Abstract:

A peer-to-peer network is a network created whenever two or more devices (usually a computer) are connected and share the resources. In a p2p network, each computer acts as both a server and a client—supplying and receiving files. The nodes (devices) are linked to each other in an overlay network. This overlay can be structured or unstructured. In this project, we'll focus on unstructured networks. Gnutella, Gossip are some examples of P2P protocols to implement unstructured networks. We will focus on the Gnutella Protocol to implement our own peer-to-peer network with a file-sharing feature. Some applications of P2P are Content Delivery, MultiMedia, Bitcoin etc…,

Gnutella:

- Each peer can ping and send packets to their neighbours. If the neighbouring peer sends back a pong packet, they are connected.
- A peer can act as both Client and Server - requesting and sending files.
- If a peer requests a file, a ping packet with the filename is sent to its neighbours.
- If any one of the neighbours contains the file, they send back a pong packet and then the file transfer happens.
- If a neighbouring peer does not contain the file, they forward the packet to their neighbours and then if it's a QueryHit, file transfer happens.

Features:

- Each peer runs a ping thread (client thread) and a pong thread (server thread)
- Ping Thread - sends out the ping packet to its neighbours
- Pong Thread - manages the incoming packets to the peer and responds to them based on various conditions and send/receive files.
- Packet - contains source ip, destination ip, needFile flag, available flag, packet ID

In the following example, 3 peers are connected in the network

Peer 1 -> Peer 2

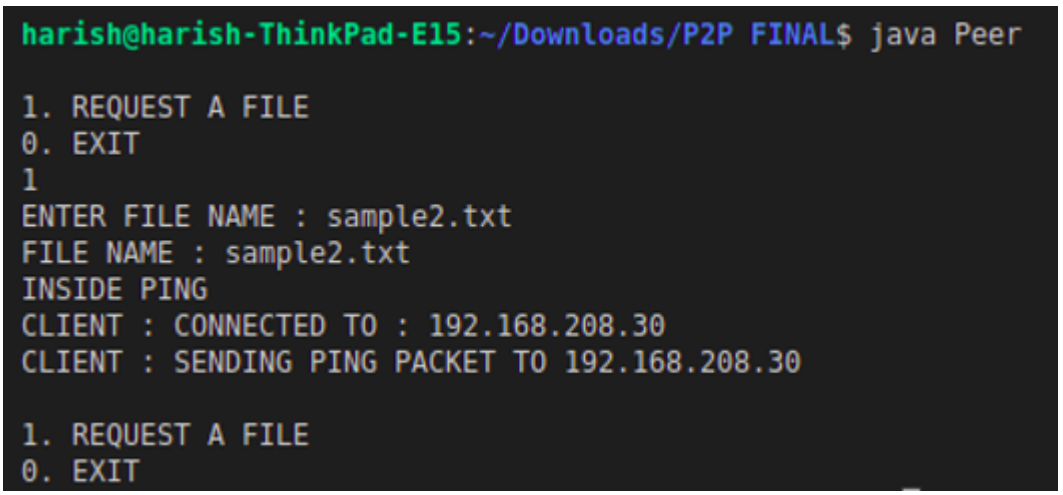
Peer 2 -> Peer 3

Peer 1 requests a file sample2.txt which resides in Peer 3. The ping packet is first received by Peer 2. As Peer 2 does not have the file, forwards the packet to Peer 3. Peer 3 sends back a packet to Peer 1 with available flag set and then file transfer happens.

OUTPUT:

Peer 1:

Peer Thread



Server Thread

```
harish@harish-ThinkPad-E15:~/Downloads/P2P FINAL$ java Server
INSIDE PONG
SERVER : CONNECTED TO /192.168.208.102
RECEIVED A PING PACKET
SERVER : CONNECTED TO : 192.168.208.140
SERVER : SENDING A NEED FILE PACKET
FILE DOWNLOADED : sample2.txt
```

Peer 2:

Peer Thread

```
C:\Users\LENOVO\Downloads\P2P>java Peer

1. REQUEST A FILE
0. EXIT
```

Server Thread

```
C:\Users\LENOVO\Downloads\P2P>java Server
INSIDE PONG
SERVER : CONNECTED TO /192.168.208.140
RECEIVED A PING PACKET
SERVER : CONNECTED TO : 192.168.208.140
SERVER : SENDING A NEXT PEER PACKET
```

Peer 3

Peer Thread

```
PS C:\Users\User\Downloads\P2P FINAL> java Peer

1. REQUEST A FILE
0. EXIT
```

Server Thread

```
PS C:\Users\User\Downloads\P2P FINAL> java Server
INSIDE PONG
SERVER : CONNECTED TO /192.168.208.30
RECEIVED A PING PACKET
SERVER : CONNECTED TO : 192.168.208.140
SERVER : SENDING A FILE FOUND PACKET
SERVER : CONNECTED TO /192.168.208.140
RECEIVED A PING PACKET
SERVER : CONNECTED TO : 192.168.208.140
SENDING FILE LENGTH : 6 BYTES
```