Evaluation of Network Performance using a Media Segment over a Transmission Control Protocol Connection

Measurement of Download time taken for a Media file through request-response interactions implemented using TCP protocol.

Ajesh Vijay Vijayaragavan
M.S Computer Science
University of New Hampshire
avv1004@wildcats.unh.edu

***Abstract*—Network performance including its quality can be evaluated through various parameters like Bandwidth, Latency, Throughput and Error Rate. Adaptive bitrate streaming (ABS) is a performance management technique for streaming multimedia over computer networks. Most of the video streaming approaches were founded on RTP or RTSP but ABS technologies are built for transmission over broad distribution networks. This paper as the aforementioned title points, focuses primarily on the Evaluation of Network performance using a Media Segment over a Transmission Control Protocol (TCP) connection. The Experiment is carried out by observing the time taken by a TCP server to send and time taken by a TCP client to download a media file. The media file (approximately 1MB in size), is then downloaded repeatedly in order to observe the changes in the download time. The Experiment is carried out on two isolated servers which does not consist of any traffic other than the one generated by the programs. The main observations and ideology are that the time taken for sending the media segment is less at server side than the time taken for receiving the segment at client side. Also, a steady improvement was observed in download time when the segment was downloaded repeatedly.**

I. Introduction

High quality of Network service is preferred mostly everywhere as Low Bandwidth and quality results in long download times especially for media segment. Bandwidth is an entity that a person perceives as the speed of a network. There are several delivery methods carried out for media segment which will be described briefly. The experiment is mainly performed to measure the performance, i.e. Time taken for downloading a media segment, over a TCP connection., For the purpose of the experiment two Linux servers were used both have 4 interfaces each named enp5s0f0, enp5s0f1, enp5s0f2, enp5sof3.

The algorithm is tested on all the four interfaces and carried out repeatedly to observe changes in download time. The amount and the type of data that is sent and received among the client and the server is same on all the four interfaces. An AVI format video file which is 1.3 MB in size and displays university logo is used for this experiment. The two Linux servers used to run the experiment are named ‘RB1’ and ‘RB2’ respectively further in the paper. For all the experiments performed the ‘RB1’ served as the server and ‘RB2’ served as the client.

II. Network Performance – transactions using a media segment

Transmission Control Protocol (TCP) is one of the principal protocols in the TCP/IP network. The IP protocol deals only with packets whereas TCP enables two hosts to establish a connection and exchange streams of data. TCP promises delivery of data and promises that packets will be transported in the same order in which they are sent. To perform the experiment, the client program is run on ‘RB2’ and server part of the code is run on the ‘RB1’. The time taken for the server to send the file and time taken for the client to receive it is noted. The method is carried out at all the four interfaces and accurate download time (in seconds) are noted down.

The following table and diagram illustrate the patterns that are observed.

Figure 2.1: Exploration of network performance using a Media Segment

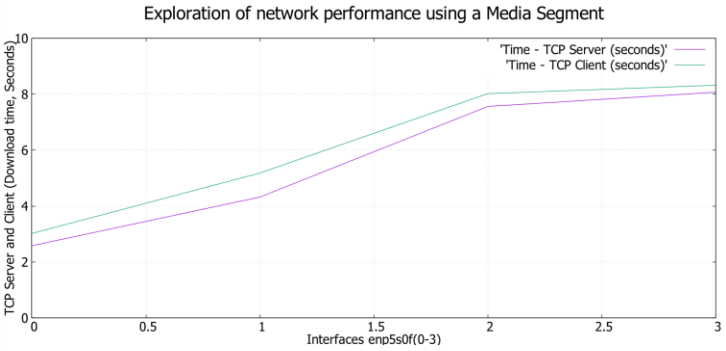


Table 2.1: Interfaces and transaction time (TCP server and Client)

Interface	Time - TCP Server (seconds)	Time - TCP Client (seconds)
enp5s0f0	2.58	3.02
enp5s0f1	4.32	5.18
enp5s0f2	7.56	8.02
enp5sof3	8.07	8.32

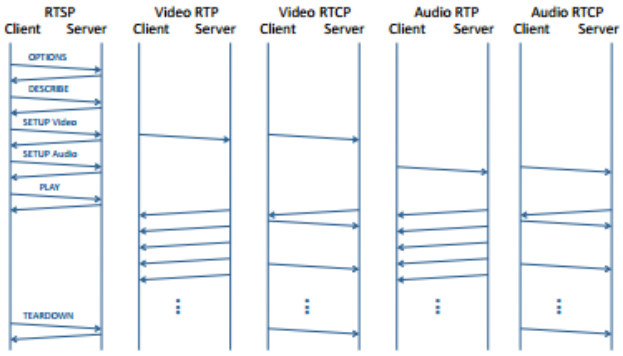
From the figure, it can be observed that the time taken by the TCP server to send the media segment is less than the time taken by the TCP client to receive the same file.

III. NETWORK PERFORMANCE – TRANSACTIONS IN A LOOP

Process carried out for Real time media especially videos uses various type of streaming and messaging protocols. Real-Time Streaming protocol (RTSP) establishes and controls media sessions. It is not responsible for streaming by itself and it can be executed over a TCP connection. RTSP is a state-full protocol. Real Time Messaging protocol (RTMP) is a proprietary protocol developed by Macromedia for communication between Flash player and server. It uses single TCP connection. The controls and streams are multiplexed together.

The following diagram is a pictorial representation of paths carried by streaming and control protocols.

Figure 3.1: Real time Streaming and control protocols.



In order to explore and evaluate the performance of network the same media segment is downloaded repeatedly and time taken by the TCP server and client was observed.

The following table and diagram illustrate the patterns that are observed.

Figure 3.2: Network performance – Transactions in a Loop

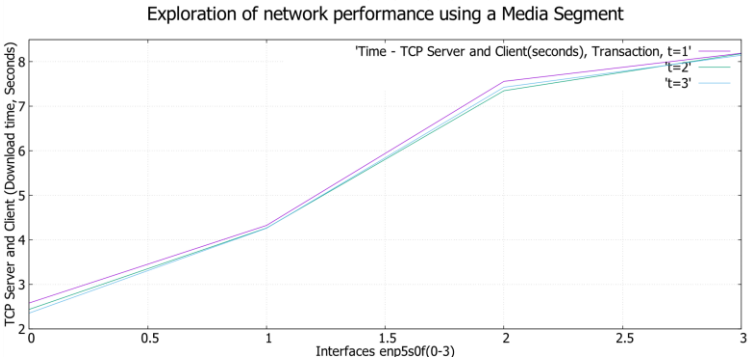


Table 3.1: Interfaces and transaction time (TCP server and Client)

Sequence	Interface	Time - TCP Server (seconds)	Time - TCP Client (seconds)
1	enp5s0f0	2.583	3.024
	enp5s0f1	4.326	5.182
	enp5s0f2	7.561	8.026
	enp5sof3	8.193	8.327
2	enp5s0f0	2.439	3.011
	enp5s0f1	4.267	5.181
	enp5s0f2	7.351	7.859
	enp5sof3	8.182	8.321
3	enp5s0f0	2.356	2.592
	enp5s0f1	4.263	5.112
	enp5s0f2	7.429	7.587
	enp5sof3	8.147	8.279

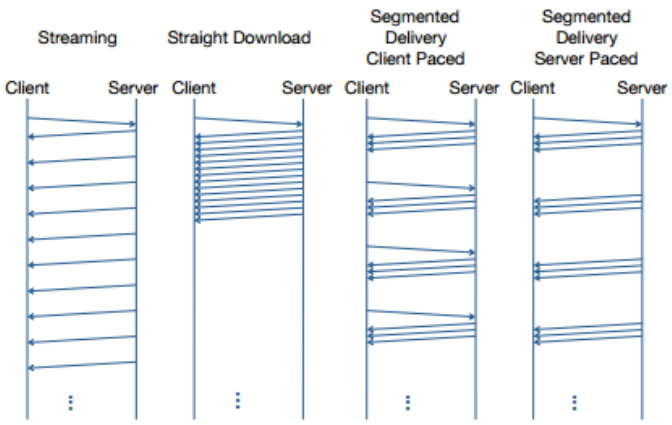
From the figure, it can be observed that the time taken by the TCP server and client to send and receive the same media segment repeatedly decreases in consecutive transactions although time taken by TCP client is greater than that of the server.

IV. VIDEO DELIVERY METHODS AND METRICS

Delivery or download of media segment is achieved by network following various delivery methods and metrics. The methods that are followed for video delivery is explained below.

1) Streaming (RTSP/RTP) – It is unreliable (User Datagram Protocol) and completes server-side pacing.
2) Straight download (HTTP) – It is reliable (Transmission Control Protocol, TCP) transport and no pacing is involved.
3) Segmented delivery (HTTP Progressive download) – It is reliable (TCP) transport, involves client-side pacing and uses Range GETs or segment files.

Figure 4.1: Video Delivery Methods



Playback Latency, Time Shift Latency, Playback Artifacts and Stoppages are some of the Video metrics. From the transactions that was performed in a sequence to download the media segment repeatedly, the Mean value taken by the client and server to download the media segment is calculated. The calculated results are provided below.

Figure 4.2: Mean time taken by TCP Server and Client

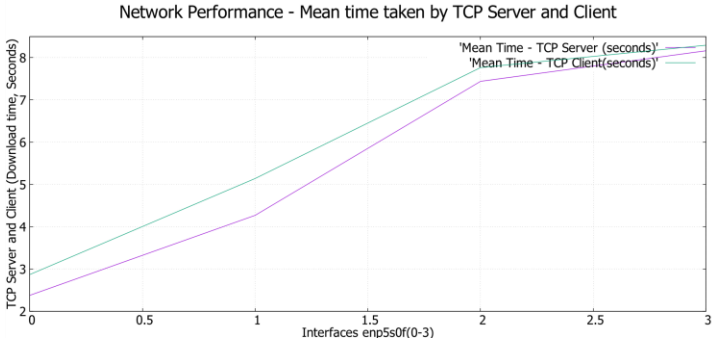


Table 4.1: The calculated mean time at each interfaces

Interface	Time - TCP Server (seconds)	Time - TCP Client (seconds)
enp5s0f0	2.379	2.871
enp5s0f1	4.272	5.143
enp5s0f2	7.441	7.767
enp5sof3	8.164	8.295

The above table and graph is obtained after the algorithm carries out transaction repeatedly. It can be observed that the Mean time taken by TCP server to send 1.3 MB media segment is less than the Mean time taken by the client to receive the same file. It was also observed that the total time taken by both client and server decreases when the file was downloaded repeatedly which is due to reduction in time taken to establishing a network line, request response acknowledgement and processing of network packets. It can be further decreased depending on network traffic, load and following consistent methods and metrics explained earlier which in turn increases the overall performance of the network.

References

[1] J. Kurose and K. Ross, *Advanced Computer Networking:* A Top-Down Approach (7th Edition).

[2] K. Hong, '*Network Programming – Client & Server*'. Available: https://plus.google.com/+khongsanfrancisco

[3] W. Stallings, '*Foundations of modern networking'*, First Edition, Addison-Wesley professional, 2016, ISBN 0134175395.