G-LAB

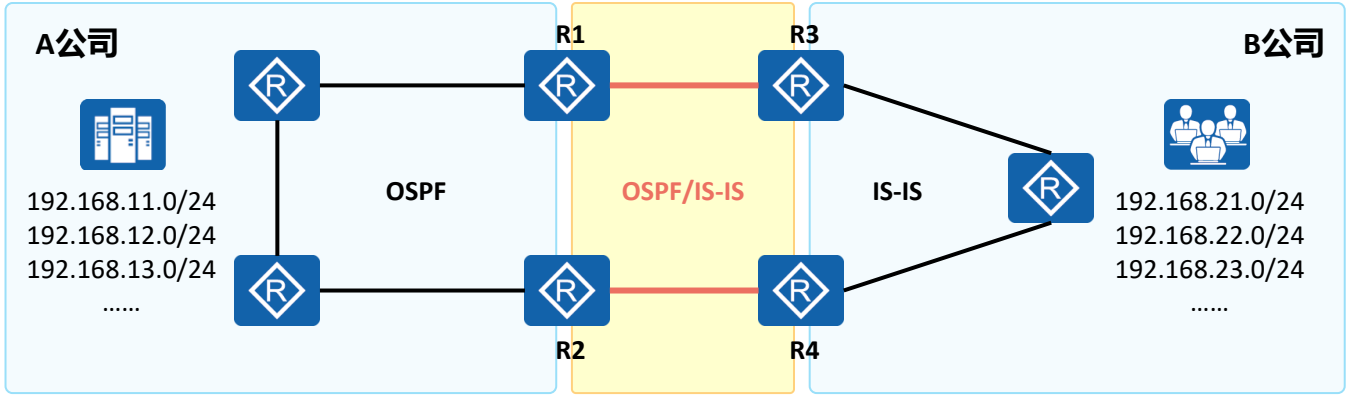
G-CNP v2.0课程

讲师：沈老师

前言

- 路由器获取路由的方式有三种，分别是动态路由（例如OSPF）、静态路由、直连路由。一个网络中可能会同时存在这三种方式，那么采用不同方式获取路由的路由器之间如何实现路由可达？
- 在复杂的数据通信网络中，根据实际组网需求，往往需要实施一些路由策略对路由信息进行过滤、属性设置等操作，通过对路由的控制，可以影响数据流量转发。
- 路由策略并非单一的技术或者协议，而是一个技术专题或方法论，里面包含了多种工具及方法。
- 本章主要介绍路由引入，网络中常用的路由选择工具以及路由策略的原理与配置。

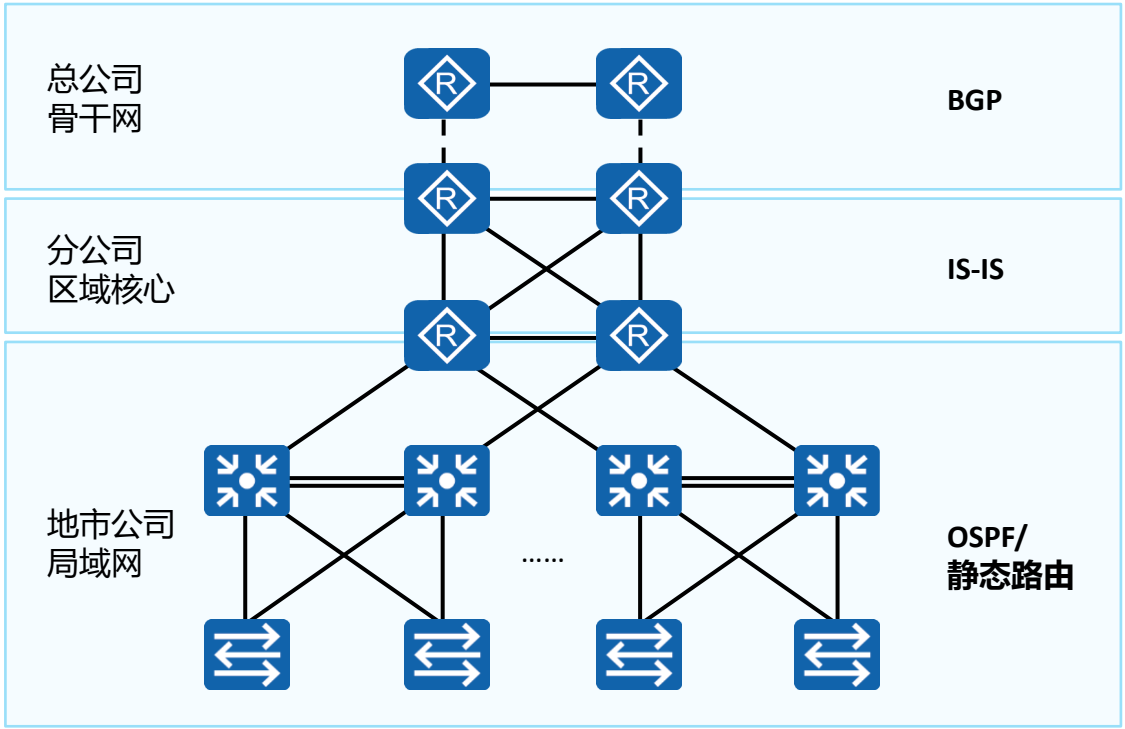
IP路由高级应用场景分析 (1)



场景描述：

- 假设A公司和B公司各有自己的网络，这两个网络被独立管理及运维，A公司网络使用的路由协议为OSPF，B公司网络使用的路由协议为IS-IS。
- 现在两家公司合并成一家公司，导致原有的两张网络不得不进行整合，为了使合并后的新公司业务流量能够正常在整合后的网络上交互，最重要的就是实现路由互通。

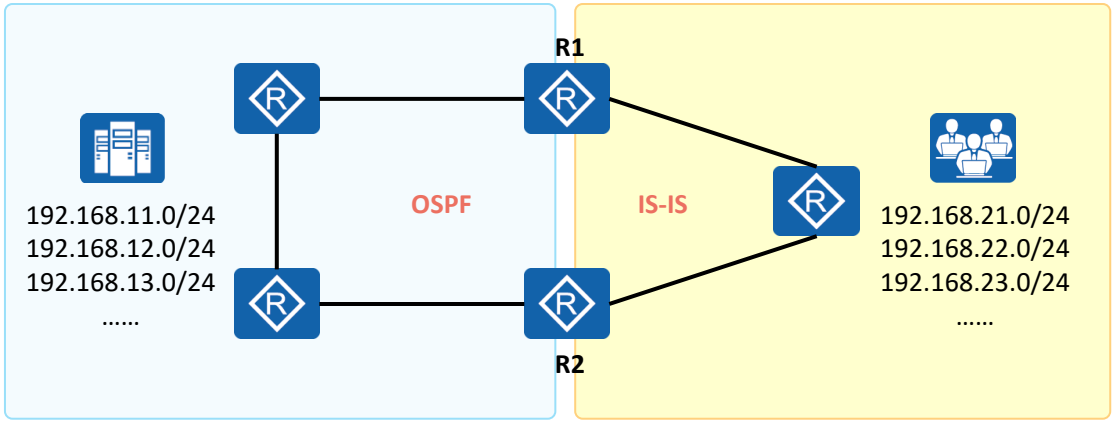
IP路由高级应用场景分析 (2)



场景描述：

- 在大型企业网络中，网络规模十分庞大，选用单一的路由协议无法满足网络的需求，因此多种路由协议共存的情况十分常见。
- 或者出于业务逻辑或行政管理的考虑，会在不同的网络结构中设计和部署不同的路由协议，使路由的层次结构更加清晰可控。
- 在这样的网络环境下，也需要实现全网路由互通。

路由引入的基本概念



两个路由协议之间路由信息彼此隔离

若要实现路由互通，可以通过以下方法：

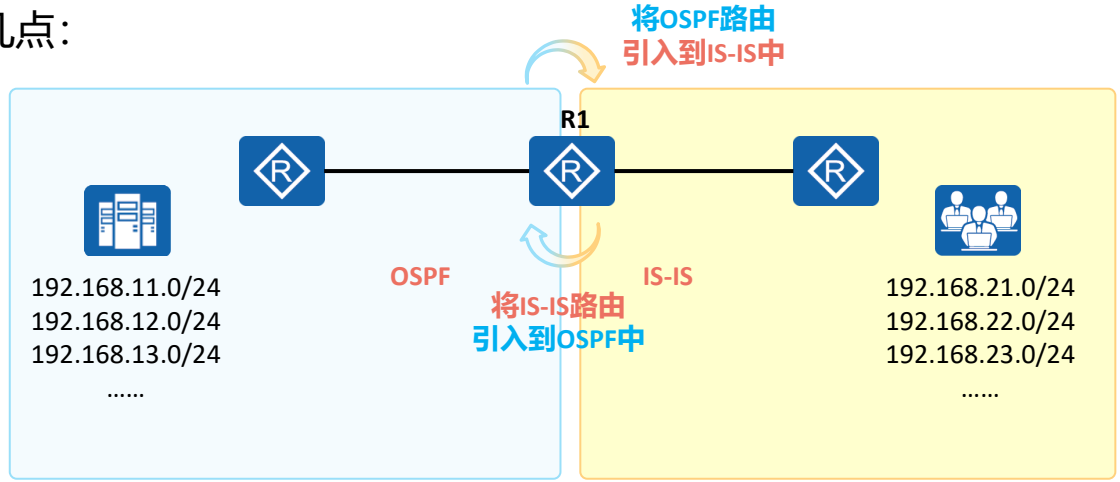
1. 重新规划及整改全网路由协议，该方案部署复杂。
2. 在OSPF和IS-IS路由域的边界设备上进行操作，使得路由信息在两个动态路由协议之间传递。该方案不需要改变原有拓扑架构，部署较为简单，但可能有环路风险。

路由引入指的是将路由信息从一种路由协议发布到另一种路由协议的操作。

- 通过路由引入，可以实现路由信息在不同路由协议间传递。
- 执行路由引入时，还可以部署路由控制，从而实现对业务流量的灵活把控。

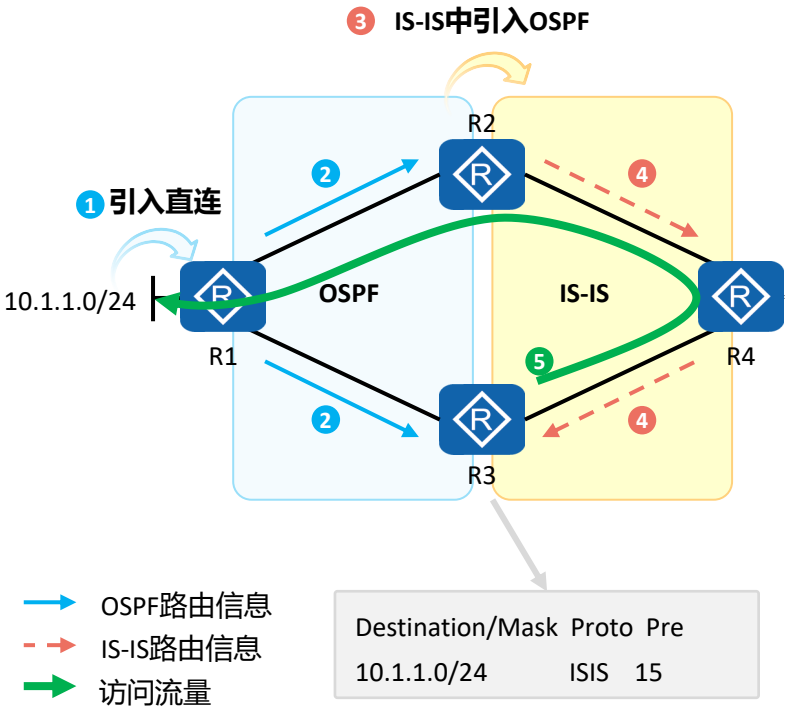
路由引入的方向性

- 路由引入是具有方向性的，将路由信息从路由协议A引入到路由协议B（A-to-B），则路由协议B可获知A中的路由信息，但是此时，A还并不知晓B路由协议中的路由信息，除非配置B-to-A的路由引入。
- 路由引入时需要注意以下几点：
  - 路由优先级
  - 路由回灌
  - 路由度量值



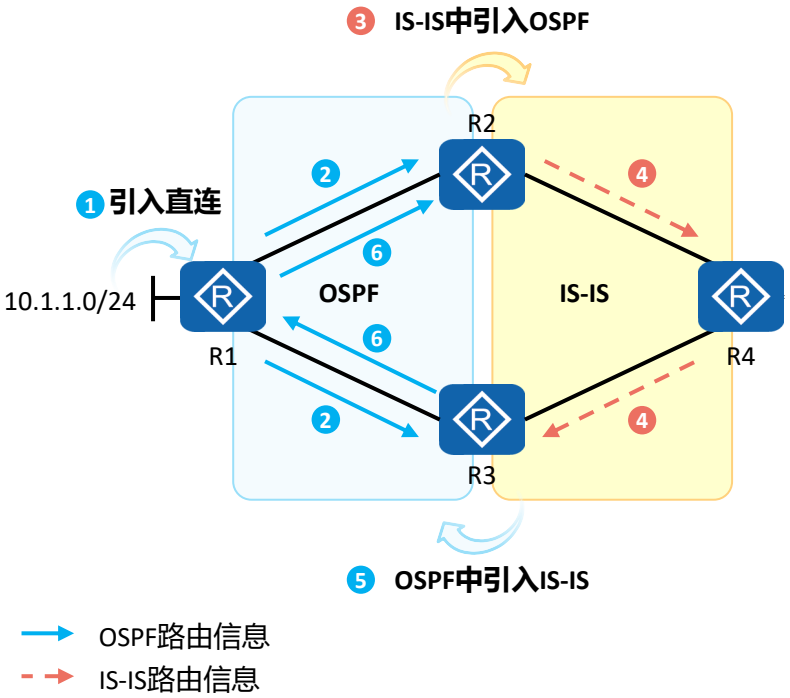
思考：如果只将OSPF路由引入到IS-IS中是否可以实现两个网络相互通信?

路由引入：路由优先级



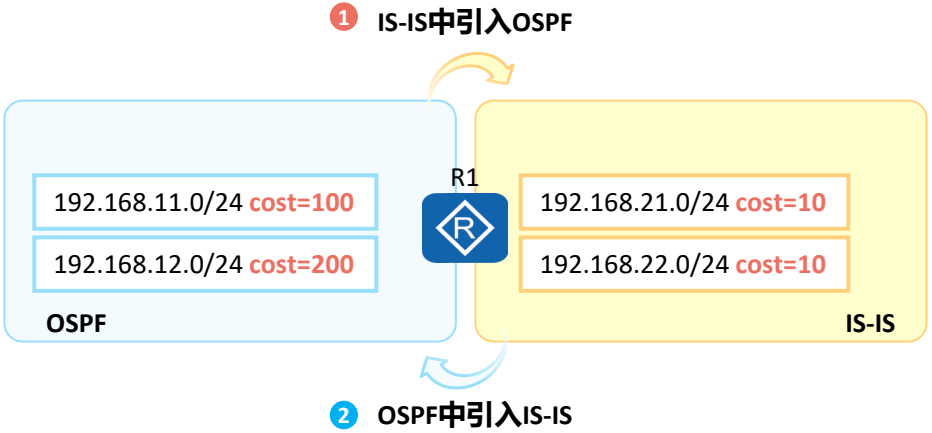
- 场景描述：
  1. R1将直连路由10.1.1.0/24引入到OSPF中。
  2. R3通过OSPF学习到10.1.1.0/24网段路由（OSPF外部路由，路由优先级为150）。
  3. R2在IS-IS进程中引入OSPF路由。
  4. R3也会通过IS-IS学习到10.1.1.0/24网段路由（路由优先级为15）。
  5. 对R3而言，IS-IS路由优于OSPF外部路由，因此优选来自R4的IS-IS路由。
- 后续R3访问10.1.1.0/24网段的路径为：R3->R4->R2->R1，这是次优路径。

路由引入：路由回灌



- 场景描述：
  1. R1将直连路由10.1.1.0/24引入到OSPF中。
  2. 10.1.1.0/24网段路由全OSPF域内通告。
  3. R2在IS-IS进程中引入OSPF路由。
  4. 10.1.1.0/24网段路由全IS-IS域内通告。
  5. R3在OSPF进程中引入IS-IS路由。
  6. 10.1.1.0/24网段路由再次被通告进OSPF域内，形成路由回灌。

路由引入：路由度量值

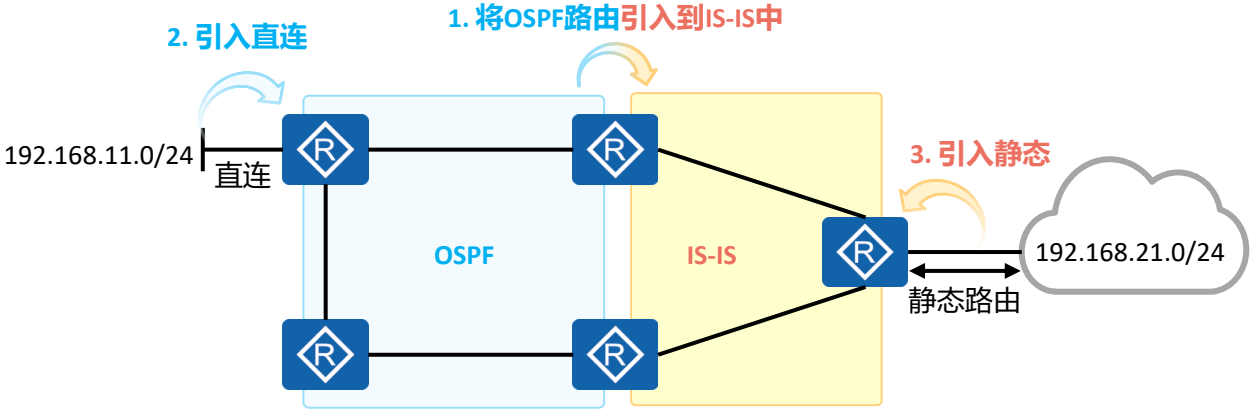


- 场景描述：
  1. 在IS-IS中引入OSPF路由。
  2. 在OSPF中引入IS-IS路由。
- 不同的路由协议对路由度量值的定义不同，那么在路由协议之间进行路由引入时，被引入的路由的度量值该如何定义？定义成多少？

路由引入场景

路由引入主要涉及以下几种场景：

1. 动态路由协议之间的路由引入
2. 引入直连路由到动态路由协议
3. 引入静态路由到动态路由协议



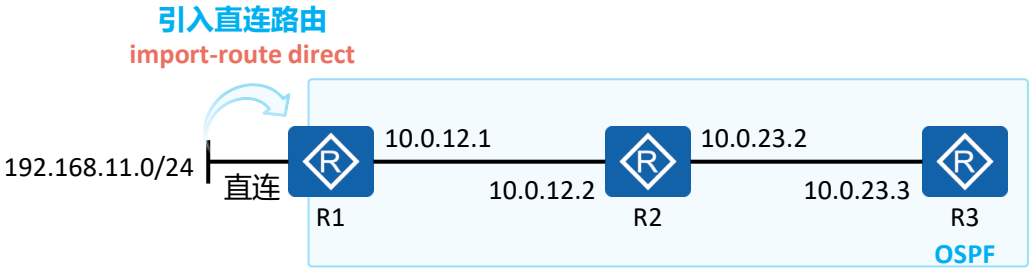
路由引入的基础配置命令

1. 配置OSPF引入外部路由

[Huawei-ospf-100] **import-route** { **bgp** | **direct** | **static** | **isis** [*process-id-isis*] | **ospf** [*process-id-ospf*]}

在OSPF视图下，引入BGP路由/直连路由/静态路由/IS-IS路由/OSPF其他进程路由。

案例1：引入直连路由到OSPF



R1的路由表

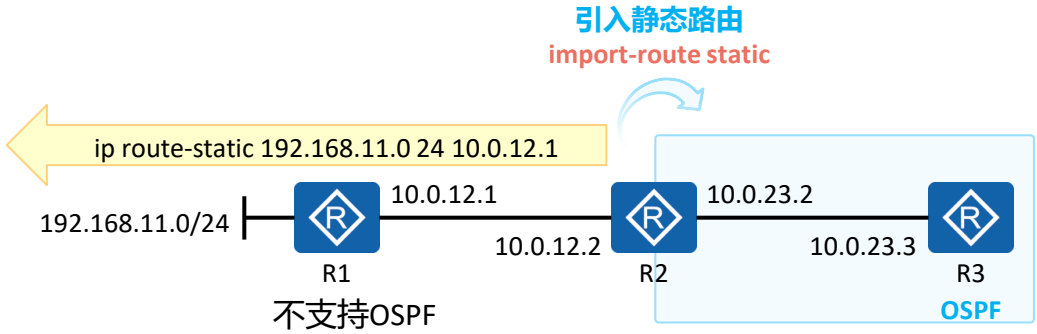
目的网络/掩码	协议	下一跳
192.168.11.0/24	Direct	192.168.11.1
10.0.12.0/24	Direct	10.0.12.1
10.0.23.0/24	OSPF	10.0.12.2

R3的路由表

目的网络/掩码	协议	下一跳
192.168.11.0/24	O_ASE	10.0.23.2
10.0.12.0/24	OSPF	10.0.23.2
10.0.23.0/24	Direct	10.0.23.3

- 可以通过使用**import-route direct**命令，将路由表中所有直连路由引入到动态路由协议。
- 引入后的路由会作为OSPF外部路由，在整个OSPF网络内通告。

案例2：引入静态路由到OSPF



R2的路由表

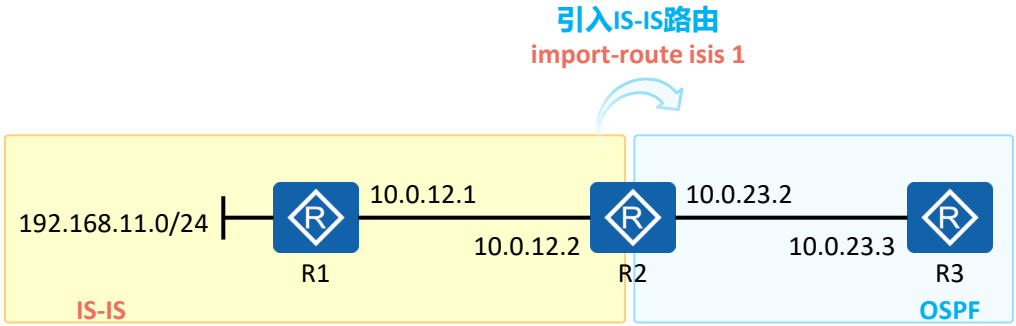
目的网络/掩码	协议	下一跳
192.168.11.0/24	Static	10.0.12.1
10.0.12.0/24	Direct	10.0.12.2
10.0.23.0/24	Direct	10.0.23.2

R3的路由表

目的网络/掩码	协议	下一跳
192.168.11.0/24	O_ASE	10.0.23.2
10.0.23.0/24	Direct	10.0.23.3

- 可以通过使用**import-route static**命令，将路由表中所有静态路由引入到动态路由协议。
- 引入后的路由会作为OSPF外部路由，在整个OSPF网络内通告。

案例3：将IS-IS路由引入到OSPF



R2的路由表

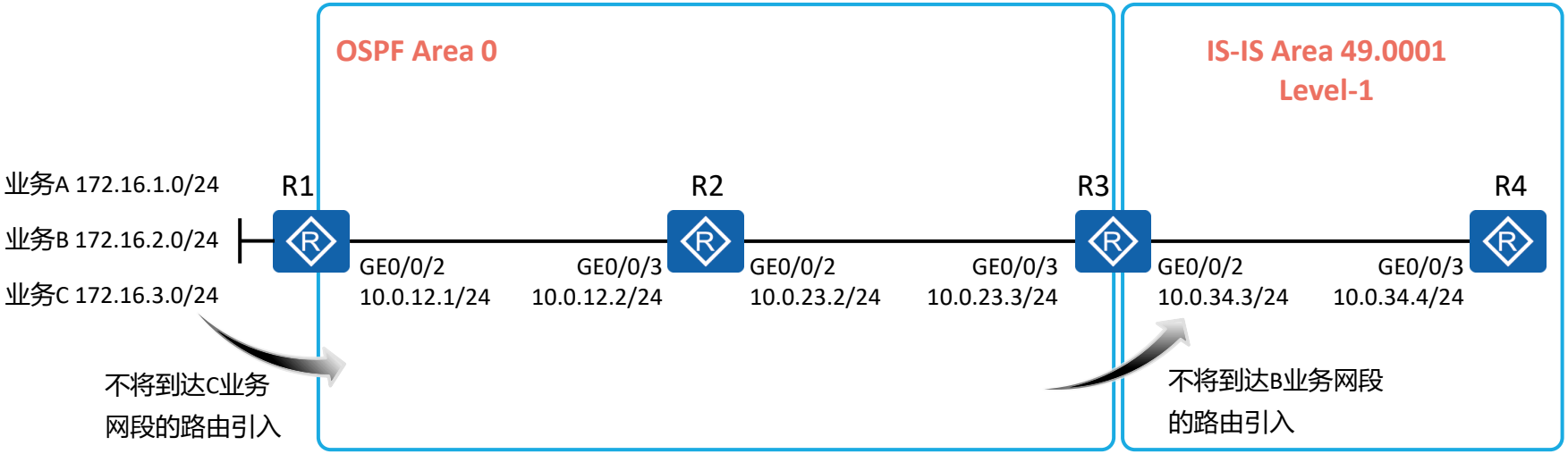
目的网络/掩码	协议	下一跳
192.168.11.0/24	IS-IS	10.0.12.1
10.0.12.0/24	Direct	10.0.12.2
10.0.23.0/24	Direct	10.0.23.2

R3的路由表

目的网络/掩码	协议	下一跳
192.168.11.0/24	O_ASE	10.0.23.2
10.0.12.0/24	O_ASE	10.0.23.2
10.0.23.0/24	Direct	10.0.23.3

- 可以通过使用**import-route isis 1**命令，将路由表中所有IS-IS路由引入到动态路由协议。
- 引入后的路由会作为OSPF外部路由，在整个OSPF网络内通告。

技术背景

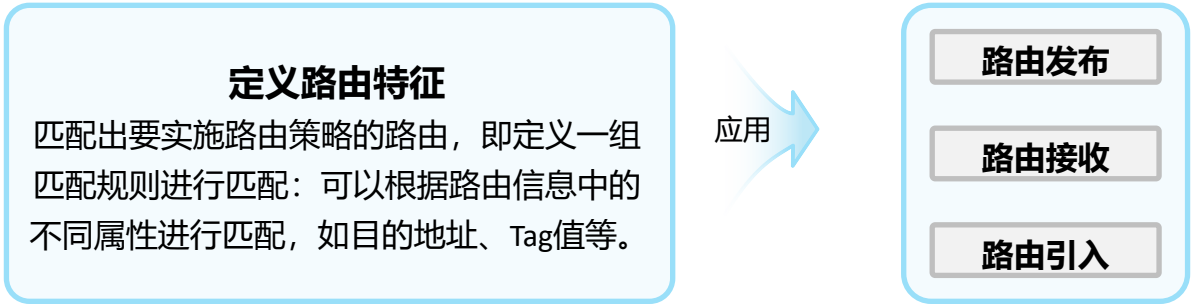


- R1上将业务A、B、C的网段路由引入时希望不引入业务C网段路由，同时R3上将OSPF路由引入到IS-IS中时，同样只希望引入B业务网段路由。
- 此时需要一个工具在引入路由时进行限制。

路由控制概述

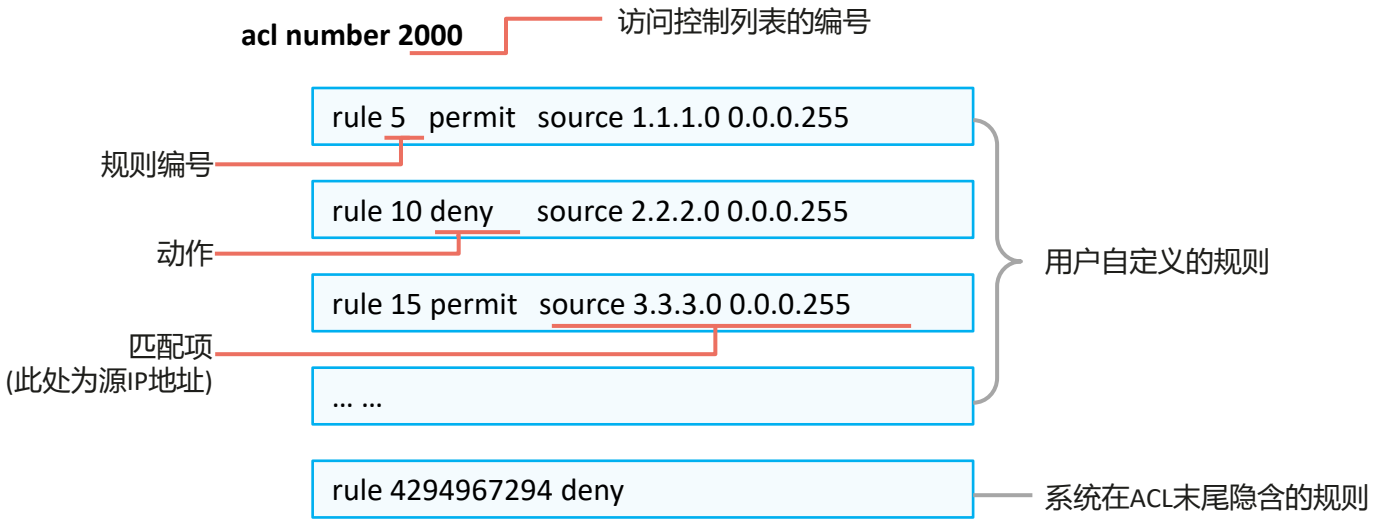
路由控制可以通过路由策略（Route-Policy）实现，路由策略应用灵活而广泛，有以下几种常见方式：

- 控制路由的发布：通过路由策略对发布的路由进行过滤，只发布满足条件的路由。
- 控制路由的接收：通过路由策略对接收的路由进行过滤，只接收满足条件的路由。
- 控制路由的引入：通过路由策略控制从其他路由协议引入的路由条目，只有满足条件的路由才会被引入。



匹配工具1：访问控制列表

- 访问控制列表（Access Control List，ACL）是一个匹配工具，能够对报文及路由进行匹配和区分。
- ACL由若干条permit或deny语句组成。每条语句就是该ACL的一条规则，每条语句中的permit或deny就是与这条规则相对应的处理动作。



通配符

```
acl number 2000
                                                    通配符
 rule      5        deny      source  10.1.1.1  0
 rule      10       deny      source  10.1.1.2  0
 rule      15       permit    source  10.1.1.0  0.0.0.255
```

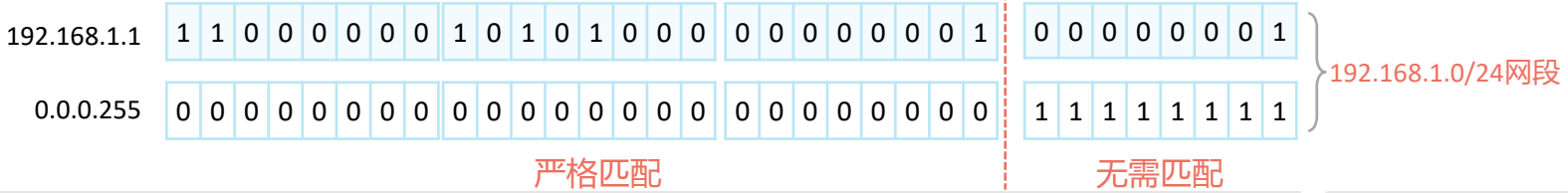
通配符 (Wildcard)

- 通配符是一个32比特长度的数值，用于指示IP地址中哪些比特位需要严格匹配，哪些比特位无需匹配。
- 通配符通常采用类似网络掩码的点分十进制形式表示，但是含义却与网络掩码完全不同。

- 匹配规则：

  “0”表示“匹配”；“1”表示“无需匹配”

如何匹配192.168.1.0/24网段内的IP地址?

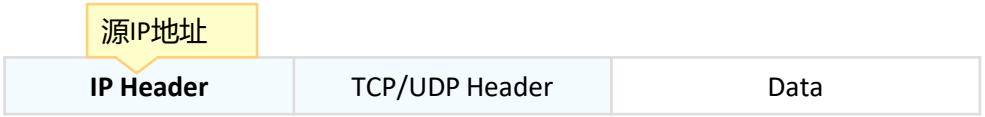


ACL的分类与基本ACL

- 基于ACL规则定义方式的划分

分类	编号范围	规则定义描述
基本ACL	2000~2999	仅使用报文的源IP地址、分片信息和生效时间段信息来定义规则。
高级ACL	3000~3999	可使用IPv4报文的源IP地址、目的IP地址、IP协议类型、ICMP类型、TCP源/目的端口、UDP源/目的端口号、生效时间段等来定义规则。
二层ACL	4000~4999	使用报文的以太网帧头信息来定义规则，如根据源MAC地址、目的MAC地址、二层协议类型等。
用户自定义ACL	5000~5999	使用报文头、偏移位置、字符串掩码和用户自定义字符串来定义规则。
用户ACL	6000~6999	既可使用IPv4报文的源IP地址或源UCL（User Control List）组，也可使用目的IP地址或目的UCL组、IP协议类型、ICMP类型、TCP源端口/目的端口、UDP源端口/目的端口号等来定义规则。

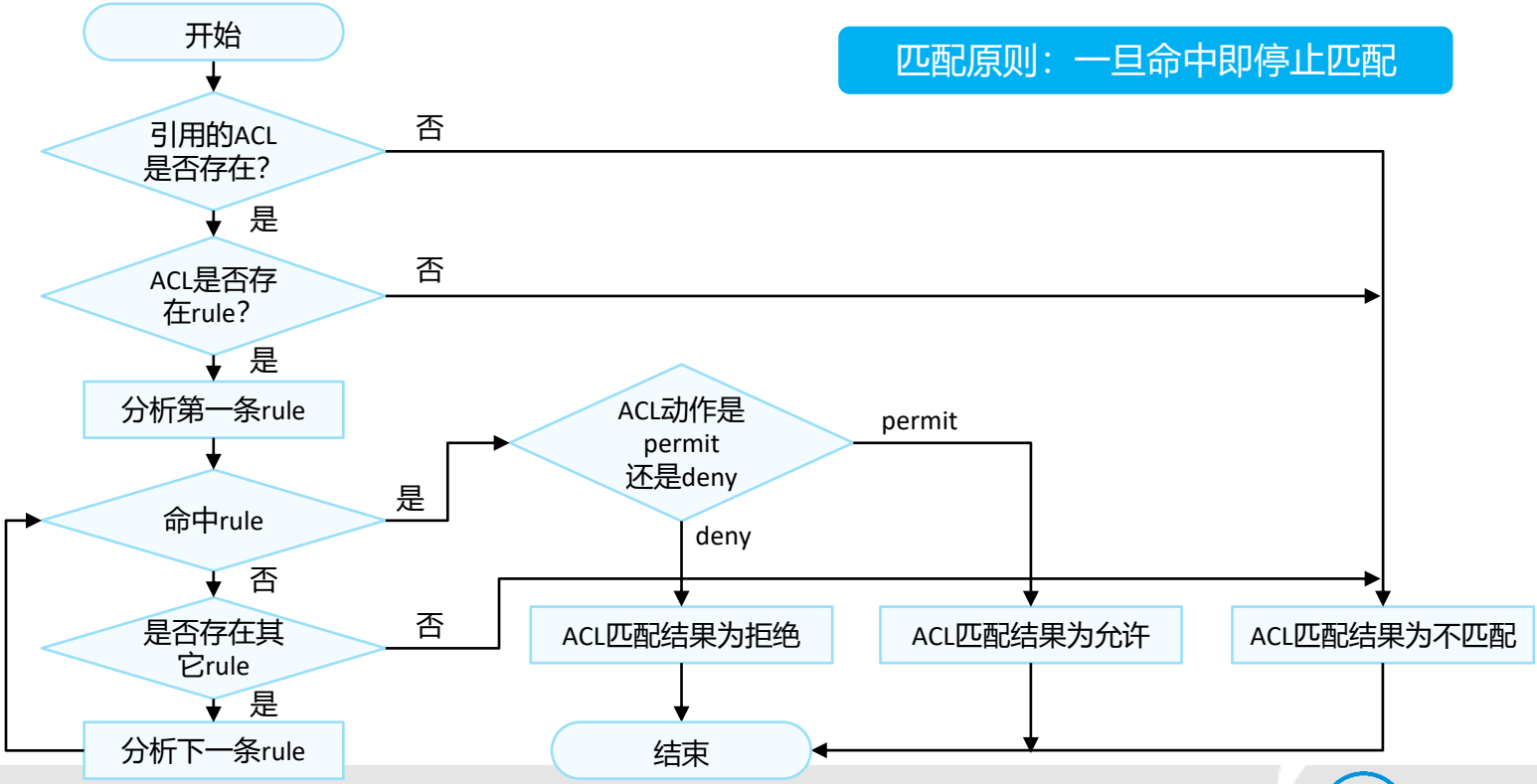
- 基本ACL




```
acl number 2000
 rule      5     deny      source  10.1.1.1  0
 rule      10    deny      source  10.1.1.2  0
 rule      15    permit    source  10.1.1.0  0.0.0.255
```

ACL的匹配机制

匹配原则：一旦命中即停止匹配



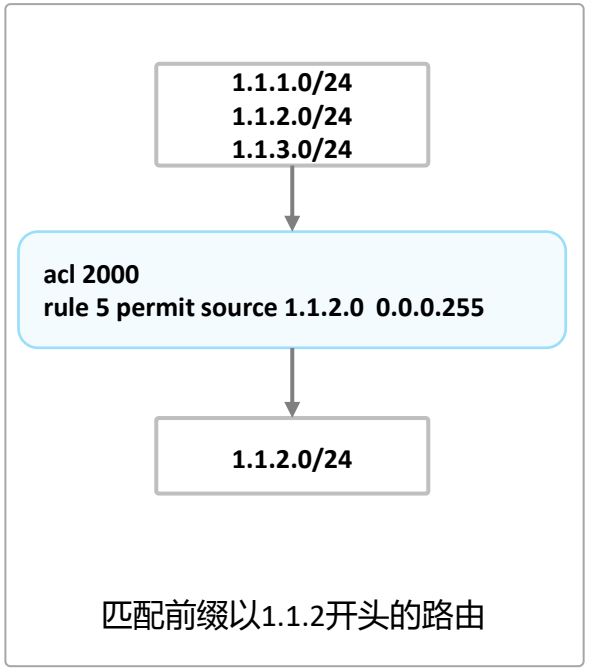
ACL的匹配顺序及匹配结果

配置顺序（config模式）

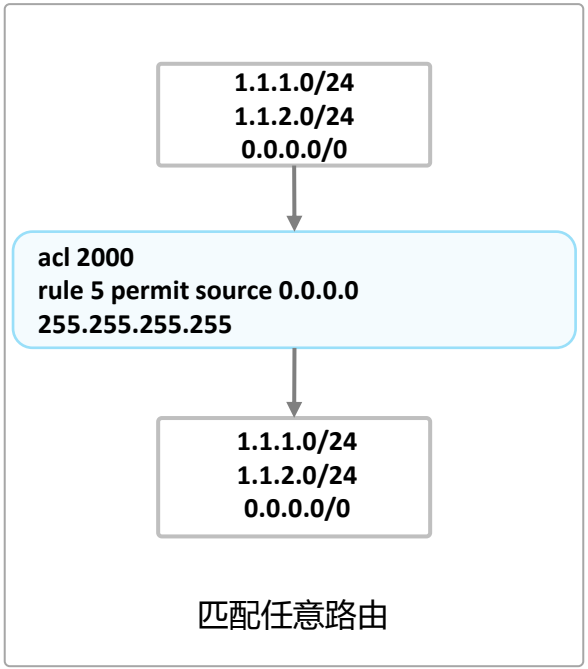
- 系统按照ACL规则编号从小到大的顺序进行报文匹配，规则编号越小越容易被匹配。

“允许”是指什么?

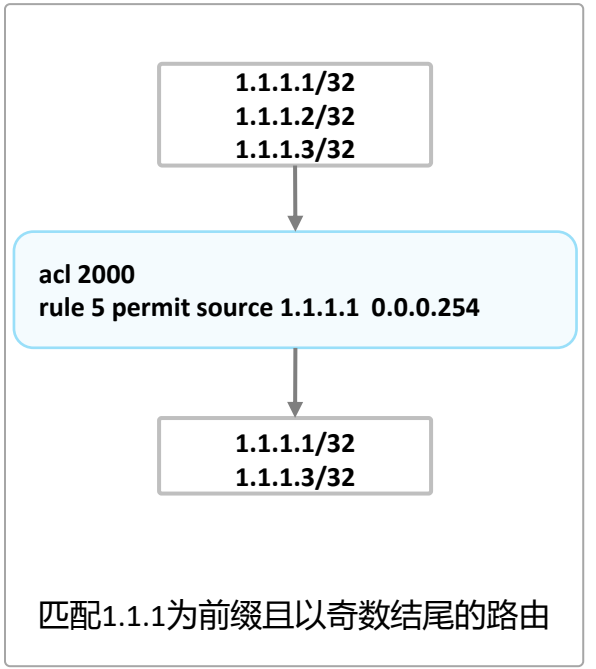
常用匹配举例



匹配前缀以1.1.2开头的路由



匹配任意路由



匹配1.1.1为前缀且以奇数结尾的路由

ACL只能匹配路由的前缀，无法匹配路由的网络掩码。

基本ACL的基础配置命令

1. 创建基本ACL

[Huawei] **acl** [**number**] *acl-number* [**match-order config**]

使用编号（2000～2999）创建一个数字型的基本ACL，并进入基本ACL视图。

[Huawei] **acl name** *acl-name* { **basic** | *acl-number* } [**match-order config**]

使用名称创建一个命名型的基本ACL，并进入基本ACL视图。

2. 配置基本ACL的规则

[Huawei-acl-basic-2000] **rule** [*rule-id*] { **deny** | **permit** } [**source** { *source-address source-wildcard* | **any** } | **time-range** *time-name*]

在基本ACL视图下，通过此命令来配置基本ACL的规则。

匹配工具2：IP前缀列表

- IP前缀列表（IP-Prefix List）是将路由条目的网络地址、掩码长度作为匹配条件的过滤器，可在各路由协议发布和接收路由时使用。
- 不同于ACL，IP-Prefix List能够同时匹配IP地址前缀长度以及掩码长度，增强了匹配的精确度。

```
[Huawei] ip ip-prefix test index 10 permit 192.168.1.0 22 greater-equal 24 less-equal 26
```

ip-prefix-name　序号　动作　IP网段与掩码　掩码范围

1、**ip-prefix-name：**地址前缀列表名称

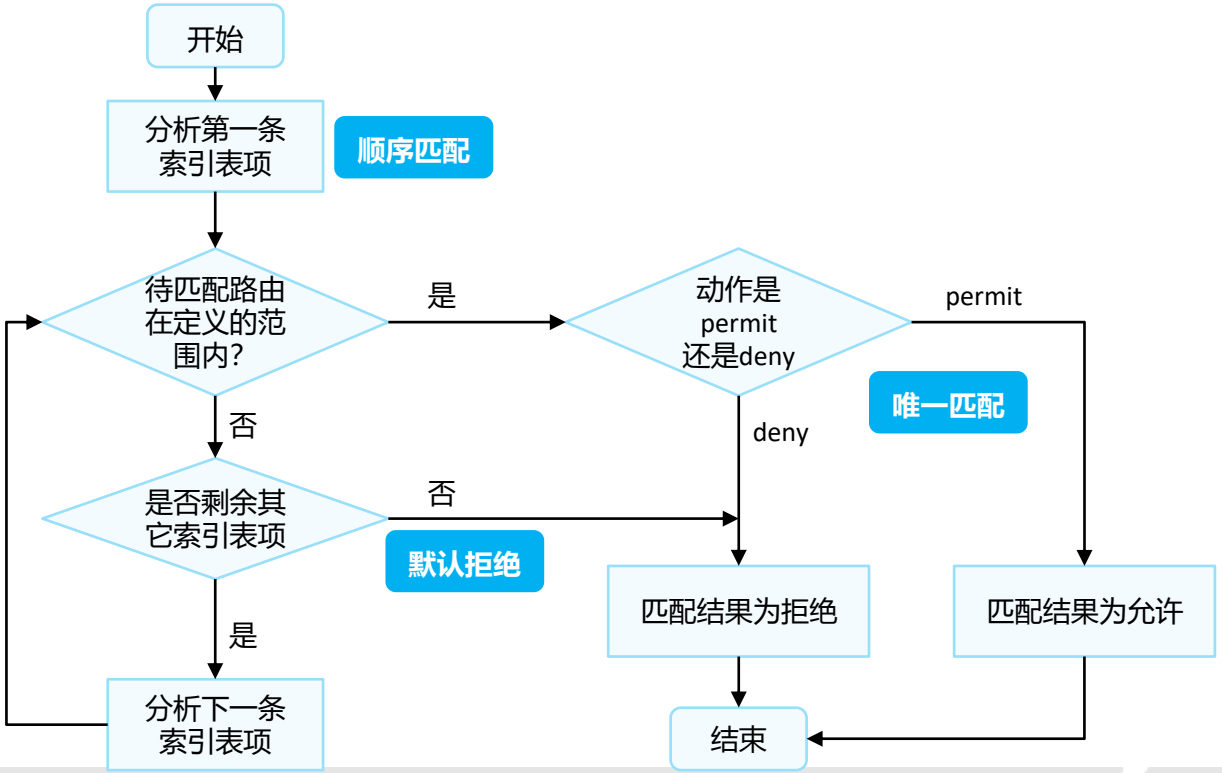
2、**序号：**本匹配项在地址前缀列表中的序号，匹配时根据序号从小到大进行顺序匹配

3、**动作：**permit/deny，地址前缀列表的匹配模式为允许/拒绝，表示匹配/不匹配

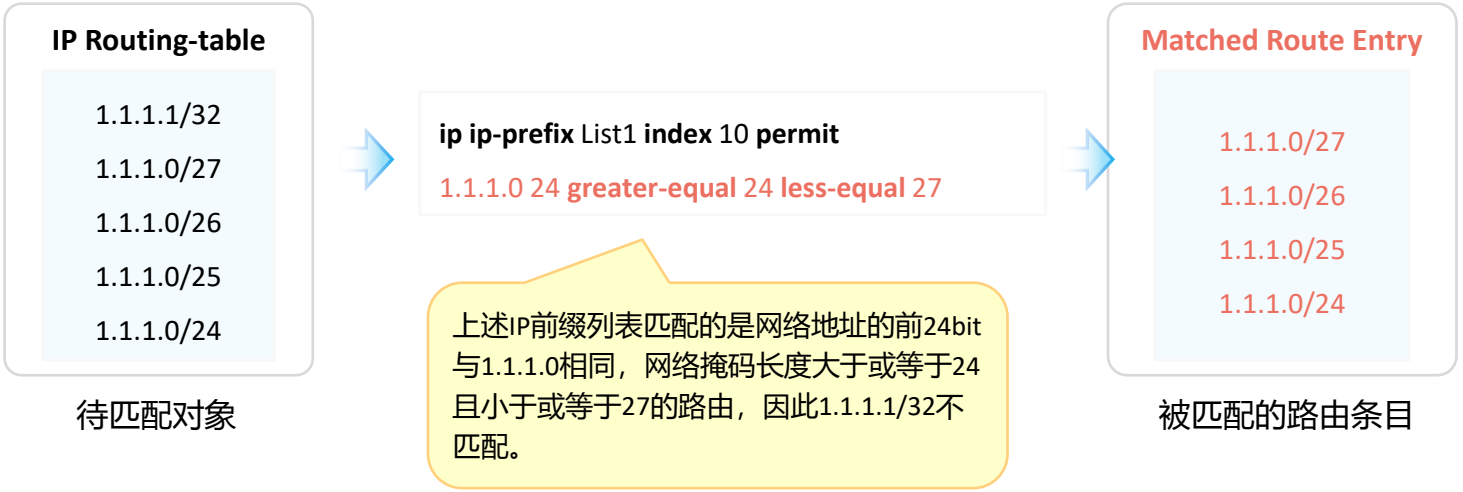
4、**IP网段与掩码：**匹配路由的网络地址，以及限定网络地址的前多少位需严格匹配

5、**掩码范围：**匹配路由前缀长度，掩码长度的匹配范围 mask-length<=greater-equal-value<=less-equal-value<=32

IP-Prefix的匹配机制



IP-Prefix的匹配示例



IP-Prefix的基础配置命令

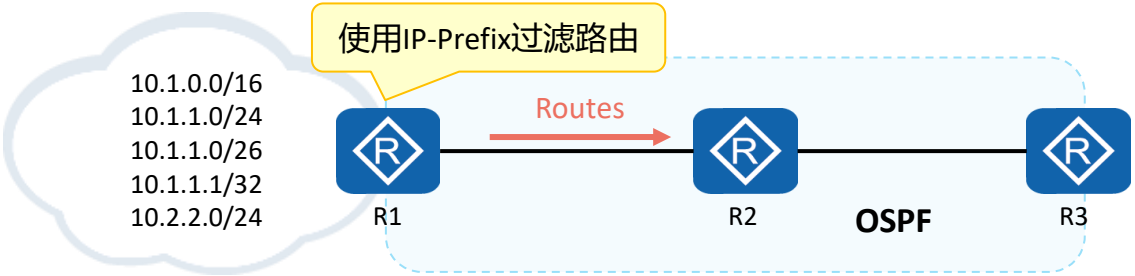
1. 创建IPv4地址前缀列表

[Huawei] **ip ip-prefix** *ip-prefix-name* [**index** *index-number*] { **permit** | **deny** } *ipv4-address mask-length* [**match-network**] [**greater-equal** *greater-equal-value*] [**less-equal** *less-equal-value*]

创建IPv4地址前缀列表或增加其中一个表项。

- *ip-prefix-name*：指定地址前缀列表的名称。
- **index** *index-number*：指定本匹配项在地址前缀列表中的序号。
- **permit**：指定地址前缀列表的匹配模式为允许。
- **deny**：指定地址前缀列表的匹配模式为拒绝。
- *ipv4-address mask-length*：指定IP地址和指定掩码长度。
- **greater-equal** *greater-equal-value*：指定掩码长度匹配范围的下限。
- **less-equal** *less-equal-value*：指定掩码长度匹配范围的上限。

IP-Prefix的配置举例 (1)



单语句匹配

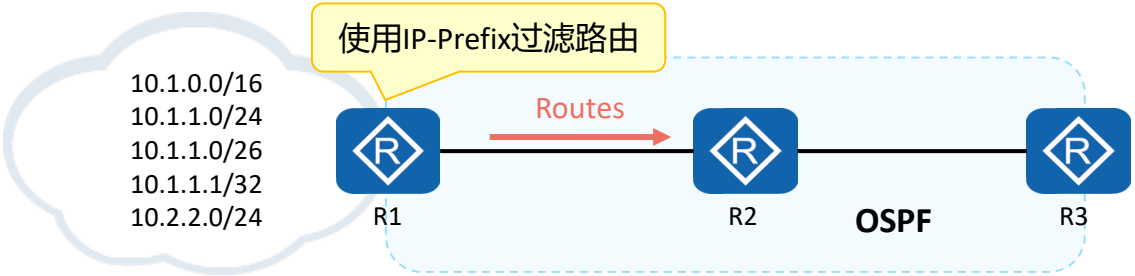
```
ip ip-prefix aa index 10 permit 10.1.1.0 24
```

– Case1：路由10.1.1.0/24被Permit，其他都被Deny。

```
ip ip-prefix bb index 10 deny 10.1.1.0 24
```

– Case2：路由全部被Deny。

IP-Prefix的配置举例 (2)



多语句匹配

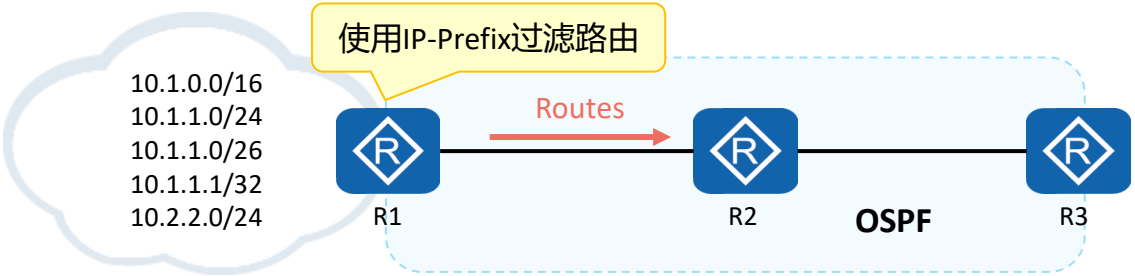
```
ip ip-prefix aa index 10 deny 10.1.1.0 24
ip ip-prefix aa index 20 permit 10.1.1.1 32
```

– Case1：路由10.1.1.0/24被Deny，路由10.1.1.1/32被Permit，其他路由都被Deny。

```
ip ip-prefix bb index 10 permit 10.1.1.0 24 greater-equal 26 less-equal 32
```

– Case2：路由10.1.1.0/26 ， 10.1.1.1/32被Permit，其他路由被Deny。

IP-Prefix的配置举例 (3)



通配地址匹配

```
ip ip-prefix aa index 10 permit 10.0.0.0 8 less-equal 32
```

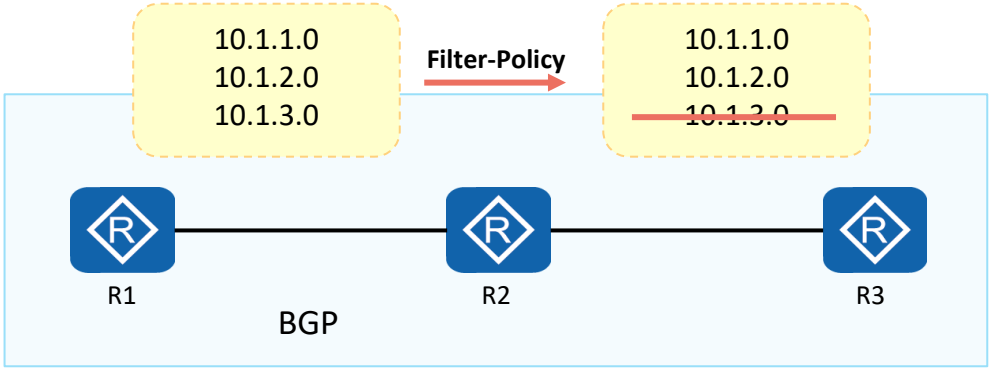
– Case1：所有掩码长度在8到32的路由都被Permit。

```
ip ip-prefix bb index 10 deny 10.1.1.0 24 less-equal 32
ip ip-prefix bb index 20 permit 10.1.0.0 16 less-equal 32
```

– Case2：路由10.1.0.0/16被Permit，其他路由被Deny。

策略工具1：Filter-Policy

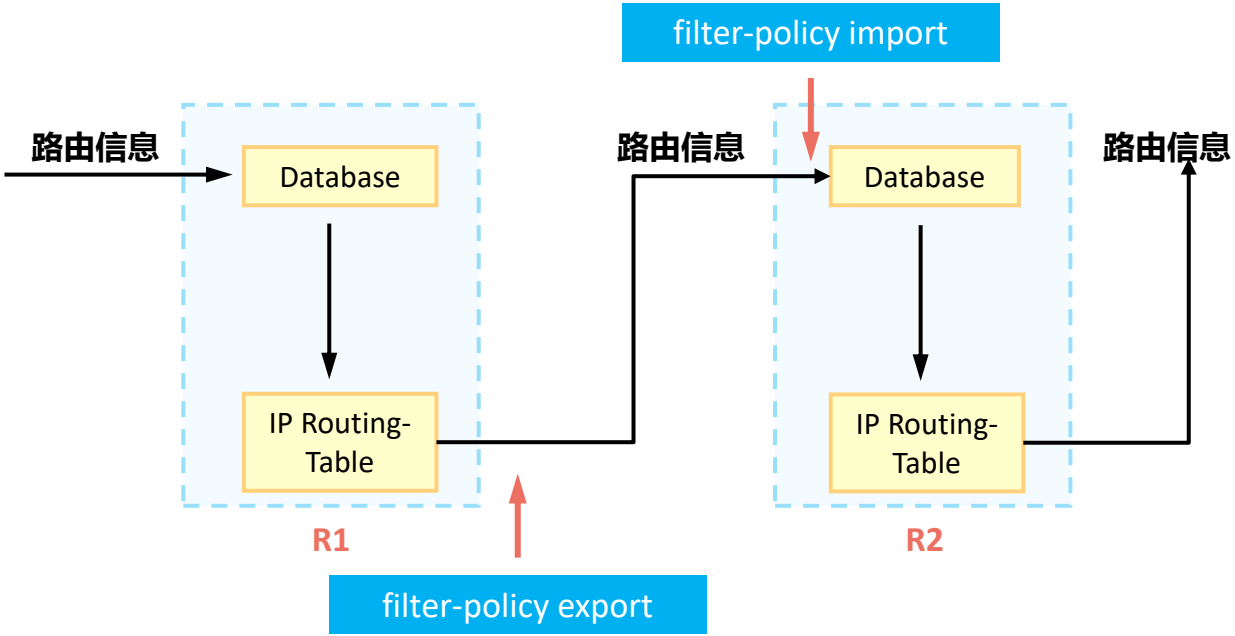
- Filter-Policy（过滤-策略）是一个很常用的路由信息过滤工具，能够对接收、发布、引入的路由进行过滤，可应用于IS-IS、OSPF、BGP等协议。



- 如图所示，R1、R2、R3之间运行BGP路由协议，路由在各个设备之间传递，当需要根据实际需求过滤某些路由信息的时候可以使用Filter-Policy实现。

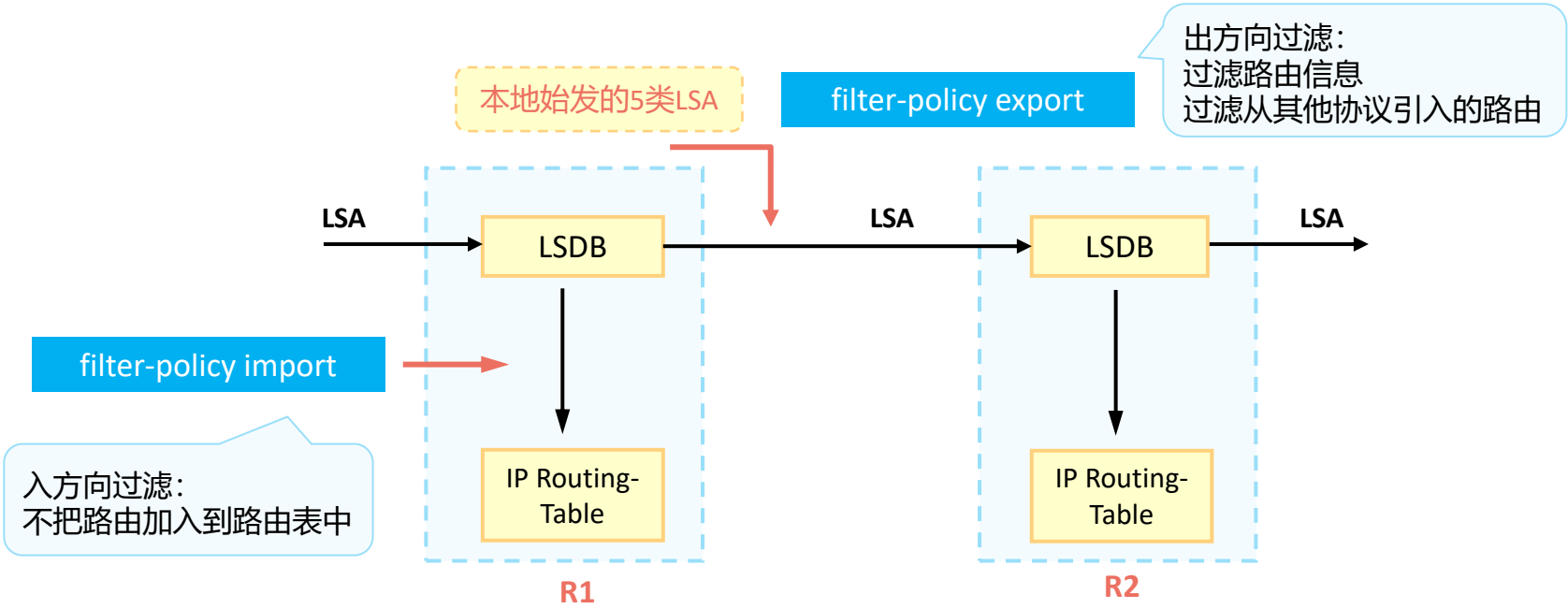
Filter-Policy在距离矢量路由协议中的应用

在距离矢量路由协议中，设备之间传递的是路由信息，如果需要对这种路由信息进行某种过滤，可以使用Filter-Policy实现，出方向和入方向的生效位置如图所示。



Filter-Policy在链路状态路由协议中的应用

在链路状态路由协议中，各路由设备之间传递的是LSA信息，然后设备根据LSA汇总成的LSDB信息计算出路由表。但是Filter-Policy只能过滤路由信息，无法过滤LSA。



Filter-Policy的基础配置命令 (1)

1. 在OSPF中的应用

[Huawei-ospf-100] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* [**secondary**] } **import**

按照过滤策略，设置OSPF对接收的路由进行过滤。

[Huawei-ospf-100] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]

按照过滤策略，设置对引入的路由在向外发布时进行过滤。

Filter-Policy的基础配置命令 (2)

2. 在IS-IS中的应用

[Huawei-isis-1] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **import**

配置IS-IS路由加入IP路由表时的过滤策略。

[Huawei-isis-1] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]

配置IS-IS对已引入的路由在向外发布时进行过滤的过滤策略。

Filter-Policy的基础配置命令 (3)

3. 在BGP中的应用

[Huawei-bgp-af-ipv4] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **import**

配置对接收的路由信息进行过滤。

[Huawei-bgp-af-ipv4] **filter-policy** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]

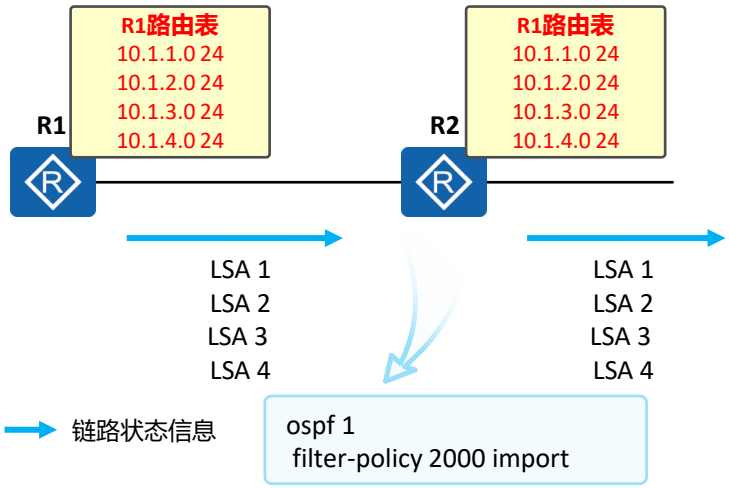
配置对发布的路由进行过滤，只有通过过滤的路由才被BGP发布。

[Huawei-bgp-af-ipv4] **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } { **import** | **export** }

配置向对等体（组）发布或从对等体（组）接收路由时的过滤策略。

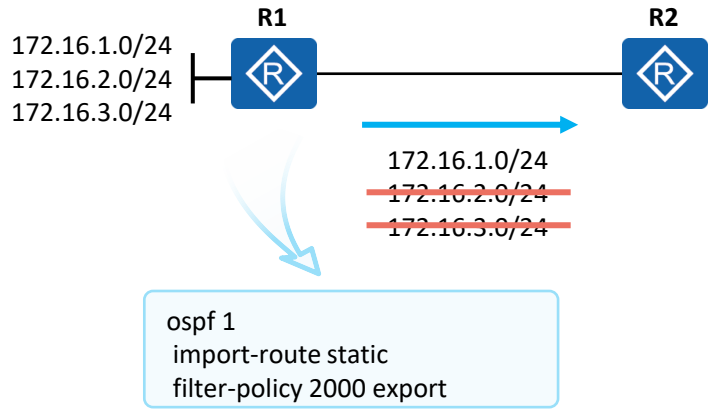
OSPF中使用Filter-Policy

filter-policy import



filter-policy import命令对接收的路由设置过滤策略，只有通过过滤策略的路由才被添加到路由表中，没有通过过滤策略的路由不会被添加进路由表，但不影响对外发布出去。

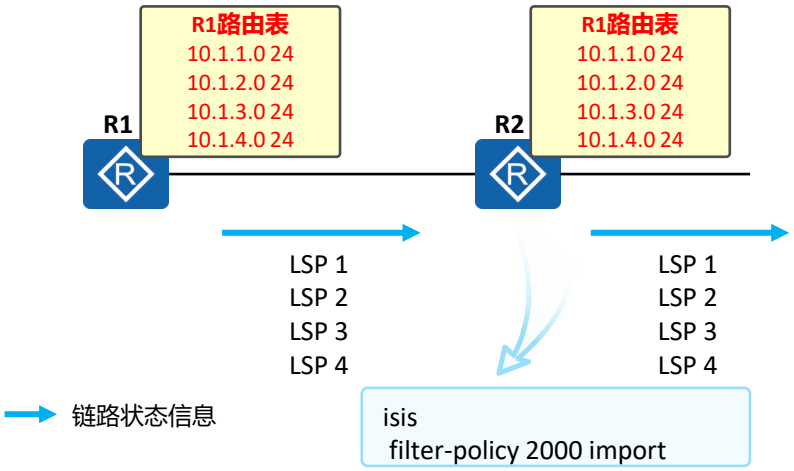
filter-policy export



OSPF通过命令**import-route**引入外部路由后，为了避免路由环路的产生，通过**filter-policy export**命令对引入的路由在发布时进行过滤，只将满足条件的外部路由转换为Type5 LSA（AS-external-LSA）并发布出去。

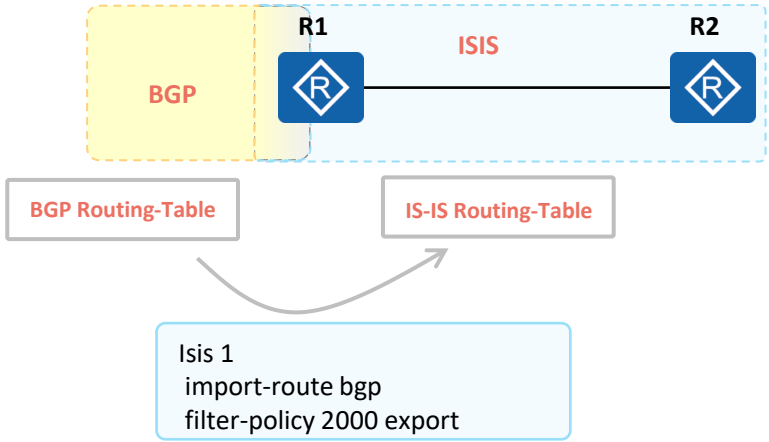
IS-IS中使用Filter-Policy

filter-policy import



与OSPF相似，**filter-policy import**命令只会对本地的路由表产生影响，不会将匹配的路由加入到路由表，不会影响本地设备的LSP的扩散和LSDB的同步。

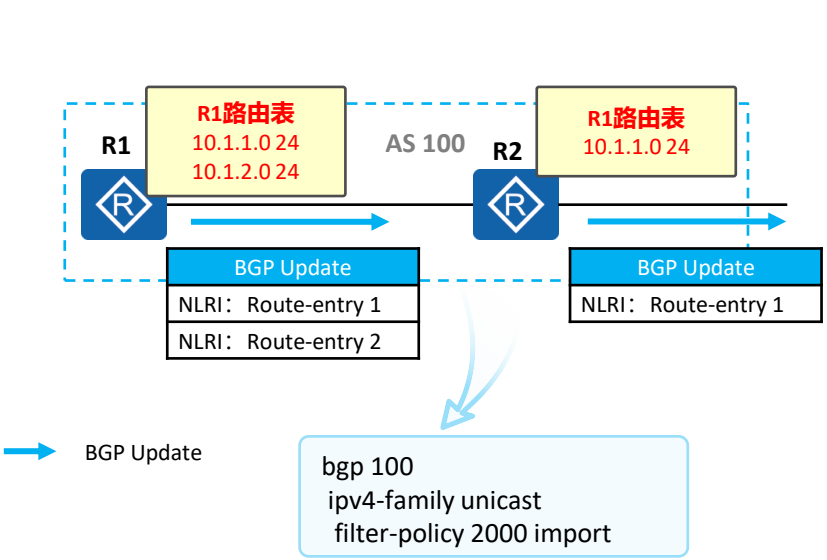
filter-policy export



当网络中同时部署了IS-IS和其他路由协议时，如果已经在边界设备上引入其他路由协议的路由，缺省情况下，该设备将把引入的全部外部路由发布给IS-IS邻居。如果只希望将引入的部分外部路由发布给邻居，可以使用**filter-policy export**命令实现。

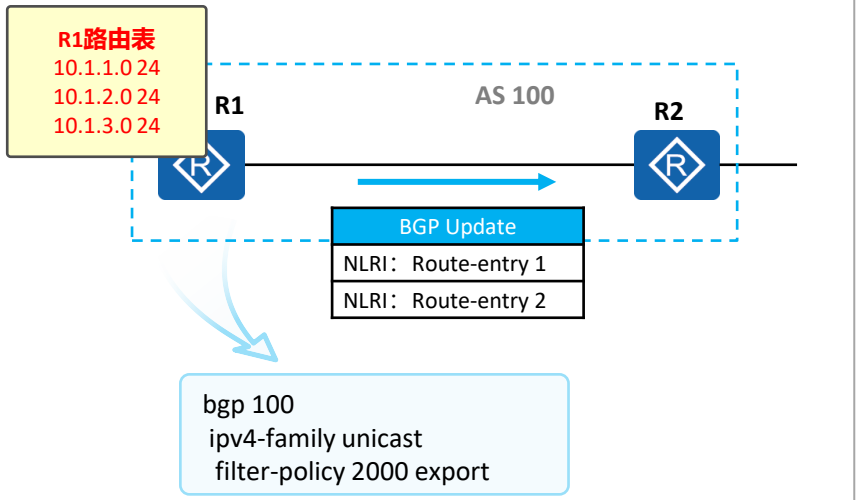
BGP中使用Filter-Policy

filter-policy import



使用**filter-policy import**命令可以对BGP设备全局接收的路由进行过滤，决定是否将路由添加到BGP路由表中。

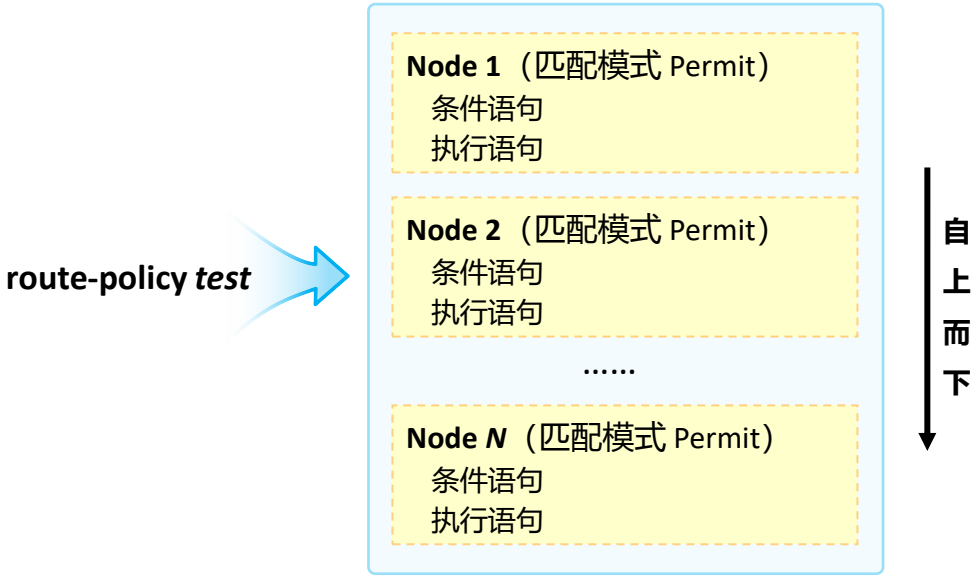
filter-policy export



使用**filter-policy export**命令可以将对外发布的路由进行过滤，只有通过过滤的路由才能加入BGP本地路由表，并被BGP发布。

策略工具2：Route-Policy

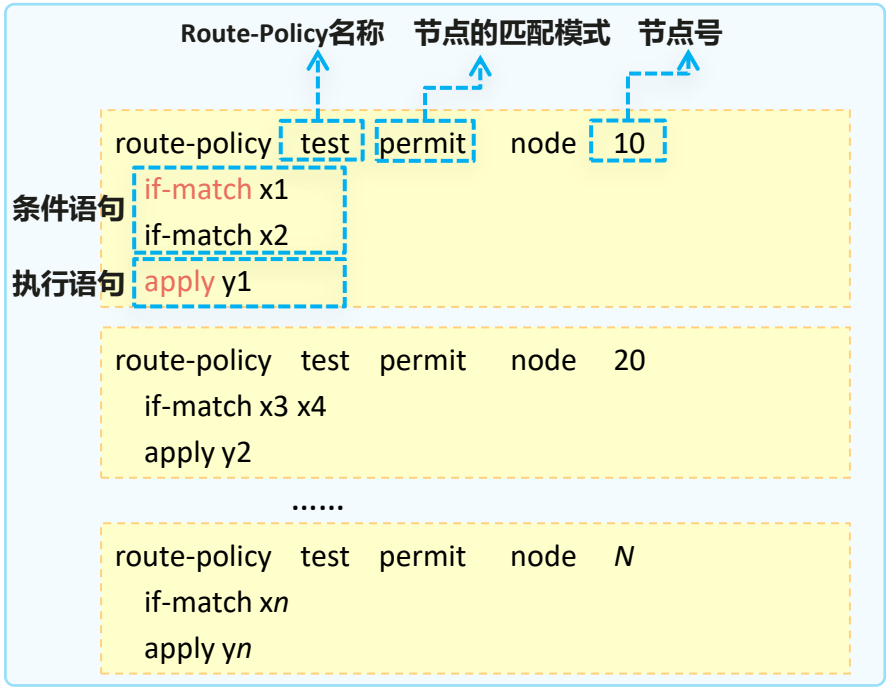
- Route-Policy是一个策略工具，用于过滤路由信息，以及为过滤后的路由信息设置路由属性。
- 一个Route-Policy由一个或多个节点（Node）构成，每个节点都可以是一系列条件语句（匹配条件）以及执行语句（执行动作）的集合，这些集合按照编号从小到大的顺序排列。



- 每个节点内可包含多个条件语句。节点内的多个条件语句之间的关系为“与”，即匹配所有条件语句才会执行本节点内的动作。
- 节点之间的关系为“或”，route-policy根据节点编号大小从小到大顺序执行，匹配中一个节点将不会继续向下匹配。

Route-Policy的组成

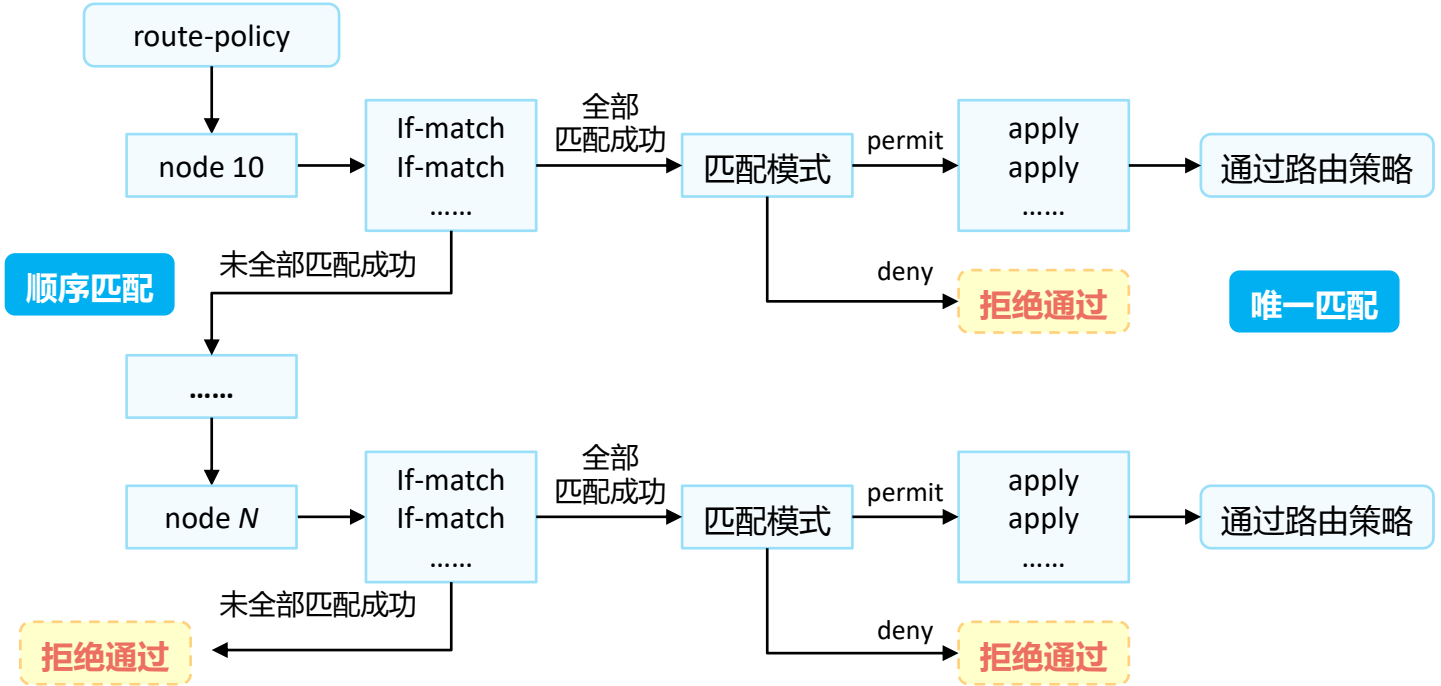
一个Route-Policy由一个或多个节点构成，每个节点包括多个if-match和apply子句。



- permit或deny：指定Route-Policy节点的匹配模式为允许或拒绝。
- node：指定Route-Policy的节点号。整数形式，取值范围是0 ~ 65535。
- if-match子句：定义该节点的匹配条件。
- apply子句：定义针对被匹配路由执行的操作。

Route-Policy的匹配顺序

路由策略使用不同的匹配条件和匹配模式选择路由和改变路由属性。



Route-Policy的基础配置命令 (1)

1. 创建Route-Policy

[Huawei] **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*

创建路由策略并进入Route-Policy视图。

2. （可选）配置if-match子句

[Huawei-route-policy] **if-match ?**

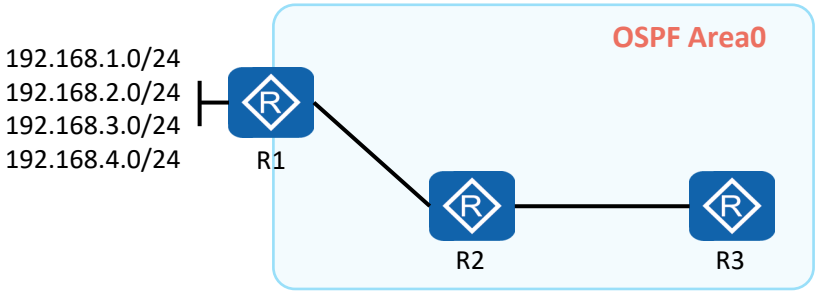
acl	匹配基本ACL
cost	匹配路由信息的cost
interface	匹配路由信息的出接口
ip-prefix	匹配前缀列表
……	

Route-Policy的基础配置命令 (2)

3. （可选）配置apply子句

```
[Huawei-route-policy] apply ?
  cost                          设置路由的cost
  cost-type {type-1 | type-2}   设置OSPF的开销类型
  ip-address next-hop           设置IPv4路由信息的下一跳地址
  preference                    设置路由协议的优先级
  tag                           设置路由信息的标记域
  ……
```

对接收的路由进行过滤



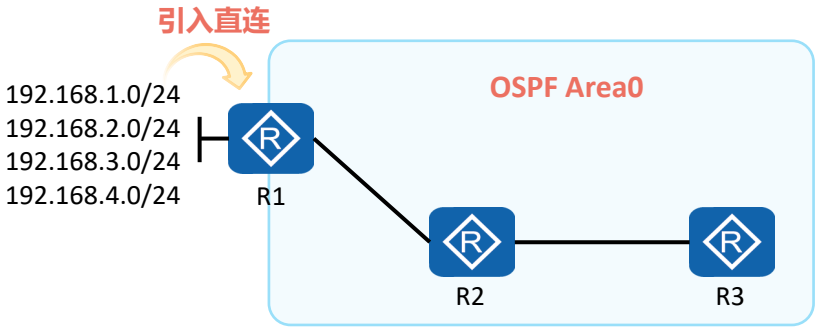
- R1、R2、R3运行OSPF，R1将192.168.1.0/24、192.168.2.0/24、192.168.3.0/24和192.168.4.0/24宣告进OSPF。
- 现在要求R2不能访问R1上192.168.1.0/24网段，但是R3可以正常访问。
- 为实现该需求，可以在R2上对接收的路由使用Filter-Policy进行过滤。

R2的配置如下：

```
[R2] ip ip-prefix in index 10 permit 192.168.2.0 24
[R2] ip ip-prefix in index 10 permit 192.168.3.0 24
[R2] ip ip-prefix in index 10 permit 192.168.4.0 24

[R2] ospf
[R2-ospf-1] filter-policy ip-prefix in import
```

注意：网络基础配置略。

对发布的路由进行过滤



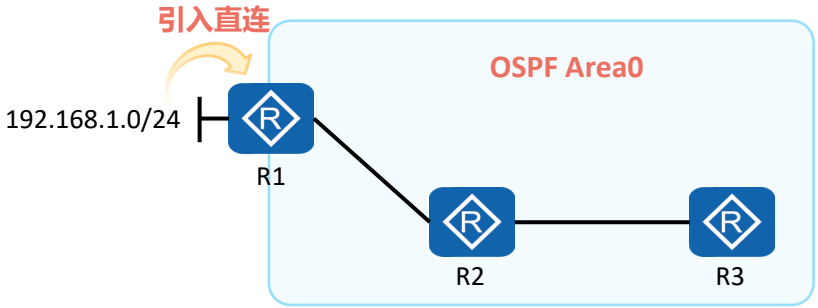
- R1、R2、R3运行OSPF，R1将直连网段192.168.1.0/24、192.168.2.0/24、192.168.3.0/24和192.168.4.0/24引入OSPF。
- 现在要求R2、R3只能学习到192.168.1.0/24网段的路由，学习不到其他三个网段的路由。
- 为实现该需求，可以在R1上使用Filter-Policy对引入的路由在发布时进行过滤。

R1的配置如下：

```
[R1] ip ip-prefix out index 10 permit 192.168.1.0 24

[R1] ospf
[R1-ospf-1] import-route direct
[R1-ospf-1] filter-policy ip-prefix out export
```

注意：网络基础配置略。

修改路由属性



- R1、R2、R3运行OSPF，R1将直连网段192.168.1.0/24引入OSPF。
- 现在要求R2、R3学到的OSPF路由192.168.1.0/24为external-type 1路由（默认为external-type 2路由）。
- 为实现该需求，可以在R1上使用Route-Policy在引入路由时修改外部路由的类型为external-type 1。

R1的配置如下：

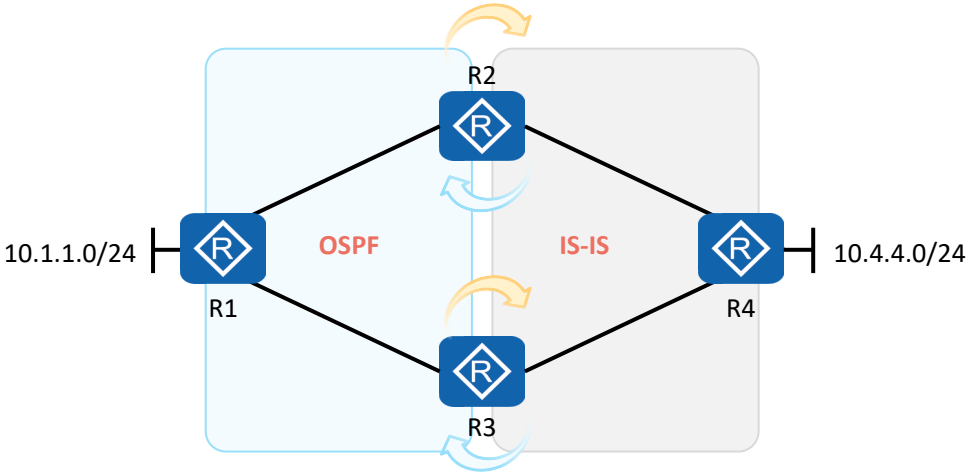
```
[R1] ip ip-prefix external index 10 permit 192.168.1.0 24

[R1] route-policy RP permit node 10
[R1-route-policy] if-match ip-prefix external
[R1-route-policy] apply cost-type type-1
[R1-route-policy] quit

[R1] ospf
[R1-ospf-1] import-route direct route-policy RP
```

注意：网络基础配置略。

双点双向路由重发布

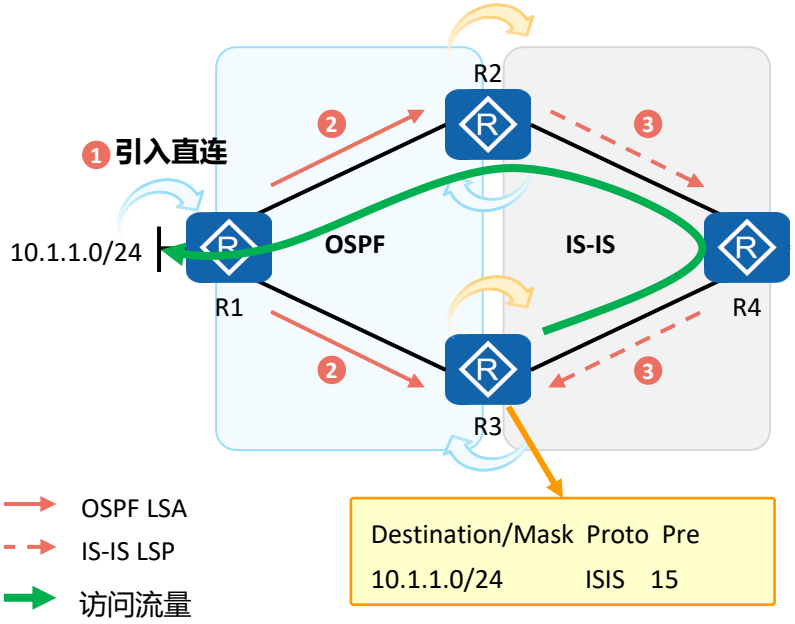
IS-IS中引入OSPF

OSPF中引入IS-IS



- 在边界路由器上把两个路由域的路由相互引入，称之为双向路由重发布。
- 两个路由域存在两个边界路由器，并且都执行双向路由重分发，此时称为双点双向路由重发布。
- 双点双向路由重发布是一种经典的路由模型，因单点的双向路由重发布缺乏冗余性，一旦单点的边界路由器故障，那么两个路由域之间的通信可能就会出现问题，因此在大型网络部署中一般采用双点双向路由重发布。
- 双点双向重路由发布虽然增强了网络的可靠性，但是容易引发：次优路径、路由环路等问题。

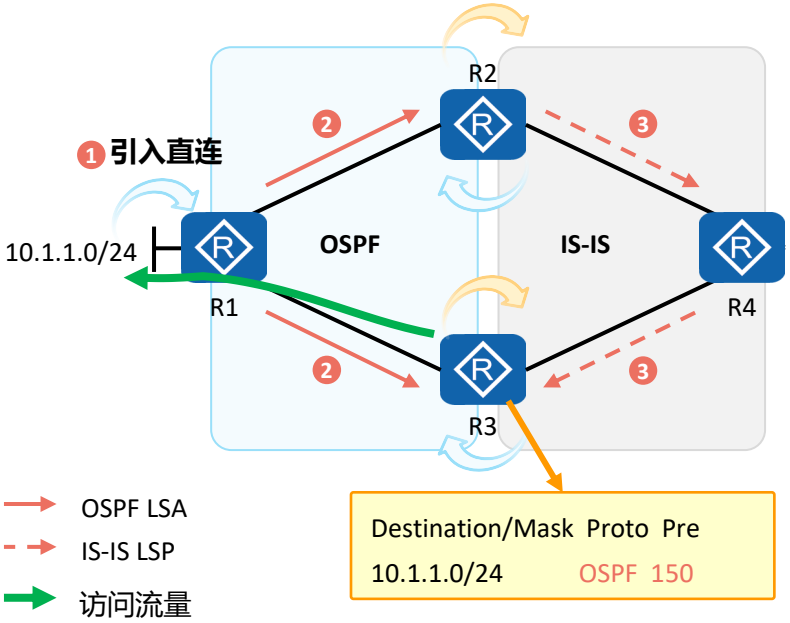
次优路径问题



以10.1.1.0/24为例：

- R1将直连路由10.1.1.0/24引入到OSPF中。
- R2、R3执行双向路由重发布，R2先将10.1.1.0/24重发布到IS-IS中，R3将会学习到来自R4的IS-IS路由。
- 对R3而言，IS-IS路由（优先级15）优于OSPF外部路由（优先级150），因此优选来自R4的IS-IS路由。后续R3访问10.1.1.0/24网段的路径为：R3->R4->R2->R1，这是次优路径。

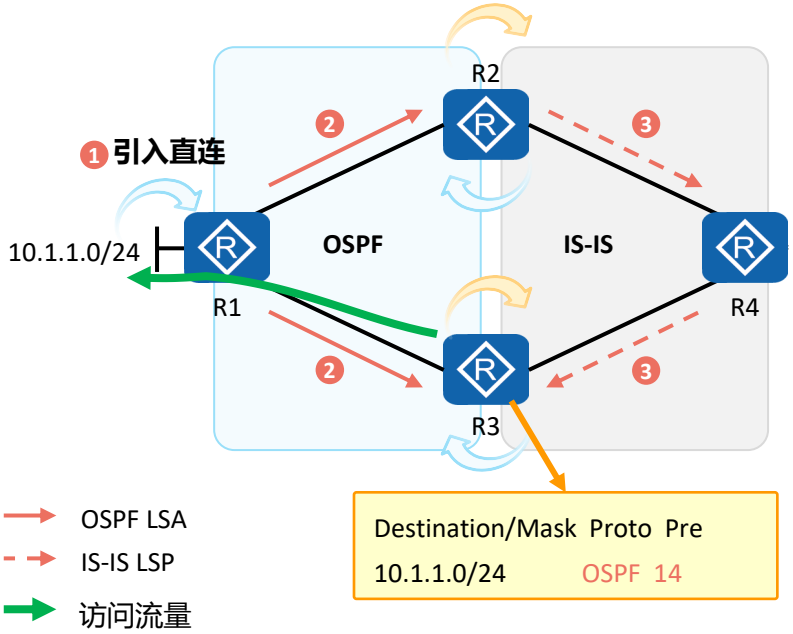
解决次优路径问题 (1)



- 解决方案一：在R3的IS-IS进程内，通过Filter-Policy禁止来自R4的10.1.1.0/24路由加入本地路由表。
- 在R3上执行以下操作：

```
[R3] acl 2001
[R3-acl-basic-2001] rule 5 deny source 10.1.1.0 0
[R3-acl-basic-2001] rule 10 permit

[R3] isis
[R3-isis-1] filter-policy 2001 import
```

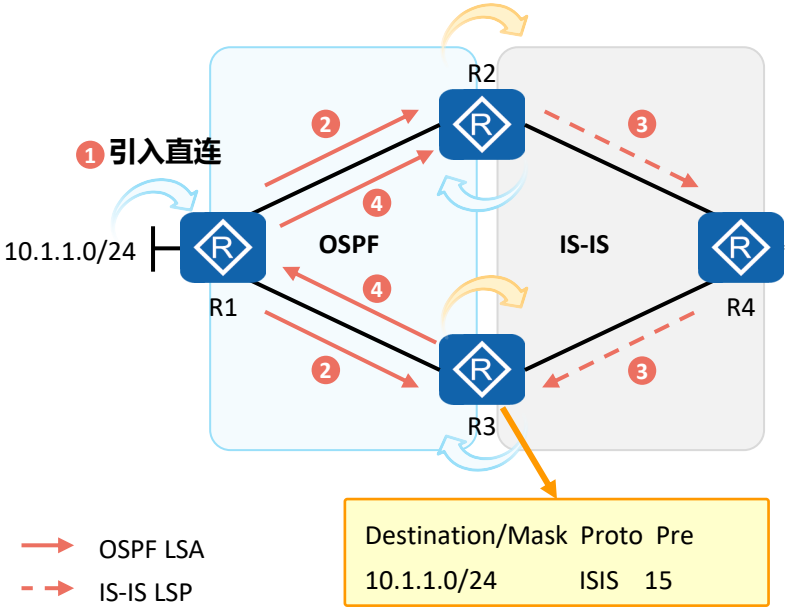
解决次优路径问题 (2)



- 解决方案二：R3通过ACL匹配10.1.1.0/24路由，在Route-Policy中调用该条ACL，将匹配这条ACL的路由的优先级设置为14（优于IS-IS）。在OSPF视图下使用**preference ase**命令调用Route-Policy修改外部路由的优先级。
- 在R3上执行以下操作：

```
[R3]acl 2000
[R3-acl-basic-2000] rule permit source 10.1.1.0 0
[R3-acl-basic-2000] quit
[R3]route-policy hcip permit node 10
[R3-route-policy] if-match acl 2000
[R3-route-policy] apply preference 14
[R3-route-policy] quit
[R3]ospf 1
[R3-ospf-1] preference ase route-policy hcip
```

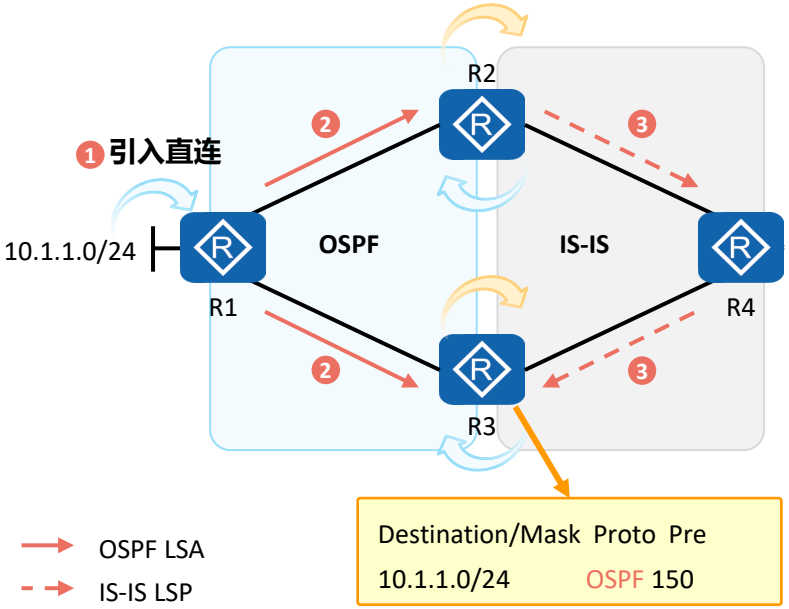
路由环路问题



场景描述：

1. R1将直连路由10.1.1.0/24引入到OSPF中。
2. R1、R2、R3运行OSPF协议，10.1.1.0/24网段路由在全OSPF域内通告。
3. R2执行了双向路由重发布。
4. R2、R3、R4运行IS-IS协议，10.1.1.0/24网段路由在全IS-IS域内通告。
5. R3执行了双向路由重发布。
6. 10.1.1.0/24网段路由再次被通告进OSPF域内，形成路由环路。

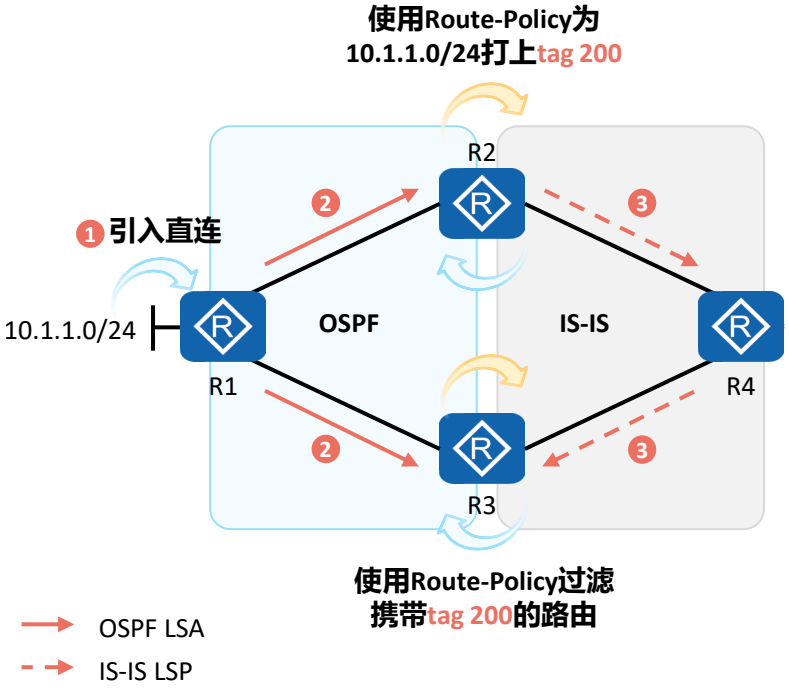
解决路由环路问题 (1)



- 解决方案一：在R3的OSPF中引入IS-IS路由时，通过Route-Policy过滤掉10.1.1.0/24路由。
- 在R3上执行以下操作：

```
[R3] acl 2001
[R3-acl-basic-2001] rule 5 deny source 10.1.1.0 0
[R3-acl-basic-2001] rule 10 permit

[R3] route-policy RP permit node 10
[R3-route-policy] if-match 2001
[R3-route-policy] quit

[R3] ospf
[R3-ospf-1] import-route isis 1 route-policy RP
```

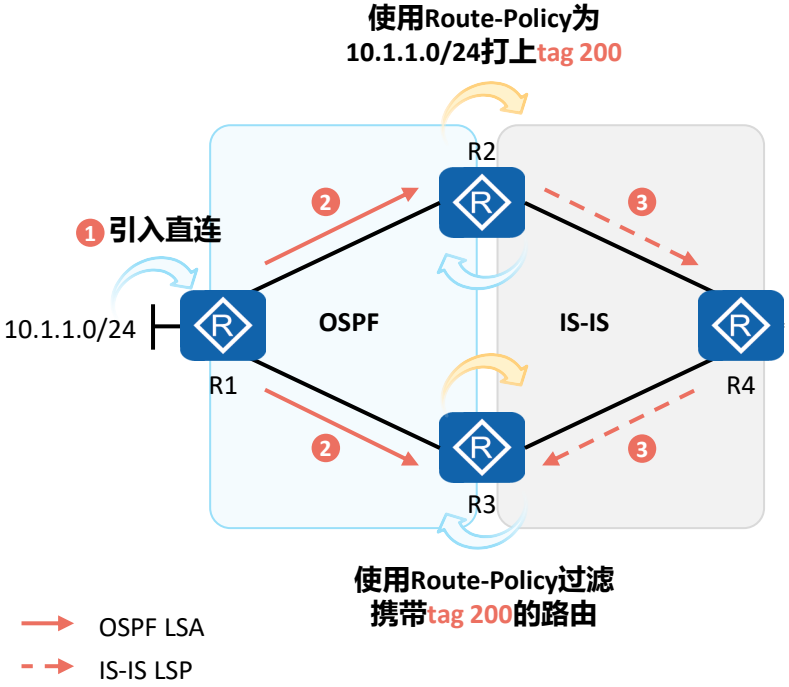
解决路由环路问题 (2)



- 解决方案二：使用Tag实现有选择性地路由引入，在R2上将路由10.1.1.0/24从OSPF引入到IS-IS中时打上Tag 200，在R3上将IS-IS引入到OSPF中时，过滤携带Tag 200的路由。
- 在R2上执行如下操作：

```
[R2]acl 2000
[R2-acl-basic-2000]rule permit source 10.1.1.0 0
[R2-acl-basic-2000]quit
[R2]route-policy hcip permit node 10
[R2-route-policy]if-match acl 2000
[R2-route-policy]apply tag 200
[R2-route-policy]quit
[R2]isis 1
[R2-isis-1]import-route ospf route-policy hcip
```

解决路由环路问题 (3)

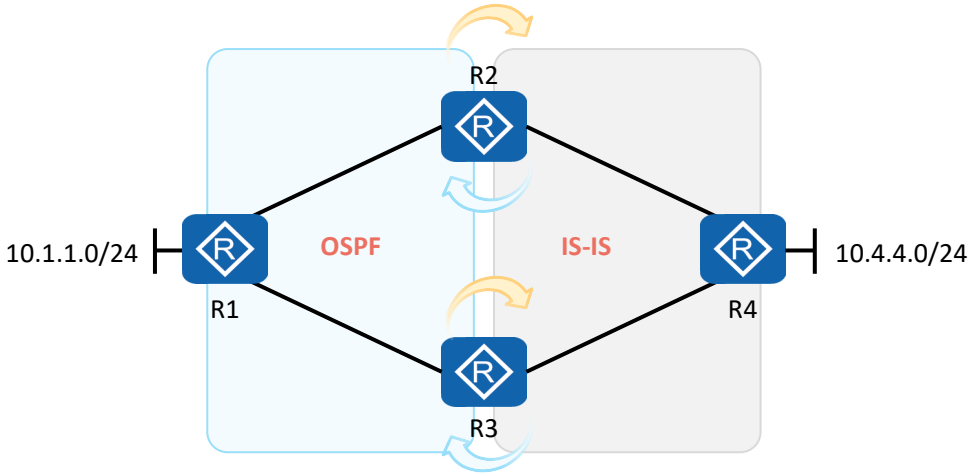


- 在R3上执行如下操作：

```
[R3]route-policy hcip deny node 10
[R3-route-policy]if-match tag 200
[R3-route-policy]quit
[R3]route-policy hcip permit node 20

[R3]ospf 1
[R3-ospf-1]import-route isis route-policy hcip
```

在路由重发布的实际应用中，通过IP前缀进行路由匹配固然可行，但当网络规模较大时，配置工作量较大；通过Tag进行路由匹配可以极大简化配置工作量。

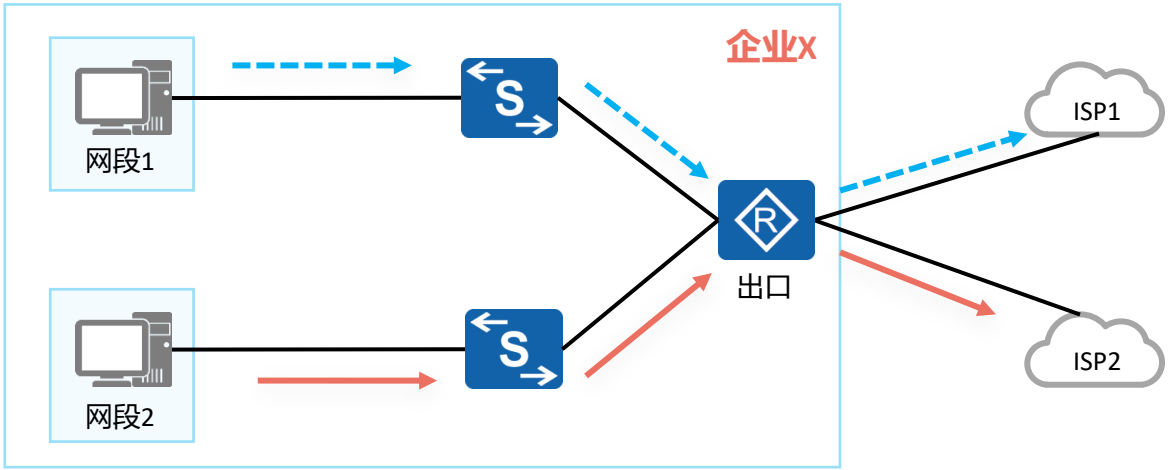
场景思考



在R1上将10.1.1.0/24引入到OSPF，在R4上将10.4.4.0/24引入到IS-IS，R2、R3上执行路由双向路由重发布，该场景下如何使用匹配Tag的方式防止路由环路?

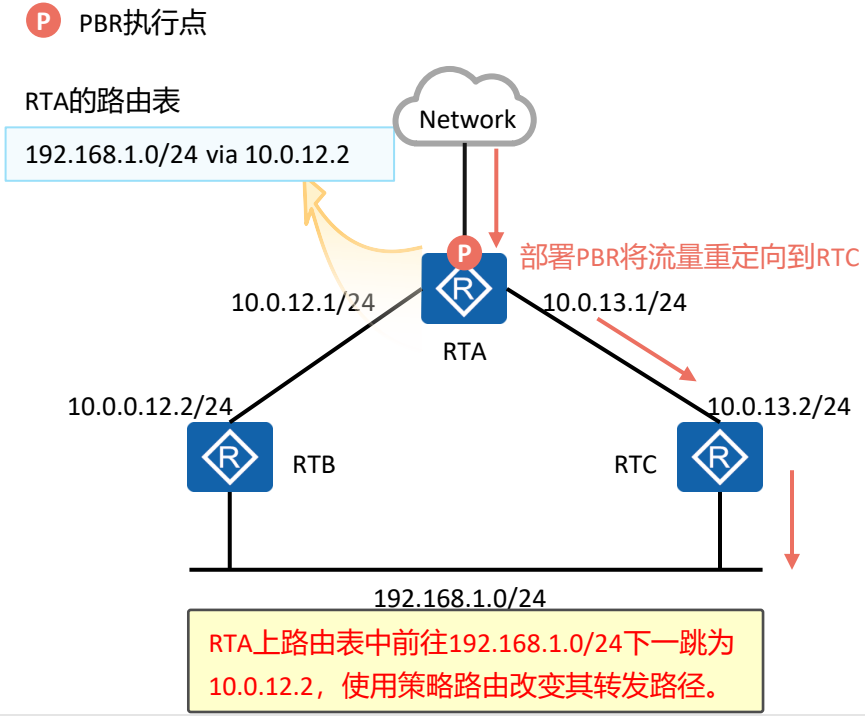
策略路由技术背景

在某些场景中我们希望一些特定用户、特定业务的流量走指定的转发路径，而其余用户或业务的流量则依旧根据路由表进行转发。



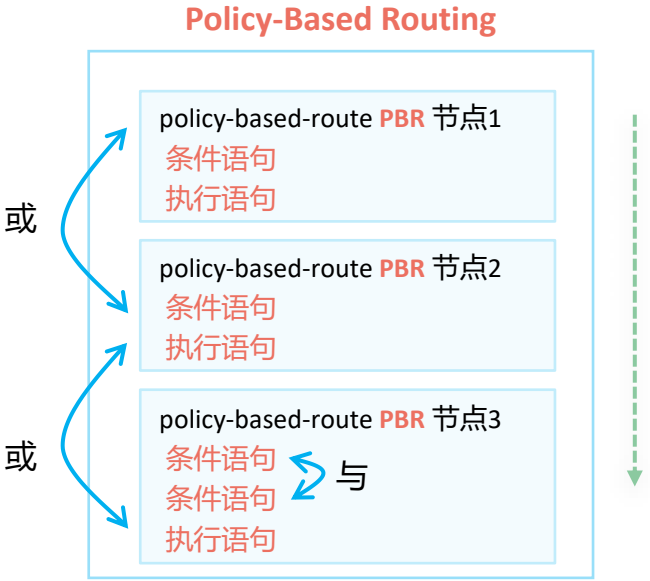
示例：双ISP接入的企业，想要实现内网网段1访问Internet通过ISP1、内网网段2访问Internet通过ISP2，该需求无法通过传统的路由技术实现。

PBR介绍 – 基本概念



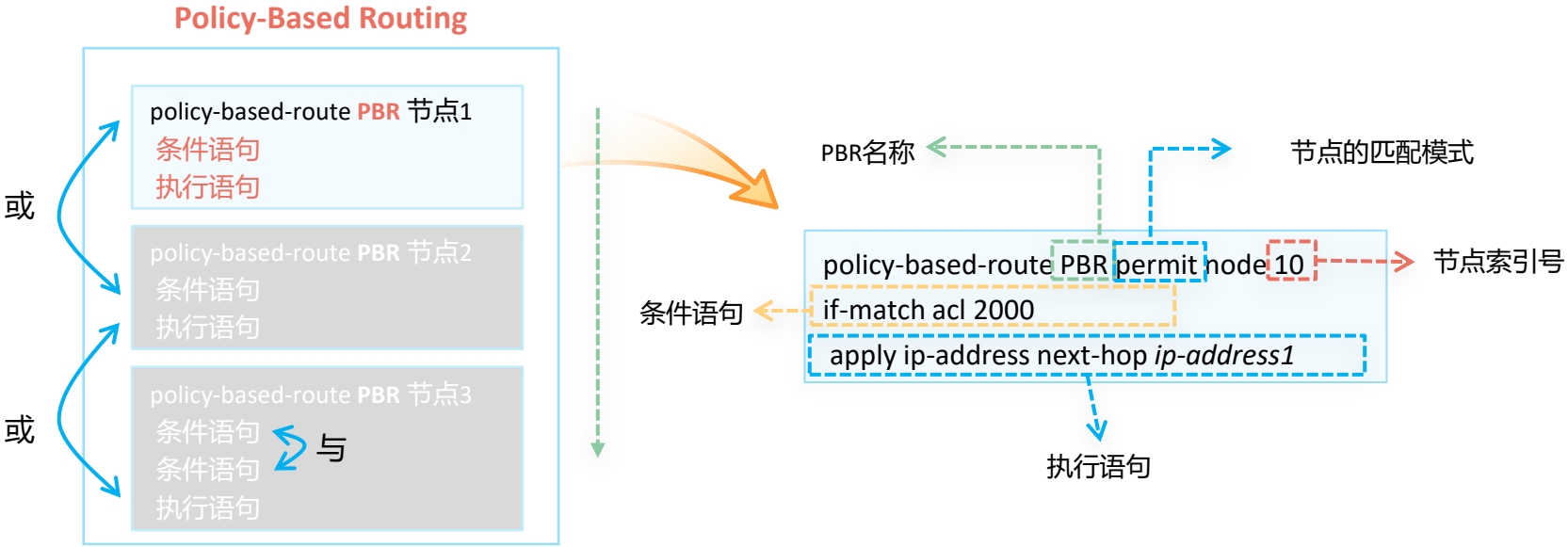
- PBR（Policy-Based Routing，策略路由）：PBR使得网络设备不仅能够基于报文的目的IP地址进行数据转发，更能基于其他元素进行数据转发，例如源IP地址、源MAC地址、目的MAC地址、源端口号、目的端口号、VLAN-ID等等。
- 用户还可以使用ACL匹配特定的报文，然后针对该ACL进行PBR部署。
- 若设备部署了PBR，则被匹配的报文优先根据PBR的策略进行转发，即PBR策略的优先级高于传统路由表。

PBR介绍 – 结构



- PBR与Route-Policy类似，由多个节点组成，每个节点由匹配条件（条件语句）和执行动作（执行语句）组成。
- 每个节点内可包含多个条件语句。
- 节点内的多个条件语句之间的关系为“与”，即匹配所有条件语句才会执行本节点内的动作。
- 节点之间的关系为“或”，PBR根据节点编号从小到大顺序执行，匹配当前节点将不会继续向下匹配。

PBR介绍 – 命令语法



PBR与路由策略区别

名称	操作对象	描述
路由策略（Route-Policy）	路由信息	路由策略是一套用于对路由信息进行过滤、属性设置等操作的方法，通过对路由的操作或控制，来影响数据报文的转发路径
PBR	数据报文	PBR直接对数据报文进行操作，通过多种手段匹配感兴趣的报文，然后执行丢弃或强制转发路径等操作

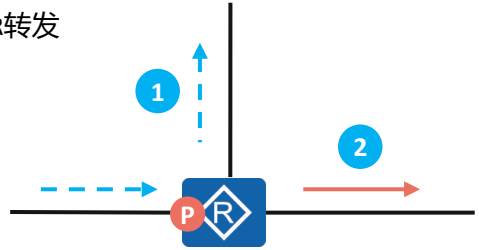
PBR的分类

接口PBR

P PBR执行点

依据路由表转发

PBR转发



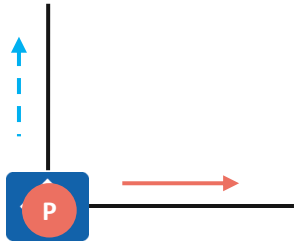
- 接口PBR只对转发的报文起作用，对本地始发的报文无效。
- 接口PBR调用在接口下，对接口的入方向报文生效。缺省情况下，设备按照路由表的下一跳进行报文转发，如果配置了接口PBR，则设备按照接口PBR指定的下一跳进行转发。

本地PBR

P PBR执行点

依据路由表转发

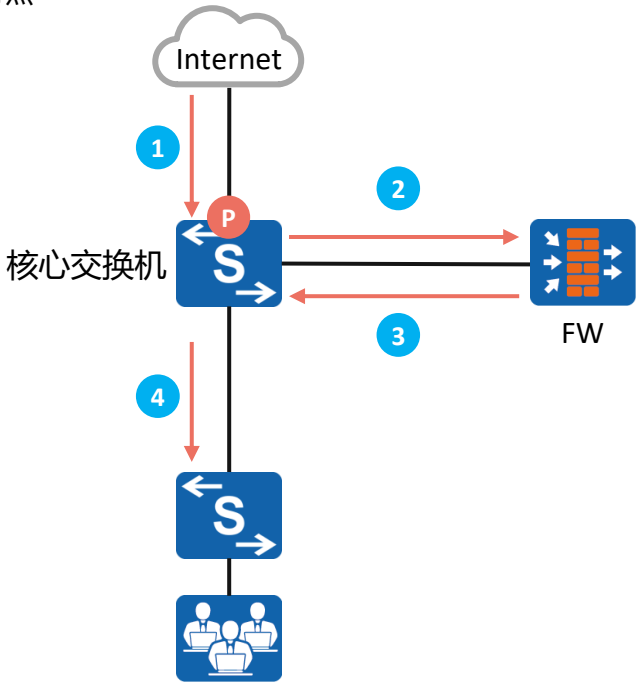
PBR转发



- 本地PBR对本地始发的流量生效，如：本地始发的ICMP报文。
- 本地PBR在系统视图调用。

PBR典型应用场景（1）

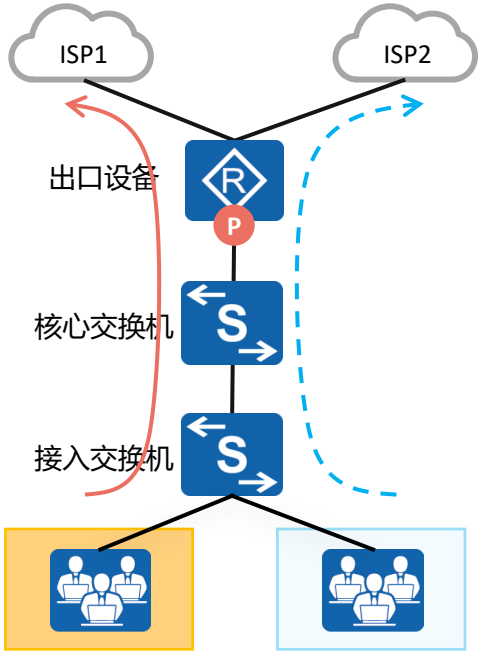
P PBR执行点



- 内网防火墙旁挂部署在核心交换机，为防护内网在核心交换机的三层接口上部署PBR，将来自外部网络的流量牵引到防火墙上进行安全检查，检查完的流量再发送回核心交换机，由核心交换机依据路由表转发到内网。
- 将流量牵引到别的设备进行安全检查等类似的行为我们称之为“引流”，PBR是一种常见的引流工具。

PBR典型应用场景（2）

P PBR执行点



当企业存在多个网络出口时，若想指定部分网段访问Internet时的网络出口，可以使用PBR：在出口设备的内网接口配置PBR，匹配来自内网的流量，为其指定不同的下一跳公网地址。

配置介绍（1）

1. 创建PBR

[Huawei] **policy-based-route** *policy-name* { **deny** | **permit** } **node** *node-id*

创建策略路由和策略点，若策略点已创建则进入本地策略路由视图。

2. 设置IP报文的匹配条件

[Huawei-policy-based-route-PBR-10] **if-match acl** *acl-number*
[Huawei-policy-based-route-PBR-10] **if-match packet-length** *min-length max-length*

缺省情况下，策略路由中未配置匹配条件，可以设置使用ACL匹配IP地址，也可以设置匹配报文长度。

3. 指定PBR中报文的出接口

[Huawei-policy-based-route-PBR-10] **apply output-interface** *interface-type interface-number*

缺省情况下，策略路由中未配置报文出接口。配置成功后，将匹配策略点的报文从指定出接口发送出去。
报文的出接口不能为以太接口等广播型接口。

配置介绍（2）

4. 设置PBR中报文的下一跳

[Huawei-policy-based-route-PBR-10] **apply ip-address next-hop** *ip-address1 [ip-address2]*

用户可以指定报文的下一跳。当该策略点未配置出接口时，匹配策略点的报文被发往指定的下一跳。

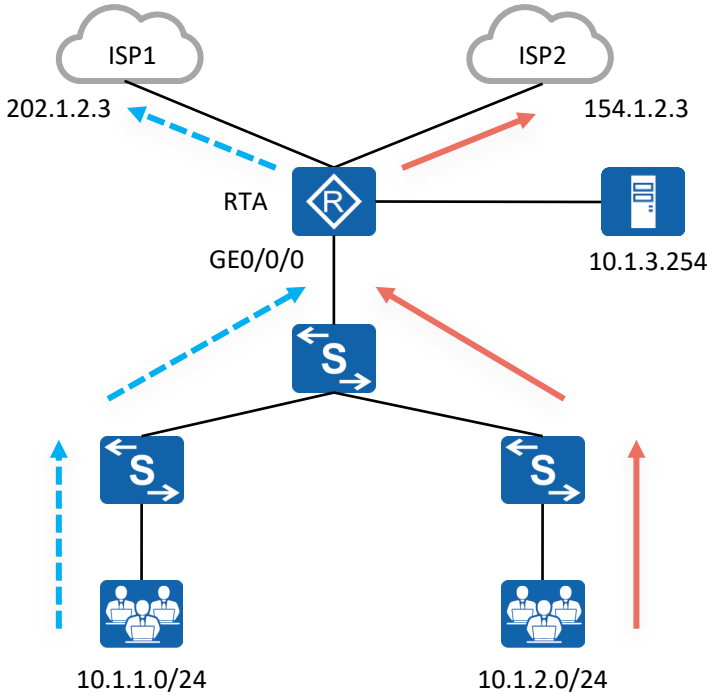
5. 全局PBR调用

[Huawei] **ip local policy-based-route** *Policy-name*

6. 接口PBR调用

[Huawei-GigabitEthernet0/0/0] **ip policy-based-route** *Policy-name*

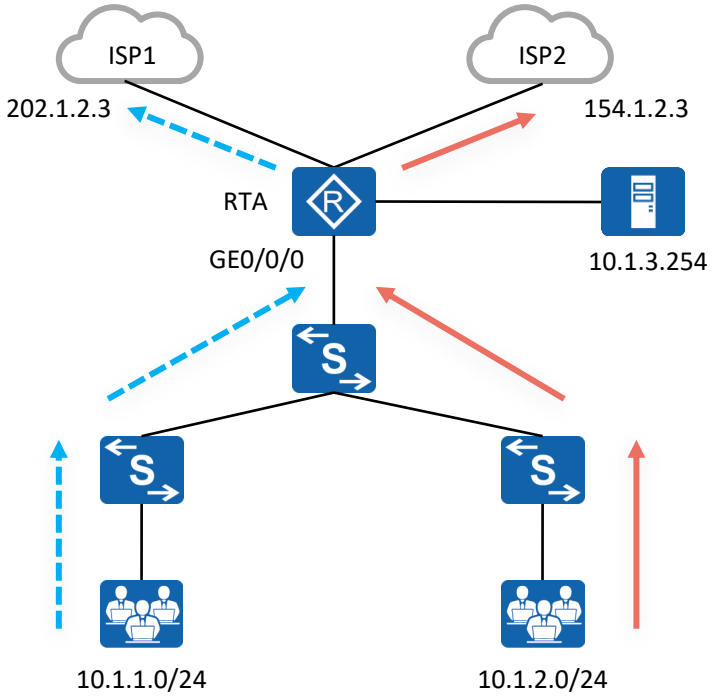
配置案例（1）



需求：

- 内网存在两个网段，网段1：10.1.1.0/24，网段2：10.1.2.0/24，在RTA的GE0/0/0接口部署PBR，实现网段1访问Internet通过ISP1、网段2访问Internet通过ISP2。
- RTA上旁挂了一台服务器，要求在RTA上部署的策略路由不影响内网用户访问该服务器。

配置案例（2）



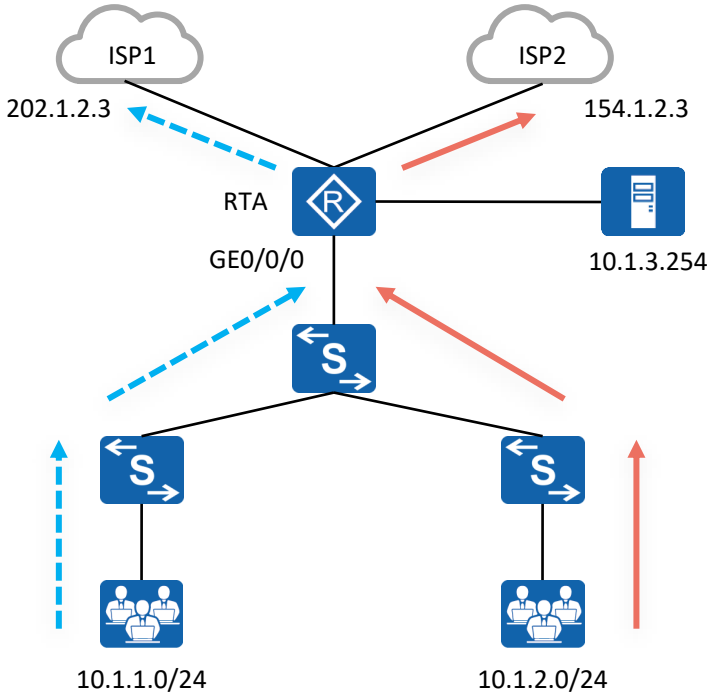
1. 配置ACL 3000，其中rule 1 deny网段1访问服务器的流量，rule 2匹配网段1访问Internet的流量。

```
[RTA] acl number 3000
[RTA-acl-adv-3000] rule 1 deny ip source 10.1.1.0 0.0.0.255 destination 10.1.3.254 0
[RTA-acl-adv-3000] rule 2 permit ip source 10.1.1.0 0.0.0.255 destination
0.0.0.0 0
```

2. 配置ACL 3001，其中rule 1 deny网段2访问服务器的流量，rule 2匹配网段2访问Internet的流量。

```
[RTA] acl number 3001
[RTA-acl-adv-3001] rule 1 deny ip source 10.1.2.0 0.0.0.255 destination 10.1.3.254 0
[RTA-acl-adv-3001] rule 2 permit ip source 10.1.2.0 0.0.0.255 destination
0.0.0.0 0
```

配置案例（3）



3. 创建PBR hcip，创建节点10，调用ACL 3000，指定其转发下一跳为202.1.2.3

```
[RTA] policy-based-route hcip permit node 10
[RTA-policy-based-route-hcip-10] if-match acl 3000
[RTA-policy-based-route-hcip-10] apply ip-address next-hop 202.1.2.3
```

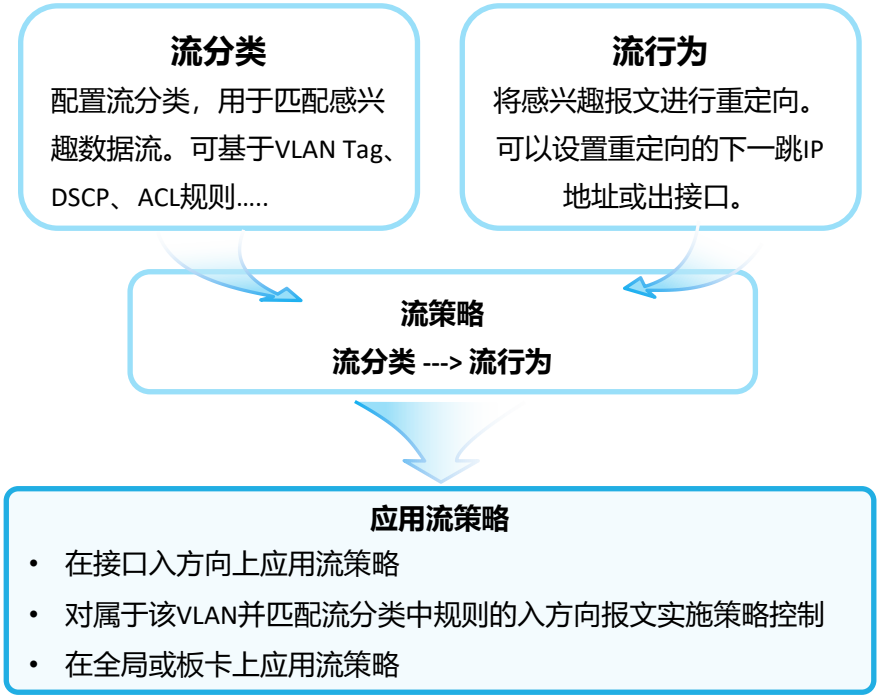
4. 创建PBR hcip节点20，调用ACL 3001，指向其转发下一跳为154.1.2.3

```
[RTA] policy-based-route hcip permit node 20
[RTA-policy-based-route-hcip-20] if-match acl 3001
[RTA-policy-based-route-hcip-20] apply ip-address next-hop 154.1.2.3
```

5. 在GE0/0/0接口调用PBR hcip

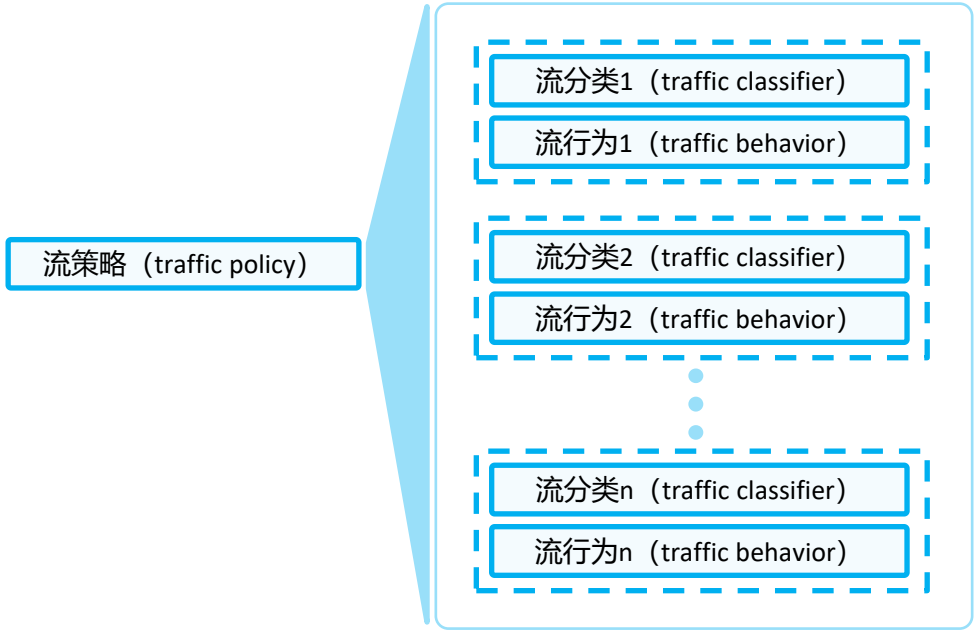
```
[RTA]interface GigabitEthernet 0/0/0
[RTA-GigabitEthernet0/0/0] ip policy-based-route hcip
```

MQC介绍（1）



- MQC（Modular QoS Command-Line Interface，模块化QoS命令行）是指通过将具有某类共同特征的数据流划分为一类，并为同一类数据流提供相同的服务，也可以对不同类的数据流提供不同的服务。
- MQC包含三个要素：流分类（traffic classifier）、流行为（traffic behavior）和流策略（traffic policy）。
- MQC的流行为支持重定向报文，因此可以使用MQC实现IP单播策略路由。

MQC介绍 (2)



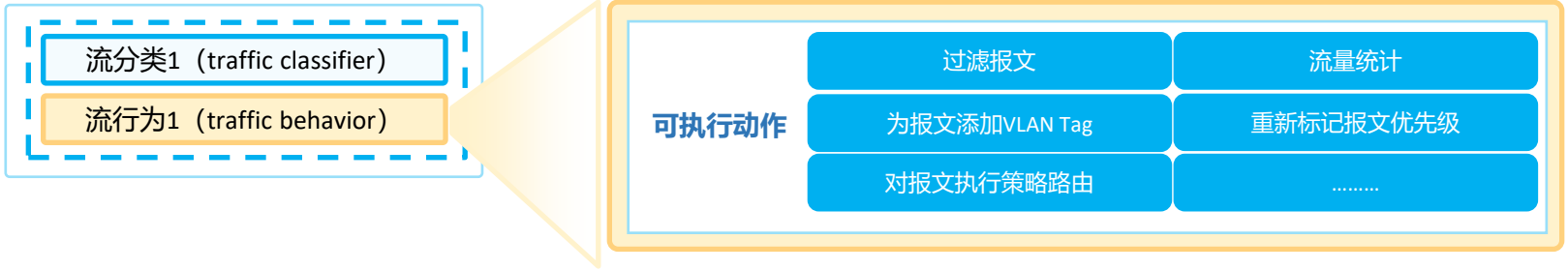
- 流策略：将流分类和流行为绑定，对分类后的报文执行对应流行为中定义的动作。
- 一个流策略可以绑定多个流分类和流行为。

MQC – 流分类

流分类：定义一组流量匹配规则，以对报文进行分类。流分类支持的匹配项如下所示。

MQC – 流行为

流行为：用来定义执行的动作，支持报文过滤、重标记优先级、重定向、流量统计等动作。

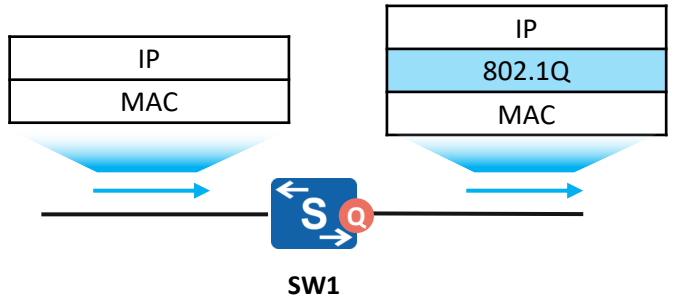


MQC – 流策略

- 流策略：流策略支持在接口上调用。
- 流策略存在方向(inbound、outbound)的概念，策略中的流行为匹配入、出方向的报文，对匹配中的报文执行相应的流动作。

流策略调用在接口出向

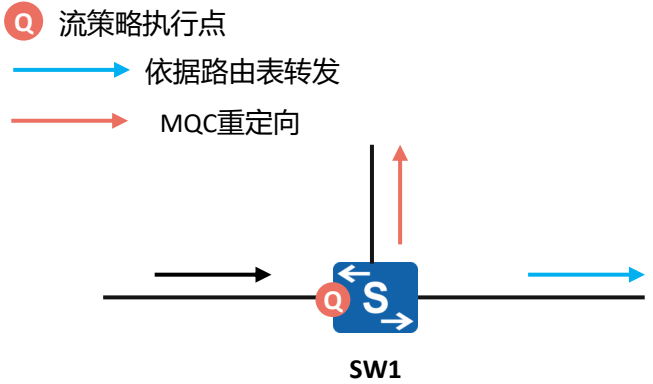
Q 流策略执行点



在接口出向使用流策略为报文打上802.1Q标签。

流策略调用在接口入向

Q 流策略执行点



在接口入向使用流策略重定向匹配中的报文

配置介绍

1. 创建流分类

[Huawei] **traffic classifier** *classifier-name* **[operator { and | or }]**

缺省情况下，流分类中各规则之间的关系为“或”（or）。流分类中的匹配规则配置可查阅产品手册。

2. 创建流行为

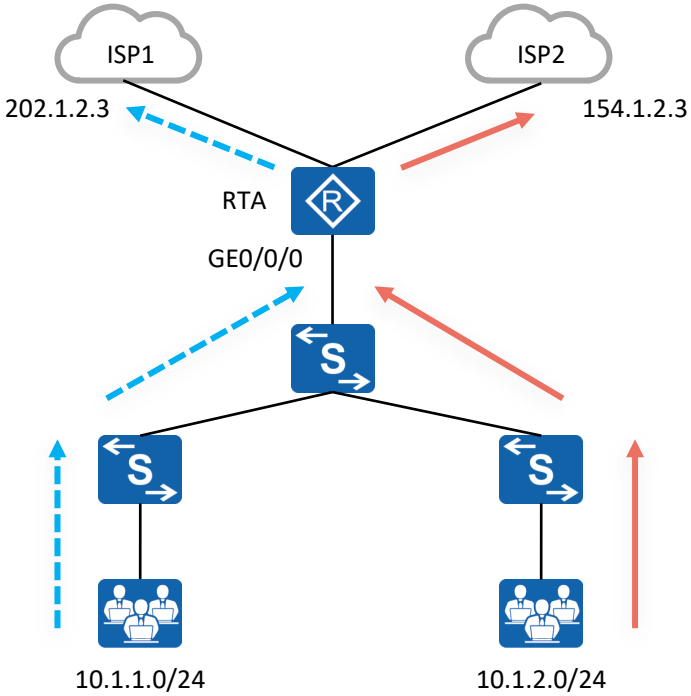
[Huawei] **traffic behavior** *behavior-name*

根据实际情况定义流行为中的动作，只要各动作不冲突，都可以在同一流行为中配置。流行为具体配置可查阅产品手册。

3. 创建流策略，并绑定流分类与流行为

[Huawei] **traffic policy** *policy-name*
[Huawei-trafficpolicy-policyname] **classifier** *classifier-name* **behavior** *behavior-name*

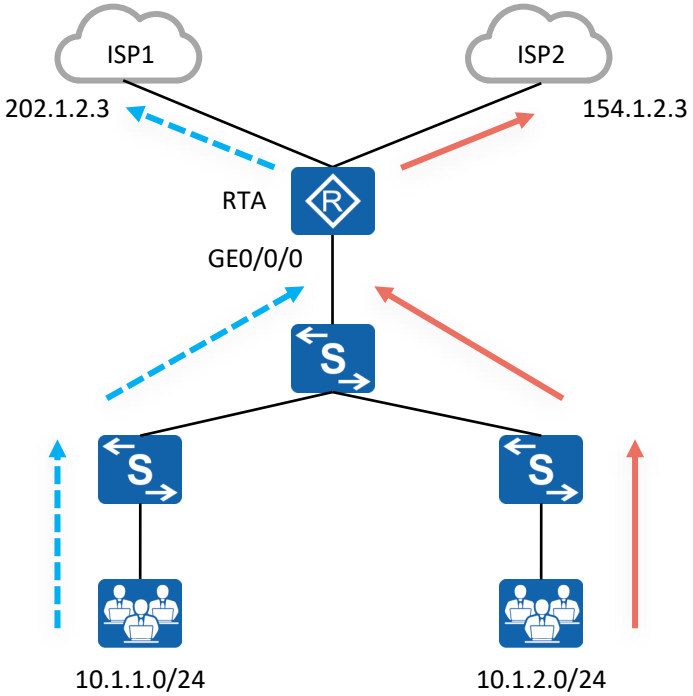
使用MQC实现策略路由 (1)



需求：

- 内网存在两个网段，网段1：10.1.1.0/24，网段2：10.1.2.0/24，在RTA上通过MQC实现策略路由，实现网段1访问Internet通过ISP1、网段2访问Internet通过ISP2。
- 将MQC调用在RTA的GE0/0/0接口

使用MQC实现策略路由 (2)



RTA的配置如下：

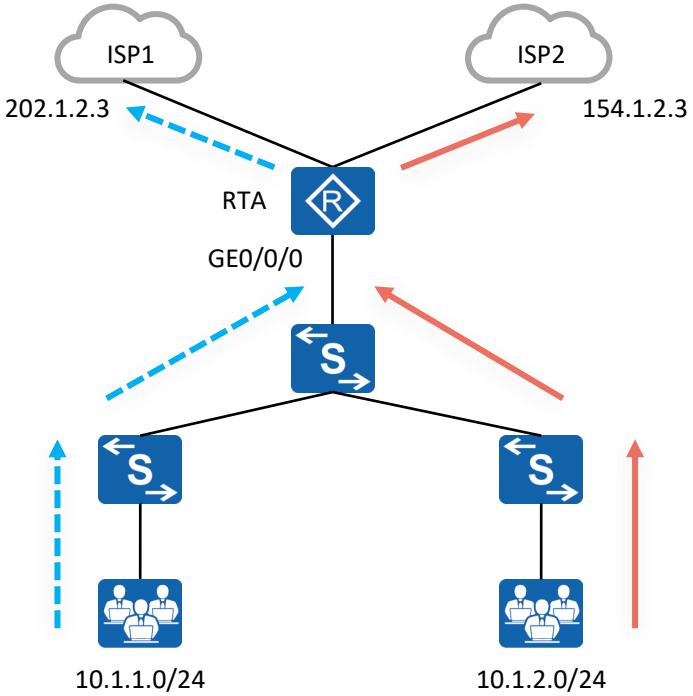
1. 配置ACL3000、3001分别匹配网段1、网段2访问Internet的流量。

```
[RTA] acl number 3000
[RTA-acl-adv-3000] rule 2 permit ip source 10.1.1.0 0.0.0.255 destination
0.0.0.0 0
[RTA] acl number 3001
[RTA-acl-adv-3001] rule 2 permit ip source 10.1.2.0 0.0.0.255 destination
0.0.0.0 0
```

2. 创建流分类1、2分别匹配ACL3000、ACL3001。

```
[RTA] traffic classifier 1
[RTA-classifier-1] if-match acl 3000
[RTA] traffic classifier 2
[RTA-classifier-2] if-match acl 3001
```

使用MQC实现策略路由 (3)



3. 创建流行为1、2分别执行将报文重定向到202.1.2.3、154.1.2.3的动作。

```
[RTA] traffic behavior 1
[RTA-behavior-1] redirect ip-nexthop 202.1.2.3
[RTA] traffic behavior 2
[RTA-behavior-2] redirect ip-nexthop 154.1.2.3
```

4. 创建流策略Redirect，将流分类1、2与流行为1、2一一绑定。

```
[RTA] traffic policy Redirect
[RTA-trafficpolicy-Redirect] classifier 1 behavior 1
[RTA-trafficpolicy-Redirect] classifier 2 behavior 2
```

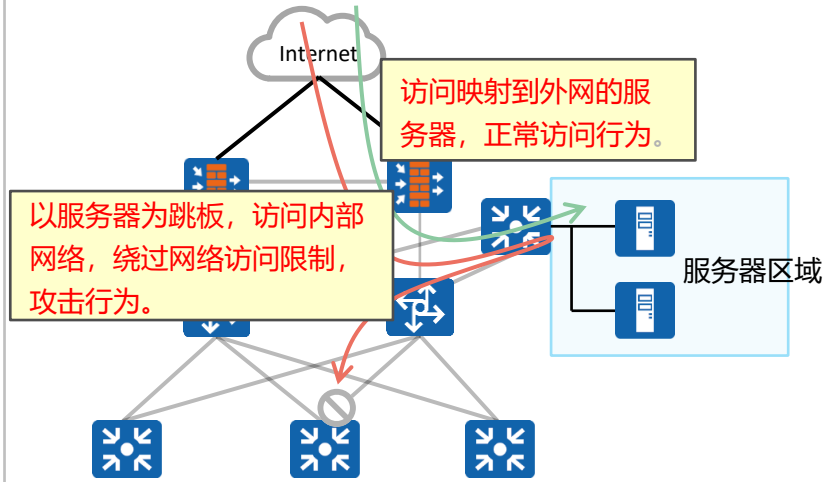
5. 在GE0/0/0接口入方向调用流策略Redirect

```
[RTA] interface GigabitEthernet 0/0/0
[RTA-GigabitEthernet0/0/0] traffic-policy Redirect inbound
```

需求背景

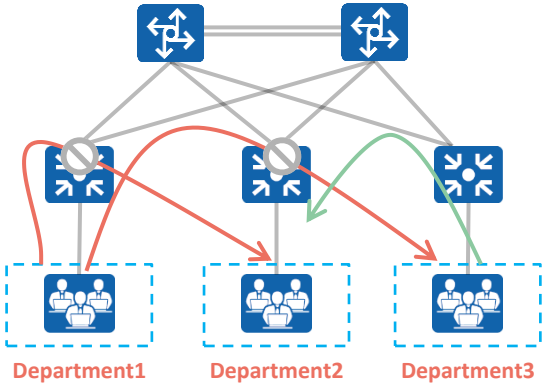
为提高网络安全性，管理人员需要控制进入网络的流量，将不信任的报文丢弃在网络边界。所谓的不信任报文是指对用户来说存在安全隐患或者不愿意接收的报文。同时保证数据访问安全性，企业网络中经常会要求一些部门之间不能相互访问。

丢弃不信任报文



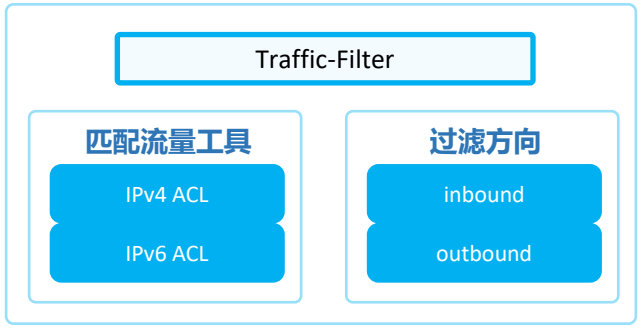
限制服务器访问内网，防止出现跳转攻击。

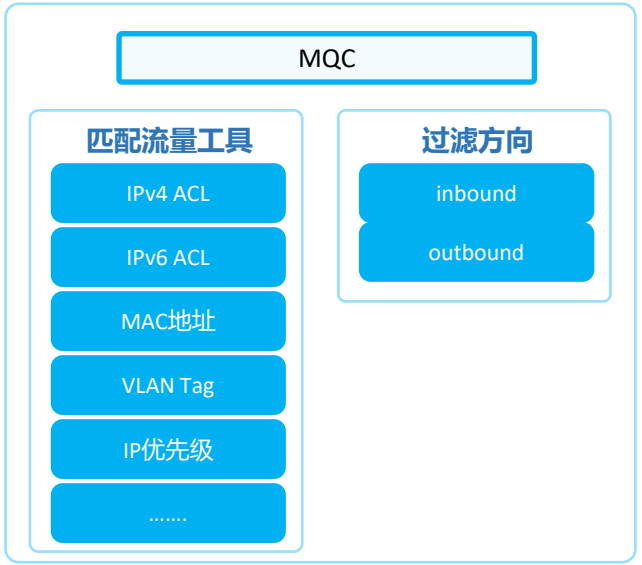
限制部门间相互访问



根据业务要求，限制不同部门之间的访问。

流量过滤工具



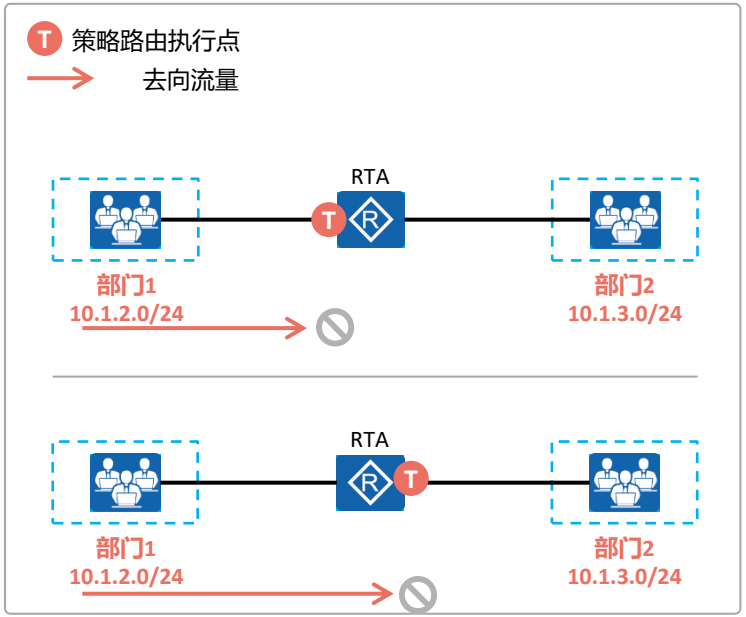


Traffic-Filter只能应用在接口视图下，而MQC可以调用在多种视图。

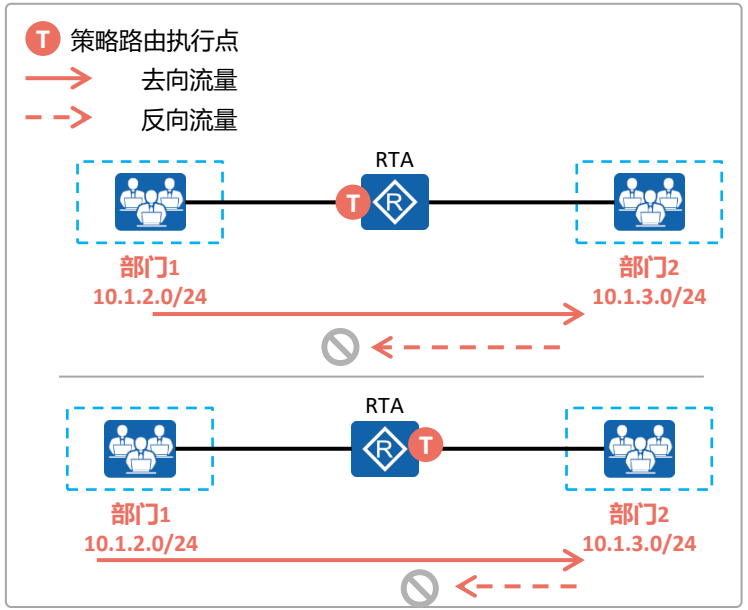
Traffic-Filter部署位置

使用Traffic-Filter过滤流量可以灵活地选择部署位置，在流量进入设备或者离开设备的接口上执行过滤动作，双向访问的业务禁止其中一个方向即可实现阻断业务的需求。

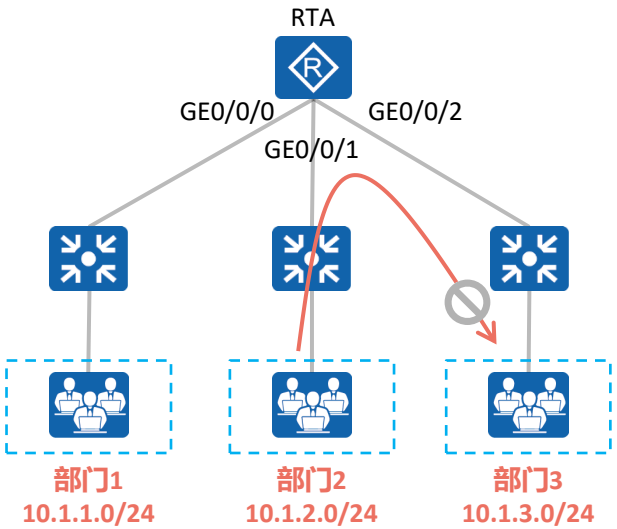
阻断部门1访问部门2的去向流量



阻断部门1访问部门2的回向流量



使用Traffic-Filter过滤流量



部门1、2、3的网关都在RTA上，现要求在RTA上使用Traffic-Filter限制部门2与部门3之间的相互访问。

RTA的配置如下：

1. 配置ACL拒绝部门2访问部门3，并放通其余所有流量。

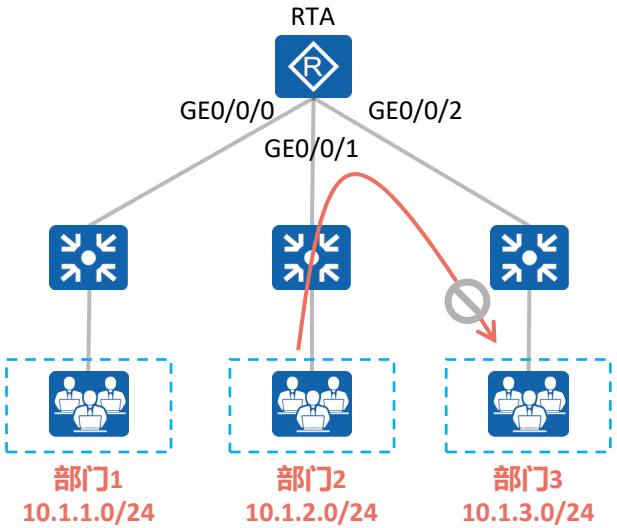
```
[RTA] acl number 3000
[RTA-acl-adv-3000] rule 1 deny ip source 10.1.2.0 0.0.0.255 destination
10.1.3.0 0.0.0.255
[RTA-acl-adv-3000] rule 2 permit ip
```

2. 在GE0/0/2接口调用Traffic-Filter

```
[RTA] interface GigabitEthernet 0/0/2
[RTA-GigabitEthernet0/0/2] traffic-filter outbound acl 3000
```

思考：ACL的写法、Traffic-Filter调用的接口是否还有别的可能?

使用MQC过滤流量 (1)



部门1、2、3的网关都在RTA上，现要求在RTA上使用Traffic-Filter限制部门2与部门3之间的相互访问。

RTA的配置如下：

1. 配置ACL匹配部门2访问部门3的流量

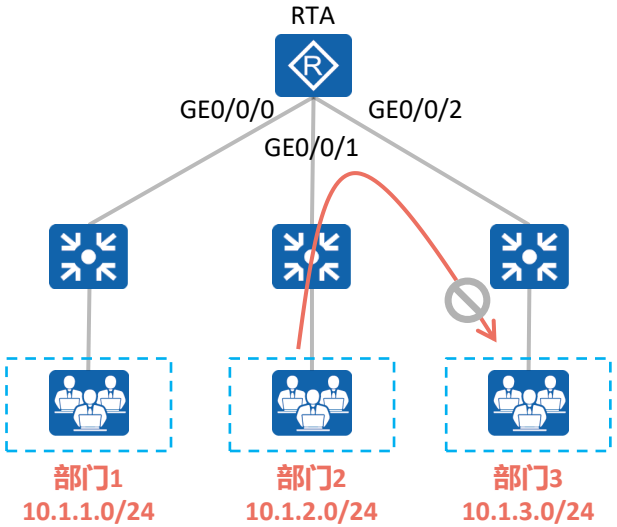
```
[RTA] acl number 3000
[RTA-acl-adv-3000] rule 1 permit ip source 10.1.2.0 0.0.0.255 destination
10.1.3.0 0.0.0.255
```

2. 创建流分类2_3、流行为2_3

```
[RTA] traffic classifier 2_3
[RTA-classifier-2_3] if-match acl 3000
[RTA] traffic behavior 2_3
[RTA-behavior-2_3] deny
```

为匹配ACL规则的报文指定报文过滤动作时，如果此ACL中的rule规则配置为permit，则设备对此报文采取的动作由流行为中配置的deny或permit决定；如果此ACL中的rule规则配置为deny，则无论流行为中配置了deny或permit，此报文都被丢弃。

使用MQC过滤流量（2）



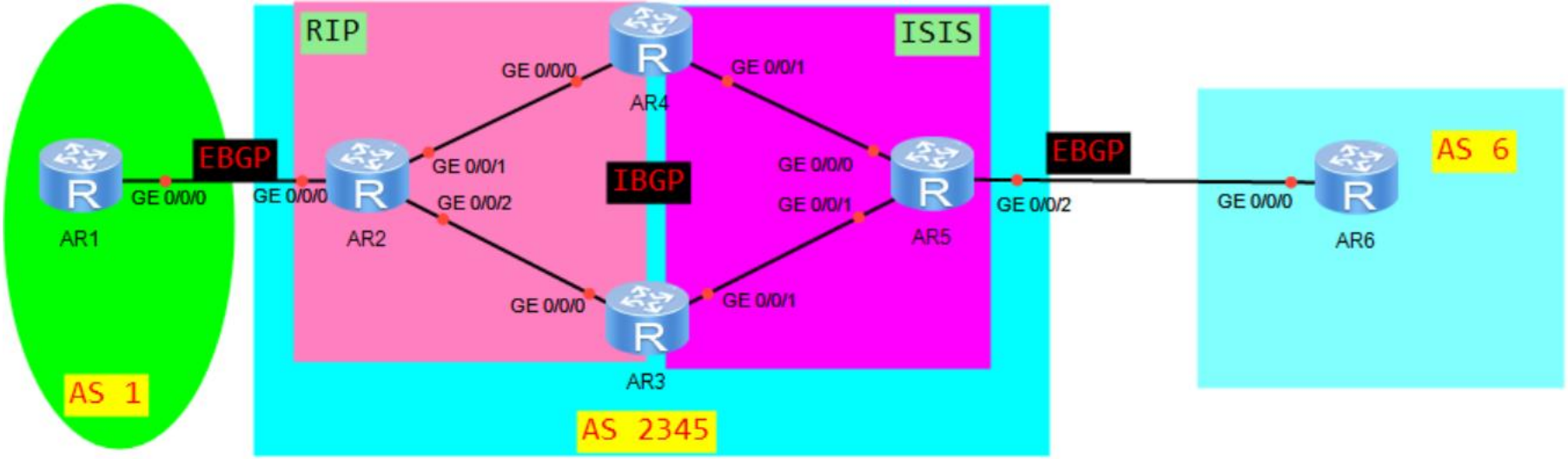
部门1、2、3的网关都在RTA上，现要求在RTA上使用Traffic-Filter限制部门2与部门3之间的相互访问。

3. 创建流策略，绑定流分类2_3与流行为2_3

```
[RTA] traffic policy 2_3
[RTA-trafficpolicy-2_3] classifier 2_3 behavior 2_3
```

4. 在接口GE0/0/1入向调用流策略2_3

```
[RTA] interface GigabitEthernet 0/0/1
[RTA-GigabitEthernet0/0/1] traffic-policy 2_3 inbound
```



实验需求：

1. R1创建4个环回口，模拟分支站点PC，宣告进BGP
2. 按照拓扑标识的路由和邻居运行相应的协议
3. R4和R3做双向双点重分布，解决次优化问题
4. R5上创建两个环回口，去往R1环回口一个从R4走，另外一个从R3走（华为）
5. R1上两个环回口去往R5环回口从R4走，另外两个从R3走（思科）
6. R6环回口到分支四个环回口的通讯Ping

THANK YOU

Ping 通您的梦想～

腾讯课堂交流群：17942636

ADD：苏州市干将东路666号和基广场401-402；Tel：0512-8188 8288；

课程咨询QQ：2853771087 ； 官网 :www.51glab.com