Traffic Engineering through MPLS in Service Provider Networks

Y Rama Devi,Ph.D
Professor, CSE Dept.
CBIT,Hyderabad, INDIA

B. Ramadasu
Asst.Professor, CSE Dept.
CBIT,Hyderabad, INDIA

B.H. Raj Gopal
CSE Dept.
CBIT,Hyderabad, INDIA

ABSTRACT

MPLS was originally designed to make IP routers are as fast as ATM switches for handling traffic. Just as in any other technology.MPLS has a specialized terminology all its own. An MPLS domain is the collection of routers running MPLS under the control of a single Administrator. An LSP (label-switched path) is a one-way (unidirectional) Flow of traffic, carrying packets from beginning to end. Packets must enterThe LSP at the beginning (ingress) of the path, and can only exit the LSP At the end (egress). Packets cannot be injected into an LSP at an intermediate hop. Generally, an LSP remains within a single MPLS domain. That is, the Entrance and exit of the LSP, and all routers in between, are ultimately in control of the same administrative authority. This ensures that MPLS LSP Traffic engineering is not done haphazardly or at cross purposes but is implemented in a coordinated fashion.

Keywords

MPLS, LSP, LSR, ERO, RSVP, QOS

1. INTRODUCTION

MPLS was originally designed to make IP routers as fast as ATM switches for handling traffic. It is still commonly believed that MPLS somehow significantly enhances the forwarding performance of label-switching routers. However, it is more accurate to say that exact-match lookups, such as those performed by MPLS and ATM switches, historically have been faster than the longest-match lookups performed by IP routers. The real benefit of MPLS is that it provides a clean separation between routing (that is, control) and forwarding (that is, moving data). This separation allows the deployment of a single forwarding algorithm—MPLS—that can be used for multiple services and traffic types. IP prefixes are totally bound to a physical next-hop destination. This next-hop resolution usually is based solely on the IGP shortest path calculation. That is, only the topology counts when link metrics are used as the basis of next-hop calculations. Without MPLS, in the traditional Layer 3 forwarding paradigm, packet travels from one router to the next, the router at each hop makes an independent forwarding decision for every packet. The router analyzes the IP network layer header and chooses the next hop based on this analysis and on information contained in the routing table. In an MPLS-based traffic engineering environment, the router performs the analysis of the packet header just once right before the packet enters the engineered path. Based on this mapping, the router assigns the packet a label, which is a short, fixed-length value placed at the front of the packet. Routers in the traffic engineering path use labels as lookup indexes into the label forwarding table. Ingress router encapsulates an IP packet that will use this LSP to destination by adding the 32-bit MPLS shim header in between layer2 and layer3. Transit routers forward a received MPLS packet to the next hop in the MPLS path. 0 or more transit router's can exist. Transit routers perform label swap operation. Penultimate router is second to last router Pop's the label stack if it has label as 3 and unlabelled packet sent to egress router . Egress router removes the MPLS encapsulation header forwards the packet toward its final destination using the normal IP forwarding table. An LSP (label-switched path) is a one-way (unidirectional) flow of traffic, carrying packets from beginning to end. Packets must enter the LSP at the beginning (ingress) of the path, and can only exit the LSP at the end (egress). Packets cannot be injected into an LSP at an intermediate hop. Another key MPLS term is the label-switching router (LSR). An LSR understands and forwards MPLS packets, which flow on, and are part of, an LSP. In addition, an LSR participates in constructingLSPs for the portion of each LSP entering and leaving the LSR. For a particular destination, an LSR can be at the start of an LSP, the end of an LSP, or in the middle of an LSP.

2. EXISTING SYSTEM

2.1 RSVP (Resource Reservation Protocol's) Signalled Lsp's

Lsp's can be established in two ways statically or dynamically via rsvp (signaling protocol). The primary problems with static LSPs are their lack of keep alive indication and their need for complex manual configuration, which is prone to error. Signaled LSP requires explicit configuration at the ingress LSR only. Once ingress node is told what it wants, the signaling protocol gets to work and establishes the required LSP's state in the transit and egress routers that makes up the LSP's path. The Resource Reservation Protocol (RSVP) is a generic signaling protocol designed originally to be used by applications to request and reserve specific quality-of-service (QOS) requirements across an internetwork. Resources are reserved hop by hop across the internetwork. Each router receives the resource reservation request, establishes and maintains the necessary state for the data flow (if the requested resources are available), and forwards the resource reservation request to the next router along the path. But how does the ingress router know which routers should be involved in the LSP's path? In the most basic case, the path through the network can be chosen at each hop by just looking in the routing table. This behavior is possible because the collection of routers comprising the MPLS domain run an IGP that provides routing to internal destinations.

3. PROPOSED SYSTEM

3.1 MPLS (Traffic Engineering)

MPLS traffic engineering is achieved by adding additional conditions to this lookup process. One such approach makes use of explicit route objects (EROs). There are two types of

EROs, strict and loose. Strict EROs must point to a directly connected downstream router. Loose EROs identifies LSRs that are not necessarily directly connected. A loose ERO relies on the underlying IGP to route between loose transit points.

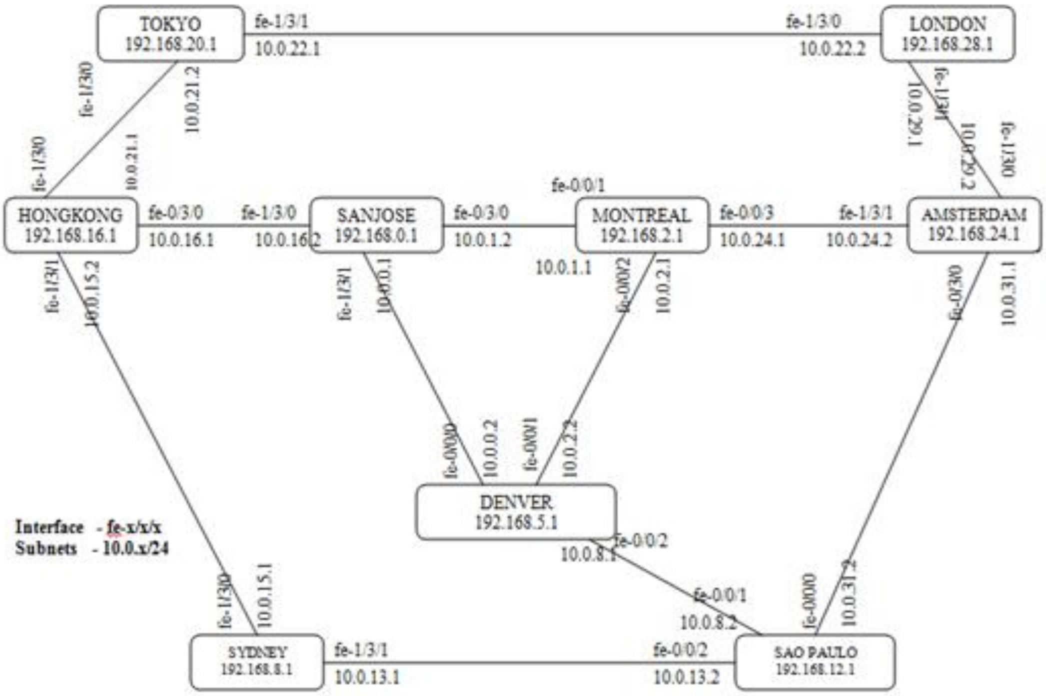

Figure 1 LAB TOPOLOGY ISILICA NETWORKS

4. CODE (in Tokyo router)

4.1 To set ip address for Tokyo router's interfaces type the following code

root@Tokyo# set interfaces fe-1/3/0 unit 0 family inet address 10.0.21.2/24
root@Tokyo# set interfaces fe-1/3/1 unit 0 family inet address 10.0.22.1/24
root@Tokyo# set interfaces lo0 unit 0 family inet address 192.168.20.1
root@Tokyo# commit

4.2 Enable ospf on the interfaces

First define the ospf area to which the router will connect and then enable ospf on the interfaces on which ospf is needed to be run.

root@Tokyo# set protocols ospf area 0 interface fe-1/3/0
root@Tokyo# set protocols ospf area 0 interface fe-1/3/1
root@Tokyo# set protocols ospf area 0 interface lo0
root@Tokyo# commit

4.3 Assign AS (autonomous system) number

A router is assigned an AS by setting the autonomous system number

root@Tokyo# set routing-options autonomous-system 100
root@Tokyo# commit

4.4 Tokyo router needs to form an internal BGP session with its neighbor's by typing following code

root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.16.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.0.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.2.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.24.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.28.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.5.1

root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.8.1
root@Tokyo# set protocols BGP group ibgp type internal local-address 192.168.20.1 neighbor 192.168.12.1
root@Tokyo# commit

4.5 Baseline MPLS

Add labeled packet support on all interfaces, add the mpls family to each logical interface that is intended to be used for mpls by typing the following code

root@Tokyo# set interfaces fe-1/3/0 unit 0 family mpls
root@Tokyo# set interfaces fe-1/3/1 unit 0 family mpls
root@Tokyo# commit

Enable the MPLS instance on the router so that labeled traffic can be processed appropriately by typing the following code

root@Tokyo# set protocols mpls interface fe-1/3/0
root@Tokyo# set protocols mpls interface fe-1/3/1
root@Tokyo# commit

Enable RSVP on the interface by typing the following code

root@Tokyo# set protocols rsvp interface fe-1/3/0
root@Tokyo# set protocols rsvp interface fe-1/3/1
root@Tokyo# commit

Adding RSVP to MPLS allows router to function as a transit or egress LSR in support of RSVP-signaled LSP's. Additional configuration is needed for the router to function as ingress LSR. In a similar manner all the operations are also performed in **Sydney, Saopaulo, Sanjose, Montreal, London, Hong-Kong, Denver, and Amsterdam**

4.5 Status of interface and protocols enabled at Tokyo router

To know status of interface enabled at Tokyo router type the following code and get the output

root@Tokyo# show interfaces | no-more

```
fe-1/3/0 {
    unit 0 {
        family inet {
            address 10.0.21.2/24;
        }
        family mpls;
    }
}
fe-1/3/1 {
    unit 0 {
        family inet {
            address 10.0.22.1/24;
        }
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.20.1/32;
        }
    }
}
```

To know status of protocols enabled at Tokyo router type the following code to get the output

root@Tokyo# show protocols | no-more

```
rsvp {
    interface fe-1/3/0.0;
    interface fe-1/3/1.0;
}
mpls {
    interface fe-1/3/1.0;
    interface fe-1/3/0.0;
}
bgp {
    group ibgp {
        type internal;
        local-address 192.168.20.1;
        neighbor 192.168.28.1;
        neighbor 192.168.16.1;
        neighbor 192.168.0.1;
        neighbor 192.168.2.1;
        neighbor 192.168.24.1;
        neighbor 192.168.5.1;
        neighbor 192.168.8.1;
        neighbor 192.168.12.1;
    }

}
ospf {
    area 0.0.0.0 {
        interface fe-1/3/0.0;
        interface fe-1/3/1.0;
        interface lo0.0;
    }
}
```

5. RESULT ANALYSIS

5.1 Establish a signaled LSP from Tokyo to Sao Paulo without traffic engineering

The following code is typed in the ingress router (Tokyo) to establish a signaled LSP from Tokyo to Sao Paulo. CSPF is discarded to avoid the need for traffic engineering database

root@Tokyo# set protocols mpls label-switch-path lsp to 192.168.12.1 no-cspf
root@Tokyo# commit

Type the following code to check whether the created tunnel is up or not.

root@Tokyo# run show mpls lsp
Ingress LSP: 1 session

To	**From**	**State**	**Rt**	**Active Path**
192.168.12.1	**192.168.20.1**	**up**	**1**	

5.2 Explicit route (traffic engineering) at ingress router (Tokyo)

Path statement is used to configure explicit route

root@Tokyo# set protocols mpls path mypath 192.168.16.1 strict
root@Tokyo# set protocols mpls path mypath 192.168.0.1 strict
root@Tokyo# set protocols mpls path mypath 192.168.5.1 strict
root@Tokyo# set protocols mpls path mypath 192.168.12.1 strict
root@Tokyo# commit

Use primary statement to tie the LSP to the route which is created explicitly. When the path is applied to the LSP it is applied as primary path so that all the traffic on the lsp uses the explicitly configured path

root@Tokyo# set protocols mpls label-switched-path lsp primary mypath
root@Tokyo# commit

root@Tokyo# run show rsvp session extensive | no-more
Ingress RSVP: 1 sessions 192.168.12.1
From: 192.168.20.1, LSPstate: Up, ActiveRoute: 1

Total 1 displayed, Up 1, Down 0
Egress LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
Transit LSP: 0 sessions
Total 0 displayed, Up 0, Down 0
root@Tokyo# run show rsvp session extensive | no-more
Ingress RSVP: 1 sessions 192.168.12.1
From: 192.168.20.1, LSPstate: Up, ActiveRoute: 1
LSPname: lsp,LSPPath: Primary
Suggested label received -, Suggested label sent:-
Recovery label received -,Recovery label sent:299776
Resv style : 1FF , Label in-,Label out:299776
Time left : - , Since : Fri jan 1 02:43:37 1999
Tspec: rate 0 bps size 0bps peak Infbps m 20 M 1500
Port number : sender 1 receiver 10680 protocol 0
PATH rcvfrom : localclient
Adspec : sent MTU 1500
PATH MTU received 1500
PATH sent to: 10.0.22.2 (fe-1/3/1.0) 8 pkts
RESV rcvfrom : 10.0.22.2 (fe-1/3/1.0) 8 pkts
Record route: <self> 10.0.22.2 10.0.29.2 10.0.31.2
Total 1 displayed, Up 1, Down 0
Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
The output (Record route) shows that the lsp takes the following default IGP path without traffic engineering i.e.TOKYO-LONDON-AMSTERDAM-SAOPAULO

LSPname: lsp,LSPPath: Primary
Suggested label received -, Suggested label sent:-
Recovery label received -,Recovery label sent:299776
Resv style : 1FF , Label in-,Label out:299776
Time left : - , Since : Fri jan 1 02:58:44 1999
Tspec: rate 0 bps size 0bps peak Infbps m 20 M 1500
Port number : sender 1 receiver 10684 protocol 0
PATH rcvfrom : localclient
Adspec : sent MTU 1500
PATH MTU received 1500
PATH sent to: 10.0.21.1 (fe-1/3/0.0) 4 pkts
RESV rcvfrom : 10.0.21.1 (fe-1/3/1.0) 4 pkts
Explct route: 192.168.16.1 192.168.0.1 192.168.5.1 192.168.12.1
Record route: <self> 10.0.21.1 10.0.16.2 10.0.0.2 10.0.8.2
Total 1 displayed, Up 1, Down 0
Egress RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0
Transit RSVP: 0 sessions
Total 0 displayed, Up 0, Down 0

The output (**Explct route**) shows that the lsp takes the MPLS traffic engineered path i.e. **TOKYO-HONGKONG-SANJOSE-DENVER-SAOPAULO**

5.3 Simulation setup

The simulations are often used for understanding and prediction of the behavior of protocols and data streams in networks. All simulation results in this paper are obtained using NS2 simulator. Figure 2 shows the network created for the simulation.

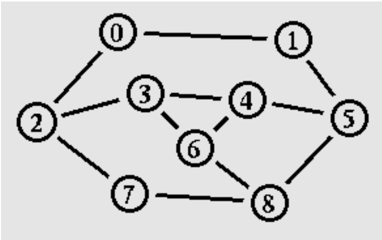

Figure 2 (NS2 topology)

5.4 Simulation results

In figure 3 the packets are sent in (MPLS) traffic engineered path . The topology has nine nodes connected to each other via TCP and UDP connections.

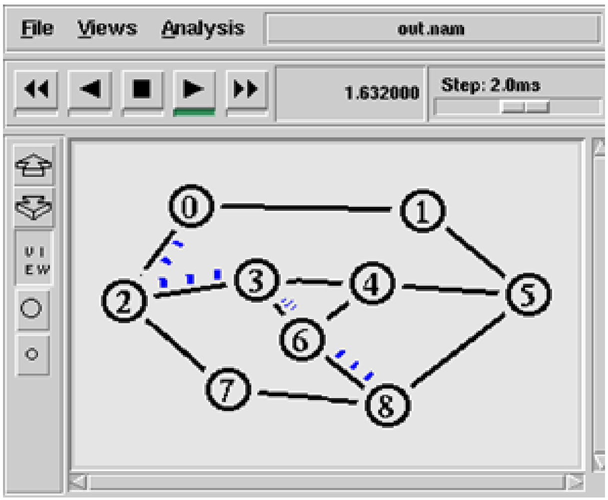

Figure 3 MPLS (traffic engineered path)

In figure 4 the packets are sent in default OSPF (open shortest path first) path .

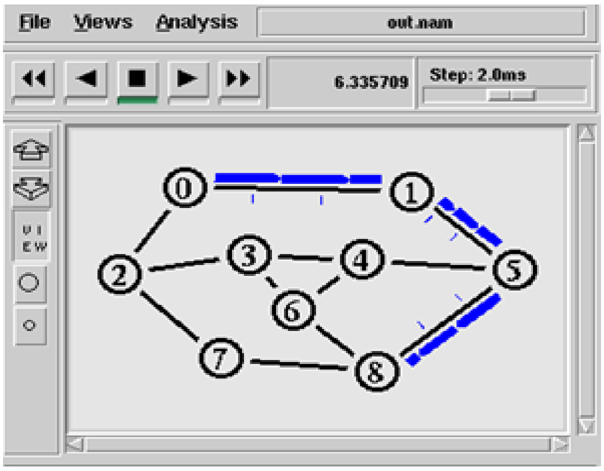

Figure 4 Default OSPF path

In figure 5 the graph shows the delay per packet in ingress router **TOKYO,** X-axis denotes packet id and Y-axis denotes delay faced by ingress router **TOKYO** in milliseconds

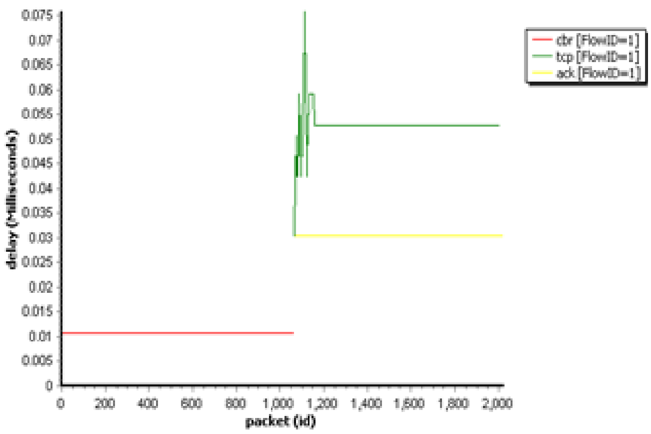

Figure 5 Delay per packet

6. CONCLUSION

Multi-Protocol Label Switching (MPLS) builds on top of that foundation and really grants you a different level of control over how your network transports traffic. By converting routed network to something closer to a switched network, MPLS offers transport efficiencies that simply aren't available in a traditional IP-routed network. In an MPLS-based traffic engineering environment, the router performs the analysis of the packet header just once right before the packet enters the engineered path. Based on this mapping, the router assigns the packet a label, which is a short, fixed-length value placed at the front of the packet. Routers in the traffic engineering path use labels as lookup indexes into the label forwarding table. An MPLS label-switched path (LSP) is unidirectional. MPLS is used in service provider networks as resources can be reserved depending on the requirements and MPLS over comes the drawback of IPV4 routing by only processing the packet in ingress and egress node and also providing traffic engineering feature for forwarding of packets

7. REFERENCES

[1] Introduction to junos operating system juniper networks

[2] JNCIS-ENT Routing Study Guide

[3] Juniper Networks Mpls Fundamentals

[4] Juniper Networks RSVP-Signaled LSPs

[5]Juniper Networks Named Paths and Routing Constraints

[6] Michael Bushong, Cathy Gadecki, Aviva Garrett "Junos for dummies" 2008

[7] "Marc Greis's ns2 tutorial"

http://www.isi.edu/nsnam/ns/tutorial/index.html

[8] Kevin Fall, Kannan Varadhan, "The ns Manual" November 4, 2011