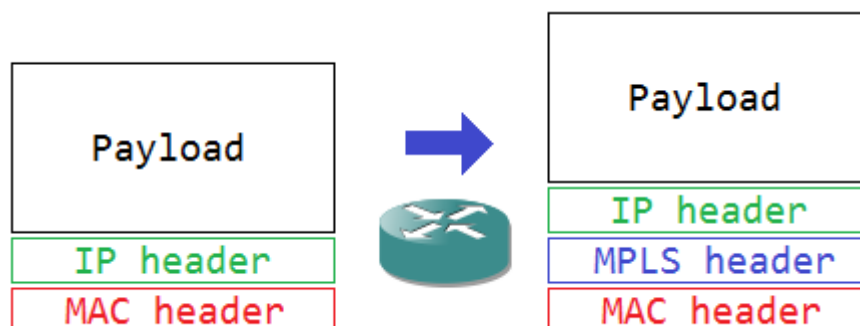


MPLS, PART I: BASIC LABELING

Multi Protocol Label Switching. Plenty of sites to explain it, yet I feel it's something to cover on my blog, it just wouldn't be complete without it. What is it? In the most simple form: it's the use of labels, inserted on frames between layer 2 and 3, and use those labels to make forwarding decisions instead of the original layer 3 header.

Why is it done? Well, in the late 1990's switching in hardware was possible, but routing (layer 3 forwarding) in hardware wasn't. By using labels as a layer 3 IP header replacement, these labels could be used in more efficient forwarding tables, and eventually also in hardware. This meant lower latency in routed networks.

So that's why up until this day, it is still discussed whether MPLS is a layer 2.5 or layer 3 protocol. I'm not going to make a judgement on that: by the standards it's a layer 3 protocol (it's above the layer 2 header, and ip-in-ip and gre over ip are not called layer 2.5 or 3.5), but on the other hand, the original use is to replace the use of the actual layer 3 header in forwarding, so 2.5 is logical in that regard.

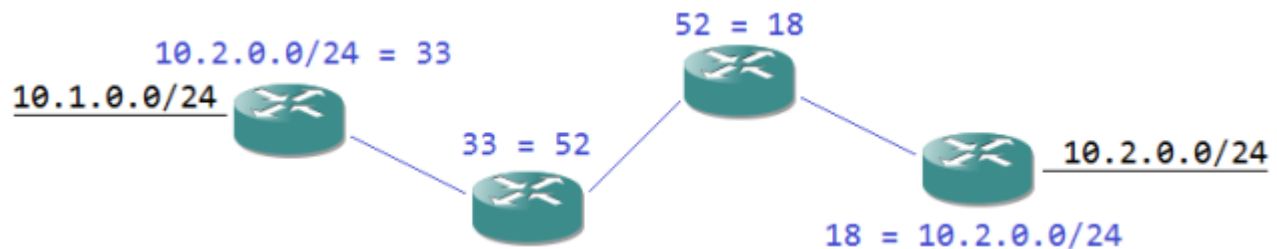


The MPLS header is 4 bytes or 32 bits (just like 802.1q!). Routers need to agree on which labels they will use towards each other. For this, they need to set up a protocol that exchanges label information: **Label Distribution Protocol (LDP)**. Cisco also has a proprietary Tag Distribution Protocol, but it's no longer widely used. LDP sends Hello/Discover frames on UDP port 646 to 224.0.0.2 (all routers multicast). Once neighbor routers see each other, they build up a session

on TCP port 646. The actual labels are automatically assigned and agreed on through these sessions. To enable MPLS and LDP on an interface, you'll need one command:

```
Router(config-if)#mpls ip
```

But the labels by themselves aren't enough. Each label represents a destination: in a basic environment, an IPv4 destination subnet. To know which label is needed for which destination, each MPLS router needs to know the possible destinations so it can assign a label to it: it needs a routing protocol. Generally speaking, it's a better move to choose a routing protocol with awareness of the topology, so a link state protocol like OSPF or IS-IS.



Once the routers know all routes and have set up TCP sessions for LDP, they can agree on which labels to use to reach a certain destination subnet. If the first router in the above picture receives a normal IPv4 packet, it sees in the forwarding table (the **F**orwarding **I**nformation **B**ase, or FIB) that the next destination is out of a MPLS enabled interface, with a label value of 33 appended to it. If the second router receives a MPLS packet with label value 33, it can do a quick lookup using its **L**abel **F**orwarding **I**nformation **B**ase (LFIB) and find a next hop out of another interface with a new label value. This process continues until it reaches the last router of the MPLS region, where the LFIB next hop points to an interface without MPLS. The MPLS header is removed and the packet is forwarded as usual.

The above explains basic MPLS functionality. But it's not everything: labels can stack, allowing for more complex configurations. In the next part, I'll explain the use of multiple labels to create VRF-aware forwarding.

Source : <http://reggle.wordpress.com/2013/03/13/mpls-part-i-basic-labeling/>