SDN lab Report

Lin Sinan

1. Introduction

In this lab, I implemented a strategy through POX controller to separate the traffic of HAS (HTTP-based adaptive streaming) and FTP to maximize the throughput of HAS traffic under two scenarios: First, HAS is the only traffic in the network; Second, both HAS and FTP traffic exist in the network.

2. Problem

As for making a baseline, the controller(of_myPOX.py) provided in the course is used at the beginning, and we record its performance in two scenarios as our baseline. The results are that the average video quality is 1.170 when there is only HAS traffic, and the average video quality is 0.567 when we transmit both FTP and HAS.

There are two aspects that can be improved. First, there is no packets going through router s2 (Figure 1). Second, throughput of HAS traffic is greatly affected by the FTP traffic when they are sharing the bandwidth (Figure 2). And my solution will aim at these two aspects to improve the video quality.

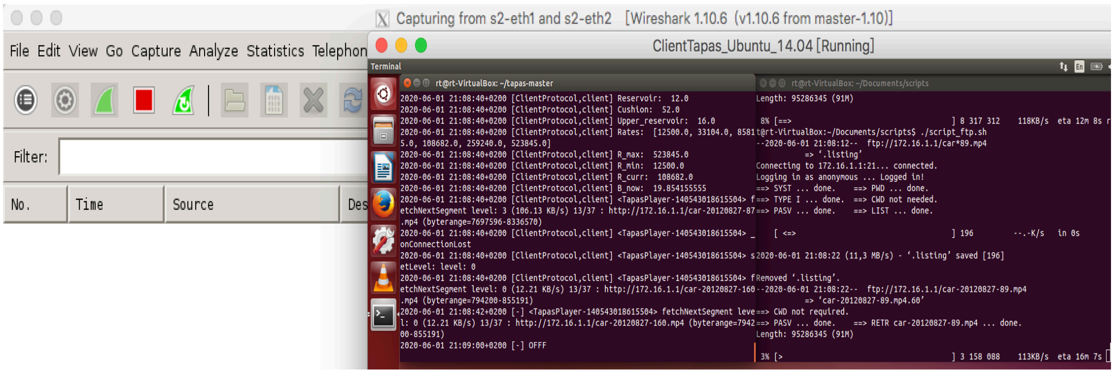

Figure 1. No traffic in s2-eth1 and s2-eht2 when using myPOX.py

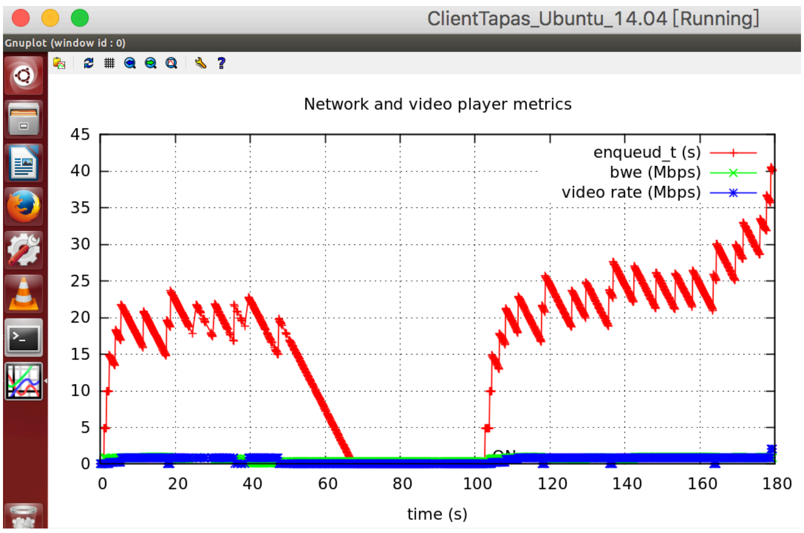

Figure 2. performance of HAS when both FTP and HAS are running

3. Solution

There is a straightforward and easy solution based on the problems we found. First, adding a route through router s2 from s1 and s2. To utilize the bandwidth of s1-s2 and s3-s2, I add new flow tables based on ingress ports, which enable them to find a route through s2 (Figure 3) and thus improve the throughput of total traffic.

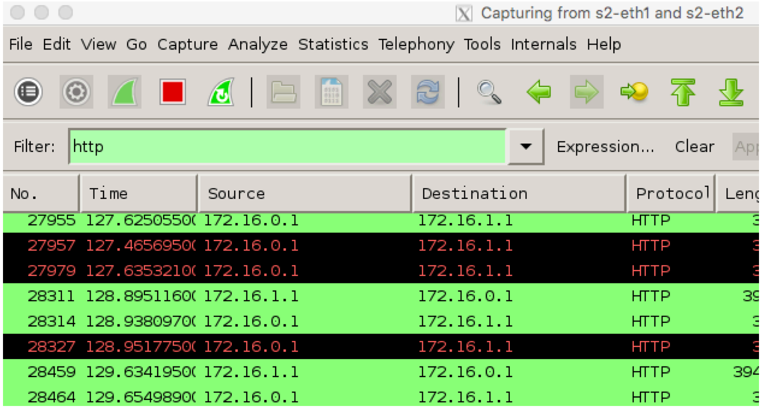

Figure 3. traffic of s2-eth1 and s2-eth2 after adding flow table

Second, separating the traffics of HTTP and FTP. In order to do so, add new flow tables and modify their match fields to distinguish HTTP packets from the traffic (Figure 4).

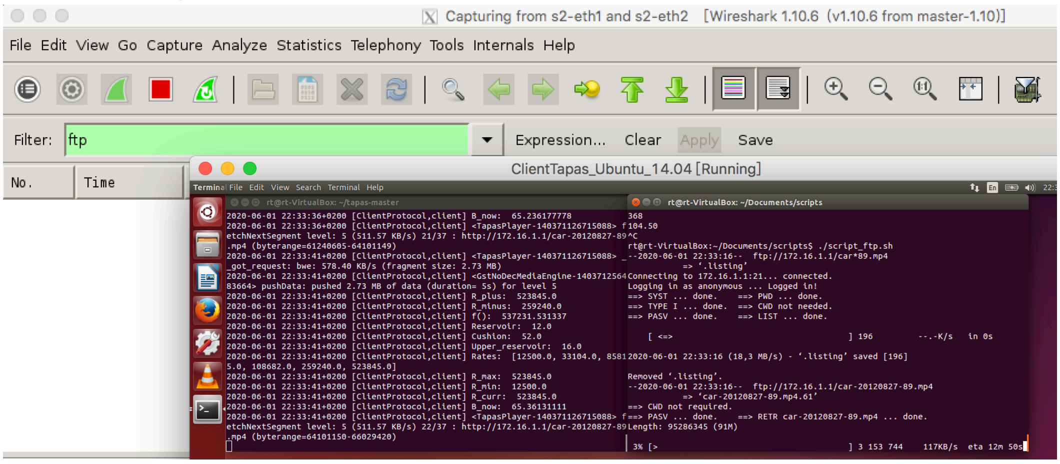

Figure 4. no FTP traffic is captured in s2-eth1 and s2-eth2

Third, assigning priority to flow tables specifying HTTP packets. The switch starts performing a table lookup from the first flow table, so if we don’t give priority to HTTP packets, they might not go through the router s2, because we did not add a match field specifying non-HTTP in the flow tables that are supposed to route FTP packet directly between s1 and s3.

And the results for these two scenarios are that the average video quality is 3.442 when there is only HAS traffic (Figure 5) and 3.257 when both FTP and HAS traffic in the network (Figure 6). As we can see, the performance of HAS is dramatically improved in these two scenarios.

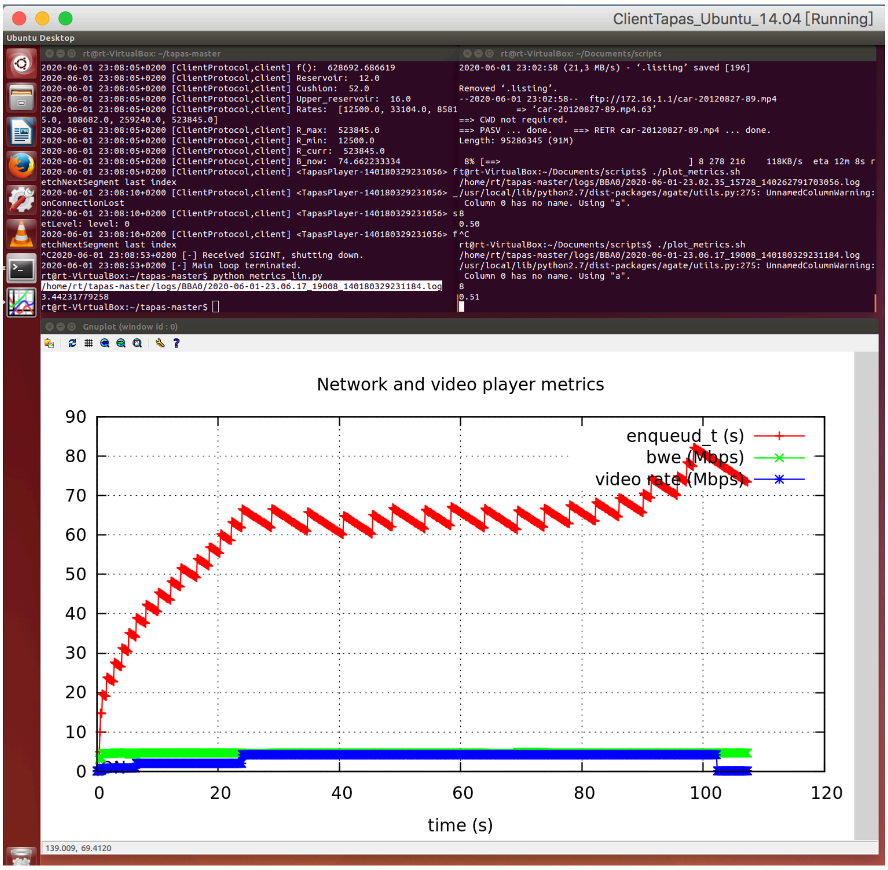

Figure 5. performance of HAS when there is only HAS traffic

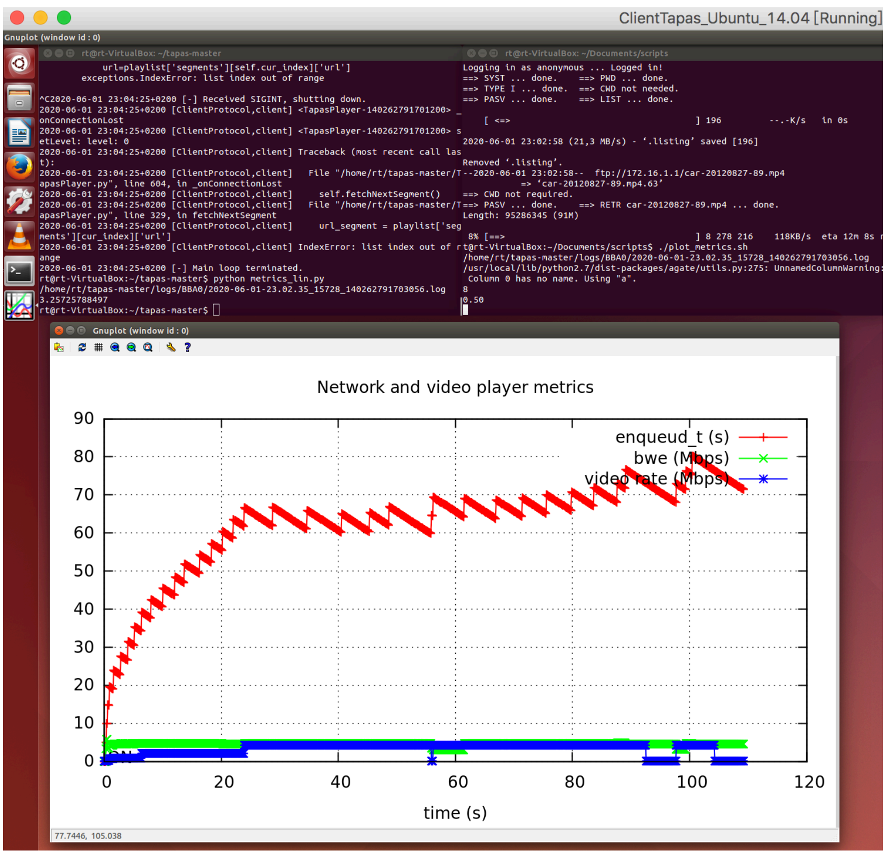

Figure 6. performance of HAS when both FTP and HAS are running

4. Conclusion

Our solution improves the performance of HAS traffic from the average video quality from 1.170 to 3.442 with only HAS traffic and from 0.567 to 3.257 with HAS and FTP sharing the bandwidth. However, since we choose the buffer-based strategy for HAS, it always starts from low resolution video segments and takes an amount of time to switch to 4.2MBps coding rate video segments, and thus the video quality in the first 20 seconds is relatively low.