

第 5 篇 控制 IGP 路由

第 13 章 路由过滤

第 14 章 路由策略

第 15 章 路由引入

第 16 章 PBR

第13章 路由过滤

路由器在发布与接收路由信息时，可能需要对路由信息进行过滤。常用的路由过滤工具有 ACL、地址前缀列表等。本章介绍了路由过滤的目的、应用、工具及其相关的配置。

13.1 本章目标

课程目标

● 学习完本课程，您应该能够：

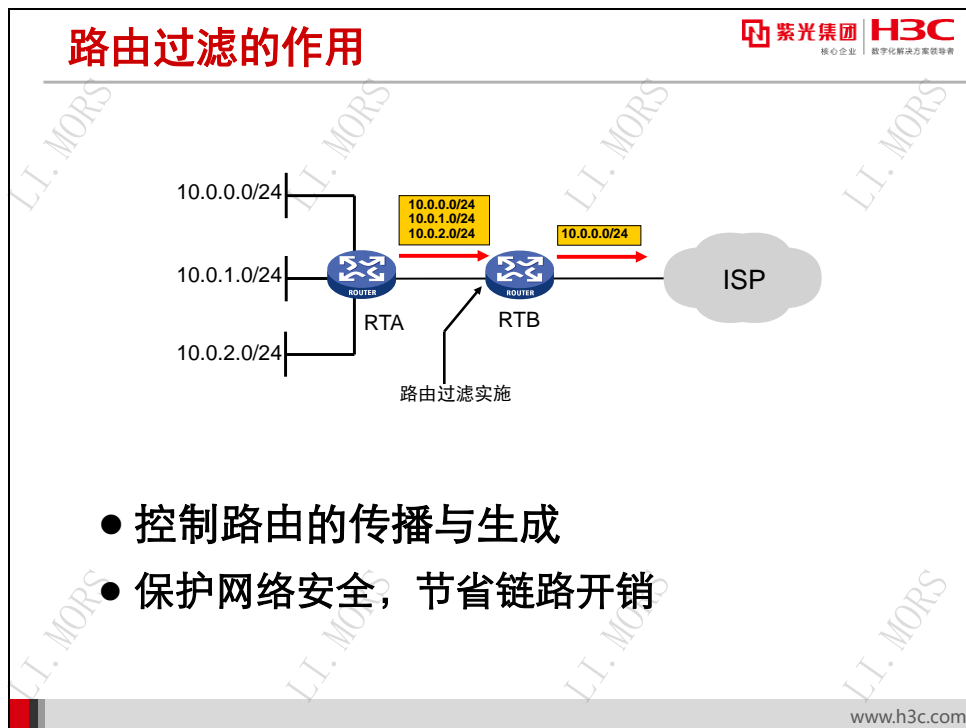
- 了解路由过滤的目的和作用
- 掌握路由过滤的原理
- 掌握过滤工具的种类和特点
- 掌握地址前缀列表的配置
- 掌握filter-policy的配置和应用



www.h3c.com

13.2 路由过滤概述

13.2.1 路由过滤的作用

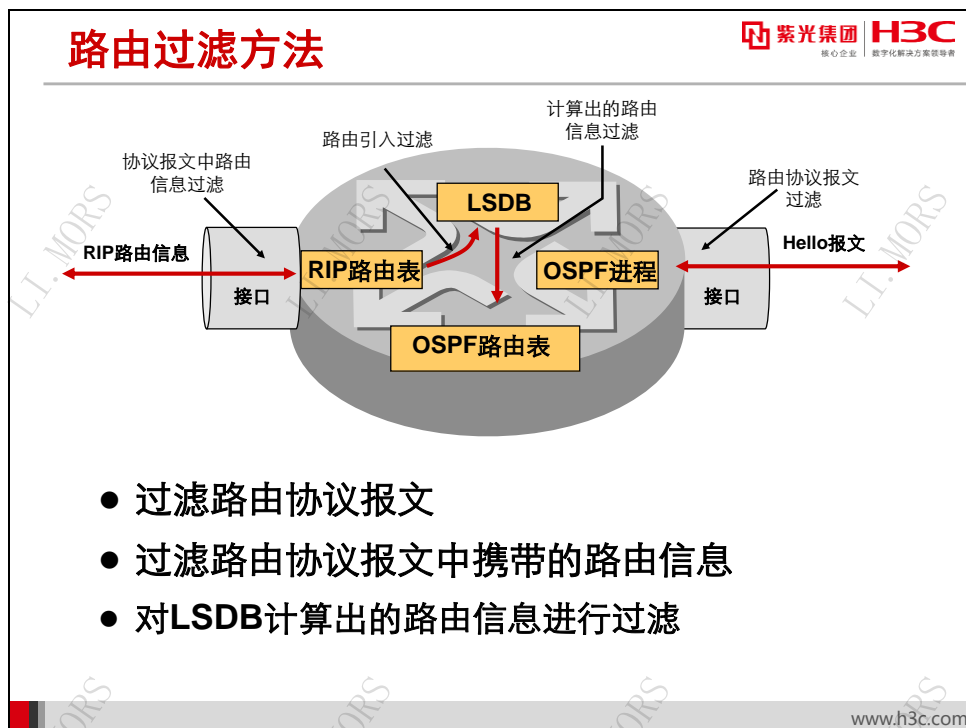


路由器在运行路由协议后，通过路由协议进行路由信息的发布与接收。通常情况下，距离矢量型路由协议会将自己的全部路由信息发布出去，同时也接收邻居路由器发来的所有路由信息；而链路状态型路由协议也会发送自己产生的 LSA，并接收邻居发来的 LSA，然后在本地构建 LSDB 数据库，根据 LSDB 计算出路由。

但是，有时为了控制报文的转发路径，路由器在发布与接收路由信息时，可能需要实施一些策略，以对路由信息进行过滤，只接收或发布满足一定条件的路由信息。路由过滤的另一个好处是节省设备和链路资源，甚至保护网络安全。

路由过滤的应用比较普遍。例如，某公司内部网络运行了路由协议，某些内部的路由信息是不希望被外部所知道的，这时可以采用路由过滤的方法把内部路由在网络边界上过滤掉。再如，某 ISP 因为某种的原因，只想把某条特定路由发送给其客户，就可以采用路由过滤的手段。

13.2.2 路由过滤方法



路由过滤主要有两种应用方式：

- **路由引入过滤。**路由协议在引入其它路由协议发现的路由时，只引入满足条件的路由信息。
- **路由发布或接收过滤。**路由协议在发布或接收路由信息时，对路由信息进行过滤，只接收或发布满足给定条件的路由信息。

本章主要讲述如何在路由协议发布或接收路由信息时进行过滤。

在进行路由过滤时，通常有如下几种过滤方法：

- 过滤路由协议报文

路由器间通过交换路由协议报文而学习路由。如果将路由协议报文过滤，则路由器间无法学习路由，也就达到过滤路由的目的。过滤路由协议报文后，所有的路由信息都被过滤了。

- 过滤路由协议报文中携带的部分路由信息

路由协议报文中包含了路由信息，路由信息携带了路由属性如目的地址、下一跳等。可以采取适当的过滤器来对其中某些路由信息进行过滤，而允许其它路由信息通过。

- 对从 LSDB 计算出的路由信息进行过滤

链路状态型路由协议首先交换 LSA 而生成本地 LSDB 数据库，再通过 SPF 算法计算出路由，再把路由加入到路由表中。所以，可以对从 LSDB 计算出的路由信息进行过滤。

13.2.3 路由过滤工具

路由过滤工具

- 静默接口
- 过滤器
 - 访问控制列表
 - 地址前缀列表
 - filter-policy
 - route-policy

 紫光集团 H3C
核心企业 数字化转型领导者

www.h3c.com

可以通过在路由器上使用静默接口来使路由器不发出协议报文，从而达到路由过滤的目的；也可以配置路由协议使用一些过滤器，来对协议报文中的路由信息进行过滤。

常见的过滤器有以下几种：

- ACL（访问控制列表）

通过使用 ACL，可以指定 IP 地址和子网范围，用于匹配路由信息的目的网段地址或下一跳地址。

- 地址前缀列表

地址前缀列表（prefix-list）的作用类似于 ACL，但比它更为灵活，且更易于被用户理解。使用地址前缀列表过滤路由信息时，其匹配对象为路由信息的目的地址信息域；另外，用户可以指定 gateway 选项，指明只接收某些路由器发布的路由信息。

- filter-policy


通过配置 filter-policy，可以制定入口或出口过滤策略，对接收和发布的路由进行过滤。在接收路由时，还可以指定只接收来自某个邻居的 RIP 报文。filter-policy 可以使用地址前缀列表（prefix-list）和访问控制列表（ACL）来定义自己的匹配规则。

- Route-policy

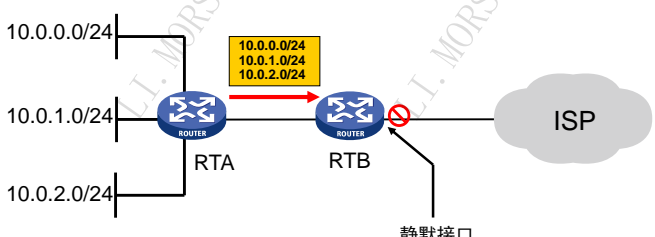
Route-policy 是一种比较复杂的过滤器，它不仅可以匹配路由信息的某些属性，还可以在条件满足时改变路由信息的属性。**Route-policy** 可以使用前面 **ACL**、地址前缀列表等过滤器来定义自己的匹配规则。

通常，**ACL** 和地址前缀列表仅对路由信息进行匹配，也就是指明哪些路由信息符合过滤的要求；而 **filter-policy** 和 **Route-policy** 用来指明对符合过滤条件的路由信息执行过滤动作，并指明是对接收还是发送的路由进行过滤。

13.3 配置静默接口过滤路由



配置静默接口过滤全部路由



- **RIP协议中，静默接口不发送路由更新**
- **OSPF协议中，静默接口不发送HELLO报文**

www.h3c.com

静默接口（silent-interface）又称为被动接口（Passive interface）。在路由器上配置静默接口是一种简单易用的过滤路由手段。通常在局域网内，主机并不需要接收路由器发出的协议报文；而且为了安全起见，管理员也不希望路由器发送协议报文给不相关的设备或区域。此时，可以通过在路由器上配置静默接口来使路由器不发送协议报文。

在 RIP 协议中，配置为静默接口的接口不会发送路由更新；而在 OSPF 和 IS-IS 协议中，配置为静默接口的接口不发送 HELLO 报文，也即不建立邻居关系。

可以在 RIP、OSPF 视图下用如下命令配置静默接口：

silent-interface { all | interface-type interface-number }

而在 IS-IS 协议中，可以通过在接口视图下禁止接口发送和接收 IS-IS 报文来达到相同的效果。其配置命令如下：

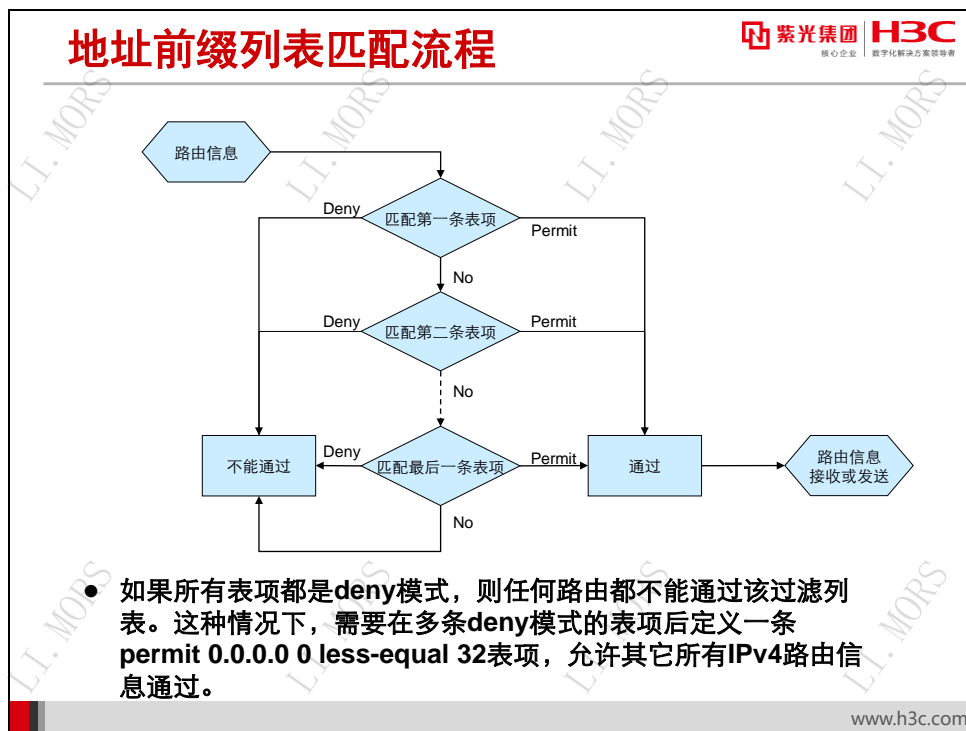
isis silent

说明：

某接口配置为静默接口后，协议仍然把该接口直连网络的路由信息从其它接口宣告出去。

13.4 地址前缀列表

13.4.1 地址前缀列表匹配流程



一个地址前缀列表由前缀列表名标识。每个前缀列表可以包含多个表项，每个表项可以独立指定一个网络前缀形式的匹配范围，并用一个索引号来标识，索引号指明了在地址前缀列表中进行匹配检查的顺序。


每个表项之间是“或”的关系，在匹配的过程中，路由器按升序依次检查由索引号标识的各个表项，只要有某一表项满足条件，就意味着通过该地址前缀列表的过滤，而不再去匹配其他表项。

每一个表项都指定了相应的匹配模式，包括允许模式（Permit）和拒绝模式（Deny）。当指定为允许模式并且待过滤的 IP 地址在该表项指定的前缀范围内时，通过该表项的过滤不进入下一个结点的测试；如待过滤的 IP 地址不在该表项指定的前缀范围内，则进行下一表项测试。当指定为拒绝模式并且待过滤的 IP 地址在该表项指定的前缀范围内时，则该 IP 地址不能通过该表项的过滤，并且不会进行下一个表项的测试，否则进入下一表项的测试。

从以上规则可以看出，如果所有表项都是拒绝模式，则任何路由都不能通过该过滤列表。这种情况下，需要在多条拒绝模式的表项后定义一条允许全部路由的表项，以允许其它 IP 路由信息通过。

13.4.2 配置地址前缀列表

配置地址前缀列表



紫光集团 H3C
核心企业 数字化转型领导者

```
[H3C] ip prefix-list prefix-list-name [ index index-number ]
{ deny | permit } ip-address mask-length [ greater-equal min-
mask-length ] [ less-equal max-mask-length ]
```

配置	结果
Permit 10.0.0.0 24	仅匹配10.0.0.0/24，不匹配任何其它网络
Permit 10.0.0.0 24 greater-equal 25	匹配10.0.0.0/24区间内的掩码大于等于25位的网络，如10.0.0.0/26、10.0.0.16/28、10.0.0.5/32等
Permit 10.0.0.0 24 greater-equal 25 less-equal 30	匹配10.0.0.0/24区间内的掩码大于等于25位但小于等于30位的网络，如10.0.0.0/26、10.0.0.16/28等
Permit 0.0.0.0 0 greater-equal 16 less-equal 24	匹配掩码大于等于16但小于等于24位的任意网络
Permit 0.0.0.0 0	仅匹配缺省路由
Permit 0.0.0.0 0 less-equal 32	匹配所有路由

www.h3c.com

配置地址前缀列表，需要在系统视图下使用如下命令：

```
ip prefix-list prefix-list-name [ index index-number ] { deny|permit} ip-address
mask-length [ greater-equal min-mask-length ] [ less-equal max-mask-length ]
```

其中的参数含义如下：

- **prefix-list-name**: 地址前缀列表名。
- **index-number**: 标识地址前缀列表中的一项。
- **permit**: 指定所定义的地址前缀列表项的匹配模式为允许模式。
- **deny**: 指定所定义的地址前缀列表项的匹配模式为拒绝模式。
- **ip-address mask-length**: 指定 IP 地址前缀和前缀长度，**mask-length** 的取值范围为 0～32。
- **min-mask-length**、**max-mask-length**: 如果 IP 地址和前缀长度都已匹配，则使用该参数来指定地址前缀范围。**greater-equal** 的含义为“大于等于”，**less-equal** 的含义为“小于等于”，其取值范围为 $mask-length \leq min-mask-length \leq max-mask-length \leq 32$ 。如果只指定 **min-mask-length** 时，则前缀长度范围为 $[min-mask-length, 32]$ ；如果只指定 **max-mask-length** 时，则前缀长度范围为 $[mask-length, max-mask-length]$ ；如果二者都指定，则前缀长度范围为 $[min-mask-length, max-mask-length]$ 。

下表中列出了一些地址前缀列表配置后的匹配结果。

表13-1 地址前缀列表匹配结果

配置	结果
Permit 10.0.0.0 24	仅匹配10.0.0.0/24，不匹配任何其它网络
Permit 10.0.0.0 24 greater-equal 25	匹配10.0.0.0/24区间内的掩码大于等于25位的网络，如10.0.0.0/26、 10.0.0.16/28、 10.0.0.5/32等
Permit 10.0.0.0 24 greater-equal 25 less-equal 30	匹配10.0.0.0/24区间内的掩码大于等于25位但小于等于30位的网络，如10.0.0.0/26、 10.0.0.16/28等
Permit 0.0.0.0 0 greater-equal 16 less-equal 24	匹配掩码大于等于16但小于等于24位的任意网络
Permit 0.0.0.0 0	仅匹配缺省路由
Permit 0.0.0.0 0 less-equal 32	匹配所有路由

如上所示为地址前缀列表配置示例及相应匹配结果。

- 当配置如下时：

```
[Router] ip prefix-list test permit 10.0.0.0 24 less-equal 32
```

匹配的结果是所有 10.0.0.0/24 范围内的路由能够通过过滤，而其它路由不能通过。

- 当配置如下时：

```
[Router] ip prefix-list test index 10 permit 10.0.0.0 24
[Router] ip prefix-list test index 20 permit 11.0.0.0 16
```

匹配的结果是只有路由 10.0.0.0/24 和 11.0.0.0/16 能够通过过滤，其它路由都不能通过。

- 当配置如下时：

```
[Router] ip prefix-list test index 10 deny 10.0.0.0 24
[Router] ip prefix-list test index 20 permit 0.0.0.0 0 less-equal 32
```

匹配的结果是只有 10.0.0.0/24 路由不能通过过滤。其它所有路由能够通过过滤。

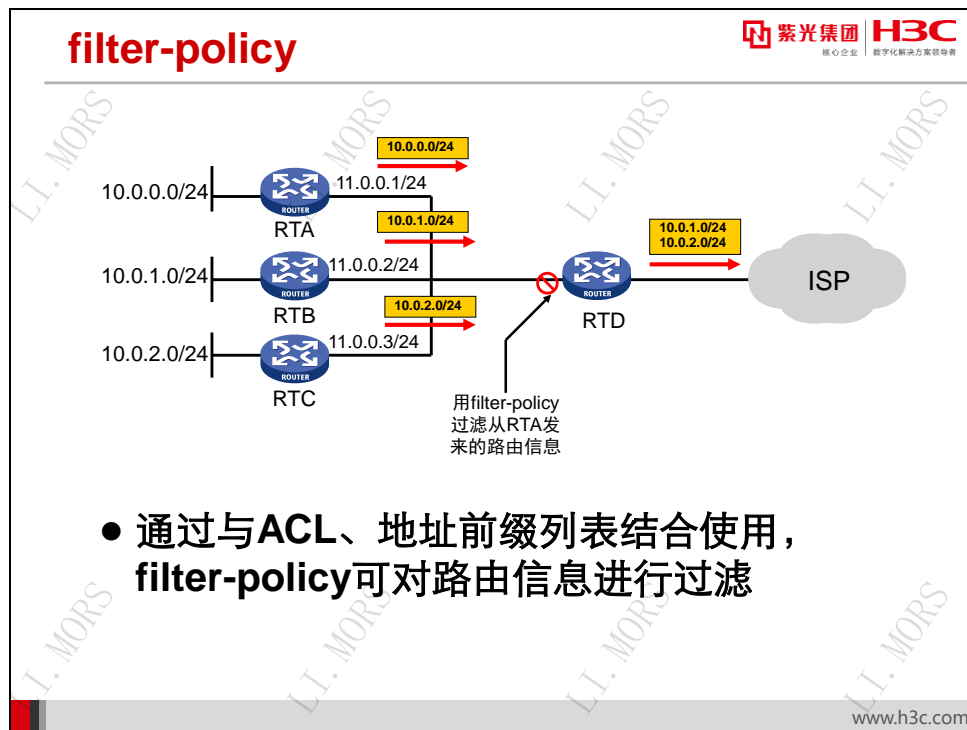
- 当配置如下时：

```
[Router] ip prefix-list test index 10 deny 10.0.0.0 30
[Router] ip prefix-list test index 20 permit 10.0.0.0 24 less-equal 32
```

匹配的结果是除了 10.0.0.0/30 外，10.0.0.0/24 区间内的其它路由能够通过过滤。10.0.0.0/24 区间外的路由不能通过过滤。

13.5 Filter-policy

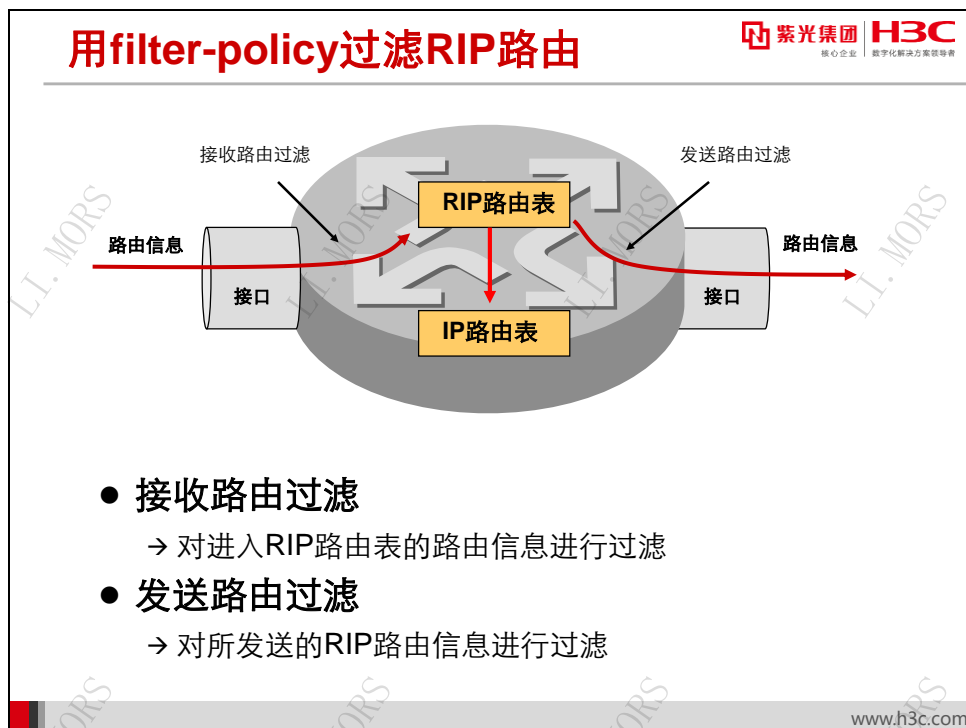
13.5.1 Filter-policy 概述



filter-policy 应用在路由协议接收或发送路由时，通过入口或出口过滤策略，对接收和发布的路由进行过滤。**filter-policy** 可以使用访问控制列表或地址前缀列表来定义自己的匹配规则。

在上图所示网络中，RTD 从 RTA、RTB、RTC 处分别收到 10.0.0.0/24、10.0.1.0/24、10.0.2.0/24 等路由更新，但因策略需要，RTD 仅需要向 ISP 发送 10.0.1.0/24 和 10.0.2.0/24。此时，可以在 RTD 上应用 **filter-policy**，通过入口过滤策略，过滤从 RTA 接收的路由；也可以通过出口过滤策略，在 RTD 发送路由时将 10.0.0.0/24 路由过滤掉。

13.5.2 配置 filter-policy 过滤 RIP 路由



使用 filter-policy 进行路由过滤时，要注意对于不同的路由协议，filter-policy 的过滤原理不同。

对于距离矢量型路由协议，协议内路由过滤可以在以下 2 个阶段实施：

- 接收路由信息的时候进行过滤
- 发送路由信息的时候进行过滤

对于 RIP 协议，因接收到的路由需要放到 RIP 路由表中，所以接收路由过滤是对进入 RIP 路由表的路由信息进行过滤；而发送路由过滤是对所发送的所有 RIP 路由信息进行过滤。

配置filter-policy过滤RIP路由



● 配置RIP对接收的路由进行过滤

```
[H3C-rip-1] filter-policy { acl-number | gateway ip-  
prefix-name | prefix-list prefix-list-name [ gateway  
prefix-list-name ] } import [ interface-type interface-  
number ]
```

● 配置RIP对发送的路由进行过滤

```
[H3C-rip-1] filter-policy { acl-number | prefix-list  
prefix-list-name } export [ protocol [ process-id ]  
[ interface-type interface-number ]
```

www.h3c.com

在 RIP 协议视图下配置对接收的路由进行过滤，其命令如下：

```
filter-policy { acl-number | gateway prefix-list-name | prefix-list prefix-list-name  
[ gateway prefix-list-name ] } import [ interface-type interface-number ]
```

其中参数含义如下：

- **acl-number**: 用于过滤接收的路由信息的访问控制列表号，取值范围为 2000～3999。
- **prefix-list prefix-list-name**: 指定用于过滤接收路由信息的 IP 地址前缀列表名称。
- **gateway prefix-list-name**: 基于发布网关过滤路由。
- **interface-type interface-number**: 接口类型和接口号。

在 RIP 协议视图下配置对发送的路由进行过滤，其命令如下：

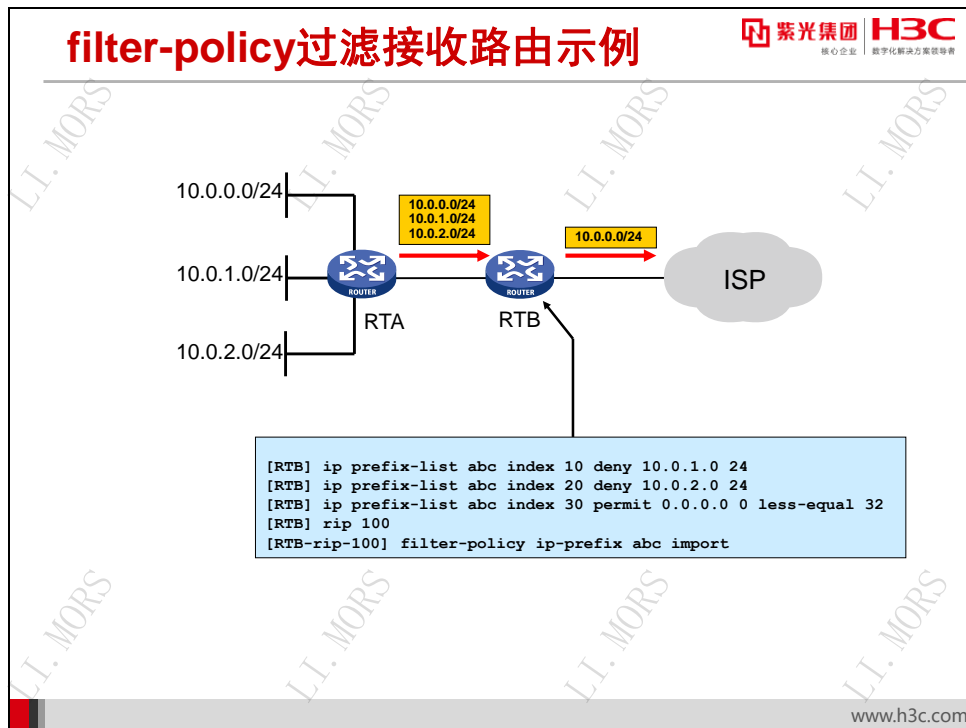
```
filter-policy { acl-number | prefix-list prefix-list-name } export [ protocol  
[ process-id ] | interface-type interface-number ]
```

其中参数含义如下：

- **protocol**: 被过滤路由信息的路由协议，如 bgp、direct、isis、ospf、rip 和 static 等。如果指定了 **protocol** 参数，则只对从指定路由协议引入的路由信息进行过滤；否则将对所有要发布的路由信息进行过滤。
- **process-id**: 被过滤路由信息的路由协议的进程号。
- **interface-type interface-number**: 接口类型和接口号。

如果指定 *interface-type interface-number* 参数，则只对从指定接口发布的路由信息进行过滤；否则将对所有 RIP 接口发布的路由信息进行过滤。

13.5.3 配置 filter-policy 过滤 RIP 路由示例

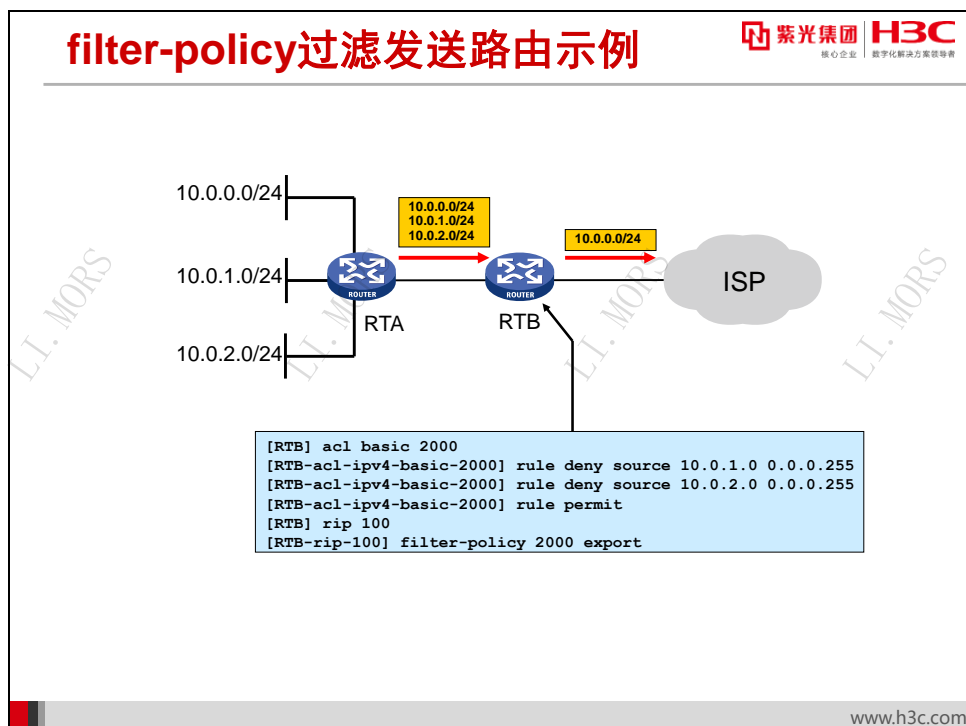


在上图所示网络中，RTA 向 RTB 发布了路由更新，包含了 10.0.0.0/24、10.0.1.0/24 和 10.0.2.0/24 路由信息。在 RTB 上配置 filter-policy，并使用地址前缀列表，使 RTB 拒绝接收其中的 10.0.1.0/24 和 10.0.2.0/24 路由，而可以接收其它路由。

RTB 配置如下：

```

[RTB] ip prefix-list abc index 10 deny 10.0.1.0 24
[RTB] ip prefix-list abc index 20 deny 10.0.2.0 24
[RTB] ip prefix-list abc index 30 permit 0.0.0.0 0 less-equal 32
[RTB] rip 100
[RTB-rip-100] filter-policy prefix-list abc import
  
```



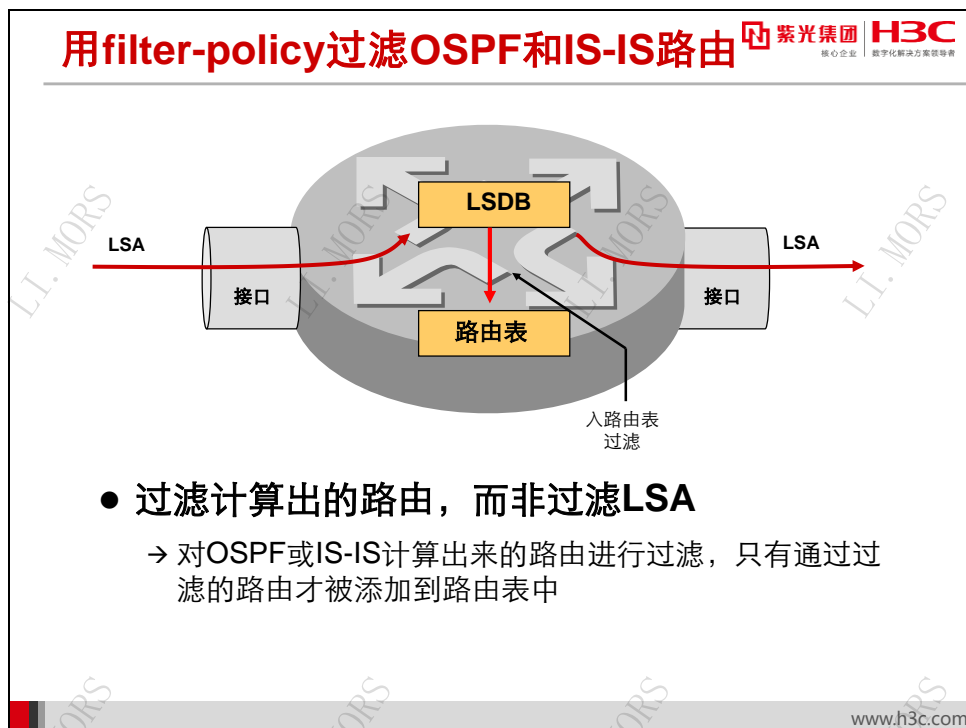
如果用 **filter-policy** 来过滤发送路由来达到相同的效果，则相应的配置如下：

```

[RTB] acl basic 2000
[RTB-acl-ipv4-basic-2000] rule deny source 10.0.1.0 0.0.0.255
[RTB-acl-ipv4-basic-2000] rule deny source 10.0.2.0 0.0.0.255
[RTB-acl-ipv4-basic-2000] rule permit
[RTB] rip 100
[RTB-rip-100] filter-policy 2000 export
  
```

以上配置中，使用 **ACL** 来过滤发送的路由信息，拒绝 10.0.1.0/24 和 10.0.2.0/24 的路由信息，而其它路由信息能够被 RTB 发送出去。

13.5.4 配置 filter-policy 过滤 OSPF 和 IS-IS 路由



OSPF 和 IS-IS 是链路状态型路由协议，协议间交换的是 LSA 而非路由信息，所以无法对协议接收和发送的路由信息进行过滤。由于 LSDB 必须同步，因而也不能过滤 LSA，而只能对依据 LSDB 计算出来的路由进行过滤，通过过滤的路由被添加到路由表中。

不过，如果路由器是 ABR（区域边界路由器），则可以通过在 ABR 上配置 Type3 LSA 过滤，对进入 ABR 所在区域或 ABR 向其它区域发布的 Type3 LSA 进行过滤。

说明：

虽然链路型状态路由协议无法过滤接收和发送的 LSA，但可以对通过路由引入方式产生的路由进行过滤。

配置filter-policy过滤OSPF及IS-IS路由

紫光集团 H3C
核心企业 数字化转型领导者

- 对OSPF通过接收到的LSA计算出来的路由信息进行过滤

```
[H3C-ospf-1] filter-policy { acl-number [ gateway  
prefix-list-name ] | gateway prefix-list-name | prefix-  
list prefix-list-name [ gateway prefix-list-name ] |  
route-policy route-policy-name } import
```

- 配置IS-IS对接收的路由在加入IP路由表时进行过滤

```
[H3C-isis-1] filter-policy { acl-number | prefix-list  
prefix-list-name | route-policy route-policy-name }  
import
```

www.h3c.com

在 OSPF 视图下，配置对 OSPF 计算出的路由进行过滤，其命令如下：

```
filter-policy { acl-number [ gateway prefix-list-name ] | gateway prefix-list-name |  
prefix-list prefix-list-name [ gateway prefix-list-name ] } import
```

其中的参数含义如下：

- **acl-number**: 用于过滤路由信息目的地址的基本或高级访问控制列表编号。
- **gateway prefix-list-name**: 指定的地址前缀列表，基于要加入到路由表的路由信息的下一跳进行过滤。
- **prefix-list prefix-list-name**: 指定的地址前缀列表，基于要加入到路由表的路由信息的目的地址进行过滤。

在 IS-IS 视图下，配置对 IS-IS 计算出的路由进行过滤，其命令如下：

```
filter-policy { acl-number | prefix-list prefix-list-name } import
```

其也是使用 ACL 或地址前缀列表对加入到路由表中的路由信息进行过滤。

13.6 本章总结

本章总结

- 利用路由过滤可控制路由在网络内传播
- ACL和地址前缀列表可用于路由信息的识别
- 地址前缀列表比ACL更加灵活
- 可利用filter-policy工具在RIP、OSPF、IS-IS等协议内过滤路由

www.h3c.com

13.7 习题和解答

13.7.1 习题

- 1、以下哪些工具属于路由过滤器？（ ）
A. ACL B. filter-policy C. Route-policy D. 地址前缀列表
- 2、在路由器上配置静默接口来使路由器不发送协议报文的命令是（ ）
A. [RTA] silent-interface serial 2/0 B. [RTA-ospf-1] silent-interface serial 2/0
C. [RTA-ospf-1-area-0.0.0.2] silent-interface serial 2/0
D. [RTA-Serial2/0] silent-interface
- 3、下列哪一个地址前缀列表匹配了缺省路由？（ ）
A. Permit 0.0.0.0 0 less-equal 32 B. Permit 0.0.0.0 0
C. Permit 0.0.0.0 255.255.255.255
D. Permit 0.0.0.0 255.255.255.255 less-equal 32
- 4、关于 filter-policy 过滤器，以下哪些说法是正确的？（ ）
A. 可以在 RIP 中使用 filter-policy 对从邻居接收的 RIP 路由信息进行过滤
B. 可以在 RIP 中使用 filter-policy 对发送给邻居的整个 IP 路由表进行过滤
C. 可以在 OSPF 中使用 filter-policy 对 LSA 计算出来的 OSPF 路由信息过滤
D. 可以在 IS-IS 中使用 filter-policy 对从邻居接收的 IS-IS 路由信息进行过滤
- 5、关于地址前缀列表中的各个表项，以下哪些说法是正确的？（ ）
A. 只要有某一表项满足条件，就意味着通过该地址前缀列表的过滤
B. 只有所有表项满足条件，才意味着通过该地址前缀列表的过滤
C. 每一个表项都指定了相应的匹配模式，包括允许模式和拒绝模式
D. 如果所有表项都是拒绝模式，则任何路由都不能通过该过滤列表

13.7.2 习题答案

1. ABCD 2. B 3. B 4. ABC 5. ABD

第14章 路由策略

Route-policy 是一种常用的、功能强大的路由策略工具。它不但能够过滤路由，还能对路由的属性进行改变。本章介绍了 Route-policy 的目的、应用和特点，Route-policy 中包含的节点匹配规则，以及相关的配置等。

14.1 本章目标

课程目标

● 学习完本课程，您应该能够：

- 掌握Route-policy的作用
- 掌握Route-policy的配置
- 掌握Route-policy的应用



www.h3c.com

14.2 路由策略概述



路由策略（Routing Policy）是为了改变网络流量所经过的途径而修改路由信息的技术，主要通过改变路由属性（包括可达性）来实现。

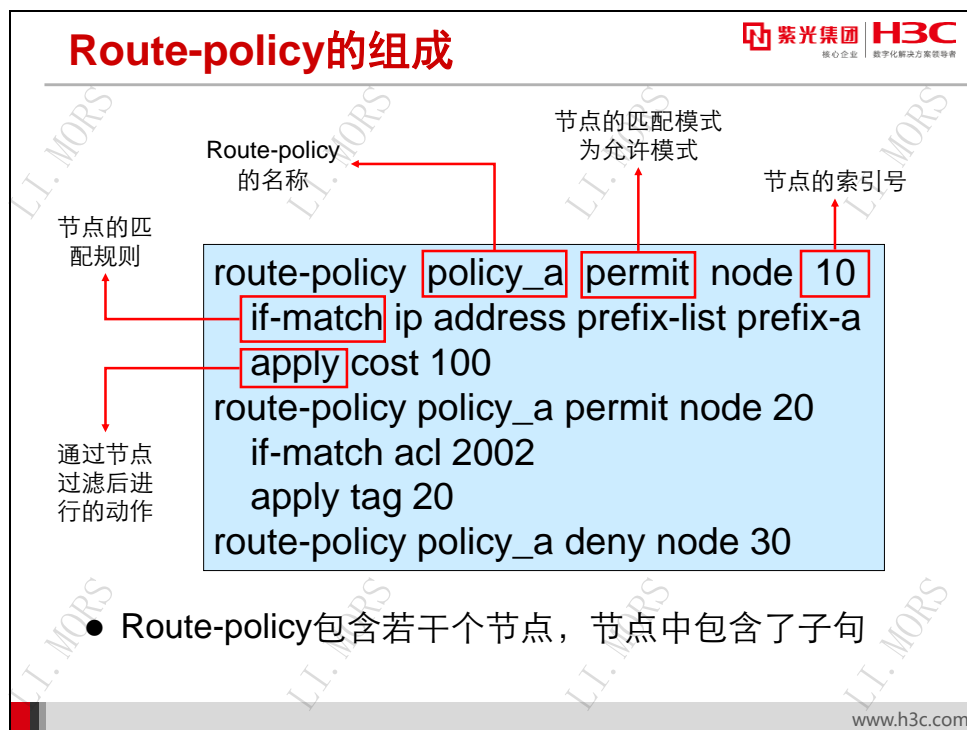
路由器在发布与接收路由信息时，可能需要实施一些策略，以便对路由信息进行过滤，例如只接收或发布满足一定条件的路由信息。一种路由协议可能需要引入其它的路由协议发现的路由信息，路由器在引入其它路由协议的路由信息时，可能只需要引入一部分满足条件的路由信息，并控制所引入的路由信息的某些属性，以使其满足本协议的要求。

为实现路由策略，首先要定义将要实施路由策略的路由信息的特征，即定义一组匹配规则。可以以路由信息中的不同属性作为匹配依据进行设置，如目的地址、发布路由信息的路由器地址等。匹配规则可以预先设置好，然后再将它们应用于路由的发布、接收和引入等过程的路由策略中。

Route-policy 是实现路由策略的工具。它实际上是一种比较复杂的过滤器，不仅可以匹配路由信息的某些属性，还可以在条件满足时改变路由信息的属性。

14.3 Route-policy组成和原理

14.3.1 Route-policy 组成

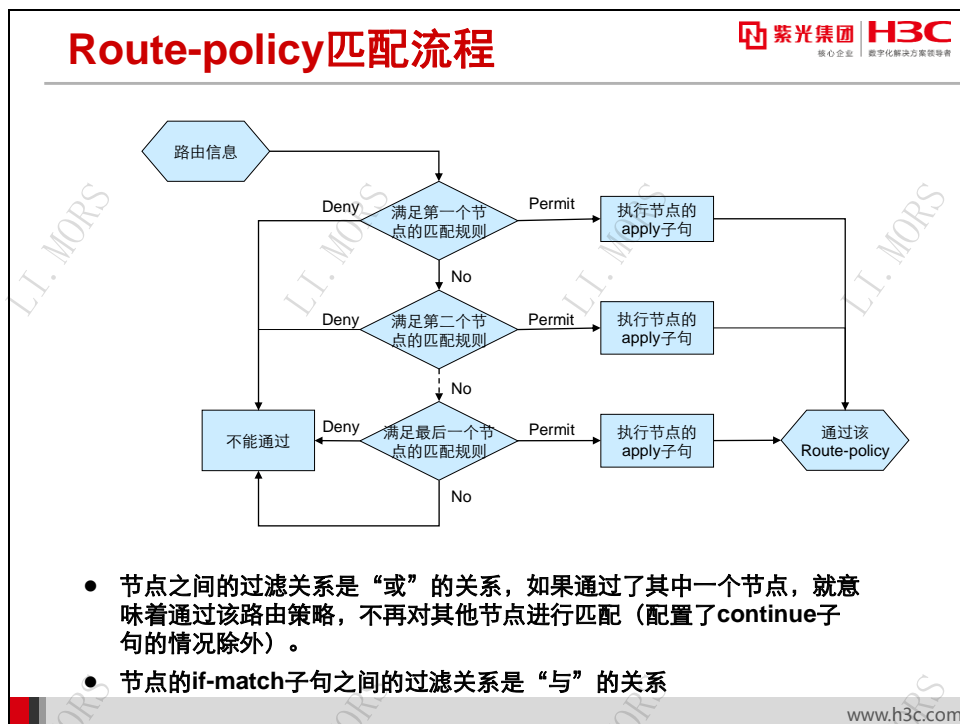


一个 Route-policy 可以由多个带有索引号的节点（node）构成，每个节点是匹配检查的一个单元，在匹配过程中，系统按节点索引号升序依次检查各个节点。

每个节点可以由一组 if-match 和 apply 子句组成。if-match 子句定义匹配规则，匹配对象是路由信息的一些属性。apply 子句指定动作，也就是在通过节点的匹配后，对路由信息的一些属性进行设置。

节点的匹配模式有允许模式（permit）和拒绝模式（deny）两种。允许模式表示当路由信息通过该节点的过滤后，将执行该节点的 apply 子句；而拒绝模式表示 apply 子句不会被执行。

14.3.2 Route-policy 匹配流程



一个 Route-policy 的不同节点间是“或”的关系，如果通过了其中一个节点，就意味着通过该路由策略，不再对其他节点进行匹配测试。

同一节点中的不同 if-match 子句是“与”的关系，只有满足节点内所有 if-match 子句指定的匹配条件，才能通过该节点的匹配测试。

如果节点的匹配模式为允许模式，则当路由信息满足该节点的匹配规则时，将执行该节点的 apply 子句，不进入下一个节点的测试；如果路由信息没有通过该节点过滤，将进入下一个节点继续测试。

如果节点的匹配模式为拒绝模式，则当路由项满足该节点的所有 if-match 子句时，将被拒绝通过该节点，不进入下一个节点的测试；如果路由项不满足该节点的 if-match 子句，将进入下一个节点继续测试。

当 Route-policy 用于路由信息过滤时，如果某路由信息没有通过任一节点，则认为该路由信息没有通过该 Route-policy。如果 Route-policy 的所有节点都是 deny 模式，则没有路由信息能通过该 Route-policy。所以，如果 Route-policy 中定义了一个以上的节点，则各节点中至少应该有一个节点的匹配模式是 permit。

节点的匹配规则

匹配规则 (if-match)	描述
ACL	路由信息的目的IP地址范围的匹配条件
prefix-list	路由信息的目的IP地址范围的匹配条件
ip next-hop	路由信息的下一跳地址的匹配条件
interface	路由信息的出接口的匹配条件
route-type	路由信息类型的匹配条件
tag	RIP、OSPF、IS-IS路由信息的标记域的匹配条件
cost	路由信息的路由开销的匹配条件

if-match 子句定义匹配规则，匹配对象是路由信息属性。最常用的路由信息属性包括目的 IP 地址范围、下一跳地址、出接口、开销（cost）、标记（tag）等。

对于 OSPF、IS-IS 等路由协议来说，路由属性还包括有路由类型（route-type）。而对于 BGP 协议，路由属性还包括有团体（community）、AS 路径（as-path）等。以上这些路由属性都可以作为匹配规则，由 if-match 子句所定义。

执行动作


动作 (apply)	描述
ip-address next-hop	设定通过过滤后路由信息的下一跳地址
preference	设定通过过滤后路由协议的优先级
tag	设定通过过滤后RIP、OSPF、IS-IS路由信息的标记域
cost	设定通过过滤后路由信息的路由开销
cost-type	设定通过过滤后路由信息的路由开销类型

使用 **apply** 子句来指定动作，对通过节点的路由信息属性进行设置。

可以对通过节点的路由信息的下一跳地址、优先级、标记、开销等进行设定。对于 OSPF 路由，还可以设定路由开销类型，以将通过过滤的路由改变为 **Type-1** 路由或 **Type-2** 路由；对于 IS-IS 路由，通过设定路由开销类型可以将通过过滤的路由改变为 IS-IS 外部路由或内部路由。

14.4 Route-policy配置与查看

Route-policy配置



紫光集团 H3C
核心企业 数字化解决方案领导者

- 创建Route-policy

```
[H3C] route-policy route-policy-name { permit | deny } node node-number ]
```

- 配置if-match子句

```
[H3C-route-policy] if-match { 匹配规则 }
```

- 配置apply子句

```
[H3C-route-policy] apply { 动作 }
```

www.h3c.com

在配置 Route-policy 之前，需要规划好 Route-policy 的名称、节点索引号，节点中子句的匹配规则，通过节点过滤后要执行的动作。

配置 Route-policy 的步骤如下：

第1步：在系统视图下创建 Route-policy，并定义名称、节点索引号、匹配模式等参数。命令如下：

```
route-policy route-policy-name { permit | deny } node node-number
```

第2步：在 Route-policy 视图下使用 if-match 子句来设定路由信息的匹配条件。命令如下：

```
if-match { 匹配规则 }
```

if-match 子句后是路由信息匹配规则的设定，可选的参数包括 ACL、IP-Prefix、cost、interface、route-type、tag、ip next-hop 等。


可通过 ACL、IP-Prefix 来对目的 IP 地址进行匹配，也可以使用 cost、interface、route-type、tag、ip next-hop 参数分别对开销、出接口、路由类型、标记域、下一跳等路由属性进行匹配。

第3步：在 Route-policy 视图下使用 apply 子句来指定通过过滤后所执行的动作。命令如下：

```
Apply { 动作 }
```

apply 子句后可选的动作参数包括 cost、cost-type、preference、tag、ip-address next-hop 等，可以分别对路由信息的开销、开销类型、优先级、标记域、下一跳地址等进行设定。

Route-policy配置示例



紫光集团 H3C
核心企业 数字化转型领导者

配置	结果
route-policy policy_a permit node 10 if-match ip address prefix-list prefix-a apply cost 100	符合地址前缀列表prefix-a的路由能够通过过滤，并设定其cost值为100； 其它路由不能通过过滤。
route-policy policy_a permit node 10 if-match ip address prefix-list prefix-a apply cost 100 route-policy policy_a permit node 20 if-match ip address acl 2002 apply tag 20	符合地址前缀列表prefix-a的路由能够通过过滤，并设定其cost值为100；符合访问控制列表2002的路由能够通过过滤，并设定其tag值为20； 其它路由不能通过过滤。
route-policy policy_a deny node 10 if-match ip address acl 2002	所有路由不能通过过滤。

- 指定节点的匹配模式为拒绝模式时，此节点下的apply子句和continue子句不会被执行。当路由信息通过该节点的过滤后，将被拒绝通过该节点，不进入下一个节点的匹配；如果路由信息没有通过该节点的过滤，将进入下一个节点继续匹配。

www.h3c.com

如图所示为 Route-policy 配置示例及相应过滤结果。

- 当配置如下时，：

```
[H3C]route-policy policy_a permit node 10
[H3C-route-policy]if-match ip address prefix-list prefix-a
[H3C-route-policy]apply cost 100
```

结果是匹配地址前缀列表 prefix-a 的路由能够通过过滤，并设定其 cost 值为 100；而其它路由不能通过过滤。

- 当配置如下时：

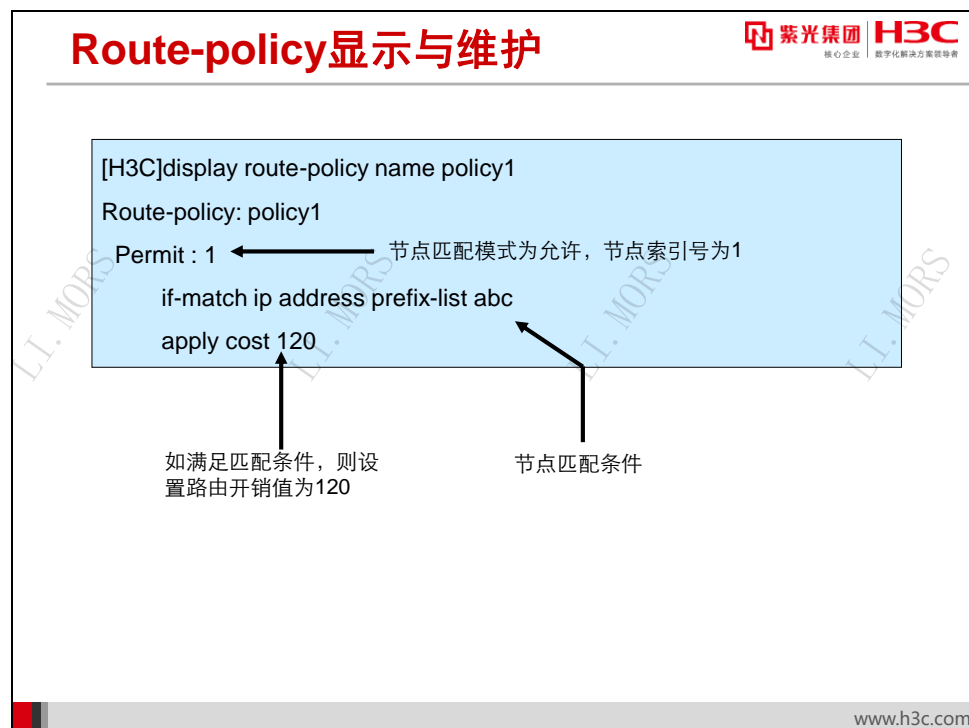
```
[H3C]route-policy policy_a permit node 10
[H3C-route-policy]if-match ip address prefix-list prefix-a
[H3C-route-policy]apply cost 100
[H3C]route-policy policy_a permit node 20
[H3C-route-policy]if-match ip address acl 2002
[H3C-route-policy]apply tag 20
```

结果是匹配地址前缀列表 prefix-a 的路由能够通过过滤，并设定其 cost 值为 100；符合访问控制列表 2002 的路由能够通过过滤，并设定其 tag 值为 20；而其它路由不能通过过滤。

- 当配置如下时：

```
[H3C]route-policy policy_a deny node 10
[H3C-route-policy]if-match ip address acl 2002
```

因此 Route-policy 仅有一个节点且节点的匹配模式是拒绝，所以结果是所有路由都不能通过过滤。



在完成 Route-policy 的配置后，在任意视图下执行 `display` 命令可以显示配置后 Route-policy 的运行情况，验证配置的效果。相关命令为：

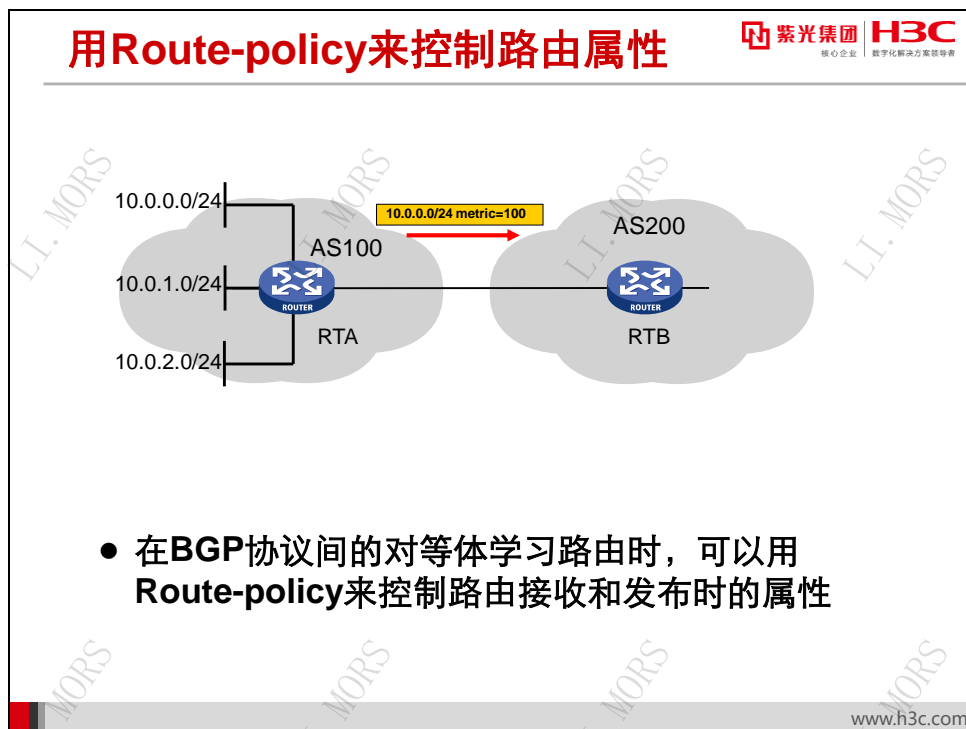
`display route-policy [name route-policy-name]`

典型的 `display route-policy` 显示输出如下：

```
[H3C]display route-policy name policy1
Route-policy : policy1
permit : 1
if-match ip address prefix-list abc
apply cost 120
```

以上输出表明，route-policy 名称为 `policy1`，包含了 1 个节点，所设定的节点索引号为 1，节点的匹配模式是允许模式。节点的匹配条件是 `prefix-list`，名称为 `abc`；如果有路由信息通过此节点的过滤，则设定开销值为 120。

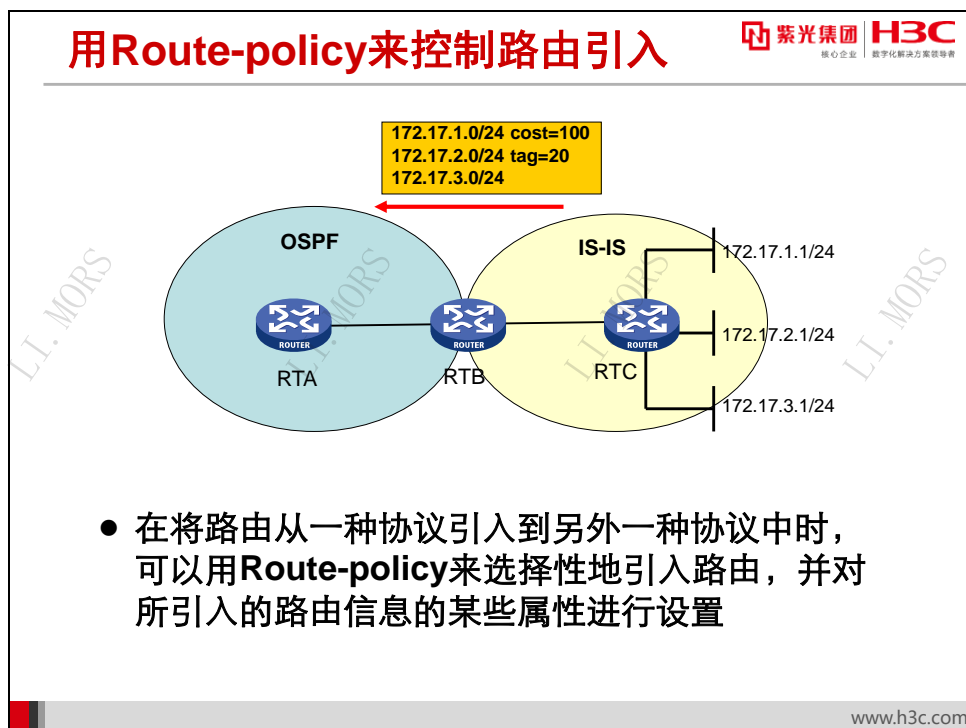
14.5 Route-policy的应用



作为实现路由策略的工具，Route-policy 被广泛应用在路由过滤、路由属性改变等场合中。其中最常用于 BGP 协议内路由学习的控制和改变接收和发送路由时的路由属性。

IGP 协议，如 RIP、OSPF，仅工作在自治系统内，路由数量较少，一般不需要进行路由学习的控制。但是 BGP 协议是在大规模网络上工作的路由协议，用来互连多个自治系统，如果没有路由学习控制，学习到的路由数量可能会极其巨大。另外，BGP 协议具有丰富的路由属性，对这些属性进行适当的调整，可以控制 BGP 的选路。

在上图所示网络中，RTA 通过 BGP 协议发布路由到 RTB。根据策略，RTB 仅需要发布 10.0.0.0/24 路由到 RTB，且在发布时修改此条路由的度量值为 100，以上需求可以使用 Route-policy 来实现。



Route-policy 的另一种常见应用是在路由引入时进行控制及路由属性的改变。

将路由从一种协议引入到另外一种协议中时，有时并不是需要把所有路由都引入，而是要有选择性的引入。此时，可以用 Route-policy 来设定匹配条件，以仅使符合匹配条件的路由能够被成功引入。比如，在从 BGP 向 OSPF 中引入路由时，可以设定仅符合某一部分地址前缀的路由被引入，以控制路由数量。

另外，在路由引入时，经常会使用 Route-policy 来改变引入后路由的属性，以达到控制路由，防止环路的目的。在上图所示网络中，RTB 作为边界路由器，负责把 IS-IS 内的路由引入到 OSPF 中。根据策略，RTB 需要把 172.17.1.0/24、172.17.2.0/24、172.17.3.0/24 等 3 条路由引入到 OSPF 中，并对 172.17.2.0/24 这条路由的标记域赋值为 20，将 172.17.1.0/24 这条路由的开销值改为 100，以上需求可以使用 Route-policy 来实现。

14.6 本章总结

本章总结

- **Route-policy**由若干个节点组成，节点中包含了if-match子句和apply子句
- 节点之间的过滤关系是“或”的关系
- 路由学习时，可使用**Route-policy**控制路由
- 路由引入时，可使用**Route-policy**改变路由属性

14.7 习题和解答

14.7.1 习题

1. Route-policy 的作用包括 ()
A. 路由过滤 B. 报文过滤 C. 改变路由的属性 D. 改变报文的内容
2. 关于 Route-policy, 下列哪些说法是正确的? ()
A. 一个 Route-policy 的不同节点间是“或”的关系
B. 同一节点中的不同 if-match 子句是“与”的关系
C. 节点的匹配模式包括允许模式和拒绝模式
D. 如果所有节点都是拒绝模式, 则没有路由信息能通过该 Route-policy
3. 在 Route-policy 配置中, 下列哪些匹配规则可以由 if-match 子句来设定? ()
A. 开销 B. 出接口 C. 路由类型 D. 标记域 E. IP 目的地址
F. 下一跳
4. 在 Route-policy 配置中, 下列哪些动作可以由 apply 子句来执行? ()
A. 开销 B. 出接口 C. 路由类型 D. 标记域 E. IP 目的地址
F. 下一跳
5. Route-policy 常应用在下列哪些场合? ()
A. 路由引入时实行路由过滤 B. IGP 路由学习时进行过滤控制
C. 路由引入时改变路由的属性 D. BGP 路由学习时进行过滤控制

14.7.2 习题答案

1. AC
2. ABCD
3. ABCDEF
4. ACDF
5. ACD

第15章 路由引入

进行网络设计时，一般都仅选择运行一种路由协议，以降低网络的复杂性，使易于维护。但是在现实中，当需要更换路由协议，或需要对运行不同路由协议的网络进行合并时，有可能在网络中同时运行多种路由协议。本章介绍了在多路由协议网络运行环境下，如何进行路由协议间的引入和部署。

15.1 本章目标

课程目标

● 学习完本课程，您应该能够：

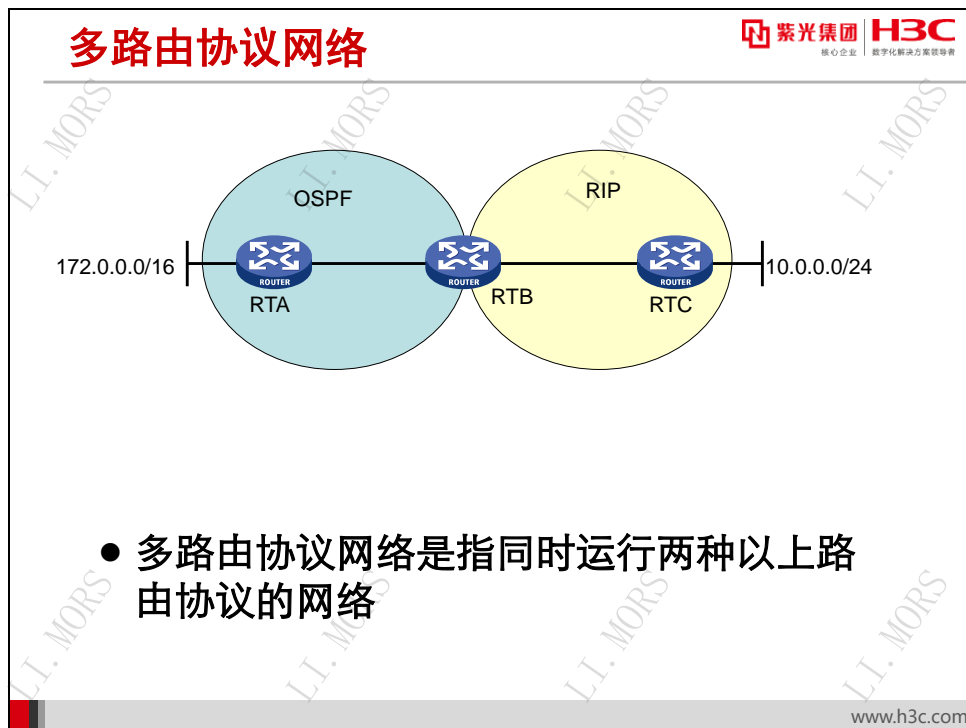
- 了解路由引入的背景
- 掌握路由引入的作用
- 掌握路由引入的规划
- 掌握在IGP中配置路由引入



www.h3c.com

15.2 多协议网络与路由引入

15.2.1 多协议网络



如果一个网络同时运行了两种以上路由协议，如同时运行了 OSPF 和 RIP 协议，或同时运行了路由协议和静态路由，则这个网络是多路由协议网络。

路由器维护了一张 IP 路由表，路由表中的路由来源于不同路由协议。由于不同路由协议之间算法不同，度量值不同，所以不同路由协议学习到的路由信息不能直接互通，一个路由协议学习的路由不能够直接传送到另一个路由协议去。

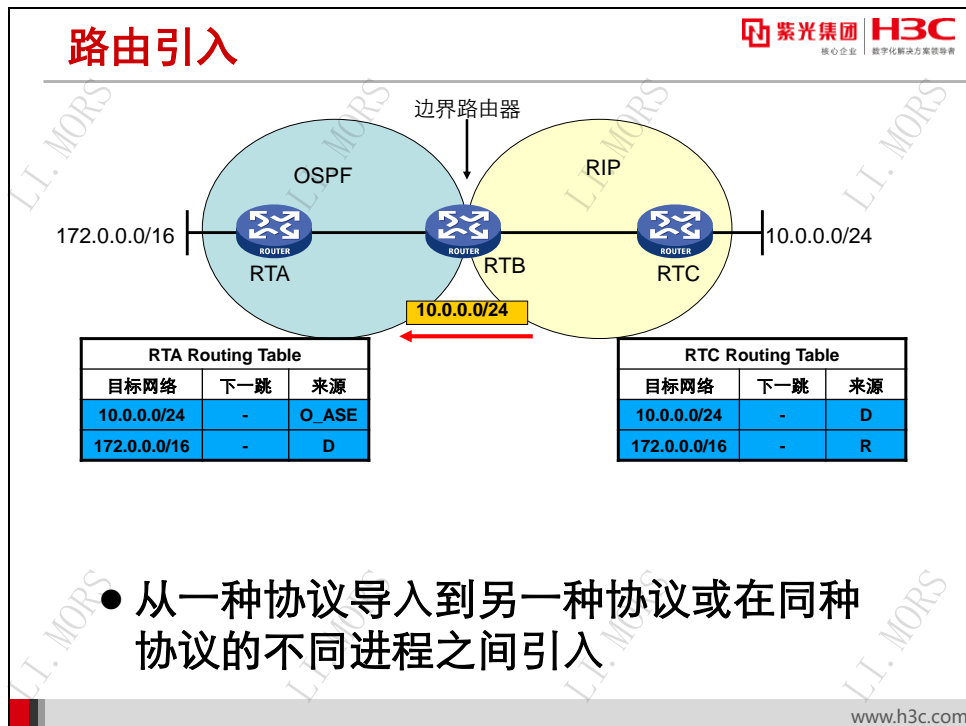
在网络合并、升级、迁移的过程中，经常会出现多路由协议的情况。比如，早期网络中使用了 RIP 协议，但随着网络规模的扩大，路由器的数量超过了 15 台，RIP 协议就变得不再适用了。此时，管理员可以将 RIP 升级成 OSPF。升级过程中可能会出现两种协议共同运行的情况。又比如，两个公司网络运行了不同的路由协议，两公司合并时，就会出现两种路由协议共同运行的情况。

网络中运行多个路由协议时，需要使用路由引入来将一种路由协议的路由信息引入到另外一种路由协议中去，以达到网络互通的目的。

在上图所示网络中，RTA 和 RTB 运行 OSPF 协议；RTB 和 RTC 运行 RIP 协议。RTA 连接到网络 172.0.0.0/16，RTC 连接到网络 10.0.0.0/24。因为 RTA 和 RTC 不是运行同一种路由协议，所以它们并不能互相学习路由信息，也就无法互通。但 RTB 既运行了 OSPF，又运行了

RIP，它能够学习到网络 172.0.0.0/16 和 10.0.0.0/24，所以可以在 RTB 上使用路由引入来使 RIP 和 OSPF 协议互相学习到对方的路由信息。

15.2.2 路由引入



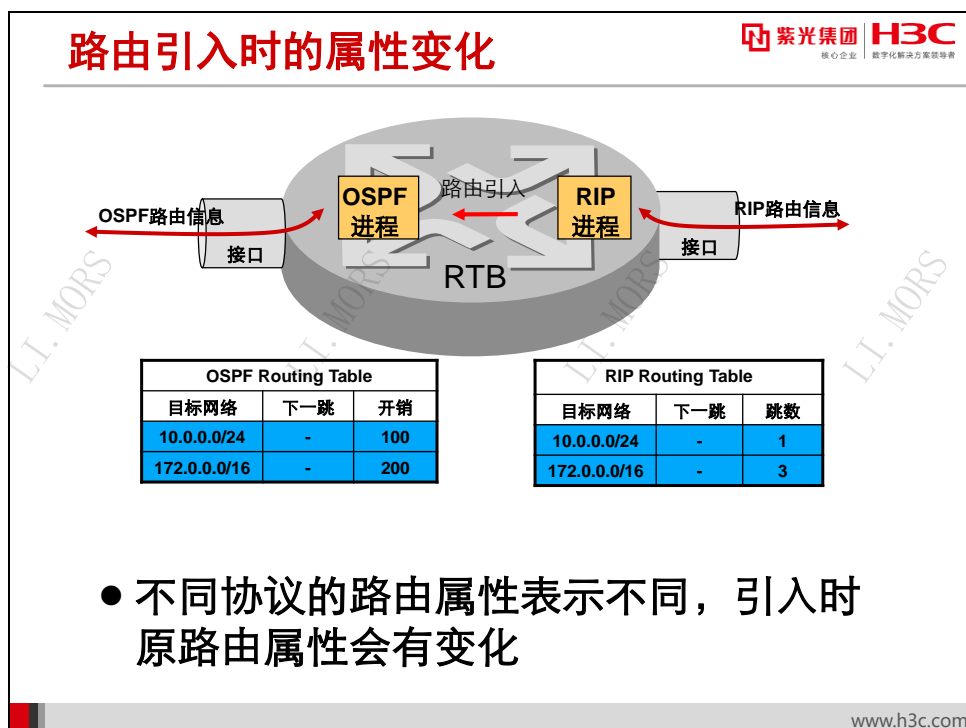
通过使用路由引入，管理员可以把路由信息从一种路由协议导入到另外一种协议；或者在同种协议的不同进程之间导入。

路由引入通常在边界路由器上进行。边界路由器是同时运行两种以上路由协议的路由器，它作为不同路由协议之间的桥梁，负责在不同路由协议间的路由引入操作。

上图所示网络中，RTB 作为边界路由器，同时运行 OSPF 和 RIP 协议。它一方面与 RTA 通过 OSPF 协议交换路由信息，另一方面与 RTC 通过 RIP 协议交换路由信息。在 RTB 上实施路由引入后，它把通过 RIP 学习到的路由导入到 OSPF 协议的 LSDB 中，然后以 LSA 的形式发送到 RTA。这样，RTA 的路由表中就有了 10.0.0.0/24 这条路由。同理，RTB 把 OSPF 路由引入到 RIP 路由表中，所以 RTC 就学到了 172.0.0.0/16 这条路由。

注意：

只有协议路由表中的有效（Active）路由才能成功引入。



在路由引入时，由于不同协议的路由属性表达方式不一样，所以原路由属性会发生变化。

不同协议的度量值算法不同，所以在路由引入时，无法将路由信息的原度量值也引入。此时，协议一般会给予路由信息一个新的缺省度量值，也称为“种子度量值”。路由信息在路由器间传播时，会以新的缺省度量值为基础进行度量值的计算。缺省度量值可以设置，以适应网络的实际情况，通常设置为大于路由域内已有路由信息的最大度量值，表示是从域外引入的路由，以避免可能出现的次优路由。

下表给出各不同协议路由引入时的缺省度量值。

表15-1 路由引入时缺省度量值

路由协议	度量值类型	缺省度量值
RIP	跳数	0
OSPF	开销 (Cost)	1
IS-IS	开销 (Cost)	0
BGP	MED	使用被引入路由的度量值作为引入BGP之后的MED值

有些路由协议会对引入的路由给予特殊的标记，以表明此路由是从其它路由协议引来的。比如，OSPF 协议会把所有引入的外部路由标记为“第二类外部路由 (Type2 External)”，并给予一个路由标记 (Tag) 值 1；而 IS-IS 协议会把引入的路由放到 Level-2 路由表中，并设定外部路由开销值为 0。

15.3 路由引入规划

15.3.1 概述

多协议网络规划

紫光集团 H3C
核心企业 数字化解决方案领导者

- 只在必要时使用多路由协议
- 路由协议的规划
 - 边缘引入到核心
 - IGP引入到BGP
- 路由引入点的规划
 - 单边界引入
 - 多边界引入

www.h3c.com

在网络中运行多路由协议给网络带来了更高的复杂度。不同路由协议算法不同，路由属性不同，收敛速度不同，混合使用可能造成次优路由或路由收敛不一致。运行多路由协议也对路由器的 CPU、内存等资源要求更高。所以，只是在必要的时候才运行多路由协议。

常见的运行多协议的网络有以下几种情况：

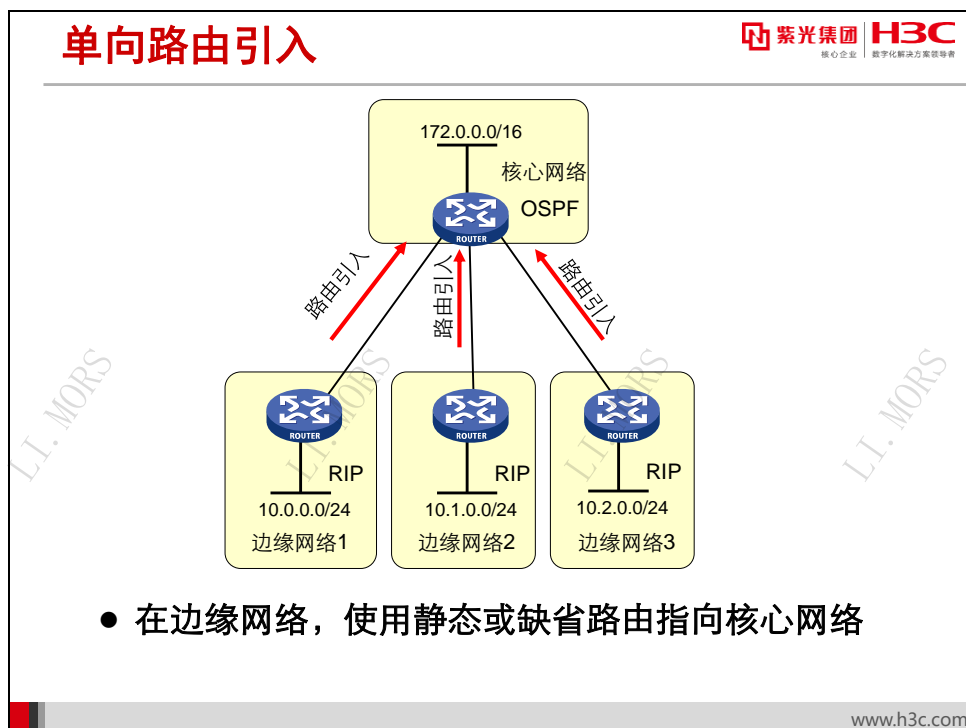
- 网络升级、合并、迁移时会出现多协议共存。此时一般会采用两个路由协议共存，并逐步切换到新路由协议。在共存期间，会使用路由引入来使两种路由协议间互相学习到路由信息。
- 网络中不是所有设备都支持同一种路由协议。小的接入层设备可能会不支持复杂的路由协议，或某个厂家的设备运行自己的私有协议。在此种情况下，规划部分设备运行一种路由协议，另一部分设备运行另外一种路由协议，然后在边界路由器上实施路由引入。
- 在不同的路由域间进行路由控制。正因为不同路由协议间不能自动学习路由，所以可以在网络中实施多协议，以划分出不同的路由域，在域的边界进行路由引入时进行路由控制。

在多协议网络规划中，通常在核心网络运行链路状态型路由协议，如 OSPF、IS-IS 等，以加快收敛速度，提高网络可靠性。而在边缘网络运行简单的路由协议如 RIP 或静态路由。此时，

实施路由引入时，通常把路由从边缘网络引入到核心网络，而在边缘网络配置静态路由指向核心网络。而如果网络中同时运行 IGP 和 BGP 时，通常是把 IGP 引入到 BGP 中，再通过 BGP 来与外界网络交换路由，以利用 BGP 协议丰富的属性来进行路由控制与选路。

路由引入时，可以仅在一台边界路由器上引入，称为单边界引入；也可以在多个边界路由器上引入，称为多边界引入。单边界引入时，相当于两个路由域间仅有一个连接点，可靠性相对较差，但优点是不会有环路或次优路由产生。在多边界引入时，不同路由域间有多条路径，可靠性增加了，但配置更加复杂，也增加了产生次优路由的可能性。

15.3.2 单向路由引入



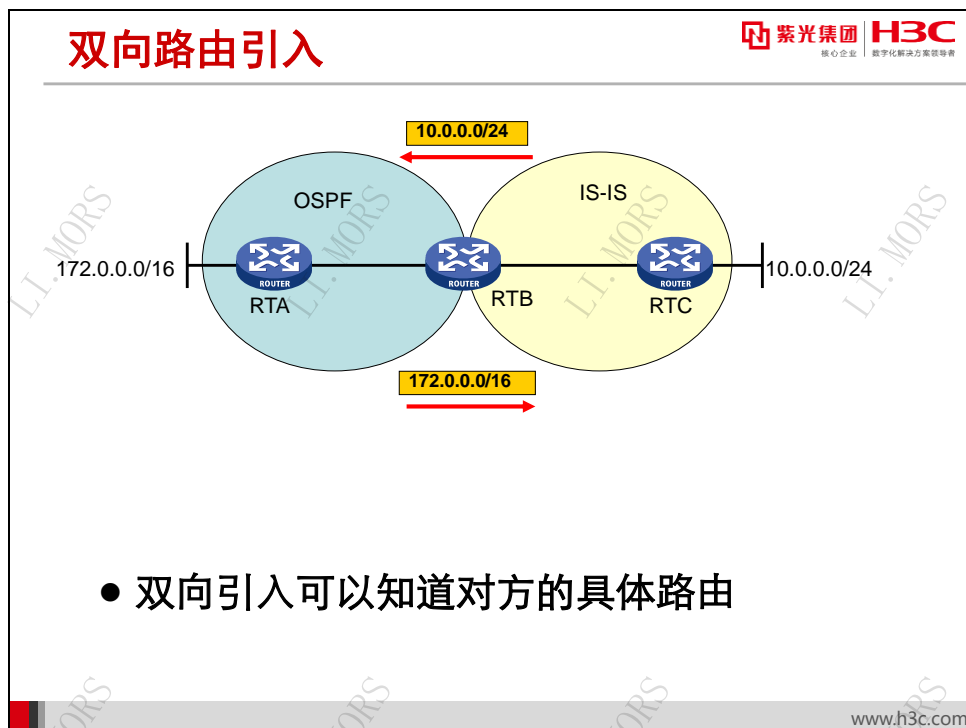
路由引入时，如果把路由信息仅从一个路由协议引入到另外一个路由协议，没有反向引入，则称为路由的单向引入。

如上图所示网络中，核心网络运行 OSPF 协议，边缘网络运行 RIP 协议。在核心网络的边界路由器上实施路由引入，把从边缘网络路由器学习到的路由信息引入到核心网络所运行的路由协议 OSPF 中。这样，核心网络就知道了边缘网络的所有路由信息，一个边缘网络发出的数据报文可以经过核心网络转发到另外的边缘网络。

单向引入会造成单向路由。如上图，核心网络通过路由引入知道了边缘网络的路由 10.0.0.0/24、10.1.0.0/24 和 10.2.0.0/24，但边缘网络并不知道核心网络的路由 172.0.0.0/16，也不知道其它边缘网络的路由。此时，需要在边缘网络路由器上配置静态或缺省路由，下一跳指向核心网络的边界路由器；或者，也可以由核心网络的边界路由器发布缺省路由。

单向路由引入适用于星形拓扑网络中。

15.3.3 双向路由引入



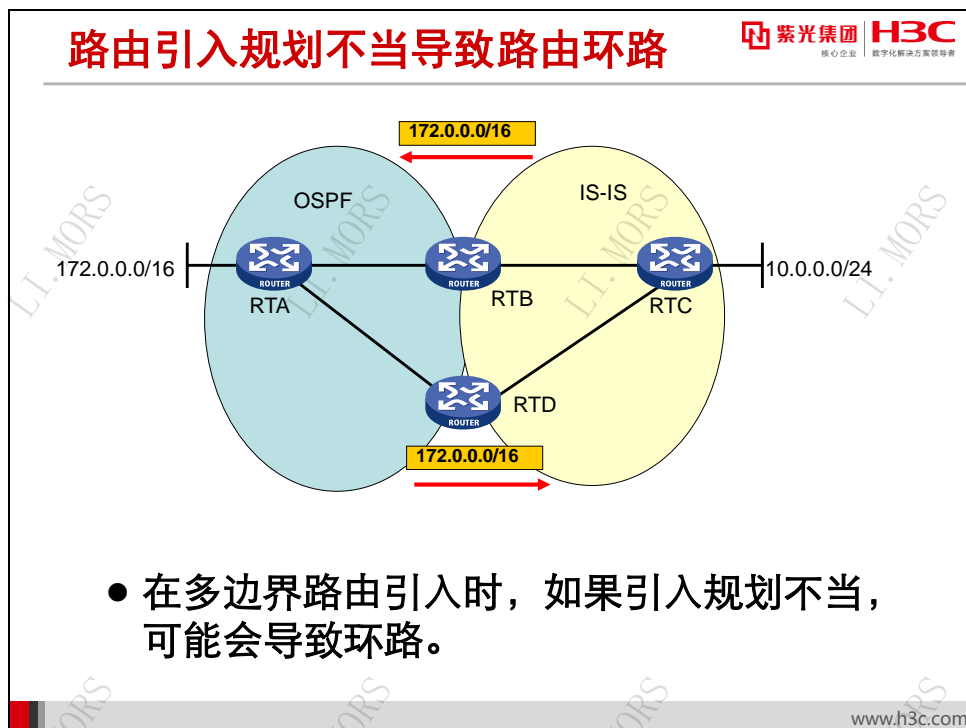
在边界路由器上把两个路由域的路由互相引入，称为双向引入。

在上图所示网络中，边界路由器 RTB 把 OSPF 路由域中的路由 172.0.0.0/16 引入到 IS-IS 路由域中，同时把 IS-IS 路由域中的路由 10.0.0.0/24 引入到 OSPF 路由域中。这样，RTA 和 RTC 就知道了彼此的具体路由。

系统在路由引入时，只会把路由表中的有效路由引入到协议中，且引入后的路由不在本地路由表中出现，只传递给其它路由器。如上图所示，RTB 从 RTC 通过 IS-IS 协议学到路由 10.0.0.0/24，作为有效路由放置在 IP 路由表中；同时把路由 10.0.0.0/24 引入到 OSPF 协议，加入 OSPF 数据库后，通过 OSPF 的 LSA 发送给 RTA。同理，RTB 把从 RTA 学到的路由 172.0.0.0/16 作为有效路由放置在 IP 路由表中，同时把它引入到 IS-IS 协议数据库后，发送到 RTC。这样，在 RTB 的本地路由表中，路由 10.0.0.0/24 和 172.0.0.0/16 仍然携带有原路由属性。

需要知道对方的具体路由时，可以使用双向引入。比如，某公司与另一公司合并，双方使用不同的路由协议，且路由数量众多，使用静态路由配置复杂；且公司都连接到 Internet，所以不适合在边界路由器发布缺省路由。此时，使用双向引入是较好的选择。

15.3.4 路由引入产生环路及解决方法

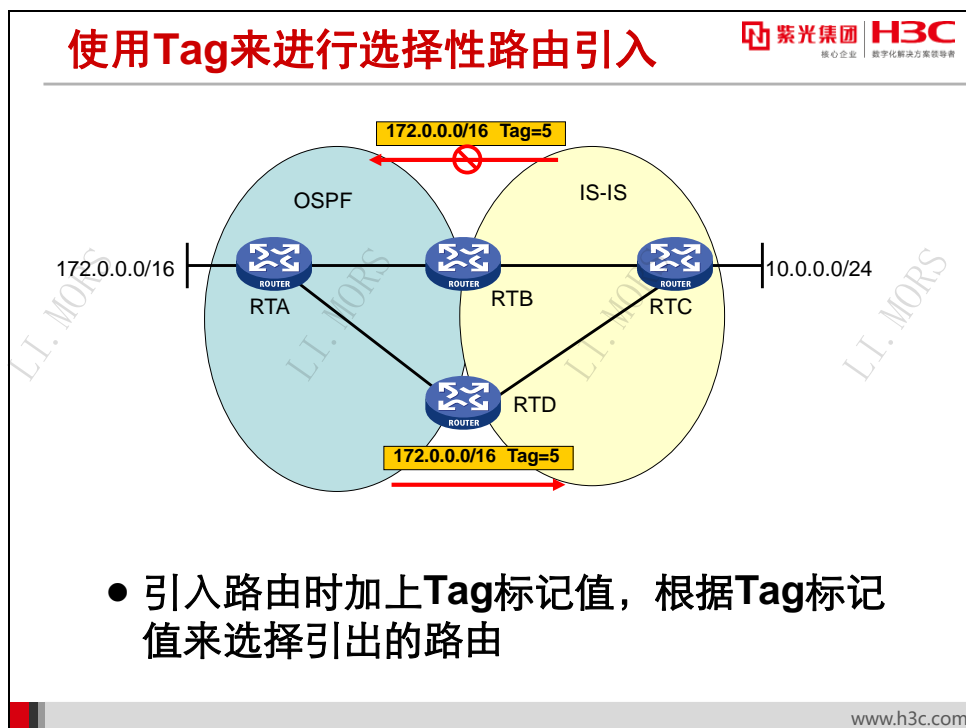


在多边界路由引入时，如果引入规划不当，可能会导致环路。

如上图所示，RTB 和 RTD 作为边界路由器，在 OSPF 和 IS-IS 间进行路由引入，RTD 配置为将 OSPF 协议路由引入到 IS-IS 协议中，而 RTB 配置为将 IS-IS 路由引入到 OSPF 协议中。RTD 从 RTA 学习到路由 172.0.0.0/16 后，引入到 IS-IS 协议中，发布到 RTC，RTC 再发布给 RTB；此时 RTB 并不知道这条路由是从 OSPF 域中引来，所以会再次引入到 OSPF 域中。

以上情况类似于距离矢量路由协议在多路径网络中，在特定条件下可能会产生路由环路。

避免以上情况发生的办法是，在边界路由器上有选择性的进行路由引入。

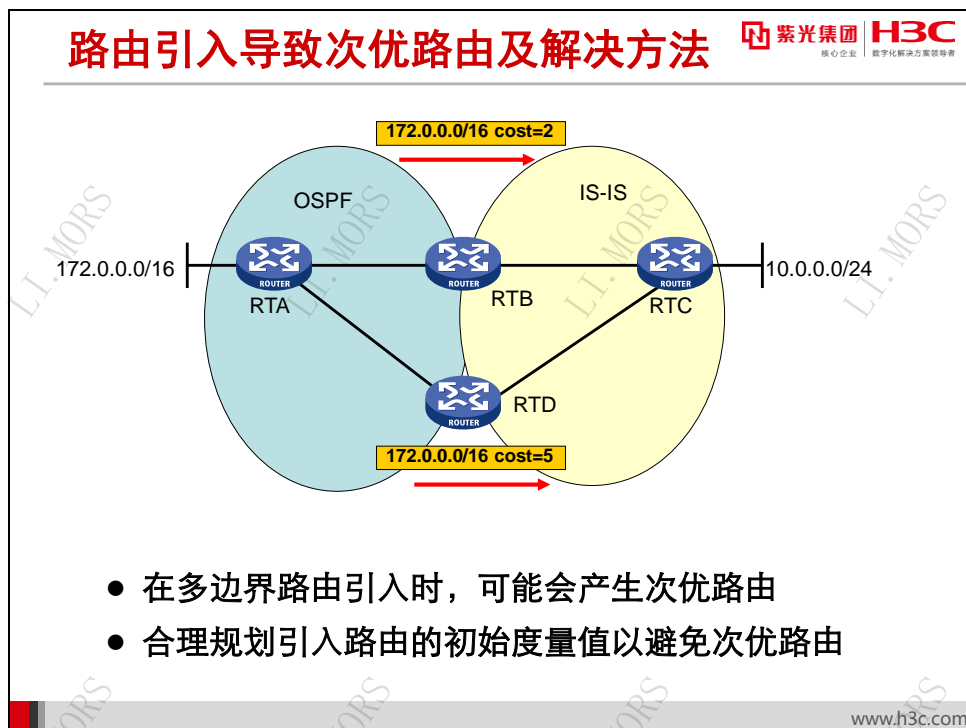


可以使用路由属性中的标记值（Tag）来实现选择性路由引入。

上图所示网络中,RTD 将从 OSPF 域中引入的路由加上 Tag 值等于 5 的标记,发布到 IS-IS 域中。在 RTB 上,配置把 IS-IS 域中除了 Tag 值等于 5 的其它路由引入到 OSPF 域中,这样,RTB 就不会把路由 172.0.0.0/16 引入到 OSPF 中,也就实现了选择性引入。

使用 Tag 来选择性引入路由简单易用,所以得到了广泛的应用。

15.3.5 路由引入产生次优路由及解决方法



路由引入的另一个常见问题是导致次优路由的产生。路由引入时，原路由属性如度量值丢失，需要协议重新给定缺省度量值或由管理员手工设定度量值，这样在网络规划不合理的情况下，会产生次优路由。

在上图所示网络中，从 RTA 到 RTC 有两条路径，假设在单路由协议环境中，RTA->RTD->RTC 的路径是最优路径。在运行多协议后，边界路由器 RTB 将路由 172.0.0.0/16 引入到 IS-IS 中，并设定开销值为 2；同时 RTD 也将路由 172.0.0.0/16 引入到 IS-IS 中，并设定开销值为 5。这样，RTC 经过开销值比较，认为经由 RTB 到达 172.0.0.0/16 的路径是最优路径，次优路由产生了。

通过合理的规划以尽量避免产生次优路由。通常在多边界引入时，给定所有引入的路由以相同的缺省度量值，这样至少在域内范围能够避免次优路由。对于域外路由，由于原路由属性在引入时丢失了，所以协议本身并不能判断出原路由的度量值大小。这时通常由管理员手工调节路由引入后的度量值，使之反映原路由的度量值，从而避免次优路由的产生。

15.4 路由引入配置

15.4.1 配置 RIP 协议引入外部路由

RIP路由引入



紫光集团 H3C
核心企业 数字化解决方案领导者

- 配置RIP引入外部路由

```
[H3C-rip-1]import-route protocol [ process-id | all-processes | allow-ibgp ] [ allow-direct | cost cost | route-policy route-policy-name | tag tag ]
```

- 配置引入路由的缺省度量值

```
[H3C-rip-1]default cost value
```

www.h3c.com

首先进入 RIP 协议视图，在 RIP 视图下配置 RIP 协议引入外部路由。其命令如下：

```
import-route protocol [ process-id | all-processes | allow-ibgp ] [ allow-direct | cost cost | route-policy route-policy-name | tag tag ]
```

其中的参数含义如下：

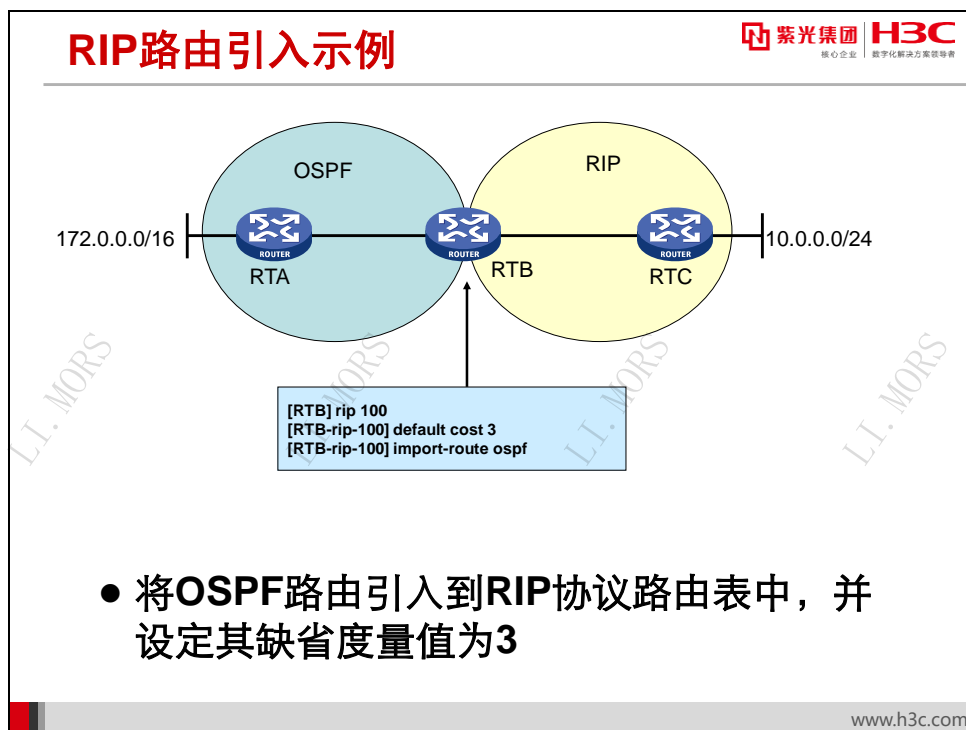
- **protocol**: 指定引入的路由协议，可以是 **bgp**、**direct**、**isis**、**ospf**、**rip** 或 **static**。
- **process-id**: 路由协议进程号，取值范围为 1~65535，缺省值为 1。只有当 **protocol** 是 **isis**、**ospf** 或 **rip** 时该参数可选。
- **all-processes**: 引入指定路由协议所有进程的路由，只有当 **protocol** 是 **rip**、**ospf** 或 **isis** 时可以指定该参数。
- **allow-ibgp**: 当 **protocol** 为 **bgp** 时，**allow-ibgp** 为可选关键字。
- **allow-direct**: 在引入的路由中包含使能了该协议的接口网段路由。缺省情况下，在引入协议路由时不会包含使能了该协议的接口网段路由。当 **allow-direct** 与 **route-policy route-policy-name** 参数一起使用时，需要注意路由策略中配置的匹配规则不要与接口路由信息存在冲突，否则会导致 **allow-direct** 配置失效。例如，当配置 **allow-direct** 参数引入 OSPF 直连时，在路由策略中不要配置 **if-match route-type** 匹配条件，否则，**allow-direct** 参数失效。

- **cost cost:** 所要引入路由的度量值，取值范围为 0~16，缺省值为 0。
- **route-policy route-policy-name:** 路由策略名称，*route-policy-name* 为 1~63 个字符的字符串，区分大小写。
- **tag tag:** 所要引入路由的标记值，取值范围为 0~65535，缺省值为 0。

因为在缺省情况下，引入路由的缺省度量值为 0，所以可根据网络情况对缺省度量值进行调整。其命令如下：

default cost value

建议缺省度量值取路由域内度量值的最大值。



在上图所示网络中，RTA 和 RTB 运行 OSPF 协议；RTB 和 RTC 运行 RIP 协议。RTA 连接到网络 172.0.0.0/16，RTC 连接到网络 10.0.0.0/24。在 RTB 上配置路由引入，将 OSPF 路由引入到 RIP 协议路由表中，并设定缺省度量值为 3。

配置 RTB:


```

[RTB] rip 100
[RTB-rip-100] default cost 3
[RTB-rip-100] import-route ospf
  
```

配置完成后，可以在 RTC 的路由表中查看到路由 172.0.0.0/16，其度量值为 4。

15.4.2 配置 OSPF 协议引入外部路由

OSPF路由引入



紫光集团 H3C
核心企业 数字化转型方案领导者

- 配置OSPF引入外部路由


```
[H3C-ospf-1]import-route protocol [ process-id | all-processes |
allow-ibgp ] [ allow-direct | cost cost | nssa-only | route-policy
route-policy-name | tag tag | type type ]
```
- 配置引入路由的缺省度量值


```
[H3C-ospf-1]default { cost cost | tag tag | type type }
```
- 配置OSPF引入缺省路由


```
[H3C-ospf-1]default-route-advertise [ [ always | permit-
calculate-other ] | cost cost | route-policy route-policy-name |
type type ] * | summary cost cost ]
```

www.h3c.com

进入 OSPF 视图后，配置 OSPF 引入其它协议的路由。命令如下：

```
import-route protocol process-id [ cost cost | type type | tag tag | route-policy
route-policy-name ]
```

```
import-route protocol [ process-id | all-processes | allow-ibgp ] [ allow-direct |
cost cost | nssa-only | route-policy route-policy-name | tag tag | type type ]
```

其中的参数含义如下：

- **protocol**: 指定引入的路由协议，可以是 **bgp**、**direct**、**isis**、**ospf**、**rip** 或 **static**。
- **process-id**: 路由协议进程号，取值范围为 1~65535，缺省值为 1。只有当 **protocol** 是 **isis**、**ospf** 或 **rip** 时该参数可选。
- **all-processes**: 引入指定路由协议所有进程的路由，只有当 **protocol** 是 **rip**、**ospf** 或 **isis** 时可以指定该参数。
- **allow-ibgp**: 允许引入 IBGP 路由。只有当 **protocol** 是 **bgp** 时该参数可选。
- **allow-direct**: 在引入的路由中包含使能了该协议的接口网段路由。
- **cost cost**: 路由开销值，取值范围为 0~16777214，缺省值为 1。
- **nssa-only**: 设置 Type-7 LSA 的 P 比特位不置位，即在对端路由器上不能转为 Type-5 LSA。缺省时，Type-7 LSA 的 P 比特位被置位，即在对端路由器上可以转为 Type-5 LSA。

LSA（如果本地路由器是 ABR，则会检查骨干区域是否存在 FULL 状态的邻居，当 FULL 状态的邻居存在时，产生的 Type-7 LSA 中 P 比特位不置位）。

- **route-policy** *route-policy-name*: 配置只能引入符合指定路由策略的路由。
route-policy-name 为路由策略名称，为 1~63 个字符的字符串，区分大小写。
- **tag** *tag*: 外部 LSA 中的标记，取值范围为 0~4294967295，缺省值为 1。
- **type** *type*: 度量值类型，取值范围为 1~2，缺省值为 2。

引入路由的缺省开销值为 1，缺省类型为 2，缺省标记为 1。管理员可以用以下命令来调整这些参数的缺省值：

```
default { cost cost | tag tag | type type }
```

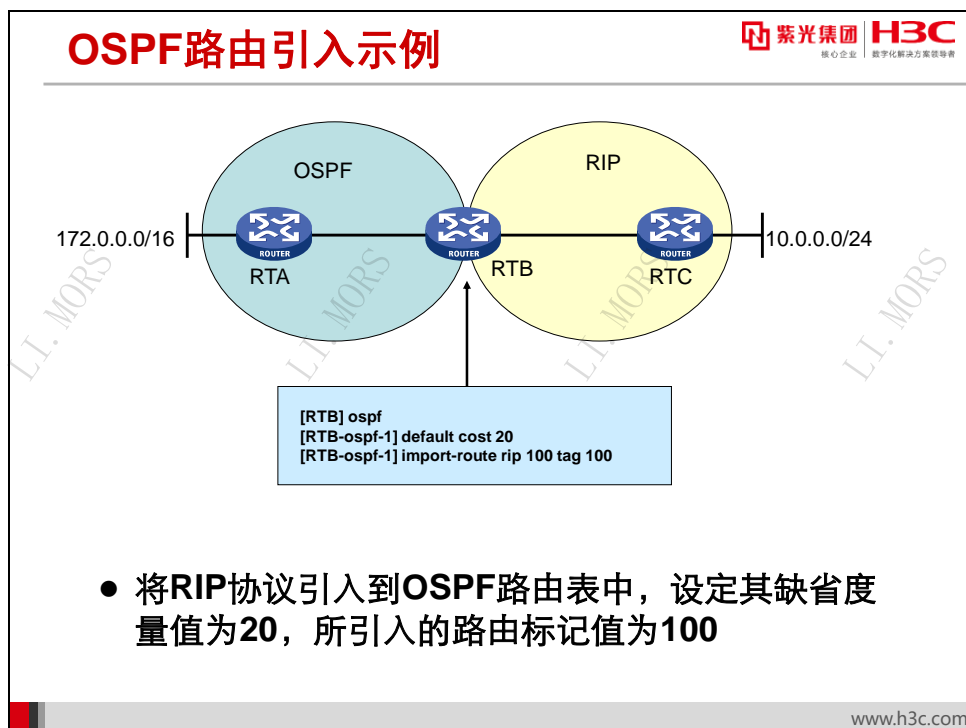
在 OSPF 中，当使用 import-route 命令引入路由时，不能引入外部路由的缺省路由。

如果要引入缺省路由，必须要使用 default-route-advertise 命令。其命令如下：

```
default-route-advertise [ [ [ always | permit-calculate-other ] | cost cost |  
route-policy route-policy-name | type type ] | summary cost cost ]
```

参数含义如下：

- **always**: 如果当前路由器的路由表中没有缺省路由，使用此参数可产生一个描述缺省路由的 Type-5 LSA 发布出去。
- **permit-calculate-other**: 当路由器产生并发布了一个描述缺省路由的 Type-5 LSA 时，指定此参数的路由器仍然会计算来自于其他路由器的缺省路由，未指定此参数的路由器不再计算来自其他路由器的缺省路由。
- **cost** *cost*: 该缺省路由的度量值，取值范围为 0~16777214，如果没有指定，缺省路由的度量值将取 **default cost** 命令配置的值。
- **route-policy** *route-policy-name*: 路由策略名，为 1~63 个字符的字符串，区分大小写。
- **type** *type*: 该 Type-5 LSA 的类型，取值范围为 1~2，如果没有指定，Type-5 LSA 的缺省类型将取 **default type** 命令配置的值。
- **summary**: 发布指定缺省路由的 Type-3 LSA。在选用该参数时，必须首先使能 VPN，否则路由不能发布。



在上图所示网络中，RTA 和 RTB 运行 OSPF 协议；RTB 和 RTC 运行 RIP 协议。RTA 连接到网络 172.0.0.0/16，RTC 连接到网络 10.0.0.0/24。在 RTB 上配置路由引入，将 RIP 路由引入到 OSPF 协议数据库中，并设定缺省度量值为 20。为了区分引入的路由，设定所引入的路由标记值为 100。

配置 RTB:

```

[RTB] ospf
[RTB-ospf-1] default cost 20
[RTB-ospf-1] import-route rip 100 tag 100
  
```

配置完成后，可以在 RTA 的路由表中查看到引入的路由 10.0.0.0/24，其标记值是 100。

15.4.3 配置 IS-IS 协议引入外部路由

IS-IS路由引入



- 配置IS-IS引入外部路由

```
[H3C-isis-1]import-route protocol [ process-id | all-processes | allow-ibgp ] [ allow-direct | cost cost | cost-type { external | internal } | [ level-1 | level-1-2 | level-2 ] | route-policy route-policy-name | tag tag ]
```

- 配置IS-IS对引入的路由进行过滤

```
[H3C-isis-1]filter-policy { acl-number | prefix-list prefix-list-name | route-policy route-policy-name } export [ protocol [ process-id ] ]
```

www.h3c.com

进入 IS-IS 视图后，配置 IS-IS 引入其它协议的路由。命令如下：

```
import-route protocol [ process-id | all-processes | allow-ibgp ] [ allow-direct | cost cost | cost-type { external | internal } | [ level-1 | level-1-2 | level-2 ] | route-policy route-policy-name | tag tag ]
```

其中的参数含义如下：

- **protocol**: 指定引入的路由协议，可以是 **bgp**、**direct**、**isis**、**ospf**、**rip** 或 **static**。
- **process-id**: 路由协议进程号，取值范围为 1~65535，缺省值为 1。只有当 **protocol** 是 **isis**、**ospf** 或 **rip** 时该参数可选。
- **all-processes**: 引入指定路由协议所有进程的路由，只有当 **protocol** 是 **rip**、**ospf** 或 **isis** 时可以指定该参数。
- **allow-ibgp**: 允许引入 IBGP 路由。只有当 **protocol** 是 **bgp** 时该参数可选。
- **allow-direct**: 在引入的路由中包含使能了该协议的接口网段路由。缺省情况下，在引入协议路由时不会包含使能了该协议的接口网段路由。
- **cost**: 引入的路由的路径开销，取值范围为 0~4261412864。
- **cost-type { external | internal }**: 表示路径开销类型：**internal** 表示内部路由；**external** 表示外部路由，配置路径开销类型为 **external** 后，通过 LSP 发布路由时路径开销会在配置的 **cost** 值的基础上加上 64，从而保证内部路由优于外部路由。缺省

情况下为 **external** 类型。只有当开销类型为 **narrow**、**narrow-compatible** 或者 **compatible** 时，该参数有效。

- **level-1**: 引入路由到 Level-1 的路由表中。
- **level-1-2**: 同时引入路由到 Level-1 和 Level-2 的路由表中。
- **level-2**: 引入路由到 Level-2 的路由表中。如果不指定引入的级别，默认为引入路由到 Level-2 路由表中。
- **route-policy route-policy-name**: 路由策略名称，只有满足指定路由策略匹配条件的路由才被引入。
- **tag tag**: 为引入路由配置 Tag 值，取值范围为 1~4294967295。

说明：

在 RIP、OSPF 协议中，也可以使用 **filter-policy** 来对引入的路由进行过滤。

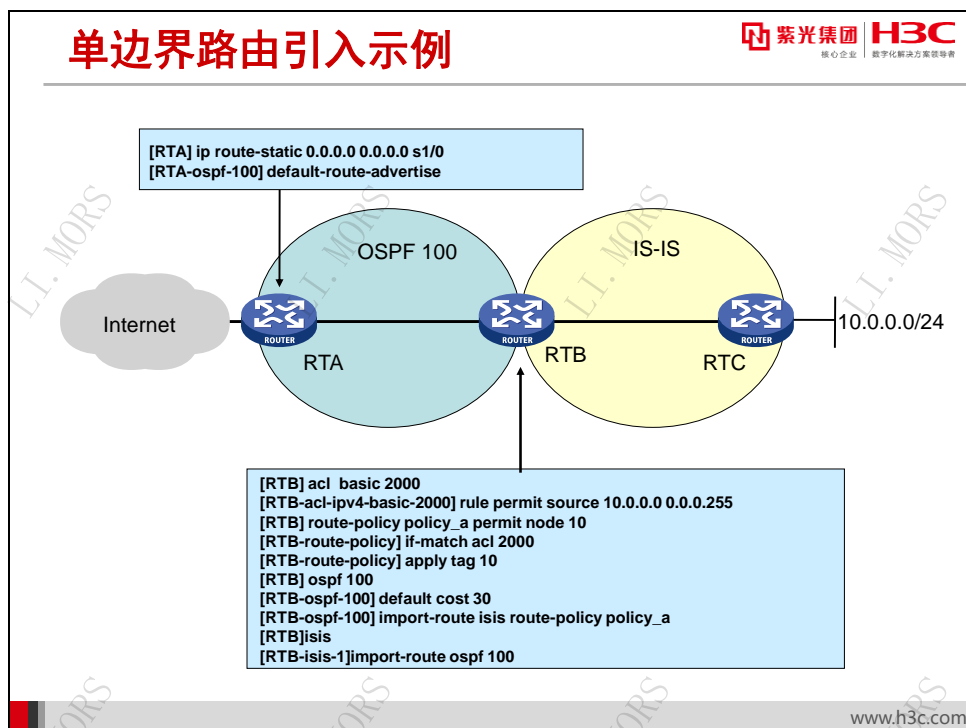
进入 IS-IS 视图后，配置 IS-IS 对引入的路由进行过滤。命令如下：

```
filter-policy { acl-number | prefix-list prefix-list-name | route-policy route-policy-name } export [ protocol [ process-id ] ]
```

其中的参数含义如下：

- **acl-number**: 指定访问控制列表序号，取值范围为 2000~3999，基于 ACL 对引入的路由信息进行过滤。
- **prefix-list prefix-list-name**: 指定 IPv4 地址前缀列表名，基于目的地址对引入的路由信息进行过滤。
- **route-policy route-policy-name**: 指定路由策略名，基于路由策略对引入的路由信息进行过滤。
- **protocol**: 路由协议名称，指定过滤从哪种路由协议引入的路由信息。
- **process-id**: 路由协议进程号，取值范围为 1~65535。

15.4.4 路由引入示例



在上图所示网络中，RTA 和 RTB 运行 OSPF 协议，OSPF 进程号为 100；RTB 和 RTC 运行 IS-IS 协议。RTA 连接到 Internet，是自治系统边界路由器。在 RTA 上配置缺省静态路由，下一跳指向连接到 Internet 的接口 S1/0，并且配置将缺省路由引入到 OSPF 中。

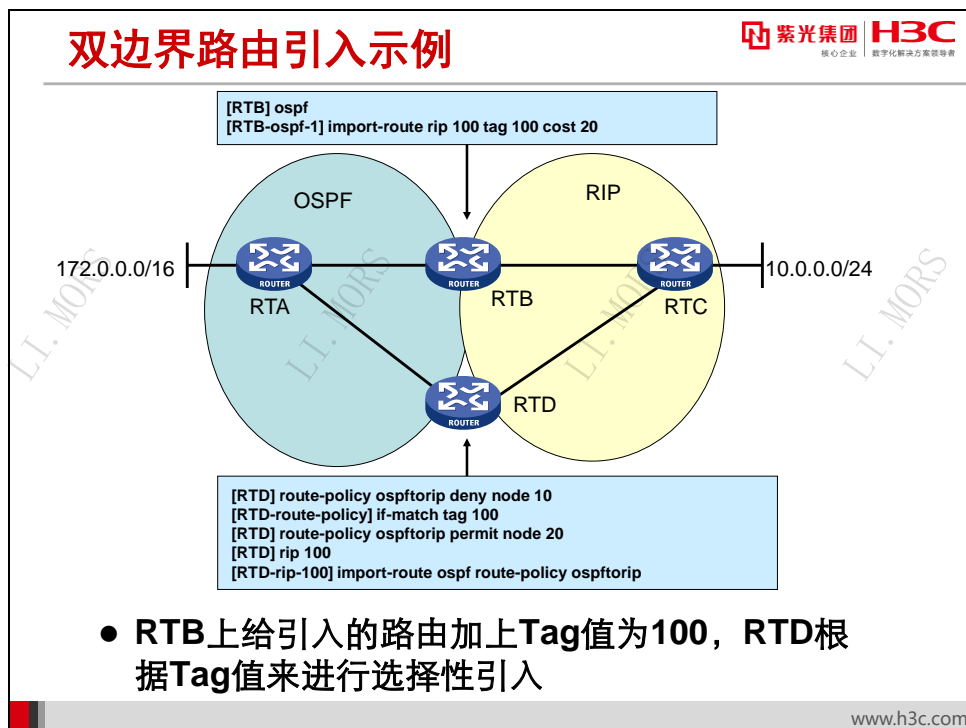
在 RTB 上配置双向路由引入。将 IS-IS 路由引入到 OSPF 协议数据库中，并设定缺省度量值是 30；并且设定引入的路由 10.0.0.0/24 的标记值为 10。同时，将 OSPF 进程 100 的路由引入到 IS-IS 协议数据库中。

RTA 的配置：

```
[RTA]ip route-static 0.0.0.0 0.0.0.0 s1/0
[RTA-ospf-100]default-route-advertise
```

RTB 的配置：

```
[RTB]acl basic 2000
[RTB-acl-ipv4-basic-2000]rule permit source 10.0.0.0 0.0.0.255
[RTB]route-policy policy_a permit node 10
[RTB-route-policy]if-match ip address acl 2000
[RTB-route-policy]apply tag 10
[RTB]ospf 100
[RTB-ospf-100]default cost 30
[RTB-ospf-100]import-route isis route-policy policy_a
[RTB]isis
[RTB-isis-1]import-route ospf 100
```



双边界情况下，可以使用 Tag 标记来防止环路产生。

在上图所示网络中，RTA 和 RTB 运行 OSPF 协议；RTB 和 RTC 运行 RIP 协议。RTA 连接到网络 172.0.0.0/16，RTC 连接到网络 10.0.0.0/24。在 RTB 上配置路由引入，将 RIP 路由引入到 OSPF 协议数据库中，并设定缺省度量值是 20，路由标记值为 100。

RTD 上配置将 OSPF 路由引入到 RIP 中，并使用 route-policy 来设定拒绝引入标记值为 100 的路由，实际上也就是不会把从本 RIP 域始发的路由再引入回 RIP 域中。

RTB 的配置：

```
[RTB]ospf
[RTB-ospf-1]import-route rip 100 tag 100 cost 20
```

RTD 的配置：

```
[RTD]route-policy ospftorip deny node 10
[RTD-route-policy]if-match tag 100
[RTD]route-policy ospftorip permit node 20
[RTD]rip 100
[RTD-rip-100]import-route ospf route-policy ospftorip
```

15.5 本章总结

本章总结

- 路由引入可解决多协议网络中的路由学习问题
- 引入的路由属性有变化，需进行合理规划
- 单向引入可避免环路
- 合理规划以在多边界引入时避免环路及次优路由

15.6 习题和解答

15.6.1 习题

- 关于路由引入，以下哪些说法是正确的？（ ）
 - 路由引入是指把路由信息从一种路由协议导入到另外一种协议
 - 路由引入也指在同种协议的不同进程之间导入路由信息
 - 可以把静态路由引入到 OSPF 路由协议中
 - 可以把 OSPF 路由信息引入到静态路由中
- 当把路由信息引入到 OSPF 协议中时，缺省度量值是（ ）
 - 0
 - 1
 - 10
 - 需要设定
- 在 RIP 中引入路由时设定缺省度量值为 3 的命令是（ ）
 - [RTA] default 3
 - [RTA] default cost 3
 - [RTA-rip-100] default 3
 - [RTA-rip-100] default cost 3
- 关于路由引入部署，以下哪些策略是正确的？（ ）
 - 只是在必要的时候才运行多路由协议
 - 需要知道互相的具体路由时，可以部署双向路由引入
 - 实施路由引入时，通常把路由从边缘区域引入到核心区域
 - 需要增加路由域间的可靠性时，采用多边界引入
- 当把路由信息引入到 IS-IS 协议中时，缺省情况下所引入路由的属性是？（ ）
 - Level-1 路由，外部开销值为 0
 - Level-1 路由，外部开销值为 1
 - Level-2 路由，外部开销值为 0
 - Level-2 路由，外部开销值为 1

15.6.2 习题答案

1. ABC 2. B 3. D 4. ABCD 5. C

第16章 PBR

通常，路由器仅根据 IP 报文中的目的地址查看路由表进行转发，报文中的其它信息不作为报文转发的依据。但在实际应用中，有时需要具有相同目的地址的数据流被分布到不同路径上。PBR(policy-based-route, 基于策略的路由)是一种依据用户制定的策略进行路由选择的机制。通过合理应用 PBR，路由器可以根据到达报文的源地址、地址长度等信息灵活地进行路由选择。本章介绍了 PBR 的适用场景、相关配置命令及维护。

16.1 本章目标

课程目标

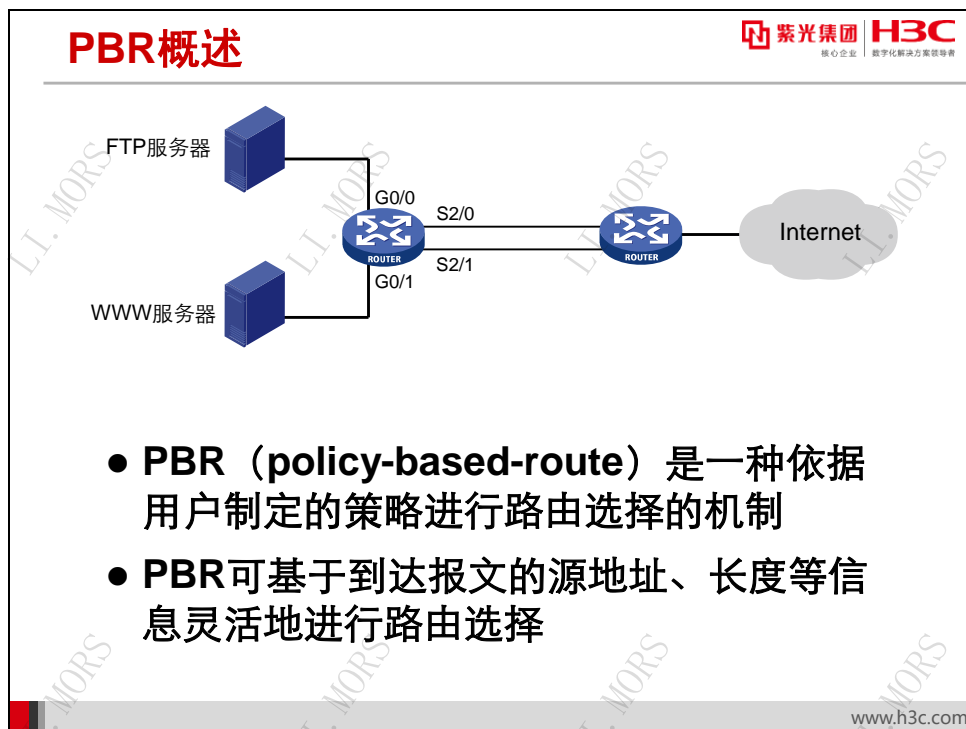
● 学习完本课程，您应该能够：

- 掌握PBR的作用
- 掌握PBR的配置
- 掌握PBR的应用



www.h3c.com

16.2 PBR概述

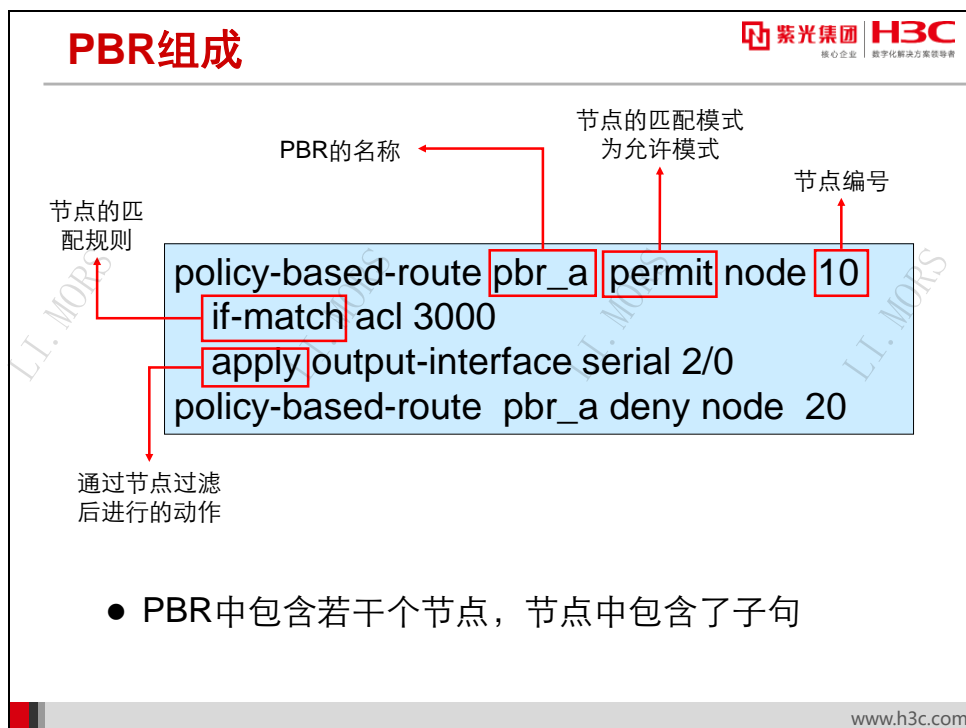


传统路由是根据报文的目的地址查找路由表进行转发操作,这样在目的地址相同的情况下,无法进行报文的选路控制。

上图是一个典型例子。公司内网络有 2 台服务器,分别向公网提供 FTP 和 WWW 服务。边界路由器有 2 条链路连接到 ISP 路由器,为了合理利用带宽,管理员想要实现不同服务数据流经由不同链路转发。此时,由于从内网到外网的数据流具有相同的目的地地址,传统路由无法对这两种数据流进行区分。

PBR (policy-based-route, 基于策略的路由,简称策略路由)是一种依据用户制定的策略进行路由选择的机制。与单纯依照 IP 报文的目的地址查找路由表进行转发不同,策略路由基于到达报文的源地址、长度等信息灵活地进行路由选择。

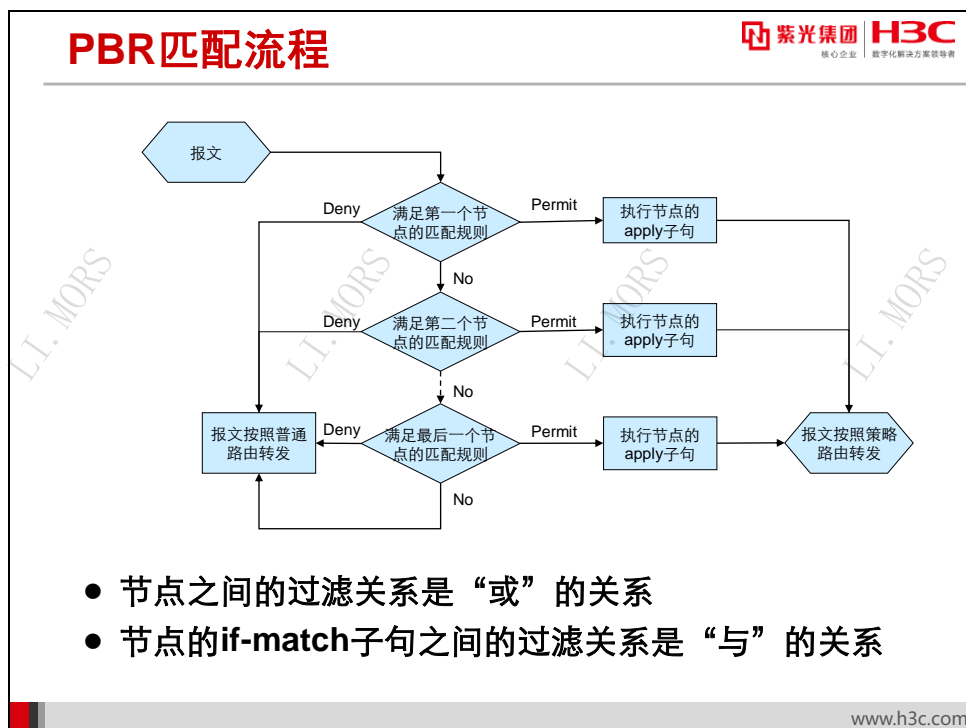
为实现策略路由,首先要定义将要实施策略路由的报文特征,即定义一组匹配规则。可以以报文中的不同特征(如源地址、长度等)作为匹配依据进行设置,然后再将策略路由应用于接口,使路由器根据预先制定的策略对报文进行转发。



一个 PBR 可以由多个带有编号的节点（node）构成，每个节点是匹配检查的一个单元，在匹配过程中，系统按节点编号升序依次检查各个节点。

每个节点可以由一组 `if-match` 和 `apply` 子句组成。`if-match` 子句定义节点的匹配规则，匹配对象是报文的特征，如源 IP 地址、报文承载的协议、端口号，报文的长度等。`apply` 子句定义通过该节点过滤后进行的动作。

节点的匹配模式有允许模式（`permit`）和拒绝模式（`deny`）两种。允许模式表示当 IP 报文通过该节点的过滤后，将执行该节点的 `apply` 子句；而拒绝模式表示 `apply` 子句不会被执行。



一个 PBR 的不同节点间是“或”的关系，即只要通过了任一节点的过滤，就意味着通过该 PBR 的过滤，不再对其他节点进行匹配。

每个节点的 if-match 子句之间的过滤关系是“与”的关系，即报文必须满足该节点的所有 if-match 子句才能执行该节点的 apply 子句。


如果节点的匹配模式为允许模式，则当 IP 报文满足该节点的所有 if-match 子句时，将执行该节点的 apply 子句，不进入下一个节点的匹配；如果 IP 报文不满足该节点的 if-match 子句，报文将会使用该条策略的下一个节点进行匹配。

如果节点的匹配模式为拒绝模式，则当 IP 报文满足该节点的所有 if-match 子句时，将被拒绝通过该节点，不进入下一个节点的匹配；如果报文不满足该节点的 if-match 子句，将进入下一个节点继续匹配。

通过一个节点所定义的策略的报文将不再参与其他节点策略的过滤和处理。如果报文不能通过一个 PBR 所有节点的过滤，则认为没有通过该 PBR，该报文按正常转发流程处理。

16.3 PBR配置与查看

PBR配置



- 创建PBR

```
[H3C] policy-based-route policy-name [ deny | permit ] node node-number
```

- 配置if-match子句

```
[H3C-pbr-aaa] if-match { 匹配规则 }
```

- 配置apply子句

```
[H3C-pbr-aaa] apply { 动作 }
```

- 使能接口策略路由

```
[H3C-GigabitEthernet1/0] ip policy-based-route policy-name
```

www.h3c.com

在配置 PBR 之前，需要规划 PBR 名称、节点编号，节点中 if-match 子句的匹配规则，通过节点过滤后要执行的动作等。

配置 PBR 的步骤如下：

第1步：在系统视图下创建 PBR，并定义名称、节点编号、匹配模式等参数。命令如下：

```
policy-based-route policy-name { permit | deny } node node-number
```

第2步：在 PBR 视图下使用 if-match 子句来设定路由信息的匹配规则。命令如下：

```
if-match { 匹配规则 }
```

if-match 子句后是报文匹配条件的设定，可选的参数包括 ACL 和 IP 报文长度。

第3步：在 PBR 视图下使用 apply 子句来指定报文通过过滤后所执行的动作。命令如下：

```
apply { 动作 }
```

apply 子句后可选的参数包括 ip-precedence、output-interface、next-hop 等，可以分别对通过节点过滤的报文的优先级、出接口、下一跳地址等进行设定。

第4步：在接口视图下使能接口策略路由。命令如下：

```
ip policy-based-route policy-name
```

策略路由可分为系统策略路由和接口策略路由：

- 系统策略路由对本地产生的报文进行策略路由，它只对本地产生的报文起作用，对转发的报文不起作用；
- 接口策略路由作用于到达该接口的报文，它只对转发的报文起作用，对本地产生的报文（比如本地的 ping 报文）不起作用。

一般情况下，使用接口策略路由。如果要使用系统策略路由，则需要在系统视图下使能，相应的配置命令为：

ip local policy-based-route *policy-name*

if-match子句的配置

紫光集团
核心企业 数字化解决方案领导者

操作	if-match配置命令（PBR视图下）
设置ACL匹配条件。匹配条件可以是报文源地址、业务流的端口号等	if-match acl <i>acl-number</i>
设置IP报文长度匹配条件	if-match packet-length <i>min-len max-len</i>

www.h3c.com

配置 if-match 子句时，可选的参数包括 ACL 和 IP 报文长度。

在 PBR 视图下进行 ACL 匹配条件的配置，其命令如下：

if-match acl *acl-number*

同样，在 PBR 视图下进行 IP 报文长度匹配条件的配置，其命令如下：

if-match packet-length *min-len max-len*

参数含义如下：

- *min-len*: 最短 IP 报文长度，取值范围为 0～65535，单位为字节。
- *max-len*: 最长 IP 报文长度，取值范围为 1～65535，单位为字节。*max-len* 应该不小于 *min-len*。

apply子句的配置	
操作	Apply子句配置命令（PBR视图下）
设置报文的优先级	apply precedence { <i>type</i> <i>value</i> }
设置报文的发送接口	apply output-interface <i>interface-type</i> <i>interface-number</i>
设置报文的下一跳	apply next-hop <i>ip-address</i>
设置报文缺省发送接口	apply default-output-interface <i>interface-type</i> <i>interface-number</i>
设置报文缺省下一跳	apply default-next-hop <i>ip-address</i>

apply 子句定义了通过节点过滤后对报文所执行的动作，包括对报文的优先级、出接口、下一跳地址等进行设定。

具体配置如下表所示：

表16-1 apply 子句配置命令表

操作	命令
设置报文的优先级	apply ip-precedence { <i>type</i> <i>value</i> }
设置报文的发送接口	apply output-interface <i>interface-type</i> <i>interface-number</i>
设置报文的下一跳	apply next-hop <i>ip-address</i>
设置报文缺省发送接口	apply default-output-interface <i>interface-type</i> <i>interface-number</i>
设置报文缺省下一跳	apply default-next-hop <i>ip-address</i>

在配置出接口和下一跳时，可以同时配置两个发送接口或两个下一跳，这两个发送接口或下一跳同时有效，可以起到负载分担的作用。

注意：

仅当报文目的 IP 地址在路由表中没有查到相应的路由，路由器才会使用 PBR 所配置的缺省下一跳或者出接口（命令为 **apply default-output-interface** 和 **apply default-next-hop**）进行 IP 报文转发。

PBR显示与维护



- 显示已经配置的PBR

```
[H3C]display ip policy-based-route [ policy policy-name ]
```

- 显示所有已经应用的PBR信息

```
[H3C]display ip policy-based-route setup
```

- 显示接口下发PBR的配置信息和统计信息

```
[H3C]display ip policy-based-route interface interface-type interface-number
```

- 显示本地PBR的配置信息和统计信息

```
[H3C]display ip policy-based-route local
```

www.h3c.com

在完成 PBR 的配置后，在任意视图下执行 `display policy-based-route` 命令可以显示已经配置的 PBR。相关命令及输出如下所示：

```
[Router] display ip policy-based-route
Policy name: abc
node 1 permit:
  apply output-interface Serial2/0
```

以上输出表明，PBR 名称为 `abc`，包含了 1 个编号为 1 的节点，节点的匹配模式是允许模式。已经匹配的报文指定发送接口为 `Serial2/0`。

在任意视图下执行 `display ip policy-based-route setup` 命令可以显示所有已经应用的 PBR 信息，相关命令及输出如下所示：

```
[Router] display ip policy-based-route setup
Policy Name      Interface Name
1                GigabitEthernet0/0
abc              local
```

以上输出表明，目前共使能了 2 个 PBR，名称为 1 的 PBR 在 `GigabitEthernet0/0` 接口被应用，名称为 `abc` 的 PBR 在本地被应用。

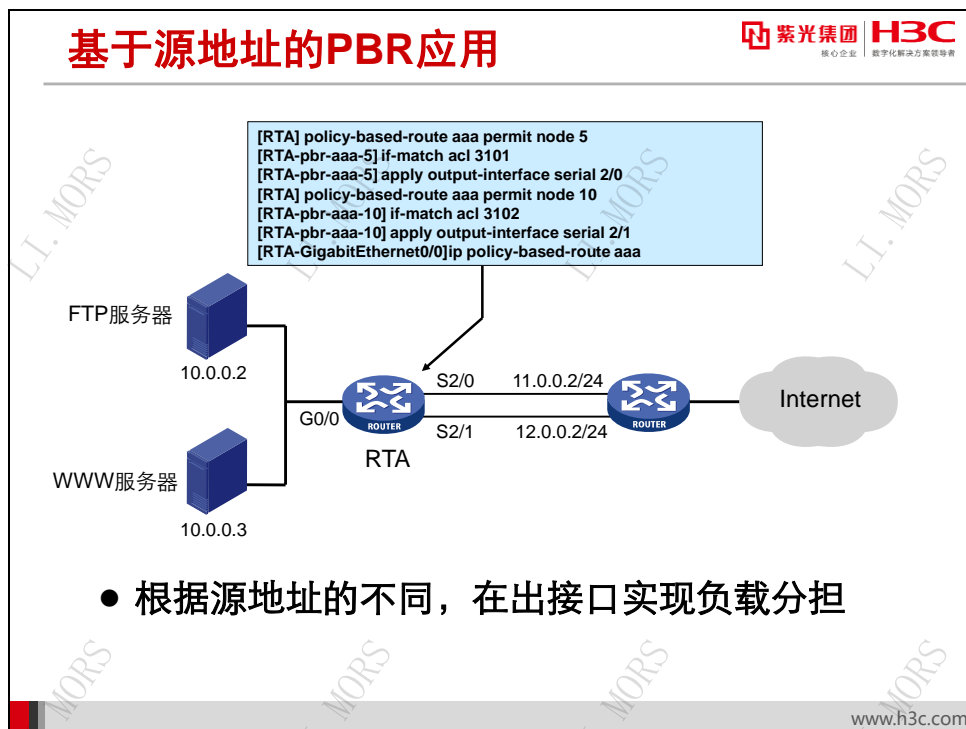
在任意视图下执行 `display ip policy-based-route interface/local` 命令可以显示已经使能的 PBR 的统计信息，相关命令及输出如下所示：

```
[Router] display ip policy-based-route interface GigabitEthernet 0/0
Policy based routing information for interface GigabitEthernet0/0:
Policy name: 1
node 1 permit:
  if-match acl 2000
  if-match packet-length 4 10
  apply next-hop 1.1.1.1
```

```
Matched: 0
node 10 permit:
Matched: 0
Total matched: 0
```

以上输出表明，在接口 **GigabitEthernet 0/0** 上应用了名称为 **1** 的 **PBR**，节点编号为 **1**，匹配模式为允许模式，已经匹配的报文发送到指定的下一跳 **1.1.1.1**。节点 **1** 匹配成功的次数为 **0**；策略 **1** 所有节点匹配成功的次数也均为 **0**。

16.4 PBR的应用



PBR 被广泛应用在源地址路由、负载分担等场合中。

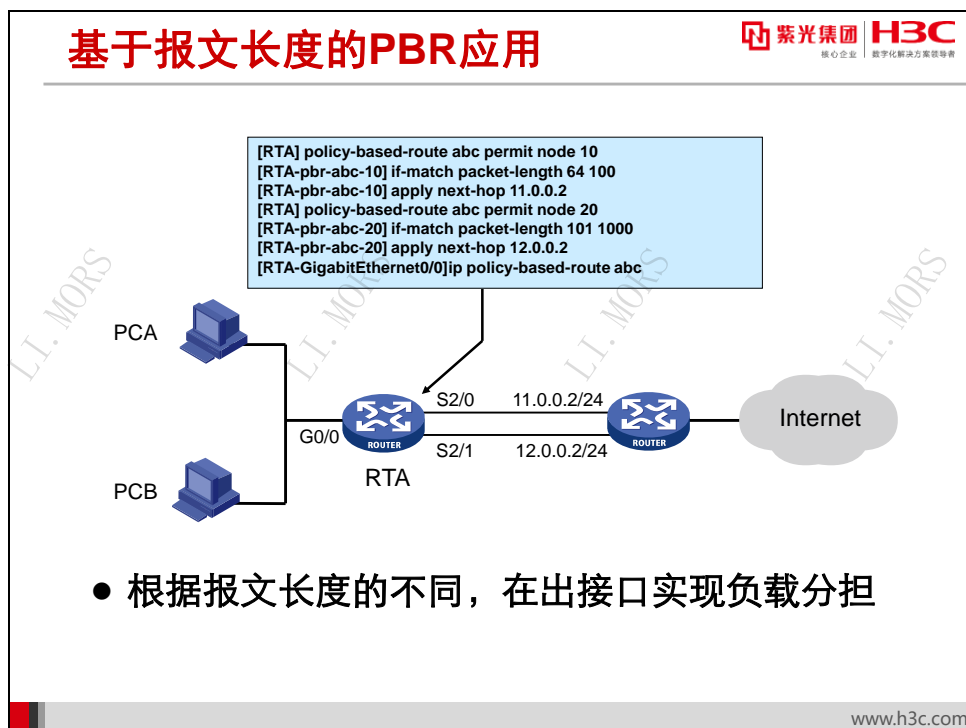
在上图所示网络中,RTA 通过 2 个接口连接到 Internet。RTA 的 G0/0 连接有 FTP 和 WWW 服务器,其 IP 地址分别为 10.0.0.2 和 10.0.0.3。在 RTA 上配置 PBR,使 FTP 服务器到 Internet 的数据流经由接口 S2/0 发送,而 WWW 服务器到 Internet 的数据流经由接口 S2/1 发送。

RTA 的配置如下:

```

[RTA]acl advanced 3101
[RTA-acl-ipv4-adv-3101]rule permit ip source 10.0.0.2 0
[RTA]acl advanced 3102
[RTA-acl-ipv4-adv-3102]rule permit ip source 10.0.0.3 0
[RTA]policy-based-route aaa permit node 5
[RTA-pbr-aaa-5]if-match acl 3101
[RTA-pbr-aaa-5]apply output-interface serial 2/0
[RTA]policy-based-route aaa permit node 10
[RTA-pbr-aaa-10]if-match acl 3102
[RTA-pbr-aaa-10]apply output-interface serial 2/1
[RTA]interface GigabitEthernet 1/0
[RTA-GigabitEthernet0/0]ip policy-based-route aaa
        
```

配置完成后,可以在 RTA 上执行 `display ip policy-based-route` 命令来观察 PBR 的运行效果。



在上图所示网络中，RTA 作为局域网出口路由器，通过 2 个接口连接到 Internet，其下一跳分别是 11.0.0.2/24 和 12.0.0.2/24。为了实现负载分担，在 RTA 上设置 PBR，将大小为 64~100 字节的报文设置 11.0.0.2/24 作为下一跳 IP 地址；而将大小为 101~1000 字节的报文设置 12.0.0.2/24 作为下一跳 IP 地址。其它长度的报文都按照查找路由表的方式转发。

RTA 的配置如下：

```

[RTA]policy-based-route abc permit node 10
[RTA-pbr-abc-10]if-match packet-length 64 100
[RTA-pbr-abc-10]apply ip-address next-hop 11.0.0.2
[RTA]policy-based-route abc permit node 20
[RTA-pbr-abc-20]if-match packet-length 101 1000
[RTA-pbr-abc-20]apply ip-address next-hop 12.0.0.2
[RTA]interface GigabitEthernet 0/0
[RTA-GigabitEthernet0/0]ip policy-based-route abc
  
```

不同数据流的报文大小会有所不同，一般管理数据流如 Telnet、ICMP、SNMP 的报文较小，在 100 字节以下；而业务数据流如 FTP 的报文较大。通过设定不同大小报文通过不同接口转发，可以使不同流经由不同接口转发，做到出接口的负载分担。

16.5 本章总结

本章总结

- PBR由若干个节点组成，节点中包含了if-match子句和apply子句
- 节点之间的过滤关系是“或”的关系
- 可使用PBR来实现源路由选择
- 使用PBR来实现根据业务进行链路间的负载分担

16.6 习题和解答

16.6.1 习题

1. 以下哪些是 PBR 的优点？（ ）
A. 可实现基于源地址的路由 B. 可实现基于目的地址的路由
C. 可实现 QoS D. 可实现负载分担
2. 在 PBR 配置中，下列哪些匹配规则可以由 `if-match` 子句来设定？（ ）
A. 报文优先级 B. 出接口 C. 报文长度 D. 报文源地址 E. 报文下一跳
3. 在 PBR 配置中，下列哪些动作可以由 `apply` 子句来执行？（ ）
A. 报文优先级 B. 出接口 C. 报文长度 D. 报文源地址 E. 报文下一跳
4. 定义了名为 `aaa` 的 PBR 后，应该使用下列哪一条命令在接口上使能之？（ ）
A. `[RTA] ip policy-based-route aaa`
B. `[RTA] ip policy-based-route interface GigabitEthernet0/0 aaa`
C. `[RTA-GigabitEthernet0/0] ip policy-based-route aaa`
D. `[RTA-GigabitEthernet0/0] ip policy-based-route interface GigabitEthernet0/0 aaa`
5. 使用下列哪一条命令来查看已经应用的 PBR 信息？（ ）
A. `display policy-based-route` B. `display ip policy-based-route`
C. `display ip policy-based-route setup`
D. `display policy-based-route statistic`

16.6.2 习题答案

1. ABD 2. CD 3. ABE 4. C 5. C