MULTIPLE FAULT TOLERANCE IN MPLS NETWORK USING OPEN SOURCE NETWORK SIMULATOR

Muhammad Kamran
FAST – National University of Computer and Emerging Sciences, Lahore, Pakistan.
kamran@inbox.pk

ABSTRACT

Multiprotocol Label Switching (MPLS) is a label based switching technique initiated by Internet Engineering Task Force to bring the speed of layer 2 switching to layer 3. Its label based switching technique allows routers to make the forwarding decision based on the contents of label instead of performing a complex route lookup table. Furthermore it allows explicit routing to overcome links and nodes failure in the network. For fault recovery multiprotocol label switching has two domains called Protection Switching and Rerouting. Explicit routing, in it the alternative paths are pre-established through MPLS core router (ingress node). We developed new protocol using explicit routing for multiple fault tolerance in the network. We simulate it in an open source network simulator ns2 and compare it with rerouting fault recovery protocol in NS that belongs to rerouting domain. The simulation shows that the proposed protection switching fault recovery protocol performed better in fault recovery time as compare to rerouting fault recovery protocol.

Keywords: mpls, fault tolerance, rerouting, protection switching, ns2.

1 INTRODUCTION

Multiprotocol Label Switching (MPLS) is a highly scalable data carrying mechanism that delivers differentiated and end-to-end IP services using simple configuration and management. This framework changes the hop-by-hop paradigm and enables devices to specify network paths based on Quality of Service (QoS) and needed bandwidth. It is a packet forwarding technology that uses labels for making data forwarding decisions in the network. It provides beneficial applications, Virtual Private Networking (VPN), Traffic Engineering (TE), Quality of Service (QoS), Any Transport over MPLS (AToM) [10]. It was designed to facilitate several problems areas in the internet domain importantly for increasing routing performance and is being adopted by service providers in their core networks [5].

In MPLS network the router that receives the packet is called label edge router (LER) or Ingress label switch router that is responsible for adding a label on the packet for further transmission. The label that is added by the ingress is based on certain criteria like IP address of the recipient and it is used to route the packet to the next routers called label switch routers (LSRs). The path through which the labelled packets are routed is called label switch path (LSP) [1]. Before the packet leaves the network the last or outgoing router that is label edge router or egress router is responsible to remove the label from it. All routers of the MPLS network excluding ingress and egress are core label switch routers.

Now-a-days MPLS network is considered to be the most efficient and reliable data carrying mechanism due to its fault tolerance ability. MPLS fault recovery techniques lies in two major domains, protection switching and rerouting. Protection switching domain pre-establishes or pre-computes the backup path before the occurrence of the fault. The backup paths are normally stored at ingress label switch router, which automatically send the data on backup path after it receives fault identification signal (FIS). Protection switching is also referred as fast reroute [4], [2]. Other fault recovery technique rerouting domain establishes or computes the backup path after the occurrence of the fault. When fault or link failure occurs in the network core router sends the FIS to ingress or the adjacent router then it dynamically computes the backup path and transfer the traffic on it [4]. MPLS allows explicit routing; it develops a pre-determined explicit path through the MPLS core. In explicit routing, the label switch path is followed by router is defined by the ingress node. The explicitly defined path consists of a series of hops defined by the ingress LSR.

The fault recovery techniques can be simulated using network simulators like Tool Box For Traffic Engineering Methods (TOTEM) [17],Optimized Network Evaluation Toil (OPNET) [18], Graphical Network Simulator (GNS) [16], Objective Modular

Network Testbed in C++ (OMNeT++) [13] and Network Simulator (NS) [15]. We used Network Simulator known as ns2 to simulate the proposed fault recovery protocol. It is an open source simulator that uses two different languages because simulator requires two different kinds of functions to perform. Firstly the detailed simulations of protocols that requires a system programming language which can efficiently manipulate bytes, packet headers and implement algorithms that run over large data sets. Secondly a large part of network research involves slightly varying parameters or configurations. In these cases iteration time, change the basic model and re-run that model is more important. NS2 meets both of these needs with two languages C++ and OTcl. C++ is fast to run but requires more time to change for making it suitable for detailed protocol implementation. OTcl runs much slower but can be changed very quickly and interactively, that makes it ideal for simulation configuration. Furthermore ns2 is the best showing thing visually and it also supports trace files that traces and save everything happening against the network, like total number of packet sent in a particular time simulation, number of packets dropped, at specific time which router sends packet to which router, which link was went down at what time etc.

In this paper we have developed a new fault recovery protocol which belongs to protection switching domain using explicit routing. We simulate the proposed fault recovery protocol using ns2 [15] and compare this with rerouting fault recovery protocol that creates new label switch path on demand after the occurrence of the fault. Proposed protocol is based on protection switching domain that pre-establishes a LSP against every link in the network. If any link between MPLS based routers fail it switches over the traffic on the pre-established backup path. For comparison we focused on two major criteria recovery time and multiple fault tolerance. Recovery time is important because if fault recovery time is less so that minimum packets will be lost.

2 BACKGROUND

2.1 Multiprotocol Label Switching (MPLS)

Multiprotocol label switching is framework by IETF for solutions to address the problems faced by the networks now-a-days like speed, scalability, quality-of-service and traffic engineering. It is used to strength the IP networks and fast packet switching or routing. It uses specific labels to forward the packets with in MPLS network. More specifically, MPLS has mechanisms to manage traffic flows of various granularities [5], [7]. It is independent of the layer-2 data link layer and layer-3 network layer protocols such as ATM and IP. It maps IP addresses to simple, fixed-length labels used by different packet-forwarding or packet-switching technologies [10]. Some specific terms related to MPLS are given below.

2.1.1 MPLS Header / Label

A header created by an edge label switch router (edge LSR) and used by label switch routers (LSR) to forward packets. Header sometimes called "shim" located between the Layer 2 and Layer 3 headers. Functionally label is a short fixed length identifier that is used to forward the packets. Within the network the labels are used to route the packets without regard to the original packets header information, now label has all the information. These labels are stacked as a last in first out (LIFO) labels enabling MPLS to be combined for transport and distribution. Within the MPLS domain, the Layer 3 header analysis is done just once when the packet enters the MPLS domain [4], [5].

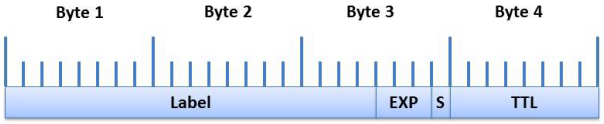

Figure 1: MPLS Header

- **Label**: Label Value (Unstructured), 20 bits.
- **Exp**: for Experimental Use, 3 bits.
- **S**: Bottom of Stack, 1 bit.
- **TTL**: Time to Live, 8 bits.

If stack bit is 1 it represents the last entry in the stack and 0 represents for all other label stack entries. The TTL field is decremented by 1 each time the packet passes through a router. The packet is discarded when the TTL reaches zero. Label is used to identify a Forwarding Equivalence Class (FEC).

2.1.2 Forward Equivalence Class

Forwarding Equivalence Class is a group of IP packets which are forwarded in the same manner, over the same path, and with the same forwarding treatment. The label gives information needed to forward the packet [5], [10].

2.1.3 Label forwarding information base

A table created by a label switch-capable device that tells where and how to forward frames with specific label values.

2.1.4 Edge label switch router (edge LSR)

Device that initially adds and then ultimately removes the labels from the packet is known as edge LSR.

2.1.5 Label switched path (LSP)

The path defined by the labels through LSRs mostly ingress router between label edge routers.

2.1.6 Label switch router (LSR)

A device such as a switch or a router that forwards labeled entities based upon the label value. Each LSR is also known as an MPLS node, must have the following:

- At least one layer 3 routing protocol.
- A label distribution protocol.
- The ability to forward packets based on their labels.

2.1.7 Upstream LSR and downstream LSR

Upstream LSR and downstream LSR are specific terms in the MPLS network, If LSR1 is forwarding label to LSR2 so LSR2 is upstream & LSR1 is downstream LSR.

The label is attached between the data link layer (Layer 2) header and network layer (Layer 3) header. The top of the label stack appears first in the packet and the bottom appears last. The layer 3 packet immediately follows the last label in the label stack. Figure 2 shows MPLS network. LER1 & LER9 are ingress & egress routers respectively; path between LER1-LSR2-LSR5-LSR7-LER9 is a label switched path.

2.2 MPLS Signaling Protocols

The way in which routers exchange their relevant information is known as signaling. The type of information exchanged between routers depends on the signaling protocol being used. At initial level, labels must be distributed to all MPLS enabled routers that are expected to forward data for a specific FEC and LSPs created. The MPLS framework does not assume any single signaling protocol and so far four methods have been specified for label distribution within the MPLS network [4], [9].

- Label Distribution Protocol (LDP)
- Resource Reservation Protocol (RSVP)
- Constrained Routing with LDP (CR-LDP)
- Distributing labels with BGP-4

2.2.1 Label Distribution Protocol

Label Distribution Protocol (LDP) is designed by IETF for the explicit purpose of distributing labels in MPLS network or setting up LSPs in the MPLS domain. LDP works closely with Interior Gateway Protocol (IGP) routing protocol. LDP often called as hop-by-hop forwarding. It always selects the same physical path that conventional IP routing would select. In conventional IP routing the next hop for each packet is found by a longest match prefix lookup on the IP header in the routing table [6]. These lookup could in some cases where the routing tables were large be time consuming and it was thought that data forwarding with label switching instead of IP lookups would speed up data forwarding. In MPLS domain two LSRs must agree on the meaning of the labels used to forward traffic between and through them. A FEC is associated with each LSP created. This FEC specifies which packets are mapped to that LSP. Two LSRs which use LDP to exchange label mapping information are known as LDP peers and they have an LDP session between them. There are four sorts of LDP messages, Discovery messages, Session messages, Advertisement messages and Notification messages [6], [10].

Discovery messages announce and maintain the presence of an LSR in an MPLS domain. This message is periodically sent as a Hello message through a UDP port with the multicast address of all routers on this subnet.

Session message is sent to establish, maintain and delete sessions between LDP peers.

Advertisement messages create, change and delete label mappings for FECs.

Notification Messages provides status, diagnostic and error information [9].

2.2.2 Constraint-Based Label Distribution Protocol

Constraint-Based Label Distribution Protocol is an extension of LDP to support constraint based routed LSPs. The term constraint implies that in a network and against each set of nodes there are a set of constraint that must be satisfied for the link or links between two nodes to be chosen for an LSP. An example of a constraint is to find a path that needs a specific amount of bandwidth [9].

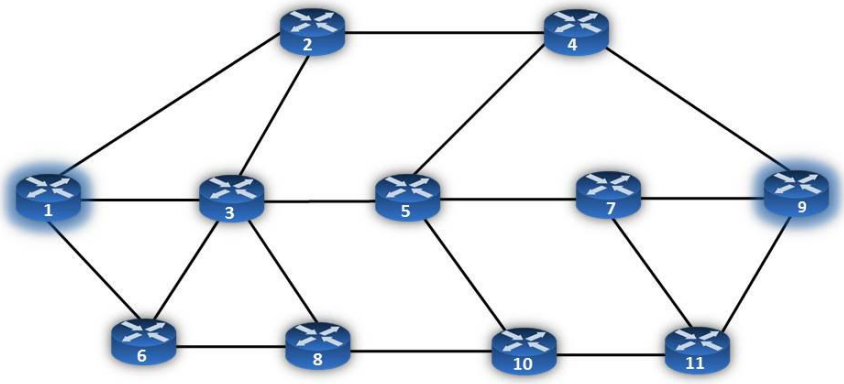

Figure 2: Example of MPLS network. LER-1 and LER-9 are Label Edge routers or Ingress and Egress respectively and all other are Label Switch Routers.

2.2.3 Resource Reservation Protocol

Resource reservation protocol is a label forwarding protocol in MPLS domain. Its best feature is its scalability. RSVP scales to very large multicast groups because it uses receiver-oriented reservation requests that merge as they progress up the multicast tree. The RSVP protocol defines a session as a data flow with a particular destination and transport-layer protocol. However, when RSVP and MPLS are combined, a flow or session can be defined with greater flexibility and generality. The ingress node of an LSP uses a number of methods to determine which packets are assigned a particular label. Once a label is assigned to a set of packets, the label effectively defines the flow through the LSP [20].

2.2.4 Border Gateway Protocol (BGP)

The Border Gateway Protocol (BGP) is also used for label distribution in MPLS network. BGP is a routing protocol used between different autonomous systems to exchange routing information. The update messages in BGP-4 that are used to distribute BGP routes can additionally carry the appropriate MPLS labels that are mapped to the same BGP route. The label mapping information for a particular route is piggybacked in the same BGP update message that is used to distribute the route itself [5].

2.3 Network Simulator NS2

In 1995 Lawrence Berkeley National Laboratory (LBNL) developed Network Simulator with support of Defense Advanced Research Projects Agency (DARPA). Later ns2 was extended and distributed by Virtual InterNetwork Testbed (VINT) [19]. Latest version of ns2 (ns-2.34) was released in June 2009. ns2 supports different platforms like Linux, Windows. ns2 has different components; NS for Simulation, Network AniMator (NAM) for visual demonstration of ns output, pre-processing for hand written TCL or topology generator and Post analysis for trace file using Perl, Tcl, AWK and MATLAB [15].

Now ns2 is used in the simulation of routing and multicast protocols and is mostly used in ad-hoc networking research. NS2 uses two types of languages system level languages C, C++ and scripting language TCL (OTcl). C++ is used for creation of objects because of its speed and efficiency. OTcl is used as a front-end to setup the simulator configures objects and schedule events.

3 RELATED WORK

3.1 MPLS Fault Recovery Techniques

For fault tolerance in MPLS networks there are several schemes and algorithms are developed, some of them are related to the domain of "Protection Switching" and some of them are "Rerouting". They are generally tested on major criteria's like, Recovery Time, Packet Loss and Multiple Fault Tolerance [1]. Most important criteria we focused are recovery time and multiple fault tolerance though multiple faults are very rare but have strong impact on reliability of the network. The two major types of recovery schemes that are used for MPLS recovery are Protection Switching and Rerouting are defined under.

3.1.1 Protection Switching

Protection switching is a recovery scheme in which recovery label switch path(s) are pre-computed or pre-established before a failure occurs on the working label switch path. When the fault occurs and Path Switch LSR (PSL) receives the Fault Identification Signal (FIS) it switches the traffic to the pre-established recovery path. As the recovery paths are pre-established so PSL immediately transfers the traffic on the backup path after receiving the FIS, this makes protection switching faster than rerouting [1], [2], [4], [11]. Resources required on establishment of recovery path are pre-reserved. Protection switching pre-establish a recovery path or path segment based upon network routing policies, the restoration requirements of the traffic on the working path & administrative considerations [1], [8].

3.1.2 Rerouting

Rerouting is a fault recovery technique where a recovery path is established on demand after a fault occurs. The recovery path can be based on fault information, network routing policies and network topology information [1], [3], [14]. An advantage with recovery by rerouting is that it does not take up any backup resources in the network before the recovery path is signaled. The new paths may be based upon fault information, network routing policies, pre-defined configurations and network topology information. Thus signaling is used to bypass the traffic. On the other hand rerouting has the disadvantage that resources may not be available at the time of computing recovery path that may leads to major failure [4], [8].

3.2 Positioning of Recovery Path

After the computation of recovery path or if the path is pre-computed by protection switching technique path can be placed locally or globally.

3.2.1 Local Repair

In local recovery, the recovery path selection or switching is done by a label switch router (LSR), which is nearest to the failed router or link. The main function of local repair is to fix the problem at the point of failure or within a very short distance from the failure for minimizing total packet loss and recovery time. In other words local repair aims to protect against a link failure or neighbour node failure and to minimize the amount of time required for propagation of failure signal [8], [14]. If a repair can be performed local to the device that detects the failure, restoration can be achieved faster. In local

repair, the immediately upstream LSR of the failure is the LSR that initiates the recovery operation (PSL) [3].

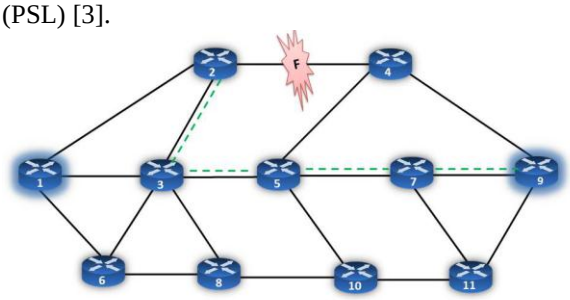

Figure 3: Local Repair.

3.2.2 Global Repair

Global recovery the alternative backup path selection is done by Protection Switch LSR. There is an alternative LSP that is pre-established or computed dynamically from ingress to egress routers. Ingress router is the entry point of MPLS network and Egress router the end point of MPLS Network. In other words global repair protect against any link or node failure on a path or on a segment of a path. In global repair the Point of Repair (POR) is distant from the failure and needs to be notified by a FIS [2], [12]. Recovery path is completely disjoint from the working path. This has the advantage that all links and nodes on the working path are protected by a single recovery path and having the disadvantage that a FIS has to be propagated all the way back to the ingress LSR before recovery can start.

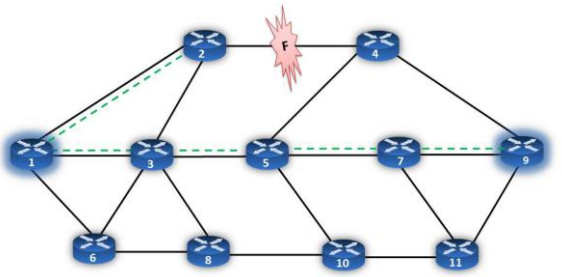

Figure 4: Global Repair.

4 PROPOSED PROTOCOL

This section describes the proposed fault recovery protocol that belongs to protection switching domain. In proposed fault recovery protocols first part, upon reception of FIS from core routers ingress router will check the explicit route against the link failure if it finds the route against that link it will transfer the traffic on it else it will terminate and that will be the worst condition. Now if fault or link failure occurs in the backup path i.e. explicit route then the core routers of that explicit route will send FIS to ingress.

Algorithm 1: Proposed Fault Recovery Protocol.

Protocol running on Ingress LSR,

```
1.  Upon reception of FIS
2.    if path Not Found against failure link
3.      then terminate algorithm
4.    else
5.      switch traffic to backup path
6.    if FIS received through backup path
7.      then
8.      if path Not Found against failure link
9.        then terminate algorithm
10.     else
11.       switch traffic to second backup path
12.   if original working path restored then
13.     switch traffic to it
```

Protocol running on Core LSR,

```
1.  send Keep alive messages
2.    if no acknowledgement then
3.      send FIS to Ingress
```

After that when ingress will receive the FIS it will again follow the same procedure and traffic will be sent on the second backup path. Meanwhile all MPLS routers will send Keepalive messages; if the original link is restored then traffic will be sent over it.

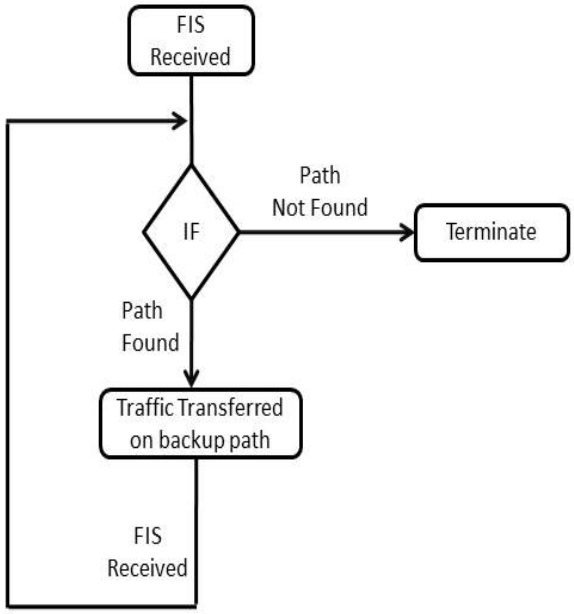

Figure 5: Flow chart diagram of proposed protocol

The second part of the proposed fault recovery protocol will be running on core LSRs of MPLS network. All the core routers of the MPLS domain will send Keepalive messages to each other, if there is no acknowledgements from any of the router the

adjacent router will FIS to ingress router so that ingress will transfer the traffic on the pre-established backup path. Figure 5 shows the flow chart diagram of proposed fault recovery protocol.

Rerouting protocol that is implemented in network simulator using the rerouting fault recovery technique which establishes the path on demand is as follows [15];

Algorithm 2: Rerouting Fault Recovery Protocol

1. **Upon** reception of FIS
2. calculate backup path using SPF **then**
3. **send** traffic to backup path
4. **if** FIS received through backup path **then**
5. calculate the shortest path using SPF
6. **send** traffic to computed backup path
7. **if** original working path restored
8. **switch** traffic to it

The above mentioned protocol is rerouting fault recovery protocol that is already in NS2. When ingress LSR receives the FIS from the core LSR then it computes the backup path (path on demand) and transfers the traffic on it. The backup path is being computed on time by using shortest path first algorithm that computes the shortest path between ingress and egress LSRs and transfer the traffic on it. If fault occurs in the backup path as well so on FIS request ingress again computes the second backup path and transfers the traffic on it. When the original link is restored then it shifts the traffic on the original working path.

5 DISCUSSION & RESULTS

We have established three different MPLS based network topologies. First two topologies are using 11 nodes, in which node0 and node10 are non-MPLS nodes, other nodes form node1 to node9 are MPLS capable nodes as shown in figure 6 and figure 7. For more reliability of the proposed protocol we created third topology having 17 nodes having 2-non MPLS and 15 MPLS nodes as shown in figure 8.

For network topology-A as shown in figure 6 we implement both protocols on it, rerouting fault recovery protocol and proposed fault recovery protocol. The first working path is LSR1-LSR7-LSR9-LSR6. After the occurrence of fault in link between LSR7-LSR9, rerouting fault recovery protocol computes the backup path LSR1-LSR7-LSR8-LSR9-LSR6 and transfers the traffic on it in 0.04ms. Whereas proposed fault recovery protocol

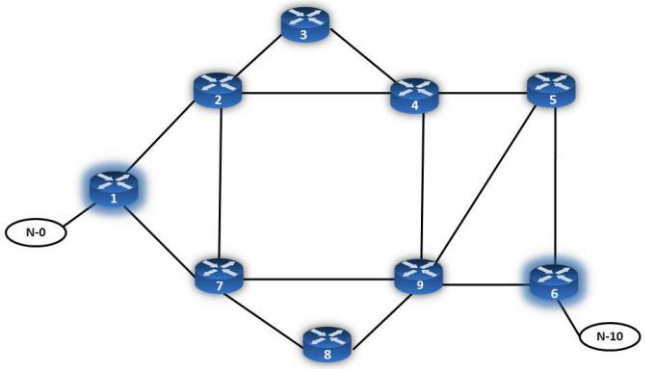

Figure 6: Network Topology A

transfers the traffic on the backup path LSR1-LSR2-LSR4-LSR5-LSR6 in 0.01ms because backup path was already computed and stored in ingress, it just took the time to send FIS to the ingress. And when we administratively down the link between LSR1-LSR7 in case of rerouting protocol and LSR2-LSR4 in case of proposed protocol, the rerouting protocol took 0.07ms to switchover the traffic on the second backup path i.e. LSR1-LSR2-LSR4-LSR5-LSR6 and proposed protocol took 0.02ms to switchover the traffic on the backup path LSR1-LSR2-LSR3-LSR4-LSR5-LSR6. For the third fault rerouting fault recovery protocol switches the traffic to path LSR1-LSR2-LSR7-LSR8-LSR9-LSR6 in 0.02ms and proposed fault recovery protocol switches the traffic to path LSR1-LSR7-LSR8-LSR9-LSR6 in 0.01ms.That is again less as compare to rerouting protocol.

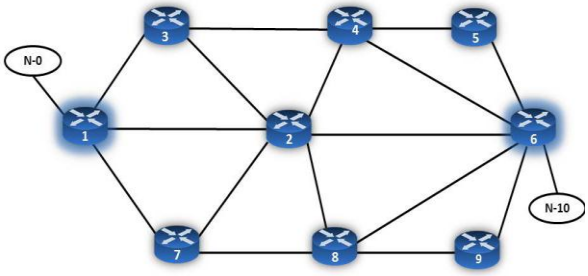

Figure 7: Network Topology B

Figure 7 shows the MPLS based network topology-B. We implemented both protocol on this topology as well. N-0 and N-10 are non-MPLS nodes and LER-1 and LER-6 are ingress and egress LSRs respectively. The first working path is LSR1-LSR2-LSR6. After the occurrence of fault in link between LSR2-LER6, rerouting fault recovery protocol computes the backup path LSR1-LSR2-LSR4-LSR6 and transfers the traffic on it in 0.03ms. Whereas proposed fault recovery protocol transfer the traffic on the backup path LSR1-LSR2-LSR4-LSR6 in 0.01ms because backup path was already

computed and stored in ingress, it just took the time to send FIS to the ingress. And when we administratively down the link between LSR1-LSR2 in case of rerouting protocol and proposed protocol, the rerouting protocol took 0.058ms to switchover the traffic on the second backup path i.e. LSR1-LSR7-LSR8-LSR6 and proposed protocol took 0.005ms to switchover the traffic on the backup path LSR1-LSR7-LSR8-LSR6. That is again less as compare to rerouting protocol.

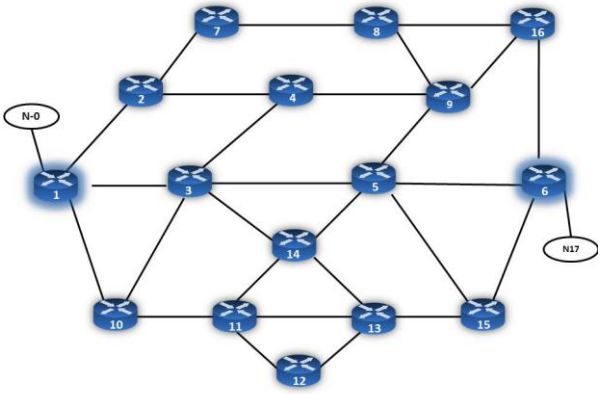

Figure 8: Network Topology-C

Figure 8 shows the third network topology having 15 MPLS nodes and 2 IP nodes. The first working path is established between LER1-LSR3-LSR5-LER6. The first link was down between LSR3-LSR5. Proposed fault recovery protocol transfers the traffic to the backup path LER1-LSR3-LSR14-LSR5-LER6 in 0.016ms while rerouting protocol took 0.049ms. The second link was down between LEr1-LSR3. The proposed protocol transferred the traffic to label switch path LER1-LSR10-LSR3-LSR14-LSR5-LER6 in 0.009ms and rerouting protocol in 0.05ms. Finally when the third link was down between LSR10 and LSR3, Proposed protocol shifts the traffic to backup path LER1-LSR2-LSR4-LSR9-LSR16-LER6 in 0.006ms and rerouting in 0.025ms.

We compared the rerouting fault recovery protocol with our proposed fault recovery protocol. Rerouting fault recovery protocol uses rerouting scheme in which LSP is computed after the occurrence of fault. Furthermore rerouting scheme recover faults in two parts (time). Firstly time taken to send FIS to the ingress in global recovery and to upstream LSR in case of local recovery plus the time for computing new LSP and transferring traffic on the backup path.

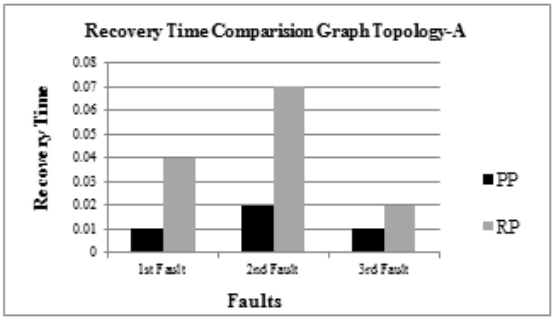

Figure 9: Comparison graph for Topology-A

Whereas in case of our proposed protocol that is purely from protection switching domain requires time to send FIS to ingress or upstream LSR. Because backup paths are already computed so when ingress LSR receives the FIS it directly transfers the traffic to backup path.

Figure 9, figure 10 and figure 11 show the fault recovery time of proposed fault recovery protocol and rerouting fault recovery protocol of network topology A,B and C respectively.

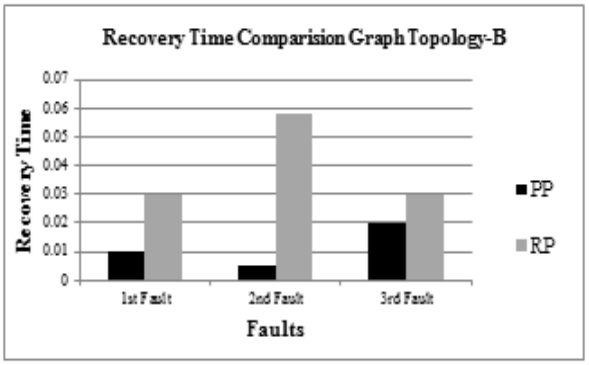

Figure 10: Comparison graph for Topology-B

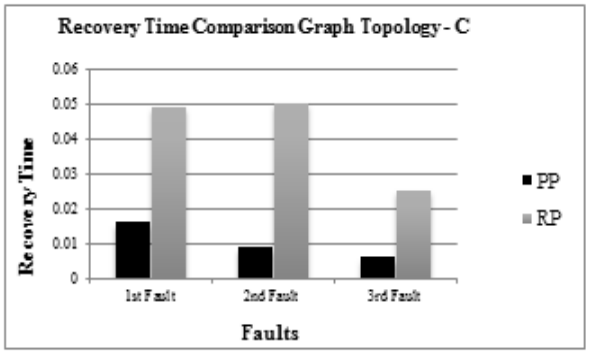

Figure 11: Comparison graph for Topology-C

6 CONCLUSION & FUTURE EXTENSIONS

In this paper we focused on multiple fault tolerance in MPLS network. We created three

different network topologies and implemented both protocol on it, rerouting fault recovery protocol and proposed fault recovery protocol. Rerouting fault recovery protocol uses rerouting domain for fault tolerance and it computes recovery path on demand after the occurrence of the fault whereas proposed fault recovery algorithm is from protection switching domain in which recovery paths are pre-computed. Through simulation in NS2 we observed that proposed fault recovery protocol took less time to switch over the traffic on the recovery path as compare to rerouting fault recovery protocol. Total time that rerouting fault recovery protocol took to recover from a particular fault is time for sending FIS to the ingress plus time to computing backup path and sending traffic on it. In proposed fault recovery protocol total time is just to send FIS to ingress then ingress will automatically transfer the traffic to the backup path.

Future extension to this work is to determine and solve the packet reordering case.

7 REFERENCES

[1] Maria Hadjiona, Chryssis Georgiou, Maria Papa, Vasos Vassiliou “A Hybrid Fault Tolerant Algorithm for MPLS Networks”. WWIC'08 Proceedings of the 6th international conference on Wired/wireless internet communications Pages 41-52 2008.

[2] Jack Foo. A Survey of Service Restoration Techniques in MPLS Networks. 2003.

[3] V. Alarcón-Aquino Y. L. Takahashi-Iturriaga J. C. Martínez-Suárez L. G. Guerrero-Ojeda. MPLS/IP Analysis and Simulation for the Implementation of Path Restoration Schemes. Proceeding AIC'04 Proceedings of the 4th WSEAS International Conference on Applied Informatics and Communications 2004 Article No.37.

[4] Johan Martin. MPLS Based Recovery Mechanisms. Master Thesis, UNIVERSITY OF OSLO May 2005.

[5] Cisco Learning Network. Learning MPLS Concepts https://learningnetwork.cisco.com.

[6] G.Ahn, W.Chun, Design and Implementation of MPLS Network Simulator Supporting LDP and CR-LDP. The Proceedings of the IEEE International Conference on Networks (ICON'00).

[7] G.Kaur, D.Kumar, MPLS Technology on IP Backbone Network, International Journal of Computer Applications (0975-8887), 2010.

[8] V.Sharma. Framework for Multi-Protocol Label Switching (MPLS)-based Recovery, RFC-3469, 2003.

[9] L.Andersson. LDP Specification, RFC-5036, October 2007.

[10] Rosen E, Viswanathan A, Callon, R. Multiprotocol Label Switching Architecture. RFC 3031, January 2001.

[11] Stephen Shew. Internet Draft. Fast Restoration of MPLS Label Switched Paths, draft-shew-lsp-restoration-00, October 1999.

[12] Yimin Qiu, Jianxun Chen, Jinguang Gu, Xin Xu, A Simulation for MPLS Global Recovery Model, First International Conference on Intelligent Networks and Intelligent Systems, Pages 259-262, 2008.

[13] http://www.omnetpp.org

[14] Svetlin Petrov. An approach for MPLS Recovery. Proceeding CompSysTech '07 Proceedings of the 2007 international conference on Computer systems and technologies 2007.

[15] http://www.isi.edu/nsnam/ns/

[16] http://www.gns3.net

[17] http://totem.run.montefiore.ulg.ac.be

[18]http://www.opnet.com/solutions/network_rd/modeler

[19] http://www.isi.edu/nsnam/vint/

[20] Swayam Prakasha, Network & Services, Resource Reservation Protocol January 2007.