

MPLS (多协议标签交换) 技术原理的基本实现思科认证 PDF 转换可能丢失图片或格式, 建议阅读原文

[https://www.100test.com/kao\\_ti2020/630/2021\\_2022\\_MPLS\\_EF\\_B\\_C\\_88\\_E5\\_A4\\_9A\\_c101\\_630752.htm](https://www.100test.com/kao_ti2020/630/2021_2022_MPLS_EF_B_C_88_E5_A4_9A_c101_630752.htm) 首先, 我们需要5台路由器 (4台也可以), 因为MPLS是主要运用于大型的网络中的一种快速转发协议, 例ISP的网络基本采用MPLS来提高数据包的转发速度。分别编号R1-R5, 接口地址见topology, 全网起OSPF协议, AREA 0, 目的是达到全网互通, 以便于LABEL信息的传递。MPLS在网络中的实现很简单 (因为是简单的配置嘛): 1、把属于MPLS网络的接口划入MPLS网络 2、启用MPLS 下面简单说一下具体配置, 首先, 进入R1的s1/3端口, 做如下命令: R1(config)#int s1/2 R1(config-if)#mpls ip 把接口划入MPLS网络 R1(config-if)#exi 至于R2-R4的S1/2, S1/3, 以及R5的s1/3在这里不一一赘述, 原理和上面的一样。这时, 我们会看到如下提示: \*Mar 1 00:13:37.163:

```
%LDP-5-NBRCHG: LDP Neighbor 23.0.0.1:0 (1) is UP 说明MPLS  
邻居建立成功, 可以相互发送LABEL信息。我们查看一下目前路由器的LABEL转发表, 也就是和每条路由一一对应的标签编号。 R1#sh mpls forwarding-table Local Outgoing Prefix Bytes  
tag Outgoing Next Hop tag tag or VC or Tunnel Id switched  
interface 16 16 34.0.0.0/24 0 Se1/2 point2point 17 Pop tag  
23.0.0.0/24 0 Se1/2 point2point 18 18 192.168.1.1/32 0 Se1/2  
point2point 19 19 45.0.0.0/24 0 Se1/2 point2point R1# 我们可以看到, LOCAL列, 表示这条路由条目在本路由器的label编号, 也是它通告给邻居LSR的编号。 OUTGOING列, 表示送往邻居LSR时应该打上的标签编号, 也就是说, 在把包交给邻
```

居LSR之前，已经打上的TAG其实是对方LSR转发表里关于这条路由的TAG编号。Pop tag表示拆除TAG，因为目的网络是下一条的直连网络，所以不用TAG继续进行转发。我们注意到，bytes tag switched这一列全部为0，这是因为我们还没有进行任何的数据包交换。下面做如下操作：

```
R1#p 192.168.1.1
sou R1#ping 192.168.1.1 source 10.1.1.1 Type escape sequence to
abort. Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2
seconds: Packet sent with a source address of 10.1.1.1 !!!!! Success
rate is 100 percent (5/5), round-trip min/avg/max = 104/198/304 ms
R1#sh mpls forwarding-table Local Outgoing Prefix Bytes tag
Outgoing Next Hop tag tag or VC or Tunnel Id switched interface
16 16 34.0.0.0/24 0 Se1/2 point2point 17 Pop tag 23.0.0.0/24 0 Se1/2
point2point 18 18 192.168.1.1/32 0 Se1/2 point2point 19 19
45.0.0.0/24 0 Se1/2 point2point
```

我们发现，PING过以后R1上依然没有交换标签的BYTE流量，这是因为MPLS是不会对自己的直连网络打上TAG标签的，粗浅的理解就是直连网络直接转发就可以了，打上TAG不是更麻烦了吗？嘿嘿。好，我们看R2-R4的转发表：

```
R2#sh mpls forwarding-table Local Outgoing
Prefix Bytes tag Outgoing Next Hop tag tag or VC or Tunnel Id
switched interface 16 Pop tag 34.0.0.0/24 0 Se1/2 point2point 17
Untagged 10.1.1.1/32 520 Se1/3 point2point 18 18 192.168.1.1/32
540 Se1/2 point2point 19 19 45.0.0.0/24 0 Se1/2 point2point
```

注意中间两行的标签字节交换数量已经出现变化，说明在R2上进行了标签交换。此现象在R3,R4上基本相同，说明在R3,R4上同样进行着标签的交换。再看R4，R5的情况：

```
R4#sh mpls
forwarding-table Local Outgoing Prefix Bytes tag Outgoing Next
```

Hop tag tag or VC or Tunnel Id switched interface 16 Pop tag  
23.0.0.0/24 0 Se1/3 point2point 17 16 10.1.1.1/32 540 Se1/3  
point2point 18 17 12.0.0.0/24 0 Se1/3 point2point 19 Untagged  
192.168.1.1/32 520 Se1/2 point2point R5#sh mpls forwarding-table  
Local Outgoing Prefix Bytes tag Outgoing Next Hop tag tag or VC  
or Tunnel Id switched interface 16 Pop tag 34.0.0.0/24 0 Se1/3  
point2point 17 16 23.0.0.0/24 0 Se1/3 point2point 18 17 10.1.1.1/32  
0 Se1/3 point2point 19 18 12.0.0.0/24 0 Se1/3 point2point 没有TAG  
交换！因为目标网络是自己的直连，所以在R4路由器把包交给R5之前，已经拆除TAG（看看R4的信息就可发现，在R4中去往192.168网段的outgoing是untagged，就是去往这个网络不应该带有标签的意思），在R4交给R5以后，IP包直接进行路由转发，而不是继续经由MPLS进行TAG交换。以上只是简单的说了说MPLS的标签交换过程，下面我总结一下流程：首先，PING包源：10.1.1.1 目的：192.168.1.1 进入R1以后，查找forwarding-table，发现去往R2，出接口时，打上R2可以正确判断的18号标签-----包进入R2后，查找forwarding-table，发现包应该交给R3，也应该打上R3识别的19号标签（即outgoing对应的出标签）-----接着，给了R3，R3照旧给了R4，此时R4查询forwarding-table，发现outgoing对应的是Untagged，即转发到192.168.1.0这个网络的数据包此时不应该继续打上TAG，而是以正常的IP数据包格式进行路由转发。Then，数据包到达R5，路由成功。更多优质资料尽在百考试题论坛 百考试题在线题库 思科认证更多详细资料 100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)