VOL. 10, NO. 13, JULY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

SUITABLE ROUTING PATH FOR PEER TO PEER FILE TRANSFER

R. Naga Priyadarsini, S. Suma and V. Dhanakoti
Department of Computer Science Engineering, Valliammai Engineering College, Kanchipuram, India

ABSTRACT

Peer-to-peer (P2P) computing or networking is a distributed application architecture. The tasks and workload are partitioned between multiple peers. Peers are equally privileged, equipotent participants in the application. P2P file sharing is the distribution and sharing of files using P2P networking technology. The peers of such networks are end-user computer systems that are interconnected via the internet. In this paper we develop a systematic methodology to identify P2P nodes, perform suitable cluster operation to transmit files between various peers. In-case of any fault nodes identified in the transmission path suitable alternate path must be identified so that the file can reach the destination safely. Finally the packets are transmitted to the destination by means of suitable path and there the packets are collected and decrypted. The results are obtained based on the moving nodes also.

Keywords: P2P, routing, network analysis, clustering, encryption, decryption.

1. INTRODUCTION

A peer-to-peer network is a network in which any node in the network can act as both a client and a server. Over the last few years, peer-to-peer (P2P) file-sharing has relentlessly grown to represent a formidable component of Internet traffic. P2P volume is sufficiently dominant on some links to incent increased local peering among Internet Service Providers [1], to observable yet unquantified effect on the global Internet topology and routing system not to mention competitive market dynamics. P2P networking refers to virtual networks of computers that replace the distinct notions of server and client nodes with the notion of peers. Despite huge differences among peers with respect to processing, connection speed, local network configuration or operating system, each member of the P2P network has the same functionality at the application layer. This peering functionality is in contrast to traditional network systems such as DNS where there is a clear distinction between the operations performed by each node. The absence of centralized authorities in P2P networks results in a totally distributed configuration of directly connected peers. Some P2P networks also have a small set of special nodes that usually handle queries.

The main application of such networks is file sharing among users. While P2P networks became popular only during the last few years, the concept of P2P networking was introduced early in the evolution of network communication systems. In fact both ARPANET in the late ‟60s and Usenet in the late 80‟s are in a sense early predecessors of today’s P2P networks; they were distributed, decentralized networks intended for file transfer and sharing among equal peers. With the dramatic growth of the Internet in the early ‟90s, the popularity of the world-wide web somewhat displaced use and development of P2P networks. However, a series of technological developments lead to the explosion of P2P applications. First, the MPEG Audio Layer-3 (i.e., the popular mp3) encoding (1995 [2]) which facilitated huge data compression gains, accompanied by the release of free mp3 players, pervasively available by 1997 (e.g., winamp [3]). Encodings that offered considerable reduction for video data were also developed later (e.g., DivX [4] in 1999). Second, the increase of available bandwidth to end users with broadband technologies that provided inexpensive high-speed Internet access. Third, the pivotal Napster network [5] fielded in 1999 revolutionized file sharing, even though Napster was technically a hybrid-P2P rather than a pure P2P network since it retained the notion of a server for indexing content of the peers. Despite this dramatic growth, reliable profiling of P2P traffic remains elusive. We no longer enjoy the fleeting benefit of first generation P2P traffic, which was relatively easily classified due to its use of well-defined port numbers.

2. RELATED WORKS

Many peer-to-peer networks have been proposed for different applications in the literature; see, for example, [6], [7], [8], [9], [10], [11], [12], and [13]. In this paper, we focus on peer-to-peer networks for efficient distributed data (file) sharing among peers.

Gnutella [11] is a decentralized unstructured peer-to-peer network. The network is formed by peers joining the network following some loose rules. There is no constraint on the network topology. To look up a data item, a peer sends a flooding query request to all neighbors within a certain radius. As Gnutella has no requirement on the network topology and data placement, it is extremely resilient to peer joining and leaving the system frequently. However, flooding is not scalable and consumes a lot of network bandwidth. Also, it is difficult to find a rare data item as it has to flood the query request to most of the peers.

Bit Torrent [12] is a centralized unstructured peer-to-peer network for file sharing. A central server called tracker keeps track of all peers who have the file. Each file has a corresponding torrent file stored in the tracker which contains the information about the file, such as its length, name, and hashing information. When receiving a download request, the tracker sends back a random list of peers which are downloading the same file. When a peer has received the complete file, it should stay in the system for other peers to download at least one copy of the file from it. Since Bit Torrent uses a central server to store all the information about the file and the peers downloading the file, it suffers called "single point of failure" problem which means that if the central server fails, the entire system is brought to a halt. Note that in some literatures, hybrid peer-to-peer networks were used to refer to the centralized peer-to-peer systems such as Bit Torrent.

Unlike previous work that proposed new centrally coordinated mechanisms [14] and new pricing mechanisms to incentivize uncoordinated p2p schemes [15], we study the effectiveness of using the popular BT algorithm for file distribution. While the performance of BT has been studied extensively as a file-sharing protocol [16], [17], [18], to the best of our knowledge, we are the first to study the performance of BT as a file distribution protocol. In file distribution, the system provides a server (BT seed) as a constant source of data content for clients that download a file and clients generally leave the system on download completion.

3. PROPOSED SYSTEM

The proposed architecture Figure-1 shows the various stages such as P2P node detection, P2P clustering using K mean clustering, Geo routing methodology Data Encryption and decryption. Firstly, analyzing the network streams by P2P-nodes detection algorithm, we can get the sets of P2P-nodes. Secondly, each P2P application has its own typical P2P protocol and the nodes in one P2P application have exchanged data frequently, the P2P-nodes clustering algorithm analysis the network streams of P2P nodes, stats the symmetry, quantity and frequency of the data exchanged between each pair of nodes, and clusters a P2P application based on K-mean clustering algorithm. The path to transfer the data must be chosen in such a way that it must not contain any fault node. The behavior of the moving nodes can also be analyzed. Finally using suitable path the packets are transmitted and are decrypted on the receiver side.

a) P2P Node detection

Any system or device connected to a network is also called a node. Each device on the network has a network address, such as a MAC address, which uniquely identifies each device. In order to get more information from the network as soon as possible, the P2P-nodes would create connections with the other nodes as many as possible, which basically exhibit the characteristic of paroxysm. Furthermore, because the P2P nodes are decentralized, each node would have connected to much more subnet and network nodes than the common nodes.

The average of the connections can be performed by

$$\overline{N_s} = \frac{1}{m}\sum_{k=1}^{m} N_{sk} \qquad (1)$$

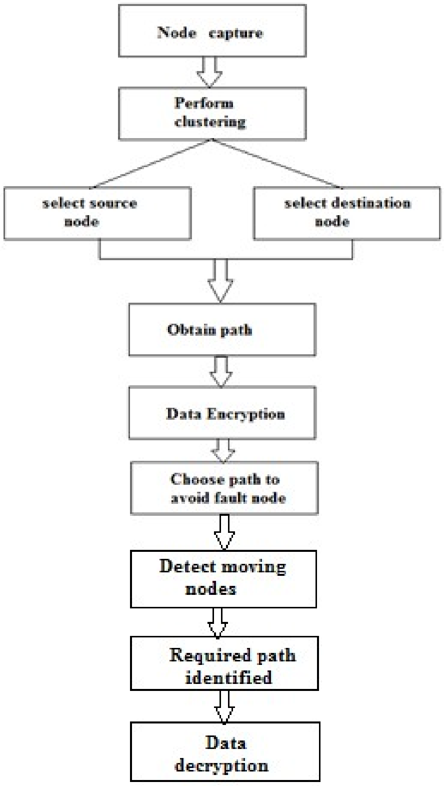


Figure-1. System flow Architecture.

b) Node clustering

K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as bar centers of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function.

In a T period, there are m times sampling. For each network node S, cumulates the connections at each sampling time, and form a collection {Ns_1Ns_2, Ns_m}. Ns refers the the number of connections of node S.

$$| = \text{[illegible]} \; ^{2} \qquad (2)$$

Where '$||x_i{}^j$- $c_j||$' is the Euclidean distance between x_i and c_j,'n' is the number of data points in i^{th} cluster and 'k' is the number of cluster centers.

c) Data encryption

Blowfish algorithm is used to perform encryption as well as decryption. Blowfish symmetric block cipher algorithm encrypts block data of 64-bits at a time. It is suitable for applications where the key does not change often, like communication link or an automatic file encryptor. It will follow the feistel network and this algorithm is divided into two parts such as Key-expansion and Data Encryption.

Blowfish is designed in consideration with,

- **Fast:** It encrypts data on large 32-bit microprocessors at a rate of 26 clock cycles per byte.
- **Compact:** It can run in less than 5K of memory.
- **Simple:** It uses addition, XOR, lookup table with 32-bit operands.
- **Secure:** The key length is variable, it can be in the range of32~448 bits: default 128 bits key length.

d) Geo routing methodology

Geographic routing also known as position-based routing is a routing principle that relies on geographic position information. It is mainly proposed for wireless networks and based on the idea that the source sends a message to the geographic location of the destination instead of using the network address. Geographic routing requires that each node can determine its own location and that the source is aware of the location of the destination. With this information a message can be routed to the destination without knowledge of the network topology or a prior route discovery.

4. RESULTS AND DISCUSSIONS

The nodes are obtained for which the connection is to be established by using equation (1). Figure-2 shows that nodes are created and then the connection is established between all the nodes by means of clustering algorithm.

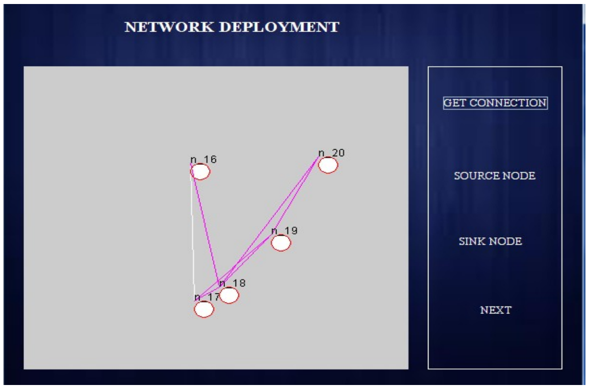
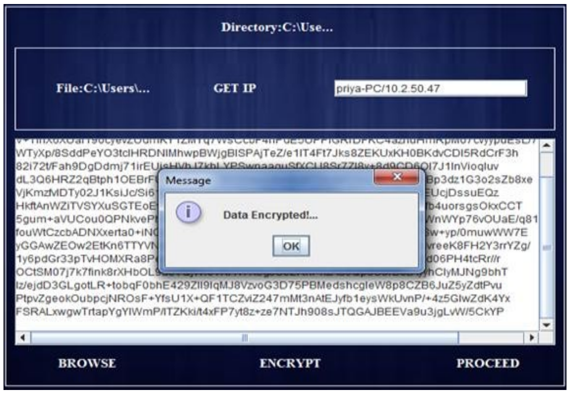


Figure-2. Connection between nodes.

Followed by the connection establishment source and destination node can be selected to transfer the file between them by dividing them into multiple packets. The file to be sent is selected and encryption is performed as shown in Figure-3.



File:C:\Users\...

priya-PC/10.2.50.47

```
WTyXp/8SddPeYO3tclHRDNIMhwpBWjgBISPAjTeZ/e1IT4Ft7Jks8ZEKUxKH0BKdvCDI5RdCrF3h
82i72t/Fah9DgDdmj71irEU                                          I7J1lnVioqluv
dL3Q6HRZ2qBtph1OEBrF                                          Bp3dz1G3o2sZb8xe
VjKmzMDTy02J1KsiJc/Si6                                          EUcjDssuEQz
HkftAnWZiTVSYXuSGTEoE                                          b4uorsgsOkxCCT
5gum+aVUCou0QPNkveP                                            WnWYp76vOUaE/q81
fouWtCzcbADNXxerta0+iNC                                         Sw+yp/0muwWW7E
yGGAwZEOw2EtKn6TTYV                                            vreeK8FH2Y3rrYZg/
1y6pdGr33pTvHOMXRa8P                                            d06PH4tcRr//r
OCtSM07j7k7fink8rXHbOL                                          hClyMJNg9bhT
Iz/ejdD3GLgotLR+tobqF0bhE429ZlI9IqMJ8VzvoG3D75PBMedshcgleW8p8CZB6JuZ5yZdtPvu
PtpvZgeokOubpcjNROsF+YfsU1X+QF1TCZviZ247mMt3nAtEJyfb1eysWkUvnP/+4z5GlwZdK4Yx
FSRALxwgwTrtapYgYIWmP/ITZKki/t4xFP7yt8z+ze7NTJh908sJTQGAJBEEVa9u3jgLvW/5CkYP
```

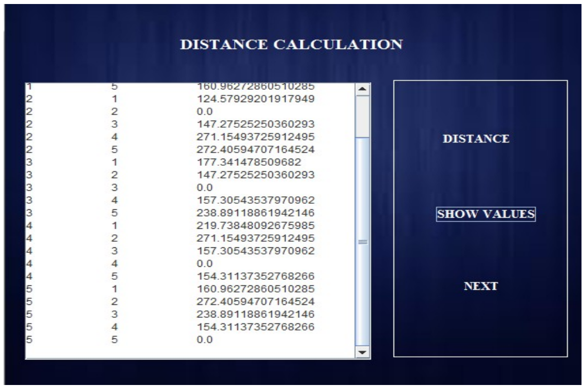
Message

Data Encrypted!...

OK

Figure-3. Data encryption.

The distance between each and every node is calculated based on the clustering algorithm and their values are obtained as shown in Figure-4.

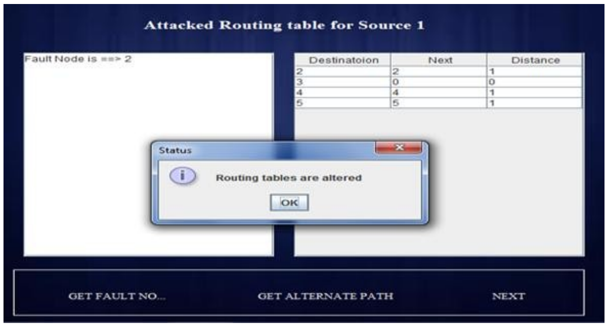


```
1          5          160.96272860510285
2          1          124.57929201917949
2          2          0.0
2          3          147.27525250360293
2          4          271.15493725912495
2          5          272.40594707164524
3          1          177.341478509682
3          2          147.27525250360293
3          3          0.0
3          4          157.30543537970962
3          5          238.89118861942146
4          1          219.73848092675985
4          2          271.15493725912495
4          3          157.30543537970962
4          4          0.0
4          5          154.31137352768266
5          1          160.96272860510285
5          2          272.40594707164524
5          3          238.89118861942146
5          4          154.31137352768266
5          5          0.0
```

Figure-4. Distance calculation.

Fault node can be obtained based on the distance. The alternative path to send the packets must be provided.

From the routing table obtained best path must be selected to transmit the data as shown in Figure-5.



Attacked Routing table for Source 1

Fault Node is ==> 2

Destinatoion	Next	Distance
2	2	1
3	0	0
4	4	1
5	5	1

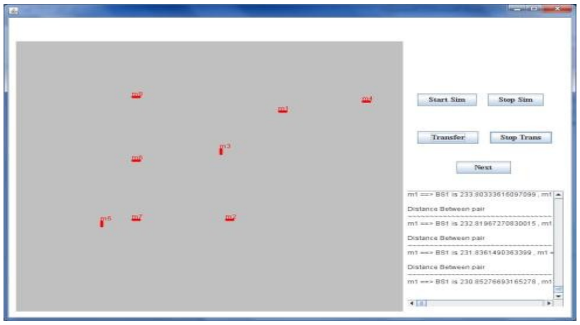
Status

Routing tables are altered

OK

Figure-5. Alternate path.

All the above results shows when the node is still. Figure-6 shows when the nodes are in moving state. The distance travelled from source to destination by each node is calculated and the exact location of the particular node can also be obtained. In case of moving nodes the packets can be transmitted to the destination based on the angle of arrival. The track of each and every movement of the nodes can be analyzed.

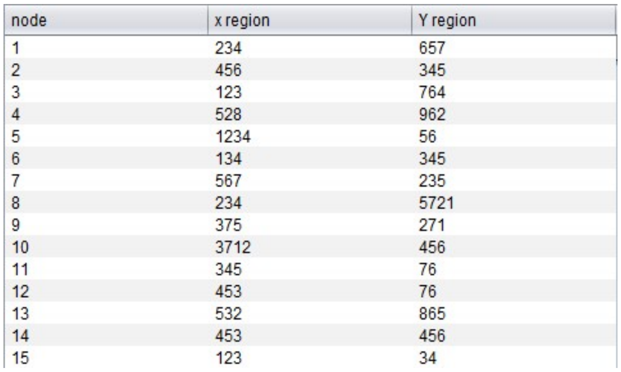


```
m1 ==> BS1 is 233.80333616097099 , m1
Distance Between pair
m1 ==> BS1 is 232.81967270830015 , m1
Distance Between pair
m1 ==> BS1 is 231.8361490363399 , m1 -
Distance Between pair
m1 ==> BS1 is 230.85276693165278 , m1
```

Figure-6. Moving nodes.

Table-1 indicated the location of each and every node in the network by specifying its exact x and y region. In case when the behavior seems to be any changes then the node can be analyzed easily.

Table-1. Node location.



node	x region	Y region
1	234	657
2	456	345
3	123	764
4	528	962
5	1234	56
6	134	345
7	567	235
8	234	5721
9	375	271
10	3712	456
11	345	76
12	453	76
13	532	865
14	453	456
15	123	34

The comparison between existing and proposed is calculated. The result obtained shows that proposed system has the highest accuracy level. The alternate path provides the good accuracy level which is shown in Figure-7.

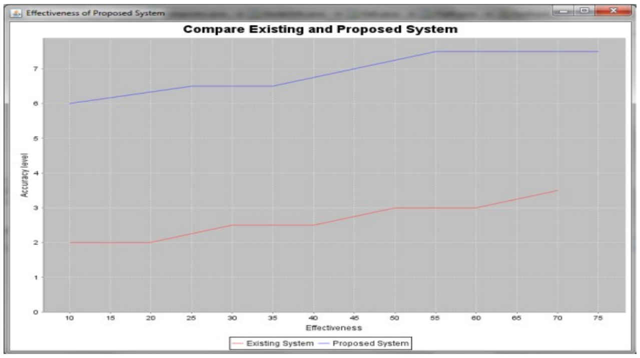


Figure-7. Comparison for existing and proposed system.

5. CONCLUSIONS

In this paper, node detection based on P2P node detection algorithm, P2P node clustering algorithm geo routing methodology and data encryption and decryption techniques are explained. The comparison shows the better result for the alternate path than the original path. In case of moving nodes the location are identified and the packets are transmitted based on the angle of arrival. Finally the file reaches the destination node in more secure and shortest way and can be decrypted to figure out the exact file.

6. FUTURE WORK

Followed by the alternative path, the behavior of the node can be analyzed. Incase if there is any presence of attackers or change in any behavior, it can be identified and eliminated. By doing this the file transfer can be done in more secure way.

REFERENCES

[1] W. B. Norton. 2003. The evolution of the U.S. internet peering ecosystem. http://www.equinix.com/pdf/whitepapers/PeeringEcosystem.pdf.

[2] Fraunhofer IIS. http://www.iis.fraunhofer.de/amm/techinf/layer3/index.html.

[3] WINAMP. http://www.winamp.com/.

[4] DivXNetowks. http://www.divxnetworks.com/.

[5] Napster. http://www.napster.com/.

[6] E.K. Lua, J. Crowcroft, M. Pias, R. Sharma, S. Lim. 2005. “A Survey and Comparison of Peer-to-Peer Overlay Network Schemes”, IEEE Comm. Surveys and Tutorials, Vol. 7, No. 2, pp. 72-93.

[7] E. Brosh and Y. Shavitt. 2004. “Approximation and Heuristic Algorithms for Minimum Delay Application-Layer Multicast Trees”, Proc. IEEE INFOCOM ‟04.

[8] A.R. Bharambe, S.G. Rao, V.N. Padmanabhan, S. Seshan, and H.Zhang. 2005. “The Impact of Heterogeneous Bandwidth Constraints on DHT-Based Multicast Protocols,” Proc. Ann. Int‟l Workshop Peer-to-Peer Systems (IPTPS ‟05), pp. 115-126, February.

[9] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek and H. Balakrishnan. 2003. “Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications,” IEEE/ACM Trans. Networking, vol. 11, no. 1, pp. 17-32, February.

[10] V. Vishnumurthy and P. Francis. 2006. “On Heterogeneous Overlay Construction and Random Node Selection in Unstructured P2P Networks", Proc. IEEE INFOCOM ‟06.

[11] Gnutella Development Forum, The Gnutella v0.6 Protocol, http://groups.yahoo.com/group/the gdf/files/, 2009.

[12]Bittorrent 2009. http://www.bittorrent.com/.

[13] P. Ganesan, Q. Sun and H. Garcia-Molina. 2003. “YAPPERS: A Peer-to-Peer Lookup Service over Arbitrary Topology,” Proc. IEEE INFOCOM ‟03, pp. 1250-1260.

[14] R.S. Peterson and E.G. Sirer. 2009. Antfarm: Efficient Content Distribution with Managed Swarms," in Proc. USENIX Symp.Netw. Syst. Design Implement., pp. 107-122.

[15] V. Misra, P. Barford and M.S. Squillante. 2010. „„„Incentivizing Peer-Assisted Services: A Fluid Shapley Value Approach,‟‟‟ ACM SIGMETRICS Performance Eval. Rev., Vol. 38, No. 1, pp. 215-226.

[16] B. Fan, J. C. S. Lui and D.-M. Chiu. 2009. The Design Trade- Offs of BitTorrent-Like File Sharing Protocols,‟‟‟ Trans. Netw., Vol. 17, No. 2, pp. 365-376.

[17] D. Qiu and R. Srikant. 2004. Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks,‟ in Proc. ACM Conf. Appl., Technol., Architectures, Protocols Comput. Commun. pp. 367-378.

[18] F. Simatos, P. Robert and F. Guillemin. 2008. “A Queueing System for Modeling a File Sharing Principle”, ACM SIGMETRICS Performance Eval. Rev., Vol. 36, No. 1, pp. 181-192.