MPLS 基础

MPLS 有哪些设备角色，它们分别有什么作用？
（1）LSR：可以进行 MPLS 标签交换和报文转发的网络设备称为标签交换路由器（LabelSwitchingRouter），由 LSR 构成的网络区域称为 MPLS 域（MPLSDomain）
（2）Core LSR：MPLS 区域内部的 LSR 称为核心 LSR（CoreLSR）
（3）LER：MPLS 域边缘、连接其他网络的 LSR 称为边缘路由器（LER）LER 负责从 IP 网络接收 IP 报文并给报文压入标签，然后送到 LSR，反之，也负责从 LSR 接收带标签的报文并弹出标签然后转发到 IP 网络；LSR 只负责按照标签进行转发

扩展问题 1：什么是 LSP？
Label Switch Path：数据转发过程中，标签交换所经过的路径。LSP 是一个单向路径，与数据流的方向一致. LSP 的建立过程实际就是将 FEC 和标签进行绑定，并将这种绑定通告 LSP 上相邻 LSR 的过程

扩展问题 2：解释一下什么是 Ingress、Transit、Egress？
LSP 的入口 LER 称为入节点（Ingress）；
LSP 中间的 LSR 称为中间节点（Transit）；
LSP 的出口 LER 称为出节点（Egress）。

标签分发的方式有哪些？
（1）静态（为 IGP 路由手动分配）
（2）LDP（默认只为 32 位主机路由分配标签）

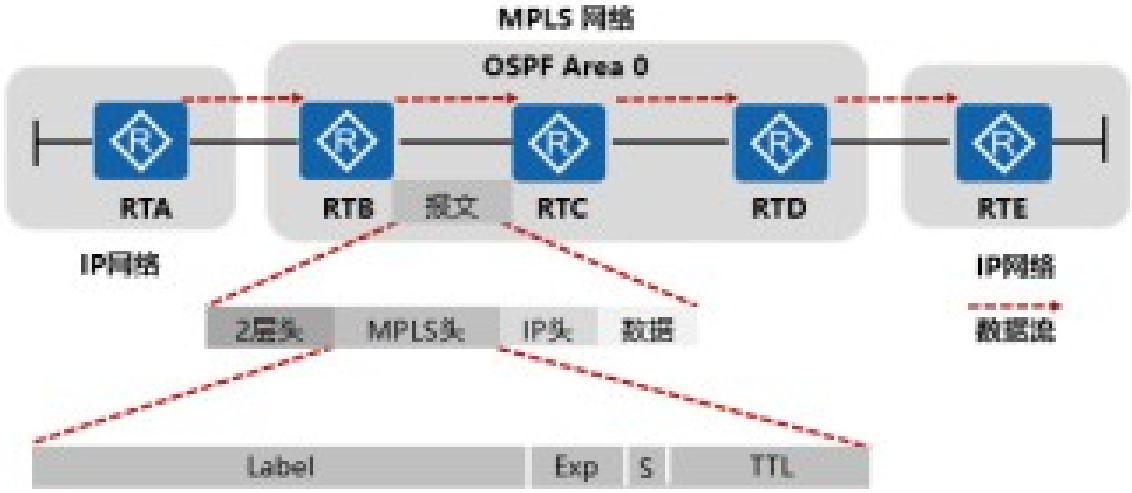
如果要使能为所有的 IGP 路由分配标签配置如下：
mpls

lsp-trigger all

（3）MP-BGP（VPNv4 路由）
（4）RSVP（QOS）

MPLS 的应用场景，举例说明？
（1）提高转发效率（早期体现，现在 IP 转发也是基于硬件转发）
MPLS 的标签格式短而定长，MPLS 转发是基于硬件的转发：
ip 转发是需要将路由表进行最长匹配，需要扫描所有路由，较消耗设备性能
（2）解决 BGP 路由黑洞
（3）MPLS VPN
（4）TE

MPLS 标签的格式



每一个 MPLS 头部总长度为 4bytes（32bits）
（1）标签 Label 长度 20bits：表示标签的编号，范围 1-2^20
1 保留标签：为特定情况保留的标签，范围 0-15
0 号：IPv4 显示空标签
2 号：IPv6 显示空标签

3 号：隐示空标签
2 静态分配标签：范围 16---1023
3 动态分配标签：范围 1024---2^20
（2）EXP（ExperimentalUse）：实验位，长度 3bits。用于表示数据包的优先级别（0-7），做 QoS 时使用
（3）S（BottomofStack）:栈底位，长度 1bits。设置为 1 时，表示为最后一层标签
1 纯 MPLS 转发：有 1 层标签
2 MPLS VPN：　有 2 层标签
3 MPLS TE：　　有 3 层标签
（4）TTL：长度 8bits，在 MPLS 域中防止数据出现环路

扩展问题 2：3 号标签和 0 号标签有什么区别？
3 号标签即是 PHP 次末跳弹出
好处：减少最后一跳路由的负担，在次末跳路由器弹出标签并且按照下一跳转发表项转发，使最后一跳路由器收的报文不带标签，只需查找一次 FIB 表。
实现方式：通过特殊的 3 号标签（隐式空标签）实现。默认为直连的 32 位主机路由分配 3 号的标签；当 FEC 对应的出标签为 3 号标签时，弹出最外层的标签再发送。
缺点：会造成最后一跳路由器无法处理 mpls 报文里的 EXP 字段，导致优先级丢失无法进行 Qos 服务。

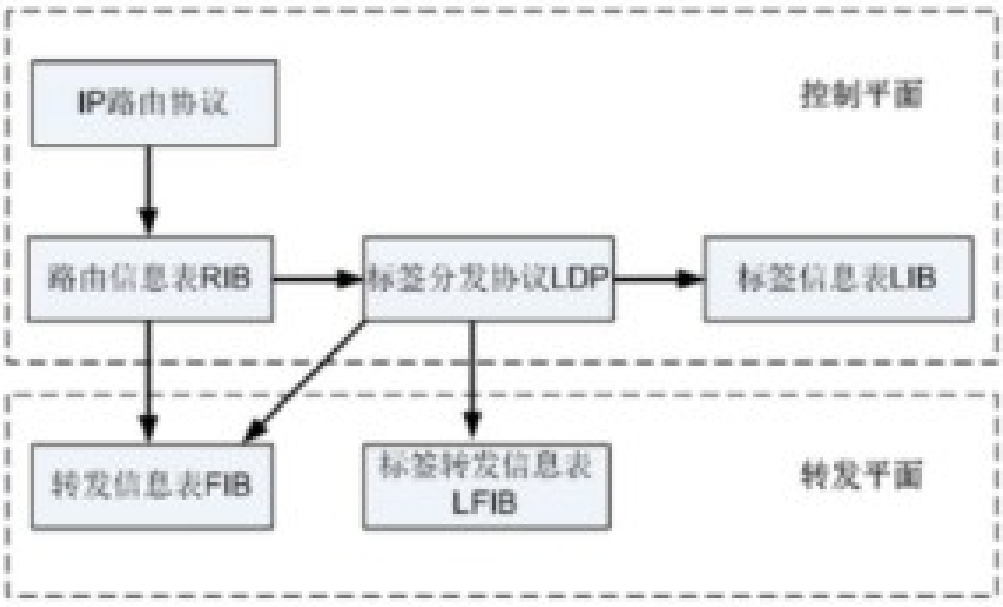
为了解决 3 号标签的缺点，提出 0 号标签（显式空标签）
出节点分配给倒数第二跳节点的标签值为 0，则倒数第二跳 LSR 需要将值为 0 的标签正常压入报文标签值顶部，转发给最后一跳。最后一跳发现报文携带的标签值为 0，则将标签弹出（无需进行查表），然后进行 IP 转发。默认使能 PHP，可在 mpls 视图下修改 Egress 节点向倒数第二跳分配显式空标签

mpls
label advertise explicit-null

label advertise 命令用来配置出节点向倒数第二跳分配何种类型的标签。推荐采用缺省配置 implicit-null，可以减少出节点的转发压力，提高转发效率。

implicit-null 支持 PHP，即在倒数第二跳节点处将标签弹出，减少出节点的负担
non-null 和 explicit-null 不支持 PHP。但是这两种方式对出节点的资源消耗较大，不推荐使用。
其中 explicit-null 支持 MPLS QoS 属性

MPLS 的转发平面和控制平面



（1）控制平面，负责产生和维护路由信息以及标签信息
1 RIB：用于选择最优路由

2 LDP：负责标签的分配、标签转发信息表的建立，标签交换路径的建立、拆除等工作
3 LIB：由标签分发协议生成，存放 FEC 和标签的对应关系

（2）转发平面也叫数据平面，负责普通 IP 报文的转发以及带 MPLS 标签报文的转发
1 FIB：转发信息库，根据 IP 路由表生成，用于决定 IP 数据包是否能带标签进行转发。属于硬件转发表
2 LFIB：标签转发信息库，由 ILM（入标签映射表）与 NHFLE（下一跳标签转发表项）关联形成，根据相关的标签发放协议（LDP，MP-BGP 等）生成。属于硬件转发表

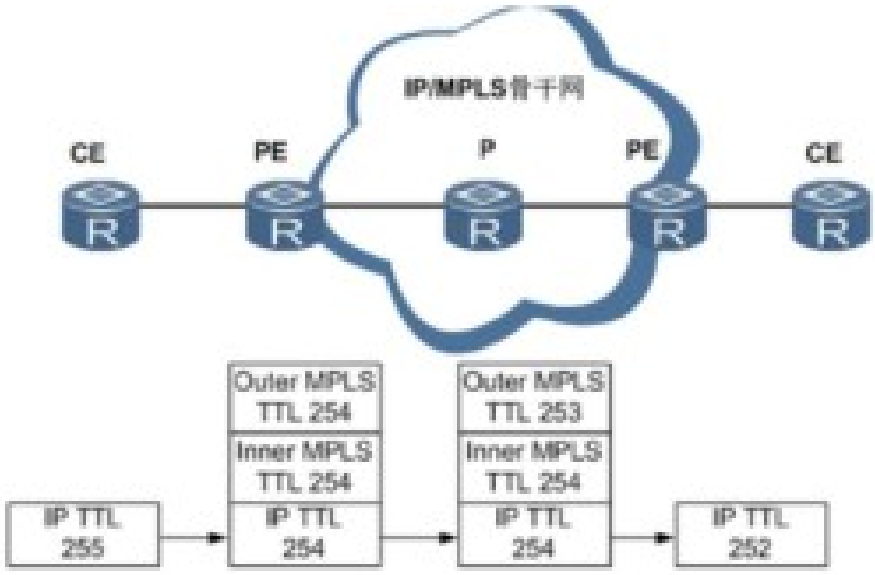
扩展问题 1：什么是 FEC？
转发等价类，MPLS 将具有相同特征的报文归为一类，属于相同 FEC 的报文在转发过程中被 LSR 以相同方式处理。
比如：在传统的采用最长匹配算法的 IP 转发中，到同一条路由的所有报文就是一个转发等价类。
所以，默认情况下把一条路由为一个 FEC。在标签分发时，针对一个 FEC 分配一个标签

MPLS 对标签的操作行为有哪些？
（1）PUSH 压入标签 IP 数据经过 MPLS 域出标签不为空
（2）SWAP 交换标签 MPLS 域的标签数据包转发出标签不为空
（3）POP 弹出标签执行弹出（或 PHP） 出标签为空或 3 号标签

MPLS 的防环机制

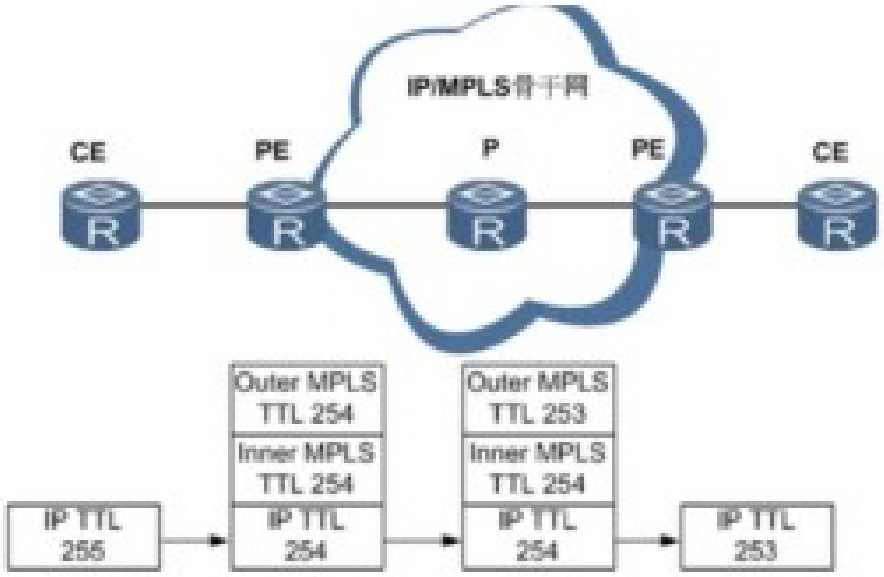
（1）控制层面使用 IGP 防环。
（2）数据层面的防环使用 TTL 值防环。MPLS 对 TTL 有两种处理方式：uniform 统一模式、pipe 管道模式
1 uniform 防环

uniform 统一模式，保证 IP 的 TTL 值和 MPLS 的 TTL 统一。根据路径时可以显示在 MPLS 网络所经过的路径，方便于排障。



a) IP 域进入 MPLS 域时，将 IP 报头的 TTL 减 1 在复制到 MPLS TTL 值中
b) MPLS 域传播时，只减 MPLS 的 TTL 值，不减 IP 的 TTL
c) 发出 MPLS 域时，将 MPLS 的 TTL 值复制回 IP 的 TTL，在减 1 发送到 IP 域

2 pipe 防环
管道模式：安全性较好，隐藏 MPLS 标签转发所经过的路径。但不易于排障



a）IP 域进入 MPLS 域时，将 IP 报头的 TTL 减 1 在复制到 MPLSTTL 值中

b）MPLS 域传播时，只减 MPLS 的 TTL 值，不减 IP 的 TTL

c）发出 MPLS 域时，不将 MPLS 的 TTL 值复制回 IP 的 TTL，只将 IP 的 TTL 值减 1 发送到 IP 域

MPLS 的数据转发流程

当数据进入 MPLS 域时：（需清楚每个节点的具体动作）

（1）会先根据 FIB 表查找相应的转发条目，转发条目中包含 tunnel id 字段。

1 如果 tunnel id 为 0X0，则进行 IP 转发；

2 如果 tunnel id 为非 0X0，则进入 MPLS 转发流程

```
Destination/Mask    Nexthop      Flag   TimeStamp
Interface TunnelID
8.8.8.8/32     192.168.17.1   DGHU      t[5839]
GE0/0/1        0x0
192.168.17.255/32  127.0.0.1    HU      t[124]
InLoop0        0x0
```

```
Destination/Mask    Nexthop  Flag  TimeStamp
Interface      TunnelID
3.3.3.3/32    192.168.12.2   DGHU  t[162]
GE0/0/0         0x3
2.2.2.2/32   192.168.12.2    DGHU  t[152]
GE0/0/0         0x1

[R1]display tunnel-info tunnel-id 0x3
Tunnel ID:                        0x3
Tunnel Token:                     3
Type:                             lsp
Destination:                      3.3.3.3
Out Slot:                         0
Instance ID:                      0
Out Interface:
GigabitEthernet0/0/0
Out Label:                        1025
Next Hop:                         192.168.12.2
Lsp Index:                        6147
```

（2）Ingress 的处理：通过查询 FIB 表和 NHLFE 表指导报文的转发。

1 查看 FIB 表，根据目的 IP 地址找到对应的 Tunnel ID。

2 根据 FIB 表的 Tunnel ID 找到对应的 NHLFE 表项，将 FIB 表项和 NHLFE 表项关联起来（FTN）。

3 查看 NHLFE 表项，可以得到出接口、下一跳、出标签和标签操作类型。

4 在 IP 报文中压入出标签，同时处理 TTL，然后将封装好的 MPLS 报文发送给下一跳。

入标签映射 ILM（Incoming Label Map）
下一跳标签转发表项 NHLFE（Next Hop Label Forwarding Entry）

（3）Transit 的处理：通过查询 ILM 表和 NHLFE 表指导 MPLS 报文的转发。
1 根据 MPLS 的标签值查看对应的 ILM 表，可以得到 Tunnel ID。
2 根据 ILM 表的 Tunnel ID 找到对应的 NHLFE 表项。
3 查看 NHLFE 表项，可以得到出接口、下一跳、出标签和标签操作类型。
4 MPLS 报文的处理方式根据不同的 Label 而不同：
a)如果 Label> = 16，则用新标签替换 MPLS 报文中的旧标签，同时处理 TTL，然后将替换完标签的 MPLS 报文发送给下一跳。
b)如果 Label 为 3，则直接弹出标签，同时处理 TTL，然后进行 IP 转发或下一层标签转发

（3）Egress 的处理：通过查询 ILM 表指导 MPLS 报文的转发或查询路由表指导 IP 报文转发。
a）如果 Egress 收到 IP 报文，则查看路由表，进行 IP 转发。（次末跳弹出）
b）如果 Egress 收到 MPLS 报文，则查看 ILM 表获得标签操作类型，同时处理 TTL：
a.如果标签中的栈底标识 S=1，表明该标签是栈底标签，直接进行 IP 转发。
b.如果标签中的栈底标识 S=0，表明还有下一层标签，继续进行下一层标签转发。

扩展问题 1：介绍一下 Tunnel-ID、ILM、NHLFE 的相关概念？
Tunnel ID：为了给使用隧道的上层应用（如 VPN、路由管理）提

供统一的接口，系统自动为隧道分配了一个 ID，也称为 Tunnel ID。该 Tunnel ID 的长度为 32 比特，只是本地有效。

ILM： 入标签映射表，入标签到一组下一跳标签转发表项的映射。包括：TunnelID、入标签、入接口、标签操作类型等信息。ILM 在 Transit 节点的作用是将标签和 NHLFE 绑定。通过标签索引 ILM 表，就相当于使用目的 IP 地址查询 FIB，能够得到所有的标签转发信息。

NHLFE：下一跳标签转发表项，用于指导 MPLS 报文的转发。
包括：TunnelID、出接口、下一跳、出标签、标签操作类型等信息。
FEC 到一组 NHLFE 的映射称为 FTN（FEC-to-NHLFE）。通过查看 FIB 表中 TunnelID 值不为 0x0 的表项，就能够获得 FTN 的详细信息。FTN
只在 Ingress 存在。

扩展问题 2：Tunnel-ID 的作用？
1、隧道标识符
2、在 Ingress 节点中，用于确定是进行 ip 转发还是标签转发
3、在 MPLS 转发过程中，用于关联 FIB、NHLFE、ILM 表项

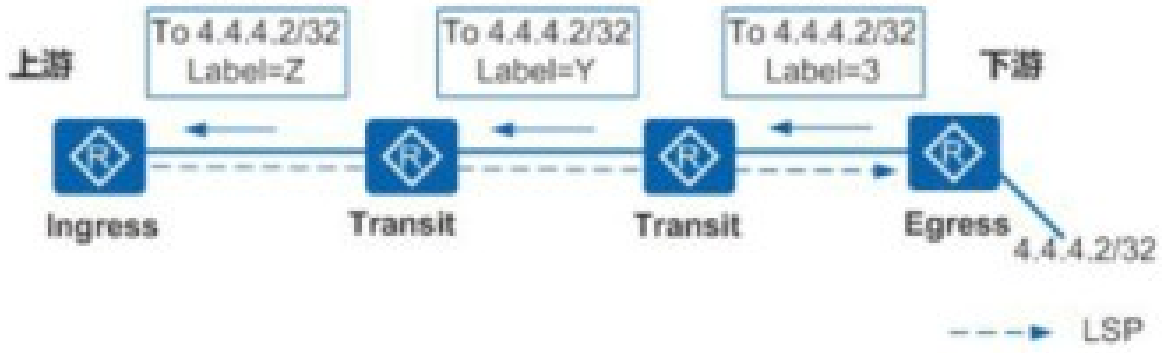
扩展问题 3：为什么 NHLFE 表项里要有下一跳，ILM 表项里要有入接口。这两个元素有什么作用？
NHLFE 需要下一跳的原因，是为了封装 MAC 地址
ILM 需要入接口的原因：基于接口的标签空间时，不同接口内的标签虽然都是不一致的。但如果每个接口内的入标签，数量都特别大。那么路由器查找标签比较麻烦。

MPLS 中 LSP 的建立方式

MPLS 需要为报文事先分配好标签，建立一条 LSP，才能进行报文

转发。LSP 分为静态 LSP 和动态 LSP 两种。
（1）静态 LSP：
由管理员手工创建的 LSP 隧道。
不能相互感知到整个 LSP 的情况，是一个本地的概念。
不使用标签发布协议，不需要交互控制报文，因此消耗资源比较小，适用于拓扑结构简单并且稳定的小型网络。但通过静态方式分配标签建立的 LSP 不能根据网络拓扑变化动态调整，需要管理员干预。
（2）动态 LSP：
动态 LSP 通过标签发布协议动态建立。
标签发布协议是 MPLS 的控制协议（也可称为信令协议），负责 FEC 的分类、标签的分发以及 LSP 的建立和维护等一系列操作。MPLS 可以使用多种标签发布协议：LDP、RSVP-TE（MPLSTE）、MP-BGP

上游 下游
上游 LSR 和下游 LSR 是根据数据报文的流向来定义的，而数据流总是由上游发往下游的。标签由下游 LSR 分配，按从下游到上游的方向分发。



LDP 有几种发现邻居的机制？
（1）基本发现机制：用于发现链路上直连的 LSR。
LSR 通过周期性（5s）地发送 LDP 链路 Hello 消息（LDP LinkHello），实现 LDP 基本发现机制，建立本地 LDP 会话。LDP 链路 Hello 消息使用 UDP 报文，目的地址是组播地址 224.0.0.2。如果 LS

R 在特定接口接收到 LDP 链路 Hello 消息，表明该接口存在 LDP 对等体。

（2）扩展发现机制：用于发现链路上非直连 LSR。LSR 周期性地发送 LDP 目标 Hello 消息（LDP TargetedHello）到指定 IP 地址，实现 LDP 扩展发现机制，建立远端 LDP 会话。LDP 目标 Hello 消息
使用 UDP 报文，目的地址是指定 IP 地址。如果 LSR 接收到 LDP 目标 Hello 消息，表明该 LSR 存在 LDP 对等体。

扩展问题 1：在基本邻居发现机制中，R1 从两条链路上都收到了 R2 的 hello 报文，这个时候 R1 和 R2 会建立两个邻居关系吗？为什么？

hello 报文中有一个 transport address 字段，路由器用这个地址来与对方建立 TCP 连接。transport address 默认和 LSR ID 一致。所以 R1 和 R2 只建立一个邻居关系。

扩展问题 2：如果 transport address 是物理接口呢？
那么 R1-2 就建立两条 TCP 连接。之后发送的初始化消息里也携带 LSR-ID，这个时候就会关闭掉一条 TCP 连接。

扩展问题 3：如果 R1 和 R2 之间 UDP 报文走上面的链路，TCP 的报文走下面的链路，会不会影响标签的分配？
不会，标签分配使用 TCP 连接。

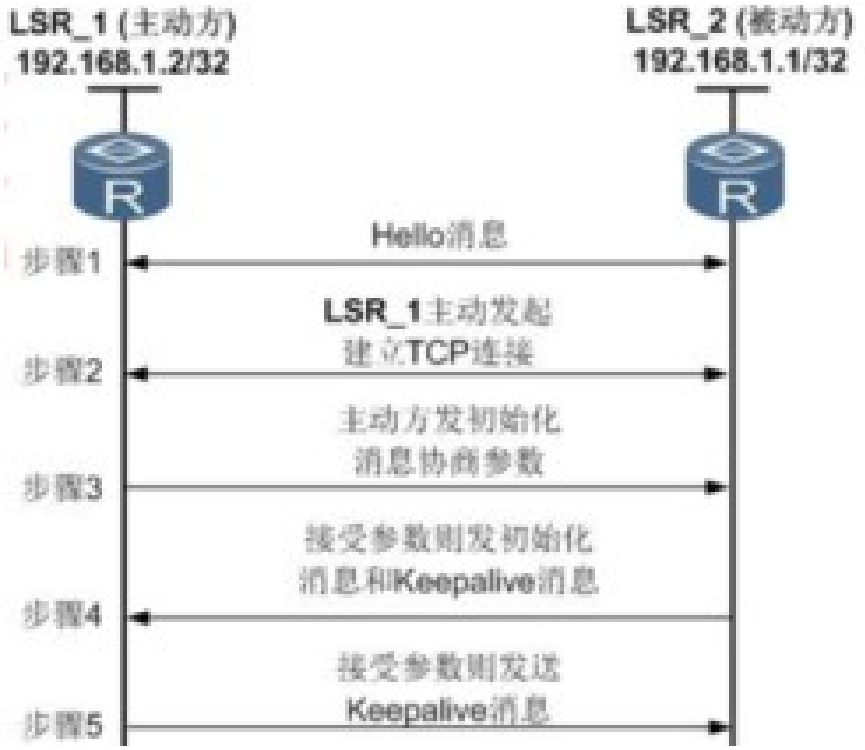
扩展问题 4：hello 包内有什么内容？
1. LS RID：本台 MPLS 路由的 routerid（不能一致）
2. 标签空间 ID：当前必须为 0，表示基于平台的标签空间
3. hello 时间间隔为 5 秒，hello 死亡时间为 15 秒
4. target hello 字段：基本发现机制时为 0，扩展发现机制时为 1（了解即可）
5. Transport address：传输地址。用于充当 TCP 发送的源 IP 地址，默认和 LSRID 的地址是一致；
注意：要保证双方的传输地址可达，否则 TCP 无法建立

标签空间取值：
“0”表示全局标签空间。
“1”表示接口标签空间。

扩展问题 5：LSR ID 和 LDP ID 的关系？
LSR ID 用于在 MPLS 域中唯一标识一台 LSR 路由器，格式与 IPv 4 相同。
每一台运行了 LDP 协议的 LSR 路由器都有 LDP ID，长度为 48bit，由 32bit 的 LSR ID+16bit 的 Label Space ID 标签空间标识符构成。

LDP 会话的建立过程

1）两台 LSR 之间交换 Hello 消息触发 LDP 会话的建立
1 两个 LSR 之间互相发送 Hello 消息。
Hello 消息中携带传输地址（即设备的 IP 地址），双方使用传输地址建立 LDP 会话。
2 传输地址较大的一方作为主动方，发起建立 TCP 连接。
如图 所示，LSR1 作为主动方发起建立 TCP 连接，LSR2 作为被动方等待对方发起连接。



3 TCP 连接建立成功后，由主动方 LSR1 发送初始化消息，协商建立 LDP 会话的相关参数。
LDP 会话的相关参数包括 LDP 协议版本、标签分发方式、Keepalive 保持定时器的值、最大 PDU 长度和标签空间等。（并非通过 hello 包进行协商）

4 被动方 LSR2 收到初始化消息后，LSR2 接受相关参数，则发送初始化消息，同时发送 Keepalive 消息给主动方 LSR1。如果被动方 LSR2 不能接受相关参数，则发送 Notification 消息终止 LDP 会话的建立。

5 主动方 LSR1 收到初始化消息后，接受相关参数，则发送 Keepalive 消息给被动方 LSR2。如果主动方 LSR1 不能接受相关参数，则发送 Notification 消息给被动方 LSR2 终止 LDP 会话的建立。

6 当双方都收到对端的 Keepalive 消息后，LDP 会话建立成功。
备注：邻居发现使用 UDP 646 快速发现邻居，标签分发使用 tcp

保证可靠

扩展问题 1：hello 和 keepalive 的区别？
hello 用于发现邻居，建立、维护 LDP 邻居关系。：
基本发现机制：hellotimer 为 5s，hellodeadtimer 为 15s
扩展发现机制：hellotiemr 为 15s，hellodeadtimer 为 45s
keepalive 用于维护 LDP 会话的 TCP 连接的完整性，发送间隔为 15s，死亡时间是 45s。

标签空间
标签空间是什么？设备有几种标签空间？
（1）标签空间，决定本台设置的入标签如何产生。有基于平台的和基于接口的。
1 基于平台标签空间：
a）设备上的所有 FEC 共同使用 1024---2^20 的标签空间
b）标签分配时并不是在每个接口下唯一
c）帧模式下使用的标签空间为基于平台
2 基于接口标签空间：
a）每个接口通告的标签范围是唯一的
b）LSR 为同一条 FEC 在不同接口通告的标签是不同的（小概率会相同）

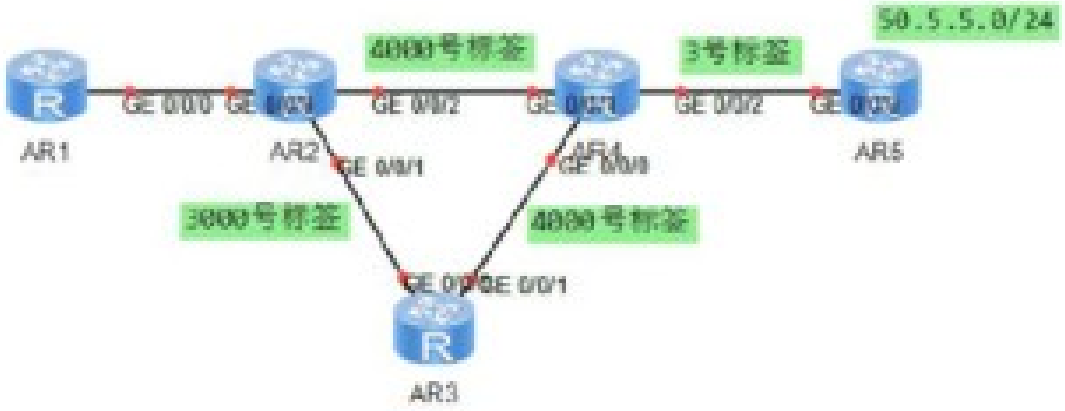
LDP 分发标签的方式？控制方式？保存方式？
1 分发标签的方式：
a）下游自主方式 DU（默认）：下游主动向上游发出标签和 FEC 的映射消息
b）下游按需方式 DoD： 由上游向下游请求后，下游才会向上游发标签映射消息

2 标签分配控制方式：

a）有序方式（Odered）标记控制（默认）：除非 LSR 是路由的始发节点，否则 LSR 必须等收到下一跳的标记映射才能向上游发出标记映射。换言之就是只有在已经收到下游标签映射消息时，才能够给上游分发
b）独立方式（Independent）标记控制：LSR 可以向上游发出标记映射，而不必等待来自 LSR 下一跳的标记映射消息。换言之就是无论有没有收到下游的标签，都能够给上游分发标签映射消息

3 标签保存方式：
a）自由方式（Liberalretentionmode）（默认）：保留来自邻居的所有发送来的标签
优点：当 IP 路由收敛、下一跳改变时减少了 lsp 收敛时间不需要请求
缺点：需要更多的内存和标签空间。
b）保守方式（Conservativeretentionmode）：只保留来自下一跳邻居的标签，丢弃所有非下一跳邻居发来的标签。
优点：节省内存和标签空间。
缺点：当 IP 路由收敛、下一跳改变时 lsp 收敛慢需再次请求

扩展问题 1：AR5 在为 50.5.5.0/24 的 FEC 分配 3 号给 AR4 后，AR4 会怎么做?它会从自己的哪几个接口分配标签？R2 收到从两个接口收到两份不同标签时如何进行优选？



假设 AR4 为自己的 2 个接口都发送 4000 号的标签。
因为帧模式下的 MPLS，默认是基于平台的标签空间，每个接口针对同一条 FEC 分配的标签号是一致的。
同时，AR4 也会为自己接收标签的接口，分配出标签。因为缺省情况下，没有为 LDP 对等体配置水平分割策略，即 LSR 会向其上游和下游 LDP 对等体都分配标签。
为 LDP 对等体配置水平分割策略，使 LSR 只向其上游 LDP 对等体分配标签，配置命令如下：
mpls ldp
outbound peer 2.2.2.2 split-horizon

AR2 上收到了两个关于 50.5.5.0/24 这条 FEC 的标签，这个时候，查看单播路由表得到 50.5.5.0/24 的下一跳和 LDP address 消息（在 LDP 会话建立时候发送）里的地址进行对比。
是自己下一跳发送过来的 4000 号标签，直接使用。非下一跳（R3）发送过来的标签 3000 号的标签作为备份，因为设备默认使用自由的标签保存方式。

如何实现 MPLS LSP 的快速切换

LDP FRR（Fast Reroute）为 MPLS 网络提供快速重路由功能，实现了链路备份；当主 LSP 故障时，流量快速切换到备份路径，从

而最大程度上避免流量的丢失。

（1）使用 LDP FRR 技术为 MPLS 网络提供快速重路由功能，实现了链路备份。
LDP FRR 原理是通过 LDP 信令的 Liberal 标签保持方式，先获取 Liberal Label，为该标签申请转发表项资源，并将转发信息下发到转发平面作为主 LSP 的备用转发表项。
（2）当接口故障（接口自己感知或者结合 BFD 检测）或者主 LSP 不通（结合 BFD 检测）时，可以快速的将流量切换至备份路径（无需重新计算），从而实现了对主 LSP 的保护。
（3）具体实现：
1 首先要设置自由的标签保存方式。
2 LDP FRR：LDP FRR（Fast Reroute）为 MPLS 网络提供快速重路由功能，实现了链路备份；当主 LSP 故障时，流量快速切换到备份路径，从而最大程度上避免流量的丢失主的 LSP 和备的 LSP，FIB 里面会有两条，正常只是主的在进行数据转发，当主链路出现故障时，无需进行收敛就可以立即切换到备链路
3 扩展：BFD for LDP LSP。BFD 可以对 LSP 进行快速的故障检测，触发 LSP 在发生故障时进行快速主备路径倒换，提高整网可靠性。

手工 FRR
int g0/0/0
mpls ldp frr nexthop 10.1.1.2

自动 FRR
mpls ldp
auto-frr lsp-trigger all