LNDP: Local Network Protocol
An Extension to BitTorrent Protocol

Dhruv Patel

May 25, 2016

1 LNDP: Local Network Download Protocol

Before describing protocol lets define few terms.

Torrent A torrent can mean either a .torrent metadata file or all files described by it, depending on context. The torrent file contains metadata about all the files it makes downloadable, including their names and sizes and checksums of all pieces in the torrent. It also contains the address of a tracker that helps to locate other peers downloading same torrent.

Swarm Together, all peers sharing a torrent are called a swarm.

Local Swarm Together, all peers sharing a torrent and having same public IP address are called Local Swarm.

LNDP Link LNDP link is a TCP connection between two peers who are communicating via LNDP.

1.1 Connection

Consider the topology shown in figure 1(on page 2). Here each computer will have different private IP address, but for this document lets identify each computer with letter of alphabet. Note that in this figure only computers that are downloading same file are shown. Other computers may exist, but we don't care about them. Computers that have identifier far in alphabet denotes that, that computer joined the local swarm later. i.e. A will be the one who started local swarm, then B joined it and then C. In the figure 1, D is the last one to join.

Now someone(in figure 1 green laptop) starts downloading a torrent file using client that support LNDP. Once he starts downloading it, he broadcasts a message asking if anybody is downloading the same torrent.[1] A

[1]The communication protocol using UDP is not described here. You may read the source code of POC available at https://github.com/DhruvPatel01/dtoc to get an idea how it can be implemented.

subset of local swarm responses to that broadcast. Response contains id of local swarm(it will be 20 bytes id of torrent client of first peer, here A),IP address and port of last peer and size of local swarm.

The subset that responses includes all peers that are connected to last peer and last peer itself(see red peers in figure 2). This is done because broadcast will be done using UDP, and UDP doesn't guarantee delivery.

If client doesn't receive response(s) in some time, it will timeout. After timeout it will retry. After few retries it will assume that it's the only one downloading that torrent. So it will initialize variables like count(number of peers in local swarm) and local swarm id, and starts listening for UDP broadcast.

If client does receive responses that have different group ids, client will choose swarm with bigger size and it will send special RESET message via UDP to peer(s) who had different group id. Whether to handle RESET or not is up to receiver.

Once client have decided a swarm to join, it creates a TCP connection with last peer of that swarm. On that TCP connection they talk BitTorrent, more specifically they talk extended BitTorrent using libtorrent extension protocol(LTEP).[2]

Clients that supports LTEP, must support one additional message on the top of BitTorrent wire protocol with id equals 20. The format for this extended message is as shown in figure 3.

After creating TCP connection and handshakes(BitTorrent, LTEP) and determining that both clients support LNDP, LNDP handshake is done. And through LNDP handshake a new client is given an integer number(here it will be 4). Once the handshake is done a new client is the last peer. Old last peer will broadcast a message saying to update last client and increment swarm size.

Note that this broadcast message is not UDP broadcast. This is LNDP broadcast. LNDP broadcast is done in following way. Each client will have a queue of limited size. Once it receives a broadcast message, it checks whether it is in queue. If it is not in queue it appends message to queue,

[2] `www.libtorrent.org/extension_protocol.html`

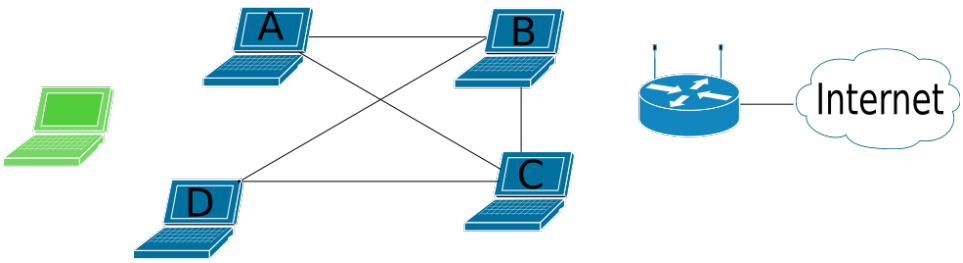

Figure 1: New peer joining the local swarm.

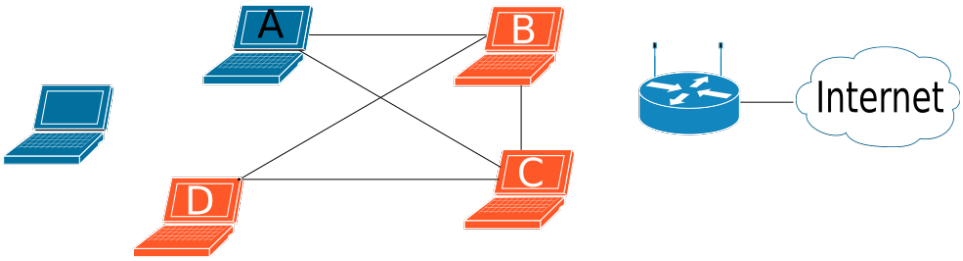

Figure 2: Subset of local swarm that responses.

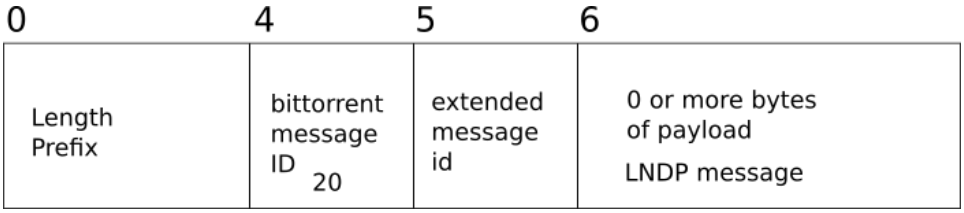

Figure 3: LTEP Format

and sends to all other links with whom it have active LNDP connection. If the message is already in queue, it does nothing. For example in figure 5, after handshake completes D generates a broadcast message and sends to all its outgoing LNDP links except to E. In figure 6, B and C receives a broadcast. This message is not in queue of either B or C. So B and C takes appropriate actions and forwards to all outgoing links. C sends message to B and A. Because message is in queue of B, B does nothing. After a while everybody in the local swarm will know about new Peer. Now that peers know about E, they may initiate TCP connection with E.

1.2 Downloading Torrent

The whole purpose of using LNDP is that no peer in local swarm downloads same piece from the Internet. How this is done in LNDP? While discussing this topic keep in mind figure 7. In this figure we also show the id of each

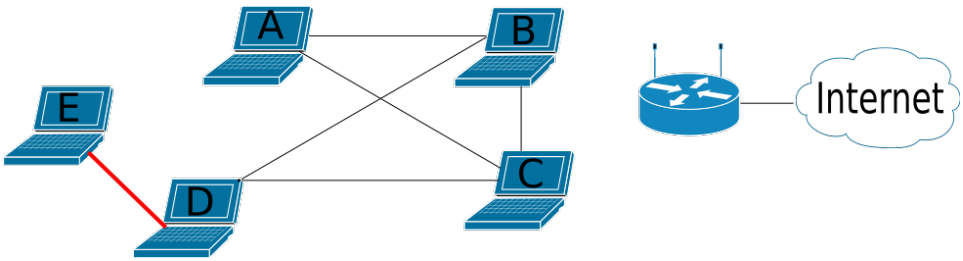

Figure 4: Client creates TCP connection with last peer

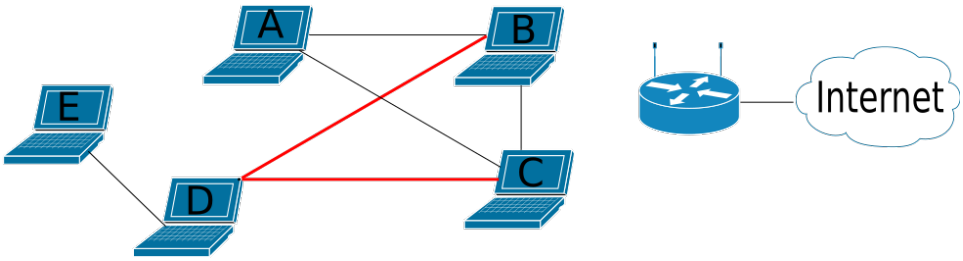

Figure 5: LNDP Broadcast 1

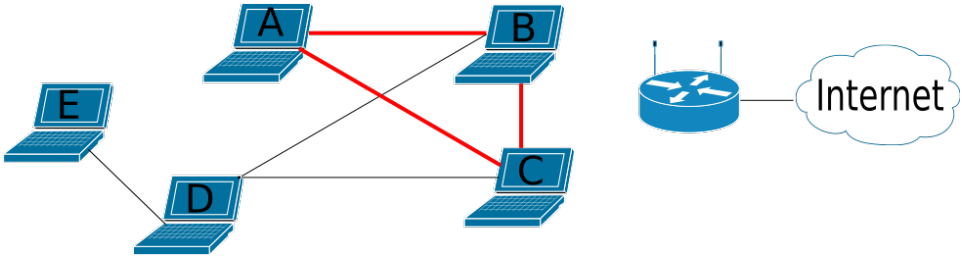

Figure 6: LNDP Broadcast 2

peer in local swarm. The range of ids is from 0 to $total - 1$. The first peer will have id 0, and the last one will have id $total - 1$. Here A have id 0, and E have id $total - 1 = 5 - 1 = 4$. Each peer will have internal variables $total$ and my_id.

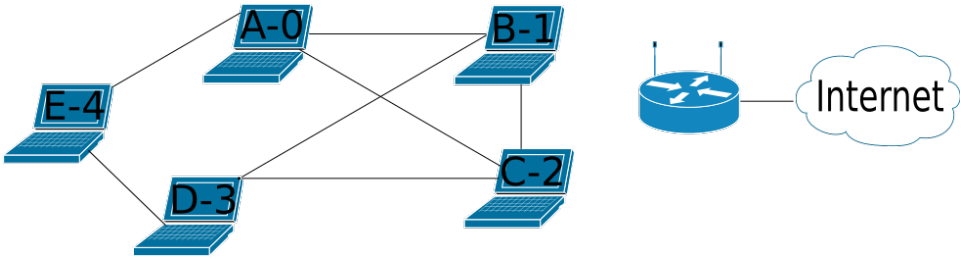

Figure 7: LNDP Network in local network

Now based on this information each peer downloads pieces calculated by following equation.

$$total * i + my\_id \qquad where \quad 0 \leq i < \left\lceil \frac{length - my\_id}{total} \right\rceil \tag{1}$$

Here $length$ is the number of pieces in torrent.

1.3 Disconnection

There are two types of disconnections.

1. Disconnect from one peer, but not from local swarm.

2. Disconnect from local swarm.

First kind of disconnection is easy. Here, a peer who is going to disconnect sends a message saying *"don't worry I am going to disconnect"* to other peer. Receiver of this message does nothing.

The second kind of disconnection is little bit complicated. Here first a peer who wants to disconnect sends a broadcast message saying *"I am going offline"*. Whoever receives this message decrements variable *total* and if $my\_id == total - 1$ then he becomes last peer and sends a message saying *"I am now the last peer"* to all of it's outgoing LNDP links.

1.4 LNDP Messages

LNDP message is encapsulated in LTEP message. Format of LTEP is shown in figure 3. The format of LNDP message is shown in figure 8.

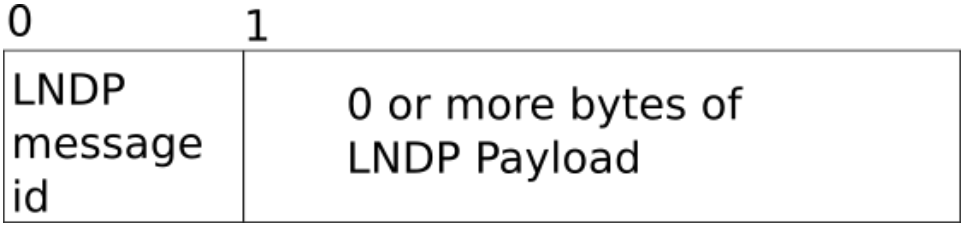

Figure 8: LNDP Message format

LNDP messages are shown in table 1.

message id	name	nature	payload
0	handshake	point to point	handshake details
1	new peer	broadcast	dictionary with keys IP, Port
2	don't worry	point to point	N/A
3	going offline	broadcast	dictionary with keys IP, Port
4	update	broadcast	dictionary with keys IP, Port, ID

Table 1: List of LNDP Messages