2001. It is also a decentralized P2P network where file transfers won't occur within the protocol itself. If the users want to locate other users, they must use a "tracker" which contains a list of IP addresses of users sharing a certain file. If the users have to download a particular file they have to discover a tracker for it, which can be discovered on tracker websites such as The Pirate Bay. The files that you download from tracker websites include a tracker file and a torrent file. The torrent file consists of the number of pieces and blocks a file has, the IP address and port number of the tracker, and also the SHA1 hash tables of the pieces for the file. The SHA1 hash tables allow users to "certify the integrity" of each piece downloaded. Files shared through a Bit Torrent network have to be broken up into "pieces" and "blocks. "The size of piece is 512 Kbytes whereas a block is 16 Kbytes. In the process of downloading files from neighboring users, this protocol allows for downloading several pieces in parallel, using many neighbors at once. A user can either be a "lecher" or a "seeder" within the Bit Torrent protocol. A lecher is a user who has not completely downloaded the file associated with the tracker whom he is a part of. A seeder usually contains full version of the file and is sharing it with lechers.

In order to address the problems that Gnutella suffered due to users who did not participate in sharing files, this protocol allows users who upload files. Seeders in a periodic manner verify the upload rates of its neighbors and only share with those who also upload. The users who do not upload have slower download speeds or even

Not allowing them from downloading from that tracker.

D. Gnutella 0.6 - A hybrid approach

It consists of a hierarchical structure comprising of leaf nodes and ultra-peers. When a new peer wants to enter the system, it is kept as a leaf at the edge of the network. A leaf simply sends queries to its ultra-peers and it is not responsible for any routing. Peers are promoted to ultra-peers when they have high capacities and processing abilities and they are responsible for routing the queries using the Query Routing Protocol (QRP). All leaves connect to these ultra-peers. Each leaf connects to 3 ultra-peers, and each ultra-peer connects to more than 32 other ultra-peers. The maximum number of hops in the system is reduced to 4 which results in high scalability and search efficiency. When a leaf tries to connect to another leaf via http request to download a file from it, the presence of a firewall around the target leaf will drop this http request. In order to prevent this the ultra-peers of a leaf act as proxies for it known as PUSH proxies. So the leaf connects to its PUSH proxy instead of connecting to the target leaf, which in turn sends a PUSH request on behalf of the sender leaf to the target leaf. The target leaf then initiates a connection with the sender leaf. Query Routing Protocol: Each leaf will create its respective Query Routing Table (QRT) which contains a mapping of each filename (of the files present at the leaf) to its hashed keyword. All the leaves will send their QRTs to their ultrapure which will combine them all along with its own QRT (if it's sharing any file) and exchanges it with other ultra-peers it's connected to. Any ultra-peer receiving a

query will hash it to its keywords and check if all the words belong to its query. If yes then it's passed along to a leaf or a subordinate ultra-peer.

III. ATTACKS ON P2P NETWORKS

Since the P2P systems primarily rely on dependence of peers among each other, security implications develop from misshaping the trust between peers. In a client-server model, internal data might not be shown to the client, but with P2P, some inwards must be uncovered to associate peers in the name of disseminating the workload. Attackers can take advantage of this in compromising P2P networks.

A. Distributed Denial-of-Service

In a conventional denial-of-service (DoS) onslaught, a server is generally the main target of bulk connections, relinquish the server defective. A traditional example of this is a TCP SYN flood onset, in which the client directs the server a SYN message, the server replies with a SYN-ACK message, and the server expects an ACK message out of the client. Nevertheless, attacking the client does not respond with an ACK message, thus confining up server resources (memory) as it fruitlessly waits. In the meantime, the client can proceed to open many more of these new non-ACK'ed connections, bringing the server solely to its knees, and thus a denial-of-service to other legalized clients. In a P2P network, on slaughters can make use of the enquiring nature of P2P networks to overburden the network. In this case of the enquiry, flooding P2P network, attack is direct: simply sending a enormous number of queries to peers, and the implied broadcast storm will provide portions of the network defective.

In recent times, attacks can utilize the P2P network as an agent to onset some *other* target, such as web sites. Basically, peers in the network are overthrown, as described in the next section, to receive files from a target, consuming the victim with massive bandwidth usage. An instance of this kind of attack aroused in 2007 in the Direct Connect network with users utilizing the DC++ file sharing application.

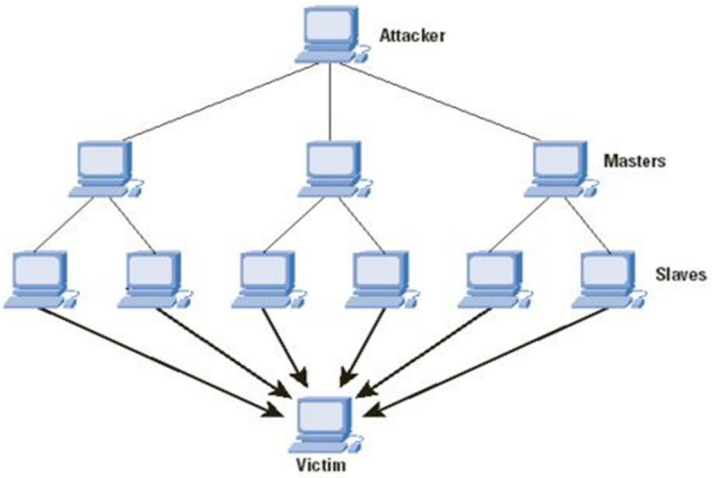

Figure 3

B. Spoiling the Network

A different approach towards on setting a P2P network is to insert useless data (poison) into the system. Since P2P networks necessarily introduce a lookup service in some manner, either it be a centralized directory or a DHT, an attacker can introduce a large amounts of futile lookup key-value sets into the index. Phony items in the index can degrade query times or, worse, produce irrational queries outcomes. Even DHTs are prone to this attack, but since DHTs have O(log n) lookup time, a massive amount of poison is needed. In fact, poisoning a P2P network has already been noticed on the Internet as giant publishing organizations endeavor to lower the potential losses of hijacked media by on setting the FastTrack P2P network.

Poisoning can also be utilized as fodder for DDoS attacks. This could be carried out in two ways, by index poisoning or route table poisoning. In index poisoning, fake records are injected into the index pointing to a target IP and port number. When a peer goes to explore a resource, it would acquire fake location information from a poisoned index, either from a central directory or from another peer. The requesting peer then makes a connection to the target, perhaps confusing the target or, if the target accepts the connection, a TCP-connection DDoS secures the outcome. In route table poisoning, the attack leverages the case that almost all P2P clients must

maintain some sort of routing state of the present peers with what it is connected. Especially in a DHT system, the route table of each peer consists of its *O(log n)* neighbors given *n* nodes in the network. The attacker cheats the peers into admitting bogus neighbors into each peer's route table, and in some situations, this as simple as devising an announcement directing at the target. The outcome is such that the target acquires a flood of connection requests, and the target would likely reject them. Typically, P2P protocols have a mechanism to remove stale peers from the routing table, updating it regularly. Hence, after the exploding of traffic to the target, the target is discarded from the route tables of connecting peers.

C. Privacy and Identity

P2P networks also represent privacy and identity issues. With respect to privacy, a peer's data stream may be accorded by neighboring peers who help in transmitting the data. A direct instance is that of VoIP applications, is of Skype, that route traffics in a P2P fashion. Even though the data stream is coded, a peer which transmits the stream that now has a direct access to data packets, which isn't the case in classical routing. Further some, Skype's encryption scheme is recovery, so there can be no verification that the method is completely secure. Also, obtain the nature of P2P applications is the openly partaking of private files. In a study, a minority were aware of the particular files that the user was sharing. In another survey on the Kazaa network, many peers were discovered to be sharing their financial, email, and web cache data unknowingly. Due to the easy use of file sharing applications, many users very well not be experienced enough to realize the privacy instructions of using a P2P application, and thus making the job of the attacker very easy.

In P2P networks which divide resources of doubtful legality, the issue of lack of privacy becomes transparent. For instance, the BitTorrent file sharing system directly exposes the IP address of peers to each other in a crowd. This would permit peers in the swarm to know the identity of other peers who are downloading certain resources, for example. Once the peer's identity is known, further attacks, whether legal or physical, can continue to be directed at that target.

D. Fairness in Sharing

Since P2P networks rely on the collaboration of its peers, a presumption is made that all peers would contribute to the resource distribution process. Nevertheless, as there is no real administrator, no authority in the system, peers are mostly free to freeload off other peers, this is called leeching and is looked down and considered as malpractice. While extremely common in classical P2P networks, including the IRC (Internet Relay Chat) network, leeching has been somewhat reduced in newer P2P applications. For example, in BitTorrent, a choke system is in place to control bandwidth to peers who do not upload a proper amount. Hence, the lechers are able to do it for a short amount of time before the other peers learn of their presence and refuse to cooperate with it or sharing with it at an increasingly slower rate.

E. Blocking of P2P Traffic

A predominant issue that hovers over P2P networks is throttling and blocking of P2P traffic. According to a 2007 Internet survey, 69% of Internet traffic in Germany is P2P, with HTTP way behind at 10%. Within P2P traffic, BitTorrent accounts for 67%, with the next highest being eDonkey at 29%. Given the astonishing amount of Internet traffic accounted by P2P applications, especially BitTorrent (from the numbers above, BitTorrent alone accounts for nearly 50% of the Germany's Internet traffic), it is not astonishing that ISPs are not allowing ports on which well-known file sharing applications run. For instance, Comcast recently introduced to smother and drop packets of Bit Torrent traffic, effectively cutting off its customers from running the software. Moving on, Ohio University newly started to cut off all P2P traffic on its campus. [5] As security issues with P2P are becoming increasingly wanton, recent efforts have been made to avoid some of the above liabilities by securing P2P networks.

IV. ENCRYPTION TECHNIQUES

There are two schools of thought when it comes to P2P encryption. First is

Encrypting user traffic in order to mask the P2P protocol from ISP's or other inquisitive eyes. Second is encrypted data shared to a neighbor and only decoding it when that neighbor uploads that data to its neighbors? This technique is used to prevail "free-loading" users.

Originally, in order to conceal traffic, users would encrypt their data and send between

Neighbors using a Diffie-Helman key exchange. [6] The original data being sent between two parties cannot be resolved while using this type of encryption, however, the type of data can still be figured out. Only encrypting the packets will allow the users further to

overcome what is called deep packet inspection, or DPI, which inspects the contents of the packets that are being sent, even though the flow of protocol can be still determined. Each and every type of data traffic being directed over the application layer of the internet has its own flow, i.e. where the data packets are going, where they are coming from, packet direction, packet size, and duration of communication. These properties can help inquisitive eyes to determine the type of data transmission occurring between two parties. A style that can be used is called Statistical Protocol Identification, or SPID which uses a statistical model to find out which protocol users are using to obtain information, like streaming, P2P, or VoiP. Users cannot only use encoding their traffic, they also have to disguise their data flow.

The authors explain the steps essential to change a BitTorrent client by allowing

Encryption plus flow of stupor. The first step would be to use a shared arbitrary secret so that it can encrypt the packet information which will fool any DPI systems. Next, in order to lessen the statistical likelihood of convincing the BitTorrent protocol flow, the authors introduce a new message type known as the padding message. Adding these new messages into the data stream hikes the packet variation, helping to retort SPID.[7] The third alteration, incidentally flushes, further increasing the packet variation and packet size, adding to the countering of SPID.

Atlas, a magic peer ID, is used in order to signal the sharing neighbors that the transmission will be encoded and complicated. If the sharing neighbor cannot support obfuscation then the connection is aborted and they will have to connect again and use plain text to complete their sharing. This suggested method will not result in complete concealment due to the frequency at which packets are sent are not modified and can be used to determine protocols nonetheless, this method does help in hiding the protocol type from most methods of protocol recognition approaches.

In a method called “Trick before Treating” it is suggested that it would drop the number of “free-riders” within a BitTorrent network by encrypting all data that is uploaded. If the massive number of users within a network are free-riders, users who do not upload any data, then the overall “capacity per user” is decremented. The Trick before treating method enforces all seeders to encrypt their file pieces when they upload them to other users. In order for the users to download these pieces will be needed to decrypt them, they have to be uploaded those pieces to other neighbors, who in turn give them a subkey as can be seen in Figure 4. The original seeder would give each leecher who downloads a file piece a sub key. There would be k sub keys and a user would need n sub keys to decrypt the file pieces. In order for users to obtain the whole file and decrypt it, they would have to upload their own file pieces to other users, forcing them to contribute within the network, eliminating a large portion of free riders.

Comparative Analysis and Secure ALM P2P Overlay Multicasting of Various Multicast Routing Techniques

Multicasting is the transfer of information or a message to a group of destination systems concurrently in a single transmission from the source. The transcripts of the messages are mechanically created in other network components like routers only when the network topology requires it. Multicast is enabled mostly in IP multicast which furthermore could be introduced in internet protocol applications of continuous media. it enables the flow of secure data between sender and reciever and vice-versa.

V. MULTICASTING TECHNIQUES

The following are various multicasting techniques we are going to discuss:

A. IP Multicasting

Multicast is a transmission pattern in which the Sender sends data to all the interested

Hosts of a particular group. There are some certain protocols which we need to know for the better understanding the entire structural design of multicast network. In a normal multicast network there is a sender which is attached to a router which needs to know its RP addresses. And on the receivers side it collects the data from those host’s which are interested in receiving data from the source. The host is connected to RP through a router. RP connects the source and host for transfer of data in form of

Packets and it maintains a track of the no. of hosts entering and exiting the network. The domain of Addresses between 224.0.0.0 – 224.0.0.225 is particularly reserved for the utilization of routing protocols and other lower level topology or maintenance protocol. Multicast routers must not forward any of the multicast datagrams with host's address in this domain.

Whenever any host is keen in receiving information from sender, it needs to know its multicast group addresses using MSDP. The sender then passes on data to first-hop router via IGMP. Then the router forwards the data to the RP which consists of data about all the feasible sources for that certain group. After receiving transferred data from the source RP copies the data and passes the copies to all the concerned receivers using the MFIB.

1) Multicast Addressing: An IP multicast group is recognized by class D address. The capacity of the Class D multicast address

space is hence 228 or 268,435,456 multicast groups. The first four bits of the address are constant and the rest 28 bits change. Thus, all the multicast addresses begin with "1110".

Fig2: Multicast Addressing

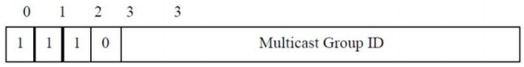

Figure 4

Multicast groups are recognized by the IP addresses in domain 224.0.0.0 – 239.255.255.255. There are some reserved and specific addresses out of the above given range that are utilized for distinct purposes.

224.0.0.1 All systems on this subnet

224.0.0.2 All routers on this subnet

224.0.1.1 NTP (Network time Protocol)

224.0.0.9 RIP-2 (Routing Protocol)

a. The domain of Addresses between 224.0.0.0 – 224.0.0.225 is fully reserved for routing protocols and other lower level topology maintenance or discovery protocol. Multicast routers must not forward any of the multicast datagram with receivers address in this range.

b. The domain of Addresses between 224.0.1.0 – 238.255.255.255 is used widely among internet for all the transmissions.

c. The domain of Addresses between 239.0.0.0 – 239.255.255.255 is utilized for the administrative uses.[7]

B. ALM Multicasting

Now a days, the preparation of network-layer multicast has not been widely accepted by most of the commercial ISP's due to the high cost of execution involved in the preparation of routers, and hence bulk parts of the Internet still are not capable of utilizing multicast applications. ALM protocols do not alter the network framework, rather they carry out multicast forwarding process solely at end hosts via ALM. basically it can be outlined as the execution of multicasting as the application service rather than network service.

C. Overlay Multicasting

It's a broad concept but is very effective and gained recognition as one of the

Procedure to overcome the obstacles of implementation to the router level results for different networks issues. The overlay results for multicasting contains the content sharing and content distribution which are studied extensively recent times. There are many number of overlay application-layer multicast proposes which have been

Approached over the last few years and the modifications are still going on for improved performances. In this multicast the hosts participation in the multicast period form an overlay network which uses only unicasts among the sets of hosts for the information dissemination and also the senders in this multicast handle routing, group management, and tree construction, solely, without any assistance from the Internet routers. The overlays primarily impose a functioning penalty over router-level options as the data packet forwarding primarily occurs at the hosts end. Also this multicast intakes more network bandwidth and also adds latency past IP multicast, so very little attention is paid to justify this overlay performance penalty. The application-level overlays do acquire performance debasement over router level results.

VI. PERFORMANCE METRICS

A. Multicast Efficiency

This will let us know which of the three multicasting techniques will multicast more effectively to all of the members of the multicast group.

B. Ease of Deployment

This will tell us which of the multicasting technique is easy to implement with accept to the environment around.

C. Complexity and Overhead

More Complex the technique, more is the difficulty to implement it primarily in the wireless environment. Same way if there is

Volume 4 Issue IV, April 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

more overhead then it lowers the performance right away.

D. Maintenance

This measure is to know which of the techniques is easy to handle or maintain. The easier to maintain a technique, easier to obtain effective performance results

E. Adaptability

There are lot of factors involved in wireless medium on which the whole communication depends. Sometimes there might be a heavy traffic so a technique must be highly adaptable so that it can adapt accordingly and yet be able to give out better results.

F. Robustness

The wireless medium is completely of uncertainties. One time the communication is going well and in the next moment there are some errors or failure in the network. Thus the technique being used for the multicasting must be reliable such that it can take care of such failures. [9]

G. Data packet Forwarding

This will let us know where in each of the mentioned techniques the data packet forwarding will happen. It is another important metric as it will inform us which multicast technique uses less time to wrap or un-wrap the data packets and hence improves the performance.

Performance Metrics	IP Multicasting	ALM	OM
1.Multicast Efficiency	High	Low	Medium
2.Ease of Deployment	Difficult	Easier	Medium
3.Complexity & Overhead	Low	High	Medium
4.Maintenance	High	Low	Low
5.Adaptability	Low	High	High
6.Robustness	High	Low	High
7.Data Packet Forwarding	At Routers	End Hosts	End Hosts

Tab. 1: Comparison of Multicasting Techniques

Figure 5

VII. CONCLUSION

After going through different papers examining about the various attacks and challenges faced by the P2P network it has been noticed that the lack of security in the network is due to the lack of central administration. To overcome these attacks there is much requirement to use encryption and multicasting techniques.

After the comparative analysis study we came to know that the multicast efficiency of IP multicasting is tremendous than that of the others with face detection. The implementation of ALM is easier than the others and also the complexity of IP multicasting is low than that of its counter parts and it also possess lower overheads than ALM and overlay multicasting. It also shows that ALM and OM are highly adaptive than IP multicasting.OM and IP multicasting are highly robust than ALM. So we conclude that when we design a routing protocol we need to keep in mind of all the metrics that result in improved efficiency of performance.

VIII. ACKNOWLEDGEMENT

We sincerely thank our professor Manjula R for providing us with this opportunity to write this paper. And also for supporting and guiding us whenever required. We also thank our university for providing us with the facilities required to complete the paper.

REFERENCES

[1] Security Issues of Reputation Management Systems for Peer-to-Peer Networks
http://www.sciencedirect.com/science/article/pii/S1574013712000123
[2] Napster, http://www.napster.com.
[3] The gnutella protocol specification http://www.gnutella2.com.
[4] SECURITY THREATS IN PEER TO PEER NETWORKS
http://jgrcs.info/index.php/jgrcs/article/viewFile/100/100
[5] [Peer07] "Peer-to-peer." Wikipedia. 2007. http://en.wikipedia.org/wiki/Peer-to-peer
[6] Analysing and classification of security issues of p2p networks
http://www.academia.edu/8991912/Analyzing_Classification_and_Security_Issues_of_Peer_to_Peer_Networks
[7] Peer-to-Peer Network Security Issues
http://www.cse.wustl.edu/~jain/cse571-07/ftp/p2p.pdf
[8] A Survey of Peer-to-Peer Security Issues
http://www.eecs.harvard.edu/~mema/courses/cs264/papers/securitySurvey-swSecurity2002.pdf
[9] PEER-TO-PEER NETWORKS AND COMPUTATION: CURRENT TRENDS AND FUTURE PERSPECTIVES
http://www.cai.sk/ojs/index.php/cai/article/viewFile/184/155
[10] Analysis and Secure ALM P2P Overlay Multicasting
http://www.ijerd.com/paper/vol11-issue3/Version_1/I1137078.pdf
[11] P2P Network Security-Logan Washbourne
http://arxiv.org/ftp/arxiv/papers/1504/1504.01358.pdf