Journal of Computer Science 5 (12): 1042-1047, 2009
ISSN 1549-3636

Comparison Study of Transmission Control Protocol and User Datagram Protocol Behavior over Multi-Protocol Label Switching Networks in Case of Failures

Taha Ahmed Al-Radaei and Zuriati Ahmad Zukarnain
Department of Communication Technology and Network, University Putra Malaysia,
43300 Serdang Selangor, Malaysia

Abstract: Problem statement: In only a few years, Multi-Protocol Label Switching (MPLS) has evolved from an exotic technology to a mainstream tool used by service providers to create revenue-generating services. MPLS provides a high reliable Label Switched Path (LSP). MPLS failures may degrade the reliability of the MPLS networks. **Approach:** For that reason, many studies have been conducted to keep the high reliability and survivability of the MPLS networks. Unlike User Datagram Protocol (UDP), Transmission Control Protocol does not perform well in case of like-failure of MPLS networks because of its inability to distinguish packet loss due to link-failure. After the recovery time, TCP takes longer time than UDP to continue as it was before the failure. **Results:** In terms of packet loss, TCP performs better than UDP. However, the receiving rate of the TCP traffic is much worse than UDP traffic. A need for a mechanism to improve the behavior of TCP after a link failure is needed. This study focused on comparing the behavior of different types TCP as well as UDP traffic over MPLS networks in case of link, node or congestion failures. **Conclusion:** Although extensions of RSVP-TE protocol support fast recovery mechanism of MPLS networks, the behavior of TCP will be affected during recovery time much more than with UDP.

Key words: MPLS, TCP, UDP, failures

INTRODUCTION

As needs for the speed and quality of service grow to carry more traffic, it is essential to maintain a high level of performance and efficiency. Traffic engineering is the process of optimization of the network to maximize performance and efficiency. MPLS is a tool for network traffic engineering and hence becoming the technology of choice for internet backbone. An MPLS network consists of two domains known as a Label Edge Routers (LERs) domain and Label Switching Routers (LSRs). A mish unidirectional tunnels, known as Label Switched Paths (LSPs) is built between the LERs and LSRs in order that a packet entering the network at the ingress LER can be transported to appropriate egress LER. Forwarding mechanism of the packets in the MPLS is carried based on fixed size labels, the path that packets traverse is pre-established according to required constraints. The path the packet traverses is called Label Switch Path (LSP). Regarding to the label distribution there are two protocols used for this propose called Label Distribution Path (LDP) and Resource Reservation Protocol (RSVP).

RSVP was invented before MPLS came into being and was originally devised as a scheme to create bandwidth reservations for individual traffic flows in networks. RSVP includes mechanisms for reserving bandwidth along each hop of a network for an end-to-end session. In context of MPLS, RSVP has been extended to allow it to be used for creation and maintenance of LSPs [RFC3209]. However, links failure or LSR failure always incurs performance degradation and packet loss in connection passing through the link or LSR to. A fast recovery mechanism is needed to support a high quality of service and to keep the reliability of the MPLS networks which is considered as one of the most important features of the MPLS[1]. Based on the recovery location, there two types of recovery, global and local protection. Global protection is accomplished by setting up an alternate path that can be used in case of failure of the working path or any LSR in the working path. Local protection is accomplished by setting up protection path around the failed link or node.

In general, on Wide Area Networks (WANs), UDP has likely been used for real-time applications, such as video and audio. UDP supplies minimized transmission

Corresponding Author: Taha Ahmed Al-Radaei, Department of Computer Science and Information Technology, University Putra Malaysia, 43300 Serdang Selangor, Malaysia

delay by omitting the connection setup process, flow control and retransmission. Meanwhile, more than 80 percent of the WAN resources are occupied by Transmission Control Protocol (TCP) traffic. As opposed to UDP's simplicity, TCP adopts a unique flow control mechanism with sliding windows. Hence, the Quality of Service (QoS) of real-time applications using UDP is affected by TCP traffic and its flow control mechanism whenever TCP and UDP share the same network resources[2].

Many researches have been conducted for improve the protection mechanisms of MPLS networks using a UDP traffic[3,4]. However, study the behavior of TCP and UDP traffic over MPLS networks in case of failure is an essential issue for fast failure detection and recovery. In this study we focused on comparing the behavior of TCP and UDP traffic over MPLS in case of any failure and what are the parameters that effect the recovery time. The label distribution path used in this study is RSVP that is defined in[6]. RSVP-TE extensions are used to establish the backup label switch path LSP tunnels for local repair of LSP tunnels. These mechanisms enable the re-direction of traffic onto backup LSP tunnels in 10 sec of milliseconds, in the event of a failure [6]. Two methods are defined in these extensions one-to-one and facility backup. We adopted the facility backup in this paper. MPLS supports label stacking which is the encapsulation of an MPLS packet inside another MPLS packet that is, adding an MPLS header on top of an existing MPLS header as in Fig. 1. The result of stacking is the ability to tunnel one MPLS LSP inside another LSP.

Instead of creating a separate LSP for every backed-up LSP, a single LSP is created that serves to back up a set of LSPs (LSP tunnel a bypass tunnel). The bypass tunnel must intersect the path of the working LSP(s) somewhere downstream of the point of local repair PLR. The two paths are composed of the transmitting path that carries TCP packets and the receiving path that carries the ACK packets. The transmitting path also carries the UDP packets in case of UDP traffic. We assumed that when a failure occurred in any working path, both the transmitting and receiving paths are switched to the backup path. Path switching for both the working and the backup paths is the responsibility of the path label-switching router PSL. Each label-switching router LSR monitors the link state upstream, when an LSR detects a failure; it sends a notification message to the PSL. When the PSL receives the notification message, it switches the traffic from the working path to the backup path. The recovery time is effected by the detection time, notification time and the switching (failover) time.

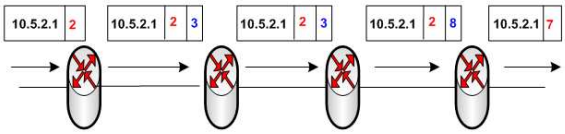

Fig. 1: MPLS label stacking

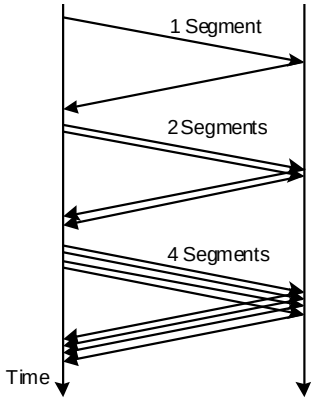

Fig. 2: TCP slow-start

Even though TCP and UDP use the same network layer (IP), TCP provides a totally different service to the application layer than UDP does. TCP provides a connection-oriented, reliable, byte stream service. TCP relies on acknowledgments from the receiver to confirm correct delivery of data. The flow control implemented in TCP prevents an overflow at the receiver by adapting the advertised window dynamically to the receiver buffer space. However, this flow control does not cope with the buffer overflows in the intermediate network nodes. To deal with network congestion, congestion control mechanisms have been implemented in TCP. In TCP, the sender starts the transmission with an initial congestion window of one segment, the congestion window can be initialized to two or four segments. Once the sender receives the acknowledgement of the transmitted segment, it increases the congestion window by one segment. As the sender receives acknowledgments of the transmitted segments, it increases the congestion window by one segment for each acknowledgement received. This procedure continues until a loss is detected, either by triple duplicate or a retransmission timeout, or until the window size reaches a threshold called slow-start threshold (Fig. 2).

When a link fails and path protection is executed, there are four possibilities that the TCP packet or ACK packet can pass or drop based on the failure timing.

As shown in Fig. 3 and Fig. 4, the failure occurred when the packet has been left the sender and it is on the path to the receiver. In Fig. 3 the packet will pass and reach the destination successfully.

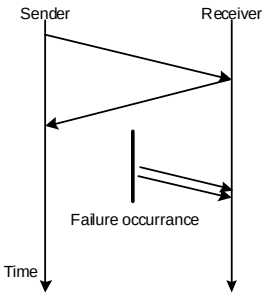

Fig. 3: The packet will pass

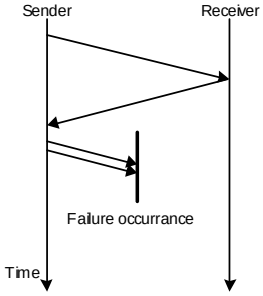

Fig. 4: The packet will drop

In Fig. 4 the packet will be dropped on the working path because the protection has not been completed. However, in some recovery mechanisms like Haskin's model, this packet will be reversed back to ingress router and forward it to the backup path. The sender will not receive the related ACK for this packet and will retransmit the packet again after the timeout timer finish.

The other two possibilities are when the receiver send ACK packet to the sender. If the ACK pass the failure location, it will reach to the sender as illustrated in Fig. 5. Otherwise the ACK will drop as in Fig. 6. Once a segment loss occurs, the behavior of TCP depends on how this loss is detected by a triple duplicate or a timeout.

When the timer expires before receiving an acknowledgment, TCP interprets this phenomenon as a severe congestion in the network. The network is overloaded and the transmitted segments are lost, which implies retransmission of the segments and a brutal reduction of the congestion window. The earliest unacknowledged segment is then retransmitted.

In addition, the congestion window CWND is set to the value of the so-called Loss Window (LW), which in general equals to its initial value.

When a loss is detected by triple duplicate ACK, TCP interprets this phenomenon as congestion in the

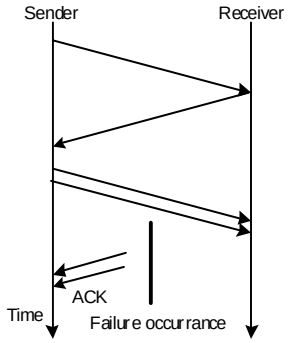

Fig. 5: The ACK pass

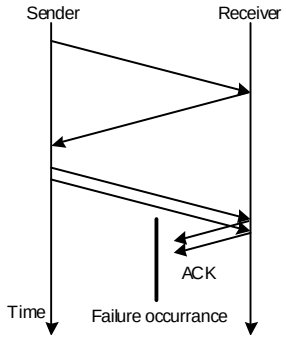

Fig. 6: The ACK drop

network and the lost segment is retransmitted. Because of the TCP flow control and congestion control, TCP needs some time to reach the steady-state in case of failure although the recovery time in MPLS is very short. And this degrades the receiving rate in the receiver side.

Unlike TCP, UDP is a connectionless, unreliable. All UDP provides is a mechanism for the application to send a short message to a given destination. However, in case of UDP, the sender will not stop sending the packets in case of failure. And it will continue sending the packet causing a packet loss much more than in case of TCP.

MATERIALS AND METHODS

NS2[7] was employed as the experimental platform in our simulation. The RSVP-TE was used as label distribution. The topology used in our simulation (Fig. 7) is a typical one for MPLS networks and has been used in a number of studies. All the nodes in the topology are LSR. The thick lines are 20 Mbps and the thin lines are 10 Mbps. The source node is node 0 and the destination node is node 12. The working path is 0-2-5-10-12. A link failure has been assumed between node 2 and node 5 at time t = 10. Traffic flows begin to transmit at time t = 2 and stop at t = 20.

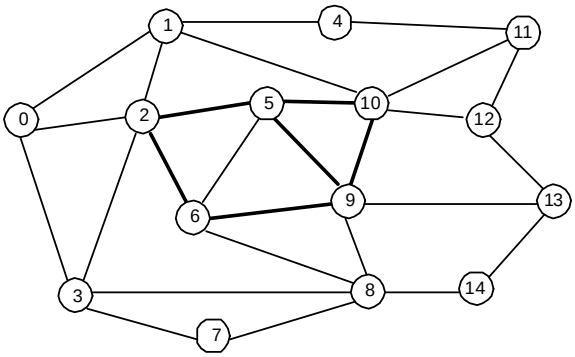

Fig. 7: Network topology

To compare and study the performance of the TCP and UDP traffic over MPLS networks in case of failure, we run the simulation once with TCP traffic and the other time with UDP traffic. For each traffic, we run the simulation with different transmission rate.

At time t = 0 node 0 sent an RSVP path message downstream along the working path 0-2-5-10-12. Path messages follow the exact paths of application data, creating path states in the routers along the way, thus enabling routers to learn the previous hop and next-hop node for the session. After the failure occurred at time 10, a Path error messages was sent to the sender that issued the Path message node 0. As a result, the traffic rerouted to backup path.

RESULTS

UDP traffic has been used in the first scenario of the simulation. The source node 0 started sending the packets at time 2. At time 10 a failure occurred and the source node did not stop sending the packets after detecting the failure. This is because the UDP is not reliable so it will not wait for the acknowledgments of the packets that have been received. Whereas in TCP, when the failure, the source node will stop sending the packets to wait for the received packets acknowledgments.

Figure 8 shows the number of lost packets in case of TCP and UDP traffics with different transmission rate.

The number of packet losses is increase with the increasing of the transmission rate of the UDP traffic. However, the number of packet losses in TCP is constant although the transmission rate increases.

Figure 9 shows the receiving rate at node 12 for UDP traffic. At time 10 sec, we can note the drop of the receiving rate due to the link failure. During the period of the failure detection, failure notification and traffic

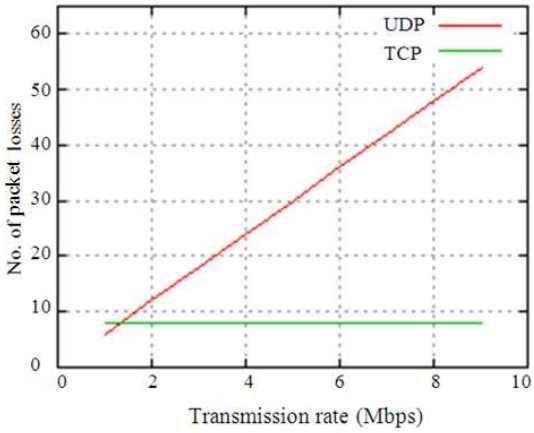

Fig. 8: Packet Losses in different transmission rate

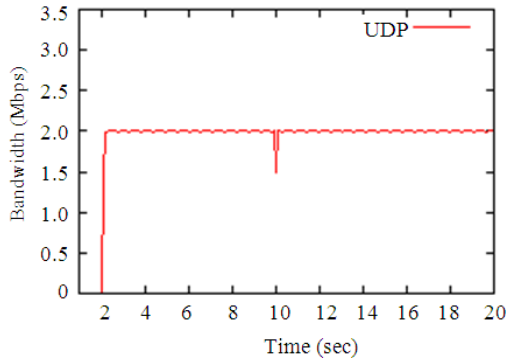

Fig. 9: UDP receiving rate

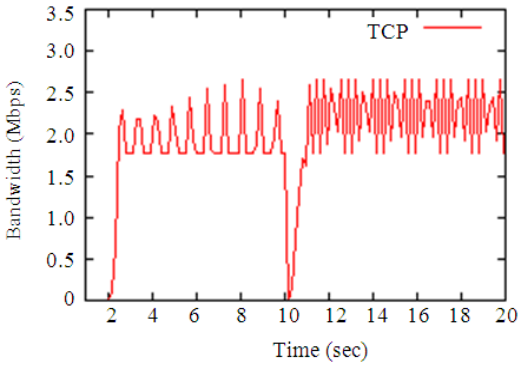

Fig. 10: TCP receiving rate

reroute, number UDP packets were lost. There is no change in the receiving rate performance before the failure and after the recovery. This is because of the continuity of sending packets during the transmission time and need lack of the acknowledgment.

Figure 10 shows the receiving rate at the receiver side (node 12). The failure has a noticeably effect of the TCP receiving rate during the recovery time and after

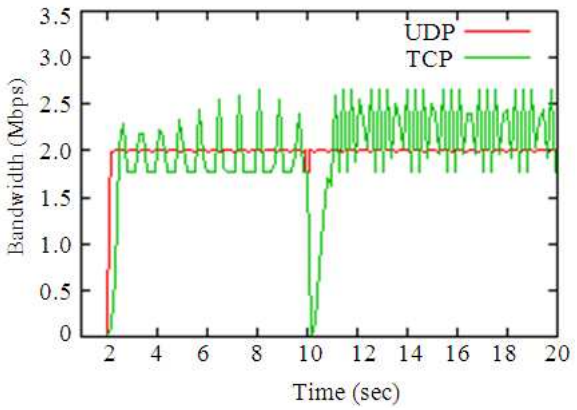

Fig. 11: Receiving Rate for both UDP and TCP

the recovery time. During the period needed to detect, notify and recover the failure, the receiving rate degraded to 0 Mbps. This is because the TCP source has stopped sending the packets waiting for the acknowledgments coming from the receiver while the ACKs packets also affected by the failure. After the recovery time, TCP multiplicatively decreased its congestion window size and needed sometime to return back to the previous congestion window size. As the traffic reroute to the backup path, TCP receiving rate suddenly has been affected by the parameters of the backup path. Figure 11 shows the differences between TCP and UDP traffic performance before, during and after the recovery time.

Also we compare the number of packets sent with and without failure for both TCP and UDP during the same simulation time. Simulation results show that the number of TCP packet sent without failure were 4162. This number increased to 4404 in case of failure. The percentage of the retransmitted packets is 5.81%. Unlike TCP, UDP traffic source is remained with the same number of sent packets (21429) in case of failure as well as without the failure during the same simulation time.

DISCUSSION

From the results above, we can see that TCP performs better than UDP in terms of the number of packet losses when the transmission rate increases. However, TCP does not perform well in terms of receiving rate because of its inability to distinguish packet loss due to link-failure or congestion. This leads TCP to start send the data packet with a small window size not resume sending the packet with same window size as it was before the failure although, the failure restoration take few milliseconds. The increment percentage of transmitted packet with failure in TCP comparing with when no failure is caused by the TCP retransmission mechanism. MPLS network may carry a huge number of LSPs and a single failure in the network may cause all the TCP traffic sources to retransmit a huge number of packets causing a congestion and consumption of network resources.A need for a mechanism for the TCP to distinguish the packet loss due to the link failure and resume sending the data packets with same parameters of the window size and start slow threshold is needed to avoid the degradation of the performance of TCP after the failure restoration stage.

CONCLUSION

Simulation results show that the high reliability of MPLS networks can survivability may degrade because of only one link or node failure in the network. In addition, after repairing the failure, more network resources are consume because of the retransmission mechanism of the TCP which is represent more than 80% of the internet traffic. A good MPLS network design may avoid the sudden changes of the traffic after the process of recovery. UDP traffic has not affected much by the failure but, the number of lost packets increased by the increasing the transmission rate. However, TCP traffic has a constant packet loss in any value of the transmission rate. TCP performs better than UDP in terms of the number of packet loss. UDP performs better than TCP in terms of consumption of network resources.

Our future research is to study the behavior of different TCP traffic in MPLS networks.

REFERENCES

1. Minei, I. and J. Lucek, 2008. MPLS-Enabled Application Emerging Developments and New Technologies. 2nd Edn., John Wiley and Sons Ltd., England, ISBN: 978-0-470-98644-8, pp: 526.
2. Tsuboi, T., H. Ueda and H. Kasai, 2005. Designing MPLS path protection networks for reduced TCP goodput degradation. http://sciencelinks.jp/j-east/article/200514/000020051405A0415504.php
3. Lai, W.K., Z.C. Zheng and C.D. Tsai, 2008. Fast reroute with pre-established bypass tunnel in MPLS. Comput. Commun., 31: 1660-1671. DOI: 10.1016/j.comcom.2007.11.008

4. Ahn, G. and W. Chun, 2001. Simulation for MPLS path restoration and performance evaluation. Proceeding of the Joint 4th IEEE International Conference on ATM (ICATM 2001) and High Speed Intelligent Internet Symposium, Apr. 22-25, IEEE Xplore Press, Seoul, South Korea, pp: 32-36. DOI: 10.1109/ICATM.2001.932052
5. Pan, P., G. Swallow and A. Atlas, 2005. RFC4090-fast reroute extensions to RSVP-TE for LSP tunnels. http://www.faqs.org/rfcs/rfc4090.html
6. Adami, D., C. Callegari, D. Ceccarelli, S. Giordano and M. Pagano, 2006. Design and development of MPLS-based recovery strategies in NS2. Proceeding of the IEEE Global Telecommunications Conference, Nov. 27-Dec. 1, IEEE Xplore Press, San Francisco, CA., USA., pp: 1-5. DOI: 10.1109/GLOCOM.2006.425
7. Hassan, M. and R. Jain, 2004. High performance tcp/ip networking concepts, issues and solutions. http://www.cs.wustl.edu/~jain/books/ftp/tcp_fm.pdf