Comparison of TCP-only and UDP-only setups in terms of throughput and number of dropped packets in wireless multi-hop networks.

CCN 2 - PROJECT REPORT II

BY

SHREYAS GAONKAR (657613409)

SAHIL SHETYE (673274841)

What is TCP? Transmission Control Protocol (TCP) is one of the most used protocol in the world of Computer Communication Networks, especially in the Internet. TCP along with UDP (User Datagram Protocol) works on Transport Layer of Layered Model serving as intermediate medium for both Application and Internet Layer [1]. TCP is major protocol used in TCP/IP networks. Internet Protocol (IP) deals only with packets but the TCP is used to set up connection and share data between two hosts [2]. TCP aims at guaranteeing data delivery keeping the order of the packets intact. TCP works by breaking the data down to small packets and then are sent over the network. These packets are them collected at the receiver and combined in the correct order [3]. TCP causes the data to be sliced and sent via the network, it can be either size bigger packets with lesser number or many packets of smaller size. This will depend totally on the channel characteristics in that particular time.

Since the data transmission from the TCP is broken into various chunks, it is necessary that each chunk contains the header so that it can accurately be transmitted to the receiver. This header file is typically large in size and causes overhead and reduces the overall efficiency. Unlike UDP, TCP is connection based, meaning that in order for TCP to work, it first needs to set up a connection before sending any data. This is generally carried out by a process known as three way handshake. The client who want to transmit to another client will first send a message indicating that it is supposed to be sending a data and will request acknowledgement from the receiver first before sending any information. Once the receiver send the acknowledgement, it will start the actual data transmission. TCP focusses on the concept of delivery report which keeps a track of the packets that are being sent. For each packet, the receiver sends an acknowledgement. This makes sure that packet loss is minimized. Whenever a packet is lost due to any arbitrary reason, the transmitter waits for a fixed value of time called as timeout. If no acknowledgment is received in this time, the transmitter assumes that the packet is lost and sends the same packet again until it is acknowledged. Each packet segment carries its own number which is used for easier access.

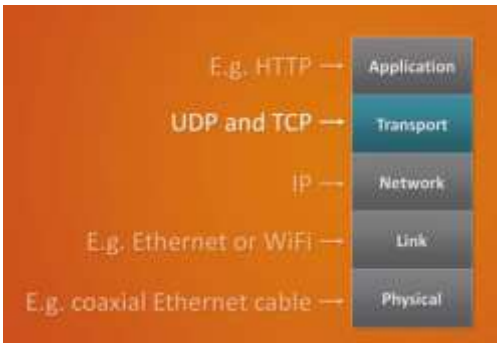

Figure 1 - TCP/UDP as Transport Layer [4]

TCP focuses on in-order delivery by collecting the packets and then arranging in the order in which they were supposed to be sent. This is a big step in the communication because most of the application requires exact data with its proper alignment like in case of messages or emails. Also, TCP uses a technique called as congestion control meaning that whenever the channel is busy, the transmission rate is altered on purpose to avoid any packet loss. The connection is resumed to its original state once the channel state improves. This can prove ineffective in applications like video calling where sending data at the appropriate moment is more important. Here UDP gets an upper hand as compared to the TCP.

UDP is the second most important and second widely used transport layer protocol after TCP that is used nowadays. UDP is lighter and easier to implement as compared to the TCP. The main reason for this is that it has a very small - 8 bytes of header which carries less overhead as compared to the TCP which effectively increases the efficiency. UDP unlike TCP is a connectionless protocol, meaning that which UDP no connection setup is required, user can simply start using the protocol without any initial connection. UDP has more control over the data which is being sent as compared to the UDP. One drawback of UDP is that it uses primitive Error Detection technique which cannot correct any data which is being transmitted. Packets lost in the transmission are lost forever and cannot be recovered or retransmitted. UDP is generally used in application which can tolerate such packet losses. Congested networks causes packet loss and reduces UDP's overall efficiency. UDP is mostly used in video calling applications since it has lesser overhead and it doesn't use congestion control.

AWK Script

AWK is a high level programming language which is used to process text files, named after its three original author's name: A : Alfred Aho W : Peter Weinberger K : Brian Kernighan. AWK Scripts are very good in processing the data from the log (trace files) which we get from NS2. If you want to process the trace file manually. AWK program structure contains mainly three parts; Begin, Content & End. To call the AWK. We use gawk function. AWK programming is used to detect pattern in each line. This pattern detection property is useful to implement various tasks such as calculating throughput or calculating packet drops. We have extensively used the AWK scripts in the project to implement various tasks.

Summary about the project:

Here, in this project, our main aim is to compare two properties of UDP and TCP viz. Throughput and packet loss: Data packet is sent over 4 nodes from node 0. Packets of size 512 bytes and having an interval of 0.01 sec are sent over from 1 sec to 6 sec using application schedulers such as CBR and FTP over UDP and TCP agents.

Average Throughput is defined as number of packets received at end node upon duration of transmission. Throughput is expressed in Bytes per second. For this example it has been observed that throughput for TCP is less than throughput for UDP. The reason can be associated to various reason such as distance between nodes is less so packet drops due to transmission are reduced. And less number of nodes has reduced the complexity thus rendering complex routing algorithm useless. Congestion control protocol also brings overhead in data transmission. All this factors result in throughput of TCP in being less than throughput of UDP.

By definition packet loss is defined as difference between generated packets at sending nodes and received packets at receiving node. In TCP, Packet loss is avoided due Checks. Thus it is observed that for this system, there is no loss of packets. But UDP experiences various loss of packets due to scheduler size at each nodes, distance etc.

We have used xgraph function to plot throughput vs time plots for both TCP and UDP. It satisfies our condition observed in comparison of Average throughput between TCP and UDP viz. avg. Throughput for UDP is more than TCP.

Note: A different throughput and packet loss comparison is possible based on position of nodes in the space. Various other factor also play an important role in these decision.

Execution step implemented in Linux terminal:

1. **ns tcpsetup.tcl** - Trace file for TCP is made and stored
2. **ns udpsetup.tcl** - Trace file for UDP is made and stored
3. **gawk -f throughput.awk tcpsetup.tr** - throughput for TCP is displayed
4. **gawk -f throughput.awk udpsetup.tr** - Throughput for UDP is displayed
5. **gawk -f packetloss.awk tcpsetup.tr** - Packet loss for TCP is found
6. **gawk -f packetloss.awk udpsetup.tr** - Packet loss for UDP is found
7. **gawk -f graphformatter.awk tcpsetup.tr>tcpsetupgraph** - Trace file is formatted in a way suitable to be read by xgraph
8. **gawk -f graphformatter.awk udpsetup.tr>udpsetupgraph** - Trace file is formatted in a way to be suitable to be read by xgraph
9. **xgraph -t THROUGHPUT -x time -y throughput tcpsetupgraph.tr udpsetupgraph.tr** - Graph is displayed
   The '-f' instructs the awk utility to get the AWK program from the source file i.e. from .awk file

Reference:

[1] – Transmission Control Protocol – http://en.wikipedia.org/wiki/Transmission_Control_Protocol

[2] – TCP - Transmission Control Protocol - http://www.webopedia.com/TERM/T/TCP.html

[3] – Web TCP/IP - http://www.w3schools.com/website/web_tcpip.asp

[4] – TCP and UDP Comparison - https://www.youtube.com/watch?v=Vdc8TCESIg8