Peer to Peer Circular Network Design

version : python 2.7
z5172461 Huixian Zhou

Demo video link

https://youtu.be/xPvayBcHSnY

code without Logical bug: https://youtu.be/m1Otuboi5IQ

Thread

1.setUDPMonitor : This function is used to build UDP monitor both server and client. As server it get the ping request message send by the peer's 2 pre - successor.As client it get the ping response message from 2 successor.
2.setTCPMonitor: This function is used to build TCP monitor both server and client. As server handle message of file request, peer leave, ask successor massage.As client handle message of file response, peer leave response , successor response message.Monitor all Tcp message send an response from network.
3.checkAlivePing:This function used to send ping messages to the 2 successor continues.Check if the send message lost more than 3 times.If it is that means that successor leave the network.

Ping successors

Get peer number and 2 successor number.setUDPMonitor() create socket that is to be used for listening ping message.checkAlivePing() send ping request to 2 successor then get the response message from them. UDP monitor get request message from the two pre successor a then send response message back to the two pre successor.The circular network build.

Request a file

Once the main() get right ‘request xxxx’ command from xterm it jump to sendFileReq(fileNum) send TCP request to the first successor.The TCP monitor of first successor get the message and jump to handleFileReq() to judge whether the file store in here or not.If it have the file than it will send a file response message back to the peer who asked for the file.If not it will send the TCP message to it’s first successor till find the file.

Peer departure

Once the main() get right ‘quit’ command from xterm it jump to sendLeave(peerNum) send the TCP message to the two pre successor tell them the changed successor .The TCP monitor of pre successor get the message jump to handleLea() change there successor.The peer exit the network.Meanwhile the thread checkAlivePing ping to initialize the new network.

Kill a peer

After the peer killed , the checkAlivePing continuous ping the 2 successor and check the number of last ping send and last ping accept.If the value larger than 3 that means that successor has been killed.After that it jump to peerKill() send TCP ask message to the left successor ask for its other successor.Here notice the second pre successor need to wait some time to make sure first pre successor update its new successor then ask the first pre successor its new successor .The TCP monitor of the other one get ask successor then send the new successor to it.When it get the TCP ask response message it update new successor.Meanwhile the thread checkAlivePing ping to initialize the new network.

Message design

mess[0] : pingReq , pingRes , fileReq, fileRes ,sendLea , resLea , askS ,ResS
mess[1] & mess[2] is peer number of target and source
mess[3] = 1/2 the order of successor