

Gabriel Gorospe
SID: 1696580
2/17/2023

Description:

The client side of my application implements a Go-back-n approach to retransmitting packets where if any packet in the retransmission window is not ACK'd, the entire window will be retransmitted. The `recvfrom` and `sendto` functions are called in separate threads where the first thread is used to transmit packets and update the sequence number data which is defined globally and the second thread is used to receive ACKs and update an array which maintains whether sequence numbers are ACK'd or not (this array is also defined globally).

The server side of my application can read messages from multiple clients at once. Each client is assigned an id with the `find_cli_id()` function which takes the head of the packet received as an argument and either finds an existing ID for a packet or assigns the packet a new ID. The ID's are stored in a global array `cli_id`. Once the client id is assigned or found for a packet the id is used to access data for that specific client such as total packets received, file path, an array of strings representing the file before it is written to output, etc. Once the total received packets or a specific client reaches the amount of total packets expected from that client, the array of strings representing the file for that client is written to output.

Test Cases:

- 1) Transferred single file of size 36KB with drop rate = 0 percent, mtu = 512, winsz = 10
`./bin/myserver 9030 0`
`./bin/myclient <server ip>.101 9030 512 10 dir/APALILogo.png outfile.png`
`diff APALILogo.png outfile.png`
- 2) Transferred single file of size 208MB with drop rate = 20 percent, mtu = 32000, winsz = 10 to cause max retransmission error.
- 3) Transferred single file of size 208MB with drop rate = 4 percent, mtu = 20480, winsz = 2
`./bin/myserver 9030 4`
`./bin/myclient <server ip>.101 9030 20480 2 dir/Brave-Browser.dmg outfile.dmg`
`diff Discord.dmg outfile.dmg`
- 4) Transferred two files from different clients one after the other with drop rate = 4 percent. First file transfer size 208MB with mtu = 32000, winsz = 5, second file transfer size 78MB with mtu = 30000, winsz = 5.
`./bin/myserver 9030 4`
`./bin/myclient <server ip> 9030 32000 5 dir/BraveBrowser.dmg out1.dmg`

```
./bin/myclient <server ip> 9030 30000 5 dir/Discord.dmg out2.dmg  
diff Brave-Browser.dmg out1.dmg  
diff Discord.dmg out2.dmg
```

- 5) Transferred two files from different clients **simultaneously** with drop rate = 4 percent.
First file transfer size 208MB with mtu = 32000, winsz = 5, second file transfer size
78MB with mtu = 30000, winsz = 5.

```
./bin/myserver 9030 4  
./bin/myclient <server ip> 9030 32000 5 dir/BraveBrowser.dmg out3.dmg  
./bin/myclient <server ip> 9030 30000 5 dir/Discord.dmg out4.dmg  
diff Brave-Browser.dmg out3.dmg  
diff Discord.dmg out4.dmg
```

From this graph it can be seen that the ACK is always slightly delayed after the DATA seq num.
Around 1 ms there is a packet loss that causes a delay of about 140 ms between seq num 7
and 8.

