index
peer c:\users\finn\351project\p2p_fileshare\peer.py

Modules

ast
re
socket
sys
threading
time

Functions

broadcast(server_ip, server_port, port, filename, sock)
Function that broadcasts the current peers file contents
Params:
listenPort: Port of the current Peer
udp_sock: UDP socket used to receive messages

checksum(data)
compiutes checksum
data: data to compute checksum on
returns: checksum of data

connectToPeer(filename, pieces, peer_port)
Function that begins file exchange with discovered peers
Params:
filename: Name of movie being torrented
pieces: Size of the overall movie
peer_port: port of the current peer we are in

main(port, metainfo, file)
Main Function that starts all threads needed by the Peers.
The Three threads are:
1)Broadcast to the tracker the current contents of my file
` 2)Listen to messages coming from the tracker telling me the (port, IP) of peers I need
3)Start Listening for incoming TCP requests from other peers requesting file chucks.

parse_torrent_file(metainfo)
Parse_torrent_file is a function that reads in the metainfo file
and builds a dictionary of configs needed to start the torrent

Params:
Metainfo: Dictionary within a dictionary that tells me where to find the tracker
as well as general params for the torrent like piece Size and Indexes.

receiveFromPeers(listenPort, p_len)
Function that opens a TCP connection and exchanges file chucks with other peers
Params:
listenPort: New port currently used by the Peer to listen to requests
P_len: Packet Length Used throughout the whole torrent

receiveFromTracker(listenPort, udp_sock)
Function that receives packets from the tracker
Params:

listenPort: Port of the current Peer
udp_sock: UDP socket used to receive messages

startBroadcast(server_ip, server_port, port, fileName, sock)
Function starts the thread to start broadcasting to the tracker

Params:
server_ip = tracker IP
server_port = tracker port
port = port number for the current peer
filename = name of the movie you want to torrent
sock = socket initialized in main

startListeningForPeers(port, p_len)
Function starts the thread to listen for imcoming request from other peers

Params:
Port: Port of the current Peer
P_len: Packet Length Used throughout the whole torrent

startListeningForTracker(port, sock)
Function starts the thread to listen for imcoming messages from the tracker

Params:
Port: Port of the current Peer
Sock: UDP socket that was created in the main method

writeToFile(filename, data_to_write, port)
Function that write the file to the directory once its confirmed
file download is complete
Params:
filename: name of movie
data_to_write: Content downloaded via P2P
port: Port of the current peer

Data

host = '127.0.0.1'
incoming_peers_to_connect = []
keep_downloading_file = True
keep_seeding = True
received_file = ''
received_index = []