H3C

IToIP Solutions Expert

2008年第二季度　总第8期

H3C
IToIP Solutions Expert

contents
目录

2008年第二季度
总第8期

综述篇 1

BGP综述/2

基础篇 7

BGP基础/8

BGP属性简介/15

深入讨论 19

BGP FAQ/20

团体属性/44

BGP路由聚合/51

BGP路由过滤/62

RR、联盟及同步/71

BGP选路解析/91

BGP Graceful Restart/105

常用BGP AS_PATH正则表达式应用/111

MBGP扩展/116

网络应用 127

BGP网络性能优化浅析/128

BGP流量负载分担规划/131

测试方法 141

BGP测试工具及测试仪器介绍/142

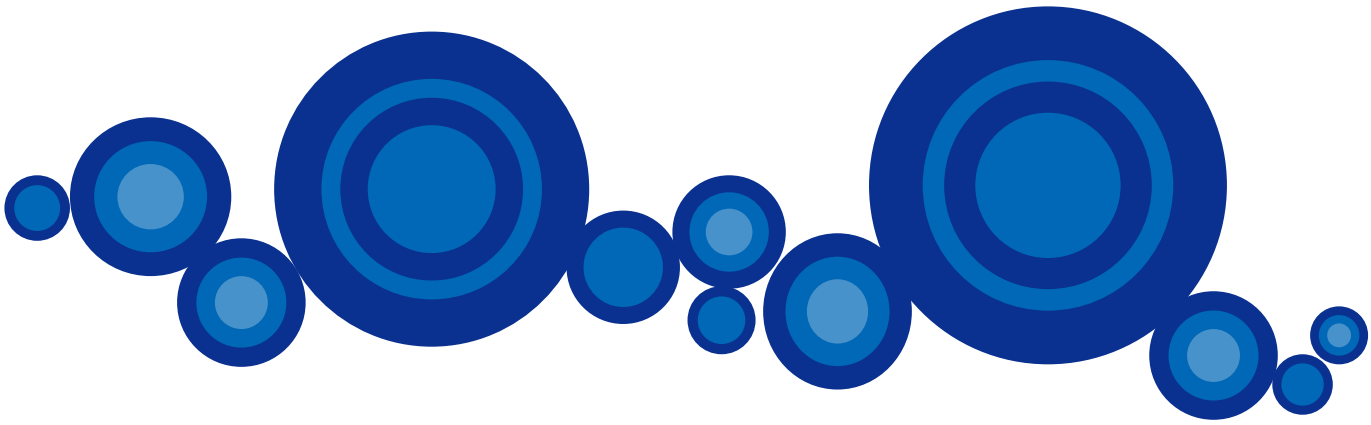
BGP性能测试方法/153

最新进展 165

BGP最新发展/166

BGP新特性/183

[综述篇]



BGP综述

文/朱皓

BGP的出现

要全面了解BGP，首先我们要回答以下看上去很简单的问题：为什么需要BGP，也就是说BGP是如何产生的，它解决了什么问题。带着以上问题，我们先简单的回顾一个路由协议发展的轨迹。

首先路由的实质是描述一个网络结构的表达方式，路由表其实是一个结果的集合。在早期的ARPANet网络时代，网络规模有限，路由数量也不大，因此所有的路由器可以维护整个网络拓扑，那时候使用的路由协议叫GGP（Gateway-to-Gateway Protocol）。GGP自然成为第一个内部网关协议（IGP）。在1980年左右。当时的网络管理者遇到了与今天类似的问题：网络规模扩大导致的路由数量不断增长。为了解决这种网络规模的增长问题，提出了自治系统的概念（AS），也可以叫做路由管理域。在AS内部使用一种路由协议，然后在AS之间使用另一种路由协议。这样做的好处显而易见，不同的网络可以自己选择IGP协议，然后再通过一个统一的AS间协议进行互连就可以了。

在IGP的发展领域中，先是RIP成为IP路由的主流，后期发展出更高级的IGP协议包括OSPF和ISIS，这些协议自动化程度更高、更智能更可靠。而IGRP和EIGRP是CISCO的私有协议，也是属于IGP的范畴。同一个AS的路由器间是有相互信任关系的，而且这些路由器往往由同样的管理人员维护，因此IGP的自动发现和路由计算信息泛洪处于完全开放的状态，人工干预的行为是比较少的。

不同AS互相连接的需求，推动产生了外部网关协议（EGP），EGP的主要目的是在不同的AS之间传递路由协议。而不同的AS之间往往是直接相连，大多数AS互联行为只涉及少量的边界路由器（ASBR），所以EGP的设计也非常简单。EGP的RFC827发布于1982年，看上去似

乎早于RIP的第一个标准FRC1058，但其实在RFC1058之前，RIP已经被广泛的使用。在当时，RIP+EGP成为一种标准的路由组合。

EGP被设计的如此简单，以至于很快就不能满足网络管理的要求。EGP单纯的发布网络可达信息，不做任何优选，也没有考虑环路避免。有人甚至认为EGP算不上是一个路由协议，EGP的众多缺陷，最终导致被BGP所取代。BGP的第一个FRC1105是1989年发布的，和EGP相比，BGP更像是一个路由协议，具有很多路由协议的特征，比如解决环路问题、收敛问题、触发更新等等。

就像是不同的企业有各自的企业文化和标准，但是企业间的交往却要遵循统一的行为规范和标准一样。对于AS间的路由交互，也必须有一个统一的标准。BGP相比EGP的众多优势，使BGP成为唯一的外部网关协议，并广泛的使用在互联网上。

综上所述，BGP是为了替代EGP而出现的一个外部网关协议，它必须能够进行路由优选、路由环路的避免、能够更高效率的传递路由和维护大量的路由。因为BGP部署在不具有完全信任关系的AS之间，因此需要BGP有丰富的路由控制能力，并且可以通过一些简单统一的方法对BGP进行扩展。

BGP的发展

BGPv1（RFC1105）定义了BGP最基础的一些协议特征。BGP在AS间传递路由，因此它非常重要。为了保证BGP的可靠传输，使用了TCP作为传输层协议。使用TCP的好处是显而易见的，BGP可以利用TCP现成的可靠性传输机制、重传、排序等机制来保证协议报文交互的可靠性。对于TCP扩展带来的好处也可以被继承，比如TCP的MD5认证就可以为BGP所用。

BGP是建立在两个不同的AS间，存在信任问题，所以BGP不能通过自动发现，而是需要手动配置邻居，使用指定地址建立TCP关系。与AS外部节点建立的BGP关系叫做EBGP关系，与AS内部节点建立的BGP关系是IBGP关系。

BGP最重要的一个概念就是使用AS号来解决AS间的环路问题，如果收到某个路由信息携带了自己的AS号，那么说明这个路由是已知路由，就不再处理它。如果AS号重复，那么说明出现了路由环路。在BGPv1中并没有AS-path的概念，这个概念在BGPv2中被明确下来。BGP从v1、v2、v3、到现在的v4，也是不断地进行改良。BGP4+则主要是进行了多协议BGP的扩展，也叫MP-BGP。关于MP-BGP的概念不是本季刊的讨论内容。

在AS内部，因为没有AS号的变化，防止环路需要采用其他的方法。BGP规定从IBGP邻居学习到的路由不会传递给另一个IBGP邻居，简单地说就是IBGP间路由只传一跳，路由只传递一次当然就不存在成环的问题。同时就要求AS内部的所有路由器都要两两建立IBGP关系，这就是BGP技术中的BGP全连接。全连接在大型网络中是不可想象的，因此后来衍生出路由反射器和BGP联盟两种技术（RFC1966和RFC1965）。路由反射器是在AS中指定一个节点作为反射器，所有的其他节点都与反射器建立IBGP关系，反射器作为一个中间节点，在其他任何两个IBGP间传递路由。所以反射器从理论上讲，在传递路由的时候，不应该改变路径属性信息，否则

就破坏了BGP在AS内部避免环路的原则。但是基于实际应用的角度，不同的厂商对反射器的功能做了很多特性，需要BGP的部署者谨慎使用。BGP联盟则是在AS内部做了重新规划，把一个扁平化的AS又分为多个私有的AS，这样做的好处一方面可以分层的管理一个庞大的AS，另一方面通过层次的划分，自然减少了全连接的需求。

BGP报文采用了TLV的结构，这种结构是非常利于扩展和向下兼容的。所以，随着网络的发展，产生了大量关于BGP扩展的RFC，这更使得BGP成为永葆青春的外部网关协议。

从最初的BGPv1到BGPv4，协议报文的种类和格式都做过调整，但是BGP的设计一直以简单明了作为原则。从BGPv2开始，消息种类确定为4种。建立TCP连接后，使用OPEN消息触发BGP关系建立过程，使用UPDATE消息进行路由的发布和撤销，使用NOTIFICATION消息通告出现错误，使用KEEPALIVE消息对BGP关系进行保活。可见BGP的路由通告是触发更新模式的，只有更新的时候才发送UPDATE，所以，需要KEEPALIVE消息对BGP关系进行保活。BGP状态机也是在BGPv2开始被确定为6种。

1990年出现的BGPv2（RFC1163）是一个重要的分水岭。首次出现了BGP路径属性的概念，而且其中对属性的分类方法沿用至今，成为BGP路由策略的主要手段，为各种对路由的过滤、标识、选择提供了多种多样的方法。

路径属性分为4种，最基本的就是公认必遵属性。顾名思义这类属性必须在发布路由的时候携带，描述了所发布路由的一些基本信息，包括：下一跳、AS_PATH和ORIGIN。下一跳用于路由计算、AS_PATH用于环路避免，ORIGIN则用于路由选择。在BGP中，路由信息通过网络前缀的方式进行描述，被叫做网络层可达信息（NLRI，本文后面有些地方使用了‘BGP路由’这个通俗的叫法，以方便理解）。NLRI这个描述更贴切一些，实际上BGP只传递了一些信息，用于计算出路由，所以NLRI必然是存在于UPDATE消息中。在UPDATE消息中，路由信息通过路径属性+NLRI的方式表达出来。既然NLRI必须附加公认必遵的3种属性，那么一个UPDATE只传递相同路径属性的NLRI信息，这样实现就比较简单。

公认必遵属性相对应的就是公认可选，也就是说这些属性必须被所有的BGP路由器所识别，携带与否是可以选择的。其他两种可选可传递属性和可选不传递属性，很显然，两种属性考虑到了协议的扩展性，对于设备不识别的属性，是可以透传或者忽略的。

各种各样的属性，一方面用于路由选择，另一方面相当于给路由做了标志，在不同的节点，根据这些标志对路由做相应的过滤、修改等操作。也可以根据属性来实现一些BGP的扩展特性。正是因为BGP使用在不完全可信的路由管理域之间，所以需要BGP具有对路由信息灵活的控制手段，这是BGP最重要的特点之一。

网络设备根据NLRI中的网络前缀和对应的路径属性，进行路由计算，计算有可能需要依赖IGP来完成。因为BGP关系可以建立在非直连的网络节点，只要建立TCP连接的地址通过IGP可达，就可以建立BGP关系并交互NLRI信息。另外，NLRI对应的路径属性中携带了下一跳的信息，下一跳也要通过IGP进行查找，如果找不到到达下一跳的路由，那么就说明无法到达发布NLRI的BGP节点，因此该BGP路由处于inactive状态。很显然，inactive的路由信息是不应该发送给其他任何一个对等体的。所以有必要在全局路由表之外，保留一个BGP路由表，通过这

个表，可以看到BGP路由决策的部分结果。

在BGP实际的使用中，还有一个同步的概念，也就是IGP路由必须与BGP路由同步。前面已经提到BGP关系是可以在非直连邻居间建立的，路由信息可以在BGP对等体之间传递，但是没有配置BGP的中间链路节点，并没有这些BGP路由信息。路由归根结底是为转发报文而服务的，当报文转发到这些中间链路节点时，会因为没有路由而被丢弃。这个现象很形象的被称为BGP黑洞。解决BGP黑洞其实很简单，就是保证如果某个节点没有配置BGP的话，必须可以通过IGP获得这些BGP路由信息，这就叫IGP路由与BGP路由同步。在真实的网络设计中BGP黑洞是很少出现的，因为如果AS是一个边缘AS，那么BGP多数只部署在AS连接其他AS的边界上。如果AS是位于多个AS中间的区域，那么这个AS是一个核心区域，其中所有的路由器都会部署BGP并且配置了FULL-MASH的IBGP关系。所以BGP黑洞是很少出现的特例，因此当前大多数厂商的实现都是默认关闭了这种同步的检查。

1991年RFC1267定义了BGPv3，一个重要的补充是增加了连接的冲突处理机制，当两个节点同时发起连接时，BGP ID大的一方发起的连接会被保留。

BGPv4（RFC1771）最重要的改变是BGP终于由有类的路由协议成为一个无类的路由协议。这个改变源于有类地址的枯竭，为解决这个问题，1993发布的RFC1520定义了CIDR（Classless Inter-Domain Routing）。而BGP作为唯一的AS间路由协议，支持CIDR是必然的。

最新关于BGP4的RFC是4271，其中对于一些细节进行了进一步的说明，比如对NEXT-HOP属性的处理原则、事件和状态机以及BGP的路由决策流程等等。4271相比于1771的变化还是比较多的，详细情况可以参看RFC4271的附录A。

BGP协议基本概念并不复杂，如果从BGP的使用场景和使用特点来看，理解起来很容易。BGP的困难主要在于部署时如何适应网络拓扑的要求，和对路由进行控制，因为BGP控制路由的手段非常多，而且这些控制都是需要管理员定义和部署。各种特性和控制手段组合在一起使用时，会互相影响。

BGP的扩展

前面提到了BGP的一些基本概念：基于TCP的可靠连接、触发更新、AS内和AS间防止环路的技术，通过各种属性实现的路由优选和路由控制策略、消息的类型等等。可以看出BGP的各种特征，都是来源于BGP使用的场合，需求决定了最终的实现，协议不过是统一了实现的方法而已。

关于BGP扩展的RFC非常多，多数都是实现了不同的特性，也有一些是对BGP的分析，甚至是BGP部署方面的建议和经验。下面列出的只是关于BGP的小部分RFC而已。

前面提到的BGP联盟出现在RFC1965（最新版为RFC5065），路由反射出现在RFC1966（最新版为RFC4456）。

BGP路由扩展团体属性出现在RFC1997（最新版本为RFC4360）。

路由刷新功能出现在RFC2918，路由刷新定义了一种新的BGP消息Route-REFRESH，使用这个消息可以要求对等体更新某个地址族的路由信息。

RFC2439定义BGP路由惩罚机制解决了路由不稳定对网络造成的影响。惩罚机制是通过两个定时器来实现的，每次振荡会导致一个惩罚值的累加。如果超过某一个固定值，该路由就不计算不发布，处于抑制状态。

RFC2842定义了BGP携带能力集的方式（最新版为RFC3392），通过在OPEN消息中的能力集字段，可以在BGP建链阶段完成双方能力的通告，并决定是否建立BGP关系和后续协议报文的处理。

RFC2858（最新版为RFC4760）定义了多协议的BGP，扩展以支持非IPv4的网络层可达信息。目前最重要的MPLS VPN技术就是通过BGP的多协议扩展，实现了路由交互。

RFC2385定义了使用TCP的MD5保护BGP连接的方法，而为了加强key的处理，由RFC3562又做了进一步的分析。

RFC4724定义了BGP GR，在双主控和控制转发分离的设备上实现协议的平滑重启，可以保证转发不中断。RFC4781进一步定义了MPLS环境中的BGP GR。

RFC1772描述了BGP4在Internet上的使用方法，包括给出了关于拓扑的建议和一些路由处理的过程。

RFC4272是研究BGP安全性的一个分析文档。

RFC4451甚至对MED属性进行了详细的分析，给出一些实现的建议。

……

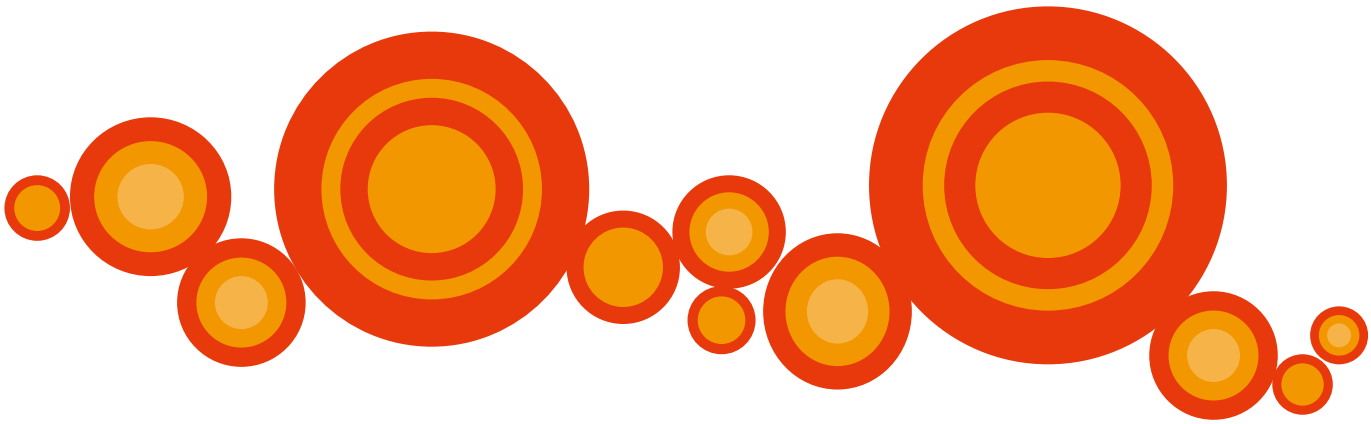
对BGP的诸多扩展是为了适应不断发展的网络结构，包括网络攻击的可能性。由这些RFC支撑着的BGP毫无疑问已经成为Internet上无法替代的一个路由协议，而且有越来越复杂的趋势。笔者认为对于BGP的理解应该是基于应用的，从上面的一些RFC就可以看出来，BGP的协议文档中很多是来源于应用的实践。

在今天的BGP技术中，在NLRI上附加了很多增量的信息，用于实现各种差异化的需求。好在BGP的协议基本构架决定了BGP是一个拥有良好扩展性和兼容性的路由协议。而且BGP技术可以说是一种模块化的技术，在一个基本的协议构架上，可以通过各种扩展增加模块以支持各种新的应用，而且有可能允许不同组件的共存。在最新的草案中，有一个是关于BGP能力集的动态发布。允许在一个已经建立的BGP关系上通过OPEN消息宣称自己支持新的能力集，并且如果不支持该能力集，可以忽略这个消息。并不会影响原有的BGP关系或路由通告。

对BGP的理解过程是一个渐进的过程，BGP的发展也是如此。还是套用刘宇在IP路由技术胶片最后写的那句话做结尾：

Routing is like a box of chocolate，you'd never know what you are going to get...

[基础篇]



BGP基础

文/叶翀

概述

在开始之前，假设让我们来给通信协议画一个简单的素描。

抛开对上层协议和复杂应用的支持不谈，通信协议最基本的功能是运行在两台或多台设备之间，通过收集、发布、交换一些信息，来为设备间的通信建立通道，即实现支撑上层数据互通。为了实现通信协议这个基本功能，需要解决三个问题：

- 单台设备需要收集和存储哪些信息？
- 设备之间如何交互和通信，来汇总出完整的信息？
- 信息汇总之后，进行怎样的决策，来得出通信通道？

对如上三个问题进一步分解，可以得出：

一、协议发布哪些信息？

- 协议需要哪些信息？
- 如何描述和存储这些信息？

二、与谁交换信息？

- 选择哪一个底层协议作为承载进行通信交互？
- 动态发现交换对象——邻居，还是静态配置邻居？
- 邻居建立过程是怎样的？

三、如何交换信息？

- 交换信息报文格式如何？
- 报文格式怎么做可以具备比较好的扩展性？

四、如何从信息中决策出最佳通信通道？

五、其他必要的考虑

通过这些问题，我们将在下文描述BGP的简单框架。当然这些仅仅是框架，它们只是BGP协议最基础的部分。BGP最强大的功能在于灵活的策略和多协议扩展支持，这些在本文中都没有涉及，在本刊的其他文章中都会有详尽的介绍。

信息收集和存储

在BGP的前身EGP中，引入一个概念，叫做自治系统（AS，Autonomous System）。AS指的是在统一技术管理下的一系列路由器。比如运行OSPF的一张网络，或者运行ISIS的一张网络，都可以作为一个AS。实际网络中，自治系统是人工规划的，AS号由专门的机构分配。通常在AS内部运行某种IGP协议，用于AS内部的路由学习和管理；而在AS的边界运行BGP，用于AS之间交换路由信息。借助BGP，各AS可以独立选择自己适合的IGP协议，并通过BGP来获得其它AS的路由信息。

从这个用途来看，对BGP来说，需要做到以下几点：

- 能够支持从各类IGP（包括直连路由）引入路由信息
- 能够从这些数据中决策出最优路由
- 不论从哪类IGP引入，将最优路由对外发布时，采用统一的格式

路由信息的引入过程不在本文的讨论范围内，决策过程见本文第5节。我们来看看上面三点中信息的收集和存储问题。BGP需要收集哪些信息，来支撑下一步的决策？

最基本的是IP前缀和掩码、下一跳，这是一条路由的最简描述，任何一种路由协议都需要收集这些信息。为了支持路由优选，需要考虑路由的优先级（至少一种度量），并记录路由的来源（哪个AS发布，什么方式引入），这是我们需要收集的可供下一步决策的最小信息集合。后面我们会看到，这些其实就是BGP UPDATE报文中的主要字段。IP前缀和掩码对应UPDATE中的NLRI，下一跳对应NEXT_HOP，优先级对应LOCAL_PREF和MED，路由来源对应AS_PATH和ORIGIN。

BGP存储路由信息的数据库叫做RIB，Routing Information Base。这个数据库分为三个部分：

- Adj-RIBs-In，保存BGP Speaker从邻居学到的路由信息，即初始路由
- Loc-RIB，保存经过决策从Adj-RIBs-In选取的路由信息，即最优路由
- Adj-RIBs-Out，保存BGP Speaker发给邻居的路由信息，即发布路由

上面三个数据库，仅仅是协议关于BGP路由管理的概念性设想，实际实现中，不要求必须保留路由的三套拷贝。

与谁交换信息

初步设想了单台设备需要收集和存储哪些信息之后，我们来看看设备之间如何通信。考虑到BGP用在AS之间，是作为大型网络之间接口的角色存在，对报文传输的稳定性有很高的要求，BGP选择了TCP作为承载协议，使用端口号179。由于TCP提供了稳定可靠的传输，BGP不需要专门的机制来处理复杂的报文分片、重传、确认等细节。

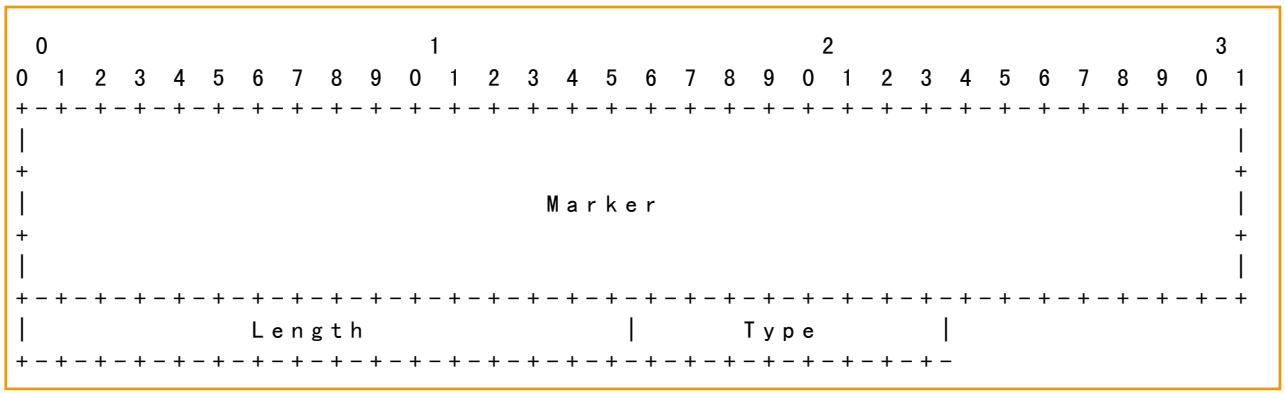


图1 BGP报文头

在TCP之上，BGP报文头如图1所示，共19个字节。

承载不同BGP协议报文类型通过Type字段标识，Length字段标识BGP消息的字节长度，含BGP报文头，虽然是16Bit数，合法范围只从19到4096。Marker字段用来探测对端与本端是否同步。

承载协议确立为TCP之后，下一个问题是，采用普通路由协议的动态发现邻居方式呢，还是采用手工静态配置方式？BGP采用了后者，只要双方指定地址路由可达，就可以建立邻居。这么做至少有两个好处：

1、可以与对端设备用任何IP地址建立邻居，而不限于某个固定的接口IP。这样，当两台设备采用环回地址而非直连地址建立BGP邻居时，即使主链路中断了，也可以切换到备份链路上，保持邻居不断。这种稳定性正是BGP作为大型网络路由承载的必要特质。

2、可以跨越多台设备建立邻居。当一个AS有多个设备运行BGP 建立域内全连接时，不必每台设备物理直连，只要用IGP保证建立邻居的地址可达，即可建立全网连接，减少不必要的链路建设。

同一个AS内，设备之间的邻居叫做IBGP（Interior BGP）邻居，不同AS间，设备之间的邻居叫做EBGP（Exterior BGP）邻居。运行BGP的设备叫做BGP发言者（BGP Speaker），相互之间称作BGP对等体（BGP peer）。

BGP用来建立邻居的OPEN消息格式如图2所示。

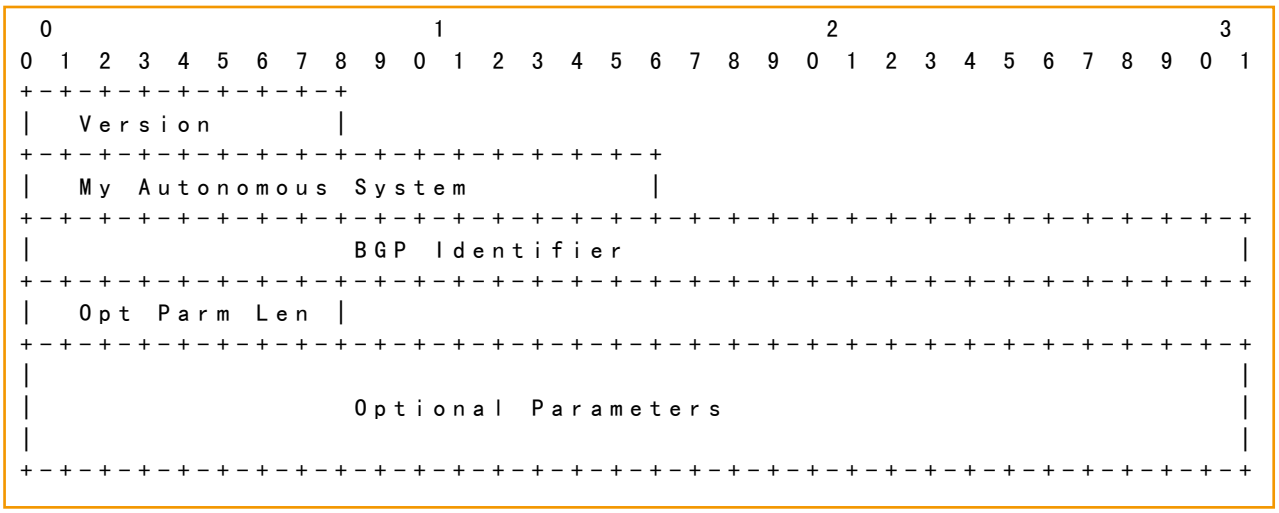


图2 OPEN消息

Version：标识运行的BGP版本。如果一个对等体的版本比对方旧，它会拒绝新版本的连接，于是对方降低版本号重新进行协商，直到双方对版本达成一致为止。

My AS：邻居建立发起者的AS号。用来决定双方是IBGP邻居，还是EBGP邻居。

Hold Time：对等体通过定期发送KEEPALIVE消息通知对端本端还在，以保持邻居。由于KEEPALIVE纯粹是一个通信知会，不需要携带什么信息，因此KEEPALIVE报文实际上是不带数据的BGP报文头。Hold Time是设备收到一个KEEPALIVE之前允许经过的最大秒数。这个时间或者是0秒（不发送KEEPALIVE），或者是至少3秒。一般默认KEEPALIVE每60秒发送一次，Hold Time为180秒。协商时，采用OPEN消息中较小的那个Hold Time作为双方的Hold Time。

BGP Identifier：用来标识邻居的IP地址。

Optional Parameters：公布对一些可选功能的支持，如认证、多协议支持等等。

建立邻居时，BGP先尝试与对等体建立一个TCP连接。如果TCP连接建立成功，BGP发送一个OPEN消息给对端，并等待从对端发来的OPEN消息。收到一个OPEN消息以后，BGP检查该消息的所有字段，如果没有发现错误，则向对端发送一个KEEPALIVE消息并启动KEEPALIVE定时器。收到KEEPALIVE消息，则邻居建立。

当邻居检测到错误需要中断连接时，BGP发送NOTIFICATION消息通知对端，消息格式如图3所示：

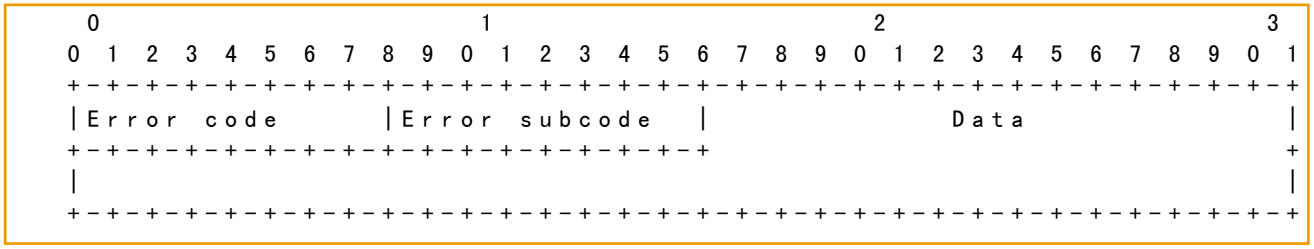


图3 NOTIFICATION消息

相关错误码和含义的列表请参考协议，这里不展开讨论。

综上我们可以得出BGP四种消息报文的用途：OPEN用来建立邻居，KEEPALIVE维持邻居，UPDATE发布路由信息，NOTIFICATION通知对端检测到错误。

BGP建立邻居采用有限状态机，共有6种状态。BGP的运行流程就是在这6种状态之间根据资源和事件的要求作转换。它们分别是：

1. Idle

BGP协议初始时是处于Idle状态。在这个状态时，系统不分配任何资源，也拒绝所有进入的BGP连接。只有收到Start Event时，才分配BGP资源，启动ConnectRetry计时器，启动对其它BGP对等体的传输层连接，同时也侦听是否有来自其它对等体的连接请求。

2. Connect

这个状态下，BGP等待TCP完成连接。若连接成功，本地清空ConnectRetry计时器，并向对等体发送OPEN报文，然后状态改变为OpenSent状态；否则，本地重置ConnectRetry计时器，侦听是否有对等体启动连接，并移至Active状态。

3. Active

这个状态下，BGP初始化TCP连接来获得一个对等体。如果连接成功，本地清空ConnectRetry计时器，并向对等体发送OPEN报文，并转至OpenSent状态。

4. OpenSent

这个状态下，BGP等待对等体的OPEN报文。收到报文后对报文进行检查，如果发现错误，本地发送NOTIFICATION报文给对等体，并改变状态为IDLE。如果报文正确，BGP发送KEEPALIVE报文，并转至OpenConfirm状态。

5. OpenConfirm

这个状态下，BGP等待KEEPALIVE或NOTIFICATION报文。如果收到KEEPALIVE报文，则进入Established状态，如果收到NOTIFICATION报文，则变为Idle状态。

6. Established

这个状态下， BGP可以和其他对等体交换UPDATE，NOTIFICATION，KEEPALIVE报文。如果收到了正确的UPDATE或KEEPALIVE报文，就认为对端处于正常运行状态，本地重置Hold Timer。如果收到NOTIFICATION报文，本地转到Idle状态。如果收到错误的UPDATE报文，本地发送NOTIFICATION报文通知对端，并改变本地状态为Idle。如果收到了TCP拆链通知，本地关闭BGP连接，并回到Idle状态。

综上，我们可以画出BGP的有限状态机如图4所示：

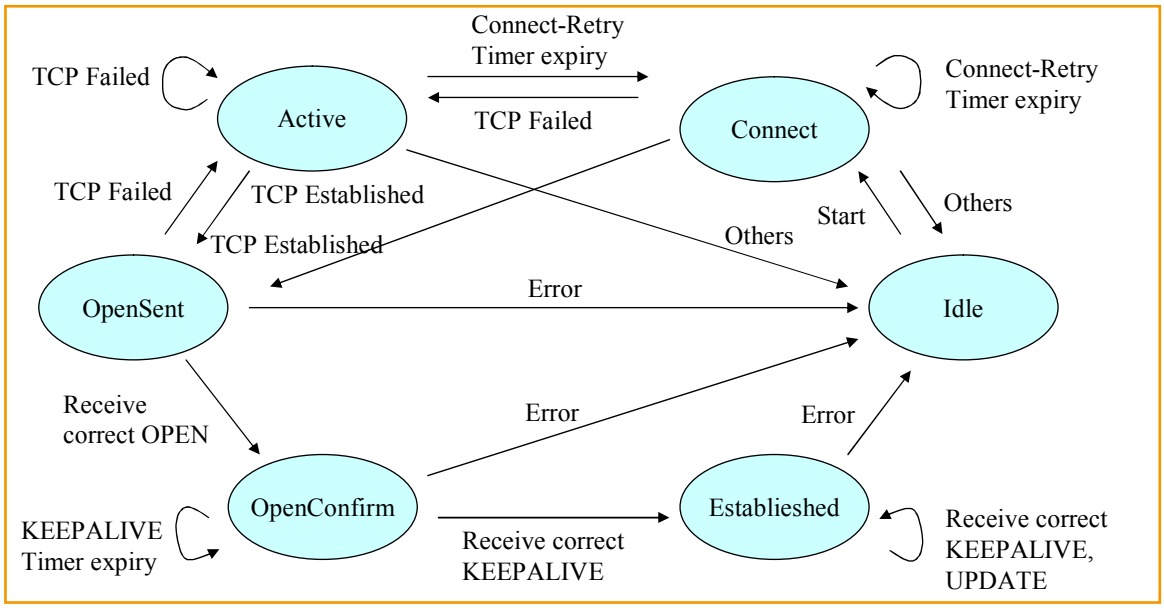


图4 BGP有限状态机

如何交换信息

邻居建立后，BGP采用UPDATE消息来发布路由或撤销路由。UPDATE消息由三部分组成：

Unfeasible Routes：之前发布过，不再有效的路由。

Path Attributes：路由信息的附加描述，是BGP用以进行路由控制和决策的重要信息。

NLRI：由一个或多个IP地址/前缀长度组成。

其格式如图5所示，一个UPDATE消息中可以携带多条路由信息：

由于BGP在报文格式中普遍采用了TLV（Type，Length，Value）的形式，而不是固定长度固定字段的形式，使得BGP具有非常好的扩展性，在后期追加新类型支持新业务时，只需要定义新的类型编码和值，报文不需要做任何更改。

关于BGP属性的具体类型和含义，请参考《BGP属性简介》一文。

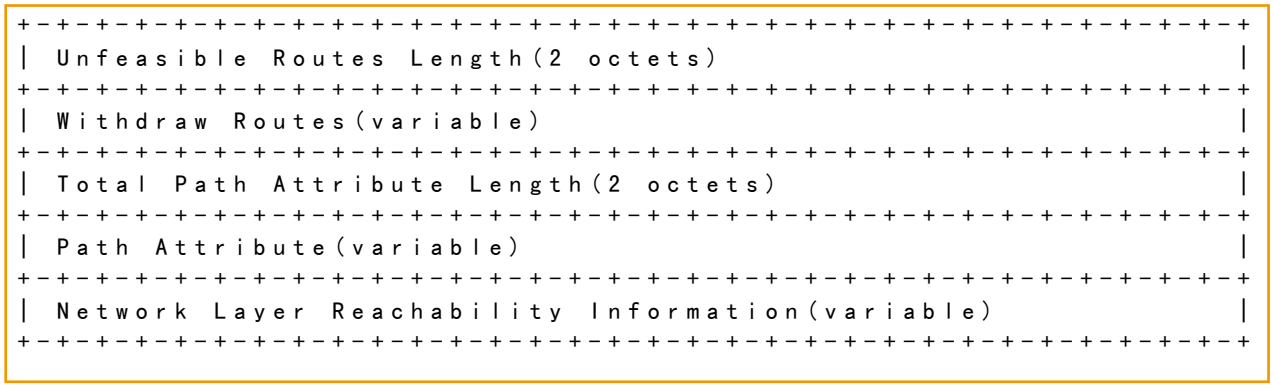


图5 UPDATE消息

决策过程

上面我们描述了BGP发言者之间交换哪些信息。决策过程选择路由用于下一步的发布，应用本地策略信息库PIB（Policy Informaiton Base）来处理Adj-RIB-In中的路由。决策过程的输出是发布到所有邻居（包括IBGP和EBGP）的路由信息集合，被选的路由存储在Adj-RIB-Out中。

决策过程分三步来进行：

1. 当本地BGP发言者接收到EBGP邻居发布过来的更新、替代或撤销路由时，为每一条路由计算优先级，并将最高优先级的路由通告到所有IBGP邻居。

2. 在步骤一完成后激活。负责从到达目的地的所有路由中选择最好的路由，同时安装每条选中的路由到相应的Loc-RIB。如果路由信息携带的下一跳路由不可达，则将该路由排除在这个决策过程之外。

3. 在步骤二完成后激活。负责根据在PIB中的规则，发布Loc_RIB中的路由到EBGP邻居的每个对端。

最优路由有三种情况：

1. 对同一个目的地集合有路由的最高优先级

2. 是到目的地的唯一路由

3. 两条或两条以上具有相同优先级，必须用更细的法则算出一条最优来。此过程称之为Tie-Break

一般来说，BGP计算路由优先级采用如下规则：

1. 选择具有最高LOCAL_PREF值的路由

2. 如果LOCAL_PREF相同，选择从本地IGP（含直连路由）引入的路由

3. 如果LOCAL_PREF相同，且没有本地引入路由，则选择AS_PATH最短的路由

4. 如果AS_PATH路径长度相同，判断ORIGIN值，IGP优于EGP，EGP优于Incomplete

5. 如果ORIGIN相同，优选MULTI_EXIT_DISC值较小的

6. 如果MED也相同，依次选择从EBGP、Confederation、IBGP发布的路由

7. 如果发布源也相同，优选下一跳IP在本地路由表中Cost值最小的路由

8. 如果下一跳Cost也相同，优选CLUSTER_LIST长度最短的路由

9. 如果CLUSTER_LIST长度也相同，优选ORIGINATOR_ID最小的路由

10. 如果ORIGINATOR_ID长度也相同，优选ROUTER_ID最小的路由

Tie-break采用如下过程：

1. 优选MULTI_EXIT_DISC值较小的.

2. 优选下一跳IP在本地路由表中Cost值最小的路由

3. 优选EBGP邻居发布的路由

4. 选择BGP标识符最小的邻居发布的路由

几点考虑

为优化协议，BGP还在如下方面做了考虑：

1. 减少路由振荡

大型网络之间，路有频繁振荡会带来严重的后果。因此BGP为尽可能减少路由振荡做了一些考虑。

● BGP邻居超时时间很长，通常是180秒。当应用环回地址建立邻居时，即便链路中断，只要备份链路能够及时发布切换环回地址路由，邻居可以保持建立，不引起振荡。

● BGP规定，如果需要撤销到一个目的地址的路由，同时更新一个掩码不同的路由，则应该把他们组合在一个UPDATE消息中。这样BGP可以一次处理，不出现路由振荡。

● BGP提供路由抑制机制（Route flap damping），它为每条路由分配一个动态的度量数字，用来反馈路由稳定程度。当一条路由出现振荡，就给他分配一个惩罚值。振荡越多，惩罚值越高。如果惩罚值超出预设的门限，该路由就不再对外发布。直到一段时间后惩罚值降低到可重新使用的门限值。

2. 节省设备资源

● 在BGP向其他邻居发布路由时，引入一个介于0.75~1的随机因子，将该因子分别与每对邻居间发布路由的最小时间间隔相乘，从而得出不同的路由发布最小时间间隔，以避免路由都挤在一个时间发布占用太多的带宽和CPU。

● BGP支持路由聚合，可根据某些属性进行灵活的聚合，减少路由发布条目。

● 引入路由反射器。IBGP要求全链接，引入反射器后，每台BGP发言者只需要与BGP反射器建立邻居，BGP反射器会把从IBGP邻居学到的路由发布给其他IBGP邻居，以节省开销。

BGP 属性简介

文/叶翀

属性分类

BGP属性是BGP进行路由决策和控制的重要信息。它可以分为如下两大类四小类：

1. 公认属性

- 公认强制（Well-known mandatory）
- 公认自选（Well-known discretionary）

公认属性是所有的BGP都必须识别支持的属性。其中，公认强制属性是BGP UPDATE消息中必须包含的必要部分。公认自选则是自由选择的部分。

2. 可选属性

- 可选转发（Optional transitive）
- 可选非转发（Optional non-transitive）

可选属性并不要求所有的BGP都识别。如果属性是可选转发的，那么，即使BGP不能识别该属性，也要接受该属性并将其发布给它的对端。而如果属性是可选非转发的，BGP可以忽略包含该属性的消息并且不向它的对端发布。

属性详述

常见的BGP属性如下：

1.ORIGIN

ORIGIN标示路径信息的来源，是公认强制属性。ORIGIN可以是以下三种值：

- IGP：网络层可达信息来源于AS内部
- EGP：网络层可达信息通过AS外部学习
- INCOMPLETE：网络层可达信息通过别的方式学习

在路由优选时，ORIGIN中，IGP优于EGP，EGP优于INCOMPLETE。

2.AS_PATH

AS_PATH由一系列AS路径组成，也是公认强制属性。AS-PATH有两种类型：

- AS_SET：在UPDATE消息中的路由经过的AS的无序集
- AS_SEQUENCE：在UPDATE消息中的路由经过的AS的有序集

当BGP发言者发布路由给IBGP邻居时，BGP不修改路由的AS_PATH属性。当BGP发言者发布路由给EBGP邻居时，对AS_PATH做如下修改：

1）如果AS_PATH的第一个路径段是AS_SEQUENCE类型，本地系统应该把自己的AS号作为序列的最后一个元素加在后面（放在最左面）；

2）如果AS_PATH的第一个路径段是AS_SET类型，本地系统应该添加一个新AS_SEQUENCE类型的路径段到 AS_PATH，包括段的内部的自己AS号码。

AS_PATH属性主要用来作为路由选路的一种度量。路由经过的AS少则优先。

它也可以用来避免环路。如果BGP发言者从EBGP邻居收到一条路由，它的AS_PATH包含发言者自己的AS号，就说明这是条环路路由，将其丢弃。

由于BGP发言者发布路由给IBGP邻居时，并不将AS号加入AS_PATH，如果邻居将路由继续转发，最终发言者自己再次收到路由时，将无法判断是否环路路由。因此，BGP要求IBGP对收到的路由不再转发。有鉴于此，AS内部BGP发言者对路由要同步，IBGP邻居必须逻辑上全连接建立邻居。

3.NEXT_HOP

它定义了到达目的地下一跳的设备IP地址，也是一个公认强制属性。

NEXT_HOP中IP地址的填写遵循如下规则：

- 如果是发布给EBGP邻居，NEXT_HOP填写BGP发言者的IP地址
- 如果是发布给IBGP邻居，且路由来自AS内部，则NEXT_HOP填写BGP发言者的IP地址
- 如果是发布给IBGP邻居，且路由来自AS外部，则NEXT_HOP保留原始的AS外部邻居的IP地址

即NEXT_HOP指向路由发布者。

4.MULTI_EXIT_DISC

MULTI_EXIT_DISC被用来区分同一个邻居AS的多个出口，是一个可选非转发属性，一般简写为MED。MED只在EBGP发布的路由中产生，接收者可以向它的IBGP邻居转发，但不允许向它的EBGP邻居转发。假设一张网络连接了邻居AS的多个出口，通过发布不同的MED给对端，就可以控制进入网络的流量从MED值最小的那个出口进来。

如图1所示，AS100的边界路由器向AS200的邻居发布10.65.47.0/24的路由时，携带的MED分别是10和20，这样从AS200访问10.65.47.0/24的流量会从左边那个路由器流入AS100。

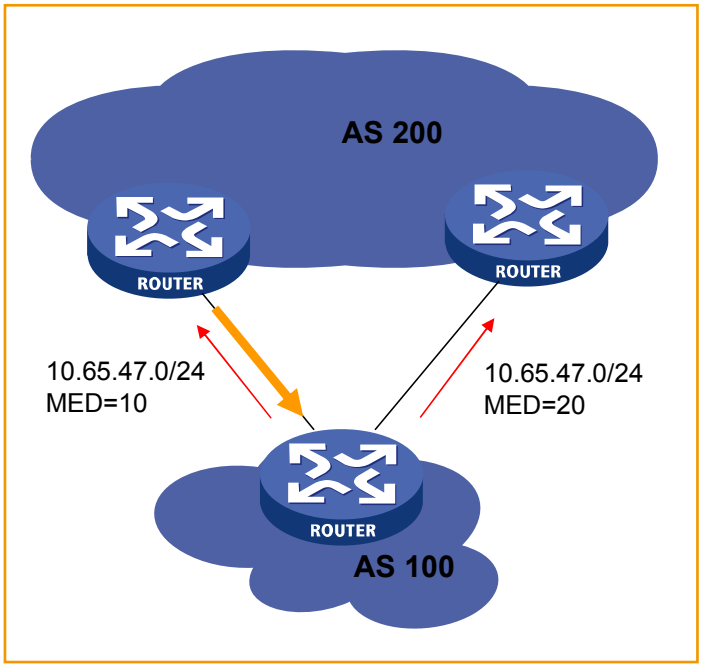


图1 MULTI_EXIT_DISC

5. LOCAL_PREF

LOCAL_PREF用来通知AS内部源发言者通告路由的优先程度，是公认自选属性。LOCAL_PREF只在IBGP发布的路由中使用，它不会传递给其他AS，除非AS建立联盟。假设一张网络连接了两个不同的AS出口，对某些特定业务，需要控制对应的流量从特定的AS出口转发，那么可以对AS边界的路由器应用LOCAL_PREF，AS内部的路由器将优选LOCAL_PREF高的路由。

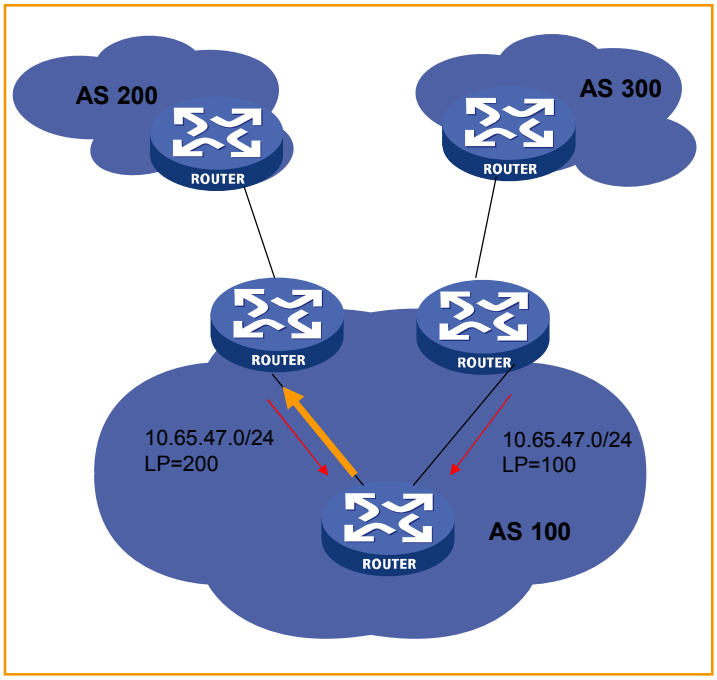


图2 LOCAL_PREF

如图2所示，AS200和AS300都向AS100发布了10.65.47.0/24路由。通过在AS100边界路由器上应用LOCAL_PREF，可以控制AS内的流量选择左边的边界路由器作为出口。

6.ATOMIC_AGGREGATE

ATOMIC_AGGREGATE是公认自选属性。有时BGP发言者会收到两条重叠的路由，其中一条路由包含的地址是另一条路由的子集。一般情况下BGP发言者会优选更精细的路由（前者），但是在对外发布时，如果它选择发布更粗略的那条路由（后者），这时需要附加上ATOMIC-AGGREGATE属性，以知会邻居。它实际上是一种警告，因为发布更粗略的路由意味着更精细的路由信息在发布过程中丢失了。

7.AGGREGATOR

AGGREGATOR是可选转发属性，它是ATOMIC_AGGREGATE属性的补充。如上所述，ATOMIC_AGGREGATE是一种路由信息丢失的警告，AGGREGATOR属性补充了路由信息在哪里丢失——它包含了发起路由聚合的AS号码和形成聚合路由的BGP发言者的IP地址。

8.COMMUNITY

COMMUNITY是可选转发属性，它是一组共享相同属性的目的地集合。例如对一组路由应用相同的团体属性值，从而通过对团体属性进行路由策略来达到对一组路由进行控制的目的。对团体属性的详细介绍请参见本刊《BGP团体属性》一文。

9.ORIGINATOR_ID

ORIGINATOR_ID是可选非转发属性，用于标识路由反射器。在讨论AS_PATH属性时，我们知道IBGP要求邻居全连接，对于大型网络来说，两两建立邻居发布路由的开销是巨大的。于是BGP提供路由反射器这个角色，每台BGP发言者只需要与BGP反射器建立邻居，BGP反射器会把从IBGP邻居学到的路由发布给其他IBGP邻居，以节省开销。通常人工选定一台或多台设备作为反射器，反射器可以是多台，形成路由层面的冗余结构。但是这样有一个问题，就是AS_PATH在AS内部无法避免路由环路。为了防止引入路由反射器之后出现环路，增加ORIGINATOR_ID这个属性来标识，反射器在发布路由时加入ORIGINATOR_ID，当反射器收到的路由信息中包含自己的ORIGINATOR_ID时，就检测到了环路。

10.CLUSTER_ID

CLUSTER_ID是可选非转发属性，用于标识路由反射器组。很容易猜到，这个属性也是用来防止环路。

[深入讨论]

BGP FAQ

文/程锋章

BGP General FAQ

BGP 有哪几种拓扑结构

BGP 有三种基本的网络拓扑结构：

- 单口 AS（stub AS）：一个 AS 通过单一出口点到达其域外的网络
- 多归路非过渡 AS（Multihomed AS）：一个 AS 有多于一个到达外部网络的出口点但它不允许业务量通过它过渡
- 过渡 AS（transit AS）：一个 AS 有多于一个到达外部网络的出口点并且它允许被其他 AS 用于过渡业务量

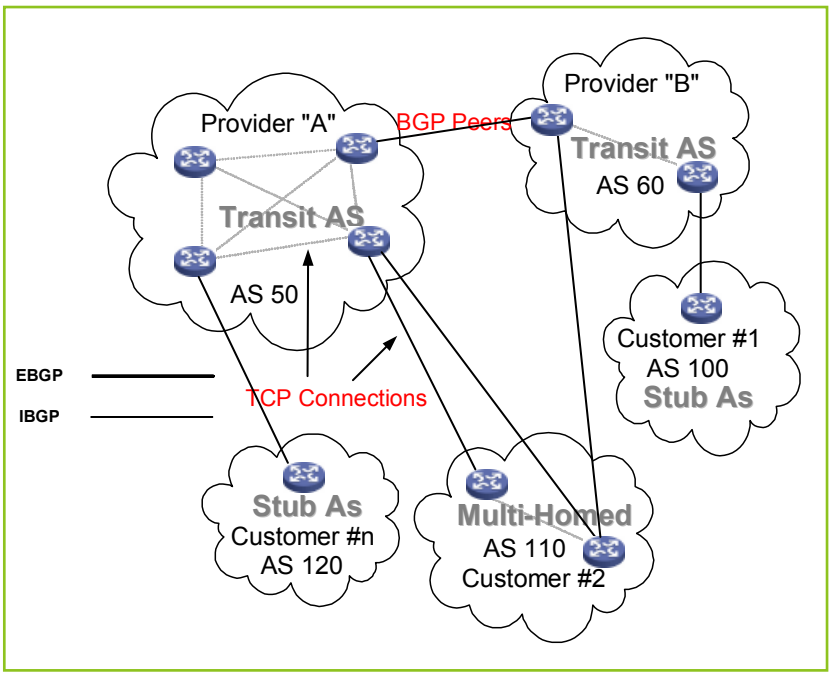


图1 BGP典型拓扑结构

从BGP的观点上来看整个Internet的拓扑就是由一系列单口AS多归路非过渡AS和过渡AS组成的连通图。每个AS用AS号码来识别两个AS之间的连接形成一个路径，BGP保证无循环域间选路路径信息的汇集形成一棵路径树，这棵路径树就是到达特定目的地的路由。

如何学习 BGP

BGP是否很难学无法给出一个确切的说法，但是可以肯定的是学习和使用BGP绝对很过瘾。BGP在路由协议中的地位是非常高的，在核心路由器上应用极多，与之相关的特性如L2vpn、L3vpn以及路由策略等比较多，其主要是用来控制路由的发送和接收，而IGP路由协议皆属于其可控制的对象，良好的BGP基础也是学习MPLS VPN的重要基础之一。

首先对BGP协议的设计思想比如防止环路、路由传播、出口流量控制、安全特性等有初步了解，理解BGP协议的基础。《TCP/IP routing II》对各种报文、属性以及相关应用场景做了非常详尽的阐述，值得刚开始好好学习；在实际测试中可以先进行基础的配置和属性了解，然后根据特性进行测试。同时在实际测试中体会我司与友商实现的差异，有助于深入理解其本质。

当然一门协议的掌握还是需要大量的积累和实践，对于相关RFC的了解也必不可少，BGP相关RFC如下：

- RFC 4808, Mar 2007 Key Change Strategies for TCP-MD5
- RFC 4797, Jan 2007 Use of Provider Edge to Provider Edge (PE-PE) Generic Routing Encapsulation (GRE) or IP in BGP/MPLS IP Virtual Private Networks
- RFC 4781, Jan 2007 Graceful Restart Mechanism for BGP with MPLS
- RFC 4761, Jan 2007 Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling
- RFC 4760, Jan 2007 Multiprotocol Extensions for BGP-4
- RFC 4724, Jan 2007 Graceful Restart Mechanism for BGP
- RFC 4684, Nov 2006 Constrained Route Distribution for Border Gateway Protocol/Multiprotocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)
- RFC 4698, Oct 2006 IRIS: An Address Registry (areg) Type for the Internet Registry Information Service
- RFC 4659, Sep 2006 BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN
- RFC 4632, Aug 2006 Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan
- RFC 4486,Apr 2006 Subcodes for BGP Cease Notification Message
- RFC 4456, Apr 2006 BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)
- RFC 4451, Mar 2006 BGP MULTI_EXIT_DISC (MED) Considerations
- RFC 4384, Feb 2006 BGP Communities for Data Collection
- RFC 4382, Feb 2006 MPLS/BGP Layer 3 Virtual Private Network (VPN) Management Information Base
- RFC 4381, Feb 2006 Analysis of the Security of BGP/MPLS IP Virtual Private Networks (VPNs)
- RFC 4365,Feb 2006 Applicability Statement for BGP/MPLS IP Virtual Private Networks (VPNs)
- RFC 4364, Feb 2006 BGP/MPLS IP Virtual Private Networks (VPNs)

- RFC 4360, Feb 2006 BGP Extended Communities Attribute
- RFC 4278, Jan 2006 Standards Maturity Variance Regarding the TCP MD5 Signature Option (RFC 2385) and the BGP-4 Specification
- RFC 4277, Jan 2006 Experience with the BGP-4 Protocol
- RFC 4276, Jan 2006 BGP-4 Implementation Report
- RFC 4275, Jan 2006 BGP-4 MIB Implementation Survey
- RFC 4274, Jan 2006 BGP-4 Protocol Analysis
- RFC 4273, Jan 2006 Definitions of Managed Objects for BGP-4, This document obsoletes RFC 1269 and RFC 1657.
- RFC 4272, Jan 2006 BGP Security Vulnerabilities Analysis
- RFC 4271, Jan 2006 A Border Gateway Protocol 4 (BGP-4).This document obsoletes RFC 1771.（最新的BGP RFC，废除了1771）
- RFC 4098, Jun 2005 Terminology for Benchmarking BGP Device Convergence in the Control Plane
- RFC 3913, Sep 2004 Border Gateway Multicast Protocol (BGMP): Protocol Specification
- RFC 3882, Sep 2004 Configuring BGP to Block Denial-of-Service Attacks
- RFC 3779, Jun 2004 X.509 Extensions for IP Addresses and AS Identifiers
- RFC 3765, Apr 2004 NOPEER Community for Border Gateway Protocol (BGP) Route Scope Control
- RFC 3562, Jul 2003 Key Management Considerations for the TCP MD5 Signature Option
- RFC 3392, Nov 2002 Capabilities Advertisement with BGP-4.This document obsoletes RFC-2842（废除了2842）.
- RFC 3345, Aug 2002 Border Gateway Protocol (BGP) Persistent Route Oscillation Condition
- RFC 3221, Dec 2001 Commentary on Inter-Domain Routing in the Internet
- RFC 3107, May 2001 Carrying Label Information in BGP-4（BGP开始支持标签）
- RFC 3065, Feb 2001 Autonomous System Confederations for BGP
- RFC 2918, Sep 2000 Route Refresh Capability for BGP-4（BGP支持路由刷新能力）
- RFC 2545, Mar 1999 Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing
- RFC 2519, Feb 1999 A Framework for Inter-Domain Route Aggregation
- RFC 2439, Nov 1998 BGP Route Flap Damping
- RFC 2385, Aug 1998 Protection of BGP Sessions via the TCP MD5 Signature Option
- RFC 2270, Jan 1998 Using a Dedicated AS for Sites Homed to a Single Provider
- RFC 1998, Aug 1996 An Application of the BGP Community Attribute in Multi-home Routing
- RFC 1997, Aug 1996 BGP Communities Attribute
- RFC 1930, Mar 1996 Guidelines for creation, selection, and registration of an Autonomous System (AS)

- RFC 1774，Mar 1995 BGP-4 Protocol Analysis
- RFC 1773，Mar 1995 Experience with the BGP-4 protocol
- RFC 1772，Mar 1995 Application of the Border Gateway Protocol in the Internet
- RFC 1771，Mar 1995 A Border Gateway Protocol 4 (BGP-4)
- RFC 1745，Dec 1994 BGP4/IDRP for IP---OSPF Interaction

BGP 的 RouterID 如何配置，如何自动选择

全局Router ID可以在全局模式下通过配置命令router id来配置，如果没有通过命令指定，系统会从当前接口的IP地址中自动选取一个作为路由器的ID号。其选择顺序是：优先从Loopback地址中选择最大的IP地址作为路由器的ID号，如果没有配置Loopback接口，则选取接口中最大的IP地址作为路由器的ID号。只有在路由器的Router ID所在接口被删除或去除手工配置的Router ID的情况下才会重新选择路由器的Router ID。为了增加网络的可靠性，建议将Router ID手工配置为Loopback接口的IP地址。

BGP的Router ID可以在启动BGP进程后通过BGP全局模式下的配置命令router-id来指定，若未配置则采用全局Router ID值。通过BGP的配置命令修改RouterID会导致已经建立的BGP peer会全部重启，如：

```
bgp 100
 router-id 1.1.1.1
```

H3C COMWAREV5平台支持手工配置。

BGP 路由的基本使用原则有哪些

BGP路由往往很多，但是并不是都会进行转发处理，大概有如下几条规则：

1）多条路径时，BGP Speaker只选最优的给自己使用，当然也可以设置负载分担，负载分担请见本文ECMP章节；

2）BGP Speaker只把自己使用的路由通告给相邻体，也就是说本地BGP路由表里面不是最优的路由不会再进行转发处理等操作；

3）BGP Speaker从EBGP获得的路由会向它所有BGP相邻体通告（包括EBGP和IBGP），当然路由不会再从原路发送回去；

4）BGP Speaker从IBGP获得的路由不向它的IBGP相邻体通告（反射和联盟可以解决这个问题）；

5）BGP Speaker从IBGP获得的路由默认会向它所有EBGP相邻体通告；若配置了同步，是否通告给它的EBGP相邻体要依IGP和BGP同步的情况来决定。

BGP 路由协议是如何避免路由环路的

BGP不同于其他IGP协议，其路由都包含了丰富的路由属性，并通过路由属性来对路由进行过滤，其中一个属性为AS_PATH，该属性为该路由经过的所有AS的序列，这样对于收到的路由，通过对AS_PATH进行检查，如果发现自身的AS号已经出现在AS_PATH属性中，那么就表示自身发布的路由又重新回到自己所处的AS中，已经出现了路由环路，这时就会丢弃接收到的路由，从而避免继续对外发布路由，导致环路产生。

当然还可以参考上面的BGP使用原则，比如BGP从IBGP收到的路由不继续往IBGP邻居发送，这也是协议上避免环路的一种方法（反射路由的处理类似，请参看后续章节）。

正常情况下BGP会丢弃AS_PATH中包含自身AS号的路由。对于特定情况下导致的AS号重复的合理环境，可以通过如下命令来进行控制“peer { group-name | peer-ipv4-address } allow-as-loop [number]”，其中number取值范围为<1-10>，默认值为1，即允许接收路由的AS_PATH中包含一个自身AS号。当然，在向EBGP邻居发布时，也还要在AS_PATH最后再加上自身的AS号。

有哪些原因会导致 BGP 连接建立不起来

有不少原因，最常见原因如下：

1）两边BGP peer地址不可达，一般是底层原因或者缺少可达的路由，可以使用扩展的ping命令检查TCP连接是否正常，由于一台路由器可能有多个接口能够到达对端，应使用ping -a ip-address命令指定发送ping包的源IP地址。

2）对等体IP地址和AS配置错误，常为大意所致；

3）OPEN报文协商失败，OPEN报文需要协商BGP版本、Holdtime、RouterID以及可选项参数（包括各种能力参数）等；H3C COMWAREV5平台实现当接受到不支持的能力参数时候直接进行忽略而不影响建立连接；

4）BGP的MD5验证配置错误；

5）BGP的RouterID冲突；

6）联盟与非联盟之间的BGP连接配置错误；

7）错误报文导致连接中断，比较少见的如BGP的Marker值出现错误；

8）还有一些比较特殊的情况，请参考下文。

有哪些排错手段针对 BGP 连接建立不起来的情况

有不少方法，最常见方法如下：

1）首先打开调试开关deb bgp X.X.X.X all开关，确认状态机在哪一步出现错误；

2）如果BGP状态始终在active状态徘徊，表示TCP建立不起来，首先排除底层不通和路由不可达的情况；

3）如果BGP状态始终在active状态徘徊，其次排除BGP的MD5验证问题；

4）如果BGP状态始终在active状态徘徊，最后特殊配置问题，比如IBGP的connect-interface或者EBGP peer的ebgp-max-hop等，见下文；

5）如果是Open报文协商错误，通过调试开关，H3C COMWAREV5平台能够很方便的查看到具体的错误类别和信息，然后根据错误提示采取具体措施。

Open 报文协商时候发现不可识别或不支持的能力怎么办

在实际应用过程中，H3C设备在与友商设备进行BGP能力协商过程中遇到无法识别和支持的能力参数，比如有的是最新RFC规定而我司没有实现的，有的是友商自定义的能力，在类似情况处理过程中会忽略无法识别的能力，打出信息并继续建立邻居。

在RFC 2842中BGP协议要求BGP speaker在收到的OPEN报文中带有一个或多个不认识的可选项参数时，它会中断BGP的会话连接，然后对方会继续进行BGP连接，此时不带上上述不认识的选项参数。不过此RFC已经被RFC3392（Capabilities Advertisement with BGP-4）所废除。

根据RFC3392的要求，当一个支持能力通告的BGP speaker向其BGP peer发送OPEN消息时，其消息可以包含称之为能力的选项参数，该参数列出它所支持的所有能力：

1）BGP speaker通过检查从其BGP对等体收到的OPEN消息中的能力列表来确定对方所支持的所有能力；

2）如果BGP speaker支持上述能力列表中的一种能力后直接使用该能力并保持BGP连接，这样不用发送NOTIFICATION并再次进行协商；

3）如果BGP speaker在收到对方对于本端发出的OPEN消息的响应是NOTIFICATION 并且其Error Subcode为Unsupported Optional Parameter，此时认为对方不支持先前的能力通告。本端将试图重新和对方建立连接，此时本端发送的OPEN消息中将不再携带对端不支持的能力选项参数Capabilities Optional Parameter。

4）如果一个BGP speaker支持某一特定能力发现对方不支持该能力，该BGP speaker可以向对方发送NOTIFICATION消息并终止该会话；此时的ErrorSubcode设定为Unsupported Capability，该NOTIFICATION消息将在Data域中包含引起会话中断的能力。而是否发送消息并中断会话，取决于本端BGP speaker，并且一旦中断将不再重新自动连接。

BGP Open报文中有哪些能力参数

BGP的能力参数类型有两种，即多协议能力和路由刷新能力；针对地址族的定义就比较多了，不同厂商实现也可能不一样。比如根据最新RFC4761，针对VPLS的能力已经定义为25/65（L2vpn也是25/65）。具体情况见表1。

表1 Open报文中的能力参数

	CODE	AFI	SAFI
IPv4 Unicast	Multiprotocol（1）	1	1
IPv4 Multicast	Multiprotocol（1）	1	2
IPv4 VPNV4	Multiprotocol（1）	1	128
Label IPv4	Multiprotocol（1）	1	4
MVPN	Multiprotocol（1）	1	66
L2vpn	Multiprotocol（1）	196	128
Ipv6 Unicast	Multiprotocol（1）	2	1
IPv6 Multicast	Multiprotocol（1）	2	2
IPv6 VPNv4	Multiprotocol（1）	2	128
Label IPv6	Multiprotocol（1）	2	4
VPLS（RFC4761）	Multiprotocol（1）	25	65
Refresh	Route Refresh（2）		
GR	Graceful Restart（64）		
4-AS	4-AS（65）		
Dynamic Capability	Dynamic Capability（67）		

使用 Loopback 口为什么无法建立 IBGP 邻居

当排除底层原因后，发现IBGP PEER使用loopback口建立却无法建立。这是因为BGP连接建立首先要建立起两个peer之间的TCP连接，而TCP连接的源地址缺省是路由器相应的出接口的IP地址，所以必须要指定TCP连接的源地址为相应的loopback接口地址，连接才能建立起来，peer X.X.X.X connect-interface命令的功能就是用于指定BGP会话建立TCP连接使用的接口。

直连 EBGP 使用 Loopback 口为什么无法建立连接

很简单，上面的例子是IBGP邻居关系的建立，如果使用Loopback口建立的是EBGP连接，即使是两个直连接口也需要配置ebgp-max-hop，因为两个loopback口不是直连接口。

一般情况下不推荐使用loopback建立EBGP邻居，而一般是使用物理接口地址建立，比如在L3vpn的各种跨域环境中。

为什么非直连 EBGP 邻居无法建立

如果是EBGP邻居，双方路由可达，且EBGP连接在物理上不是直连的，请检查是否配置了peer的ebgp-max-hop。默认情况下，EBGP邻居不配置这条命令，如果不是直连，必须配置peer X.X.X.X. ebgp-max-hop，该命令的默认值是64。

有哪些原因会导致 BGP 连接建立成功后再 down 掉

有不少原因，最常见原因如下：

1）BGP连接建立好后，在协商后的holdtime时间内收不到keepalive报文，导致错误代码为4/0的错误；

2）收到非法的Update报文导致BGP为了安全考虑自动中断连接，并打印错误信息；

3）MTU问题，路由器会因为一些转发芯片限制或者人为的MTU设置，导致经过多次封装后的BGP报文超过mtu而被丢弃；

4）MTU和QoS设置不当可能导致大的Update报文被丢弃，由于TCP的重传机制，当发送多个大的Update报文时，可能产生大量等待重传的Update报文，从而抑制了keepalive报文的正常发送，当连续收不到keepalive报文时，BGP认为邻居已经Down。

5）网络拥塞问题：网络拥塞可能导致Keepalive报文收发失常，邻居状态不断改变；另外，如果到达邻居的路由是通过IGP（如OSPF）发现的，网络拥塞可能导致该路由丢失，从而使邻居间的连接中断。

6）设置原因，导致TCP179端口号不可用。

当然了，BGP所支持的操作非常多，还有很多主动的原因导致BGP会话重新启动：

1）对端关闭会话，比如对peer配置ignore命令；

2）如果配置了路由数目限制（peer X.X.X.X route-limit），超过指定数目后也会down掉；不同机型其最大值也不一样。

3）远端AS改变；

4）修改路由反射器客户机配置；

5）修改对等体/组的某些策略或者能力；

6）配置和反配置BGP的Router ID；

7）由联盟改为非联盟，或反之；confederation nonstandard也可导致；

8）无法识别对端发送的BGP报文或者接受到错误的报文。

9）路由更新报文中的必遵属性缺失。

为什么使用 network 命令无法将本地路由通过 BGP 发布出去

Network命令是BGP各个视图下很强大的路由引入命令，能过将各种IGP有效路由、静态路由、直连路由等引入BGP中发布出去。比如本地存在直连路由或者IGP协议路由172.16.1.0/24，BGP视图下使用network 172.16.1.0命令，目的是准备把这条路由传递到BGP路由表中，但是查看本地BGP路由表里面没有这条路由。

使用BGP的network命令发布路由，前缀和掩码必须完全匹配才能正常发布。172.16.0.0是一个B类网段地址，如果没有mask参数的话，缺省使用16位自然掩码，而上述路由的掩码是24位，所以必须在mask参数中配置24位地址掩码才能正常发布路由。

BGP配置模式下的network命令可以带mask参数，也可以不带。不带mask参数的情况下缺省使用路由的自然掩码。在全局路由表中必须具有前缀和掩码都相同的路由，才能正常发布。

Peer Ignore 命令有什么作用

Peer ignore命令用来人为地停止指定对等体/对等体组的激活会话，并且清除所有相关路由信息，禁止与指定对等体/对等体组建立会话，BGP邻居将一直抑制在idle的状态，会话一直处于无法建立的状态。如果该命令用来对于一个对等体组，这就意味着大量与对端的会话突然终止。缺省情况下，允许与BGP对等体/对等体组建立会话。在配置peer ignore命令之后，查看peer状态如下：

```
<H3C>display bgp peer
 BGP local router ID : 19.19.19.19
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 0
4.4.4.4         4   100        0        0     0       0 02:35:59 Idle(Admin)
```

Public-as-only 在 H3C COMWAREV5 平台中的用法

参看peer public-as-only命令用来配置发送BGP更新报文时不携带私有自治系统号。举个例子，使用策略给发送路由增加属性，并针对peer使能该功能：

```
route-policy 1 permit node 0
 apply as-path 65535 65534 1000 65532 65531 65530 65530 65529
```

查看对端路由表发现：

```
[H3C]display bgp routing-table 60.1.1.0
 BGP routing table entry information of 60.1.1.0/24:
 From            : 66.1.1.1 (2.2.2.2)
 Original nexthop: 66.1.1.1
 AS-path         : 200 65535 65534 1000 65532 65531 65530 65530 65529
 Origin          : igp
 Attribute value : MED 0, pref-val 0, pre 255
 State           : valid, external,
 Not advertised to any peers yet
```

可以看到私有的AS号一个都没有去掉，为什么了？H3C COMWAREV5平台目前实现的规格如此，只要有公有as号就不进行私有as的删除。如果在策略里面去1000，那么5040收到的路由as-path只有200。

BGP 如何发布默认路由

BGP可以通过peer default-route-advertise和default-route import来控制缺省路由发布。Peer default-route-advertise不需要本地存在缺省路由而直接向peer发布缺省路由，而default-route import仅仅表示允许引入本地缺省路由，意思是必须通过import方式引入存在本地路由表里面的IGP默认路由，然后再配置default-route import才能使默认路由正确发布。

为什么从直连 EBGP 邻居向 IBGP 邻居发布路由时路由会失效

在BGP中，向IBGP和EBGP邻居发送路由时，下一跳的处理是不同的。向EBGP邻居（即在AS间传播）发送路由时，next-hop均改为该路由器的出口IP地址（当下一跳修改前后的地址符合第三方下一跳时，不做修改）；向IBGP邻居（即在AS内传播）发送路由时，next-hop是不变的。

由于BGP向其他IBGP邻居转发来自EBGP路由时不修改下一跳，这样的话若IBGP邻居所处的设备没有到该下一跳地址的路由，会导致该IBGP收到这条转发自IBGP邻居的EBGP邻居的路由后下一跳不可达，导致路由失效。

解决方案有多种：可以配置next-hop-local，这样收到EBGP路由再往IBGP邻居发送的时候会强制更改下一跳为自己的出接口地址；自治域内所有的设备都配置IBGP邻居且要全链接，通过bgp把下一跳也学过去；通过IGP协议来保证自治域内的所有设备能够知道所有部的接口地址。

为什么相同路由比较时候没有选择 MED 值小的路由

假设一个这样的场景，三台AS号不一致的MSR之间分别建立了EBGP邻居关系，其中RTC同时收到RTA和RTB发来的因特网路由。根据RTC的要求，RTA将自己发送给RTC的路由设置MED值为50，而RTB将自己发送给RTC的MED设置为100。RTC希望选择MED值小的路由作为最佳路由，从而对相同目的地来说，把通过RTA的链路作为主链路，而把通过RTB的链路作为备份链路。当时在RTC上面没有把RTA发送过来的路由选为最优，为什么？

BGP在路由优选过程中考虑若干因素，包括本地优先级、AS路径长度、起点类型、MED值等。在前几项都相同的情况下，应选择MED值小的路由作为最佳路由。需要注意的是，MED值默认只在同一个AS传来的路由之间才具备可比性。为了能够在不同AS传来的相同路由之间比较MED值，从而选择MED小的路由作为最佳路由，需要在BGP或者BGP VPN视图下配置命令compare-different-as-med。

BGP FAQ

为什么 OSPF 的路由引入到 BGP 中后 cost（MED）需要加 1

在RFC4577（OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks）中有如下描述：

MED (Multi_EXIT_DISC attribute). By default, this SHOULD be set to the value of the OSPF distance associated with the route, plus 1。即ospf路由被引入到BGP中后MED值需要加1。

因为在PE上引入到BGP中再发布到对端PE上OSPF还原后就丢失了原来生成者的信息，这条路由的原来生成者通过其他途径再收到这这条被还原的路由后，如果进行了计算就会导致环路。MED加1，被还原的路由cost会比原路由cost大，能够在某种程度上避免环路。

OSPF 多实例下 MCE 能力对 BGP MED 的影响

命令行vpn-instance-capability simple的作用并不是使能vpn能力，而是使能了多实例CE的能力；同时该能力使MCE不去检查DN位是否已经被置位（当DN位被置位时说明这条LSA由PE发给CE，值得注意的是DN位只存在于3类LSA）。

比如不配置vpn-instance-capability时候，引入到BGP多实例进程中的路由会携带下面类似属性Ext-Community :<OSPF Domain Id: 0.0.0.0:0>, <OSPF AreaNum: 0.0.0.0 RouteType: 5 Option: 1>, <OSPF Router Id: 10.10.1.1:0:0>,<RT: 1:1>，上述扩展团体属性是bgp携带发给对端PE设备的，用来让对端PE上的OSPF进程还原LSA的依据；此时候设备就作为普通的PE，PE上bgp引入ospf路由时，med的值等于ospf路由的cost加1；

```
[H3C]display bgp vpn vpn vpn-a routing-table 172.32.0.0
 BGP local router ID : 104.104.104.104
 Local AS number : 100
 Paths:   1 available, 1 best
 BGP routing table entry information of 172.32.0.0/16:
 Imported route.
 From            : 0.0.0.0 (0.0.0.0)
 Original nexthop: 10.10.1.2
 Ext-Community   :<OSPF Domain Id: 0.0.0.0:0>, <OSPF AreaNum: 0.0.0.0 RouteType:
 5 Option: 1>, <OSPF Router Id: 10.10.1.1:0:0>,  <RT: 1:1>
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 2, pref-val 0, pre 150
 State           : valid, local, best,
 Not advertised to any peers yet
```

当配置了vpn-instance-capability simple后，本地路由器就不是PE了，而成了MCE，这样所以BGP引入OSPF路由时，本地OSPF就不会组装这些属性值给bgp，只是作为普通的引入，处理，BGP路由只会携带扩展团体属性<RT:1:1>，而引入的ospf路由的其他扩展团体属性则会丢失；而这个时候bgp引入ospf路由时，med的值等于ospf路由的cost。

```
[H3C-ospf-1000]
#
ospf 1000 vpn-instance vpn-a
 vpn-instance-capability simple
 area 0.0.0.0
  network 10.10.1.0 0.0.0.255
#
[H3C-ospf-1000]display bgp vpnv4 vpn-instance vpn-a routing-table
 Total Number of Routes: 3
 BGP Local router ID is 104.104.104.104
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
Network                 NextHop        MED        LocPrf     PrefVal Path/Ogn

 *>  16.16.16.16/32     0.0.0.0          1                     0       ?
 *>  50.1.1.0/24        0.0.0.0          1                     0       ?
 *>  172.32.0.0         0.0.0.0          1                     0       ?
[H3C-ospf-1000]display ospf routing
          OSPF Process 1000 with Router ID 10.10.1.1        Routing Tables
 Routing for Network
 Destination        Cost     Type    NextHop         AdvRouter       Area
 10.10.1.0/24       10       Transit 10.10.1.1       16.16.16.16     0.0.0.0
 Routing for ASEs
 Destination        Cost     Type    Tag         NextHop         AdvRouter
 172.32.0.0/16      1        Type2   1           10.10.1.2       16.16.16.16
 50.1.1.0/24        1        Type2   1           10.10.1.2       16.16.16.16
 16.16.16.16/32     1        Type2   1           10.10.1.2       16.16.16.16
 Total Nets: 4
 Intra Area: 1  Inter Area: 0  ASE: 3  NSSA: 0
```

如何实现 BGP 多进程和网络迁移

众所周知，一个路由器只支持一个BGP进程，有着唯一的AS号，但是在某些特殊情况下比如网络迁移更换as号的时候为了保证网络切换的顺利，需要一些特性来支持，具体可以参看《BGP Support for Dual AS Configuration for Network AS Migrations》。

H3C COMWAREV5平台通过fake-as命令为指定PEER设置一个假AS号来实现，该特性只针对EBGP PEER有效。该命令用来支持BGP邻居可以配置不同于当前由使能BGP协议时指定的自治系统号，配置peer { group-name | peer-ipv4-address } fake-as [number]命令后，该peer和本地建peer关系时，要用fake-as号来代替本地的实际as号。示例说明一下，本地RTX

（本地地址57.0.0.1）的BGP配置如下：

```
bgp 100
 router-id 1.1.1.1
 undo synchronization
 peer 57.0.0.2 as-number 57
peer 57.0.0.2 fake-as 88
```

那么RTX在向57.0.0.2建立连接时的本地AS将是88，而不是100。与此同时，RTY（本地地址57.0.0.2）配置peer 57.0.0.1时对应的AS号也应该为88，而不是100。相关BGP配置如下：

```
bgp 57
 peer 57.0.0.1 as-number 88
```

实际应用中，该命令通常和peer { group-name | peer-ipv4-address } substitute-as结合使用。

什么是 BGP 的同步原则

同步的目的是为了防止在某种情况下转发“黑洞”的出现，起用同步功能后BGP Speaker在接收到IBGP邻居发过来的路由后都会查看该路由是否已经在IGP路由表中，如果IGP路由表中有这条路由，BGP路由表才会将这条路由置为有效；如果IGP路由表中没有该路由则BGP表中的该条路由是无效的。如果关闭同步功能，在收到IBGP邻居发来的路由更新后不检查IGP表是否有该路由，而直接将该路由置为有效，这样的话在以下拓扑中就会出项问题：

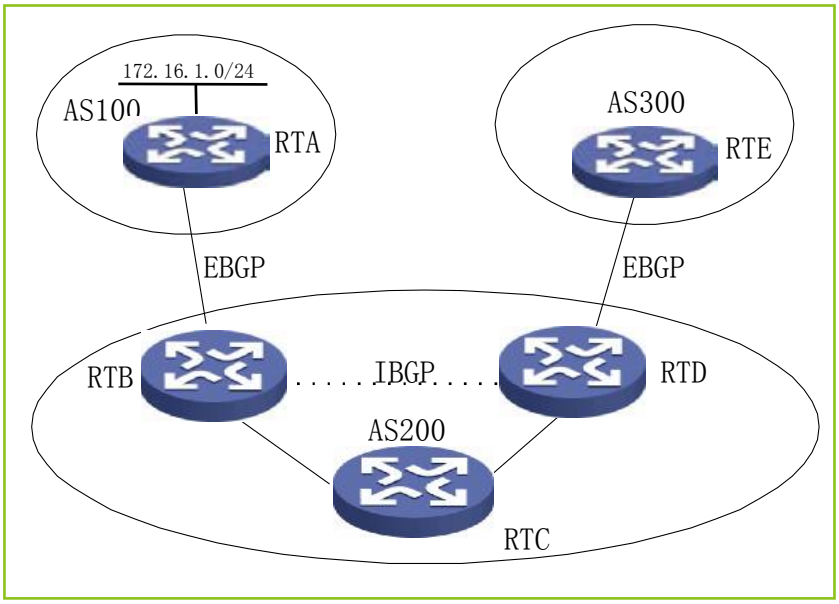


图2 BGQ同步原则

在图2中，RTC没有运行BGP，RTD关闭了同步功能。172.16.1.0/24从RTA始发，传播方向为：RTA---------->RTB---------->RTD---------->RTE。RTB、RTD、RTE收到该路由后将其置为有效，这时如果RTE要转发一个目的ip为172.16.1.10的报文的话，将会通过如下步骤转发：

- step1：RTE将目的ip为172.16.1.10的报文发给RTD；
- step2：RTD接到此报文后将向RTB转发此报文（RTB已使用next-hop-local），但由于RTD与RTB的IBGP连接为逻辑连接，因此去RTB的实际下一跳是RTC,于是又将此报文转发给RTC；
- step3：RTC收到此报文后查找路由表，但没有找到匹配项，因为RTC没有运行BGP，不知道172.16.1.10的下一跳，所以此目的ip为172.16.1.10的报文在RTC处就被丢弃了！

RTB、RTC、RTD的IGP路由表中没有172.16.1.0/24的路由，如果开启同步功能的话，RTB向RTD通告该路由时RTD不会将此路由置为有效，RTD也不会向RTE通告该路由，也就避免了上述问题的发生。

在实际环境中AS转接路径中的所有路由器都运行BGP，也就不会出现上述问题，因此可以将同步功能关闭。在具体实现上H3C ComwareV5平台可以支持同步，默认为不同步。

如何实现路由聚合

H3C COMWAREV5平台有两种聚合方式，如下：

- 自动聚合功能

通过summary automatic命令在BGP/BGP VPN视图下配置，默认不使能；自动聚合只聚合通过import-route命令引入的各协议路由（对从邻居收到的BGP路由不生效），且不对缺省路由进行聚合，同时对参与聚合的IGP引入的子网路由会自动进行抑制，从而减少路由选择信息的的数量。这种方法比较死板，而且是按照自然掩码进行聚合，有的时候不能满足需要。

- 手工聚合

通过aggregate在BGP/BGP VPN视图下配置，该命令携带的参数比较多，而且聚合时候灵活多边，可以与路由策略巧妙结合在一起以达到精确控制的目的，具体使用方法可以参看《MSR路由器BGP聚合路由测试经验小结》。

aggregate在手工聚合时候，如果不设置掩码，会以自然掩码进行聚合，这一点尤其要注意。

BGP 通过哪些改变来支持 IPv6

根据RFC2858，BGP4+增加2个新属性MP_REACH_NLRI、MP_UNREACH_NLRI用以支持BGP4＋，在update报文中只有next-hop、aggregator、NLRI三个字段与IPv4有关；继承了BGP的属性以及各种应用规则。

目前 BGP4 ＋的实现是否和 BGP 完全一致

目前H3C COMWAREV5平台全面支持IPv6功能，增加了对BGP4＋的支持，目前重点的特性目前支持BGP4+的团体和反射、单播、组播、路由聚合等功能。

BGP4+是BGP协议的扩展用来支持IPv6地址族，这实际上可以理解为Multi-protocol Extensions for BGP-4（RFC2238）针对IPv6的应用。但是因为下一跳长度等发生变化，

单纯的IP地址变化无法满足实际需求。为此，在UPDATE报文中增加了两项optional non-transitive的路由属性对对应地址族下的路由进行控制，分别是Multi-protocol Reachable NLRI - MP_REACH_NLRI (Type Code 14，十六进制：0x0E)和Multi-protocol Unreachable NLRI - MP_UNREACH_NLRI (Type Code 15，十六进制：0x0F)。其中MP_REACH_NLRI用来发布路由，MP_UNREACH_NLRI用来撤销路由。

BGP 路由如何迭代到等价路由

比如现在有两条缺省路由，一条出接口NULL0，一条是迭代到GE0/1.1上的。BGP路由在迭代时候如何处理？为什么使用display ipv6 relay-tunnel查看迭代计数的时候，只发现基于Null0的统计了迭代次数？

在H3C COMWAREV5平台实际处理中，BGP路由都是迭代到缺省路由上了，而不是直接迭代到GE0/1.1上。只不过有两条等价的缺省路由，所以生成了学过来的每条BGP路由，都多产生一条Derived路由，从而形成等价路由。但实质是迭代到::/上了，所以看到的迭代次数 55::/64仅1次（ipv6 route-static :: 0 55::1 迭代的），::/ 10000次（所有的BGP路由）。目前H3C COMWAREV5平台实现BGP迭代到等价路由可以生成8条等价路由。

BGP Route Decision FAQ

BGP 的选路规则

H3C COMWAREV5平台的选路规则如下：

- 首先丢弃下一跳（Next_Hop）不可达的路由；
- 若配置了Preffered-value值，优选值高的；
- 优选本地优先级（Local_Pref）最高的路由；
- 优选本路由器始发的路由；
- 优选AS路径（AS_Path）最短的路由；
- 依次选择Origin类型为IGP、EGP、Incomplete的路由；
- 优选MED值最低的路由；
- 依次选择从EBGP、联盟、IBGP学来的路由；
- 优选到下一跳Cost值最小的路由；
- 优选Cluster_List长度最短的路由；
- 优选Originator_ID最小的路由；
- 优选Router ID最小的路由器发布的路由。

路由优选时候如何比较 AS-Path

在路由优选的时候，比较的是AS的长度（即个数），并不比较其内容，但是形成等价路由的时候必须要求其内容也一致。

所以在进行BGP方案部署的时候可以通过使用路由策略来增加路由的AS-path或者替代AS-

path内容等手段来实现指定路由的优选、负载分担等目的。在实际应用中经常可以碰到类似情况。

EBGP 路由的 LP 属性如何参与决策

从EBGP邻居收到的路由不会携带LP（本地优先级，IBGP传递路由会携带此属性），那么路由如何参与决策？

答案是如果显示为空，默认以100参与路由优选！

为什么在 BGP/BGP 多实例应用中，没有优选 MED 值最低的路由

MED属性仅在相邻的两个AS之间交换，收到此属性的AS一方不会再将其通告给任何其他第三方AS。通过不同的EBGP学到的目的地址相同的多条路由时，在其他条件相同的情况下，优先选择MED值较小者作为最佳路由。

但需要注意的是，BGP缺省只比较来自同一个AS的路由的MED属性值。H3C COMWAREV5平台可以通过compare-different-as-med命令使BGP比较来自不同AS的路由的MED属性值。

BGP VPNV4 FAQ

标签分配方式除了传统的IGP＋LDP之外还有哪些？

除了传统的IGP+LDP方式分发标签外，Label BGP方法也是一种标签分配方式，并且简单和方便操作，在跨域或者C2C环境中经常可以使用到这种典型配置，通过BGP来分配标签，相邻两台路由器配置如下：

bgp 100 ipv4-family vpn-instance vpn200 peer 2.2.2.2 as-number 200 peer 2.2.2.2 route-policy asbr export peer 2.2.2.2 label-route-capability	bgp 200 undo synchronization peer 2.2.2.1 as-number 100 peer 2.2.2.1 route-policy RT2 export peer 2.2.2.1 label-route-capability
[RT1-bgp]dis route-policy asbr Route-policy : asbr permit : 0 apply mpls-label	[RT2-bgp]dis route-policy RT2 Route-policy : RT2 permit : 0 apply mpls-label

使用 BGP 成功应用标签分配策略，但是路由还是没有分配标签

如上配置BGP已经建立成功，并且策略应用成功，但是查看路由应该被选为最优的却不是最优，检查路由表又没有相同网段路由存在：

```
[H3C-GigabitEthernet0/1]dis bgp vpn vpn vpn200 routing-table label
*       19.19.19.19/32       2.2.2.2          NULL/1025
*       103.103.103.103/32   2.2.2.2          NULL/1024
*>i     107.107.107.107/32   104.104.104.104  1029/1028
```

Label BGP分配标签时候还要求其相应的接口使能MPLS，这样才能正确的形成下一跳的隧道，并正确转发。在接口上使能和去使能mpls功能是测试MPLS L3vpn特性的一种很常见和重要的测试方法；而基于子接口的测试也是常发现问题的手段之一。

为什么优选的 vpnv4 路由插入到 VRF 中后，唯一但不能形成最优

收到一条vpnv4路由，下一跳可达，而且在全局VRF路由表里中也是唯一，但是始终无法达到最优。

```
<H3C>display bgp vpnv4 all routing-table
 Total number of routes from all PE: 1
 Route Distinguisher: 5060:1
    Network            NextHop         In/Out Label     MED        LocPrf
*>i 2.2.2.0/24         102.102.102.102 NULL/1027        0          100
 Total routes of vpn-instance vpn200: 6
    Network            NextHop         In/Out Label     MED        LocPrf
* i 2.2.2.0/24         102.102.102.102      NULL/1027        0          100
*>  107.107.107.107/32 9.1.1.2              1028/3           0
```

VRF中的路由在优选的时候，对于普通L3VPN，H3C COMWAREV5平台的实现是必须保证其下一跳的隧道的存在，通过使用命令display tunnel-info all命令可以查看目的网段102.102.102.102的隧道是否存在，如果不存在则不会最优，请检查标签分配LDP或者BGP的配置。

VPNV4 路由经策略改变扩展团体属性不会生效

比如存在 vpn — a 和 vpn — b，相应接受 VT 属性为 1：1 和 2：1。在 vpn — a 视图下通过 export route-policy 命令强制改变本 vpn 路由的 VT 属性增加 2：1，但是 vpn — b 实例不会接受到这些 vpnv4 路由。

H3C COMWAREV5平台目前实现是通过vpnv4获取的路由在刚开始接受时即会检查其VT属性然后决定下发到哪些vpn中，如果是在接受到VPNV4路由后再改变其VT属性不会影响以前所做的决定。当然，如果是VRF本身获取的路由，改变其属性会影响到路由下发其它VRF中。

如何将 VPN 路由发布到其他 VPN 并进行策略控制

缺省情况下，某个 VPN 路由不会发布到其他 VPN 中，可以通过在 VPN 实例视图下配置 vpn-target 命令将当前 VPN 实例的路由发布到其他 VPN 实例或将其他 VPN 实例的路由引入到当前 VPN 实例。

另外，还可以使用系统视图----IP VPN实例视图下的import route-policy、export route-policy命令，以比采用扩展团体属性更精确地方式控制发布VPN实例路由。

BGP Route-Policy FAQ

路由策略支持哪些过滤规则

H3C COMWAREV5平台支持丰富的路由策略来控制路由的接受和发送，针对bgp对等体或者对等体组有如下方式：

1）as-path-acl，AS路径过滤控制列表；

2）ip-prefix，IP前缀列表(支持IPv6，即ipv6-prefix)；

3）route-policy，路由策略；

4）filter-policy(advanced acl)，路由应用过滤策略；

其中route-policy又支持多种控制规则，比如：

1）if-match as-path，匹配AS路径列表；

2）if-match community，匹配团体属性列表；

3）if-match extcommunity，匹配扩展团体属性列表；

4）if-match cost，匹配路由MED；

5）if-match interface，BGP不支持这种过滤方式；

6）if-match mpls-label，BGP支持，通过BGP分配标签可以代替IGP＋LDP模式，在L3vpn的c2c以及跨域中得到大量应用；

7）if-match acl（advanced acl），匹配访问控制列表；

8）if-match ip/IPv6，匹配下一跳，可以指定acl或者地址前缀列表；

9）if-match ip-prefix，匹配地址前缀列表，同样也支持IPv6。

路由策略的基本匹配规则有哪些

BGP号称路由中的王者，有很大一部分功劳归功于路由策略，可以说是其左右臂膀之一。针对路由策略的使用，各个厂商有其各自的规则。H3C COMWAREV5平台的路由策略配置和使用都相对比较简单，只要掌握以下几条基本原则，相关的问题就会迎刃而解。

1）一个Route-policy的所有NODE之间是"或"的关系

2）一个NODE内部所有"if-match"之间是"与"的关系

3）一个"if-match"内部的所有参数之间是"或"的关系

简单来说，一个Route-policy可以由多个节点（node）构成，每个节点是进行匹配测试的一个单元，节点间依据顺序号（node-number）进行匹配。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。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。apply子句指定动作，也就是在通过节点的匹配测试后所执行的动作——对路由信息的一些属性进行设置。

一个Route-policy的不同节点间是“或”的关系，系统依次检查Route-policy的各个节点，如果通过了Route-policy的某一节点，就意味着通过该Route-policy的匹配测试（不进入下一个节点的测试）。

而对于某些if-match子句，后面可以跟多个同类的并列参数，这些参数之间是“或”的关系，即满足其中一个参数的值，就满足了该if-match子句。

例如下面的配置：

```
route-policy 1 permit node 1
 if-match cost 20
 if-match route-type internal external-type1
route-policy 1 permit node 2
 if-match cost 30
```

在route-policy 1中配置了两个节点node 1和node 2，而在不同的node中配置了不同的if-match子句。

很容易可以看出，满足node 1的条件是cost为20并且路由类型为OSPF内部或者外部type1路由。即对于if-match route-type internal external-type1来说，由于internal和external-type1是同一个if-match子句中多个并列参数，所以它们之间是“或”的关系，只要类型为internal或者external-type1的路由均算满足该if-match子句。

而对于node 1来说，它存在多个并列的if-match子句，它们之间是“与”的关系，所以必须同时满足

```
if-match cost 20
if-match route-type internal external-type1
```

这两个条件才算正在通过node 1的测试。

而对于node 2而言，只有没有通过node 1检测的情况下才会发挥作用，否则通过了node 1的检测就不再进入node 2的检测了。

注：如果node中的if-match条件匹配成功且if-match的条件是DENY，则不论node配置的是permit和deny继续匹配下一个node；如果所有的node都没有匹配成功，则按照DENY处理。对于不存在的路由策略默认通过！

匹配了前缀列表，为什么还是没有对端发送过来的路由

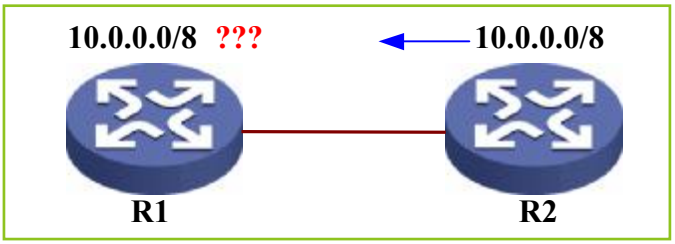


图3 BGP前缀过滤示例

如图 3，R1 上配置如下：

```
Peer X.X.X.X ip-prefix 1 import
ip ip-prefix 1 index 20 permit 10.0.0.0 16
```

很显然，配置的前缀列表是错误的，应该是permit 10.0.0.0 8，当进行前缀控制列表匹配的应该要注意掩码和规则的逻辑关系。

为什么给等体配置路由策略apply community后，对端收到的路由属性里却没有任何团体属性？

如图3，R2上配置如下：

```
Peer X.X.X.X route-policy 1 export
Route-policy : 1
  permit : 0
         apply community 1 2 3
         apply extcommunity rt 0.0.0.0:0
```

R1收到路由后不会具备团体属性和扩展团体属性，为什么了？BGP默认是不发送团体属性和扩展团体属性（当然vpnv4默认发送扩展团体属性）的，所以要想将这类属性发送出去必须针对指定的对等体或者对等体组设置命令：

```
Peer X.X.X.X advertise-community，发送团体属性；
Peer X.X.X.X advertise-ext-community，发送扩展团体属性，两者没有耦合关系。
```

如何使用正规则表达式通过 AS-path 进行路由控制

使用as-path控制列表来进行路由控制是比较复杂且难以记忆的控制方法，主要是正规则表达式的使用。

BGP Reflector FAQ

反射有什么特点，如何配置 BGP 反射

在一个AS内，IBGP必须要求在逻辑上是全连接的，但随着网络拓扑的日益复杂，IBGP的全网连接开销很大，为了解决这个问题，引入了反射机制。路由反射器概念的基本思路是：指定一个集中路由器作为内部对话的焦点。多个BGP路由器可以与一个中心点对等化，然后多个路由反射器再进行对等化。路由反射器的特点：简易理解、移植方便（不用更改现有网络拓扑结构）、兼容性好（不用所有的路由器都支持反射机制，反射器对于客户来说是透明的）

在RFC2796中规定：“In addition, when a RR reflects a route, it should not modify the following path attributes: NEXT_HOP, AS_PATH, LOCAL_PREF, and MED. Their modification could potential result in routing loops.”即反射器反射路由时，不应该修改NEXT-HOP,AS_PATH,MED以及LOCAL_PRE属性。同时在反射器上即使应用路由策略修改属性，新的属性也不应该应用到被反射的路由上。

H3C COMWAREV5平台反射器支持普通BGP、BGP4+、VPNV4、BGP VPN，在指定视图下进行如下配置：

```
reflector cluster-id 4294967295          //反射器ID
peer 104.104.104.104 reflect-client    //指定IBGP peer作为反射器客户端
reflect between-clients                 //默认已配置,反配置则取消反射功能
```

什么是冗余反射器和嵌套反射器

反射的配置相当灵活，除了普通的配置方案外，为了加强反射技术的健壮性和灵活性，还可以配置冗余反射器和嵌套反射器：

由于AS域内逻辑结构的改变，反射器成为路由发布的瓶颈，一旦反射器出问题，那么整个域内的路由传递就会受到很大的影响，在这种情况下，可以通过配置冗余反射器来解决，即：一个群内存在一个以上的反射器，各反射器CLUSER_ID是一样的，都与客户进行全连接，当一台反射器出问题时，另一台反射器仍能正常工作，相当于备份功能。冗余反射的概念可以进一步参考下文。

除此以外，还可以配置嵌套反射器，即在一个群内嵌套配置一个反射群，反射群与该群的CLUTER_ID是不同的。嵌套反射器在VPNV4用的比较多，比如MPLS L3vpn环境中，通过多级反射来分担PE的压力。

为了避免路由环路，引用了originator-id属性和cluster-list属性，originator-id属性是由反射器产生的，它的值是始发这条路由的邻居的router-id；cluster-list也是由反射器产生的，反射器如果发现update报文中有了cluster-list属性，就把自己的cluster-id添加到后面；如果没有，就创建一个cluster-list属性，把自己的cluster-id放到上面，再向其他邻居发布；如果发现与本地雷同，则会丢弃该路由以避免环路。cluster-id的值可以在反射器上配置，如果没有配置，缺省使用反射器的router-id。

为什么收到携带含有与本地 Router ID 相同的 originator-id 属性路由后会丢失

如图4，R1和R2为RR，R3和R4为RRC，R4发布一条路由，R2收到了，但是R1和R3都没有收到。为什么？

发射路由在发送过程中会携带一个originator-id属性和一个cluster-list属性。其中originator-id的值是始发路由器的ID，cluster-list的值为沿途反射器的cluster-id。当客户机收到反射路由后会检查路由的这两个属性，如果在收到的路由中的originator-id属性中发现了自己的Router ID，就会拒绝该路由。这里原因为R1和R4的BGP进程具备相同的ID。

为什么收到携带含有与本地 cluster-id 相同的 cluster-list 属性路由后会丢失

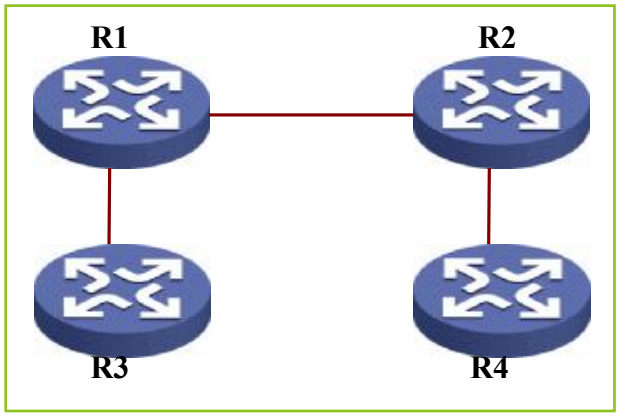


图4 BGP反射示例

如图4，R1和R2为RR，R3和R4为RRC，R4发布一条路由，R2和R1都收到了，但是R3没有收到。为什么？

发射路由在发送过程中会携带一个originator-id属性和一个cluster-list属性。其中originator-id的值是始发路由器的ID，cluster-list的值为沿途反射器的cluster-id。当客户机收到反射路由后会检查路由的这两个属性，如果在收到的路由中的cluster-list属性中发现了自己的cluster-id，就会拒绝该路由。所以原因为R1和R2设置了相同的cluster-id，R1收到

R2反射过来的路由后，会将路由丢弃而不会转发到R3。

如果R1和R2具备相同的cluster-id，而R3还要收到R4路由的话，可以用到前面提到的冗余反射概念，将R3也连到R2上，这样R1和R2都是RR而且具备相同的cluster-id，形成冗余反射环境，R4的路由会直接发射到R3上。

路由反射遵循哪些原则

反射器的IBGP邻居有两类：客户和非客户邻居，反射器同客户一起形成一个群，群内的客户不应再与群外的BGP邻居形成IBGP连接。一个AS内所有的路由反射器和非客户机构成全闭合网。

1）反射器从非客户收到的路由发向所有客户；

2）由客户收到的路由会发向所有客户以及非客户（包括发送者本身）；

3）由EBGP邻居收到的路由发向所有客户以及非客户。

反射路由的属性不应该被改变

被反射的路由其属性不应该被改变，比如在测试中经常忽略的联盟属性等都不应该被反射器改变。

配置BGP路由反射，可以减少IBGP连接，反射到客户端的路由要在CLUSTER_LIST由属性中添加自身的cluster-id，但cluster-id的配置也不是必需的。当BGP配置reflector cluster-id后，即采用所配置值，当没有配置该值时，BGP将把local router id添加到对应CLUSTER_LIST路由属性中。

BGP Confederation FAQ

如何配置联盟以及联盟的作用

在RFC3065中定义：“This document describes an extension to BGP which may be used to create a confederation of autonomous systems that is represented as a single autonomous system to BGP peers external to the confederation, thereby removing the "full mesh" requirement. The intention of this extension is to aid in policy administration and reduce the management complexity of maintaining a large autonomous system.”可见联盟同反射类似，都是为了解决大规模网络中IBGP全网连接的问题。联盟的概念基于一个AS可以被分为多个子AS，子AS内使用IBGP全闭合网，子AS之间以及联盟本身与外部AS之间使用特殊的EBGP连接。虽然子AS之间的路由经EBGP交换，所有的IBGP规则仍然适用，因此对于AS外的路由器来看一个联盟就象一个单一的AS。EBGP下个中继、量度值和本地优先值仍然在内传送。

参与联盟的路由器一般遵循如下配置：

```
confederation id 6500          //大AS号，一个联盟内一致,不能与本地AS号相同
confederation peer-as 600     //本地相连子AS的AS号
```

联盟新增的两个属性

在RFC3065中新增加了两个为联盟定制的属性，即：

1. AS_CONFED_SEQUENCE: ordered set of Member AS Numbers in the local confederation that the UPDATE message has traversed;

2. AS_CONFED_SET: unordered set of Member AS Numbers in the local confederation that the UPDATE message has traversed。

增加这两种属性是为了防止联盟内部的环路。

AS-PATH 参数在联盟中如何进行传递

对于AS_CONFED_SEQUENCE和AS_CONFED_SET，联盟内处理方式大致与AS_SEQUENCE和AS_SET相同，同时：

1. 当路由在联盟内子自治系统间传递时，不应修改AS_PATH属性。

2. 当路由在联盟内子自治系统间传递时：

a) 若第一个AS_PATH是AS_CONFED_SEQUENCE，BGP将自己的子自治系统AS号加在最左端。

b) 否则，创建一个AS_CONFED_SEQUENCE，包含自己的子自治系统AS号。

3. 当向联盟外EBGP传递路由时：

a) 若第一个AS_PATH是AS_CONFED_SEQUENCE，将后续的AS_CONFED_SEQUENCE和AS_CONFED_SET删除，至b)；

b) 若第一个AS_PATH是AS_SEQUENCE，则将联盟AS加在最左端；

c) 若第一个AS_PATH是AS_SET，增加一个AS_SEQUENCE，将联盟AS加在最左端。

4. 对于本地初始路由的传播：

a) 向本自治系统内IBGP发送，空的AS_PATH属性；

b) 向联盟内，本自治系统外EBGP发送，带有AS_CONFED_SEQUENCE属性；

c) 向联盟外EBGP发送，带有AS_SEQ属性。

H3C COMWAREV5 平台的 confederation nonstandard 命令有什么用处

RFC1965中规定：AS-PATH Segment Type 3是AS_CONFED_SET属性，Type 4是AS_CONFED_SEQUENCE属性。而过去C友商把Type 3作为AS_CONFED_SEQUENCE属性，Type 4不使用。这样导致其发送的BGP update报文中，联盟的AS-PATH属性的格式和RFC不一致，导致互通过程我司不能识别合法的带有联盟的AS-PATH属性的BGP报文。过去为了达到互通的问题，需要配置confederation nonstandard命令以兼容友商的处理。

BGP ECMP（负载分担） FAQ

为什么目的网段相同的 BGP 路由在设置 balance 以后还是无法形成等价路由

H3C COMWAREV5平台等价BGP路由的实现有着自身的实现，具体如下：

1）参与BGP负载分担特性的路由必须为有效路由；

2）参与负载分担的BGP路由ORIGIN，LOCAL-PREFERENCE，MED以及AS-PATH路径属性必须相同。BGP根据路由来源分为IBGP学到的路由，EBGP学到的路由，NETWORK命令引入路由，IMPORT-ROUTE命令引入路由，自动聚合路由以及手动聚合路由。不同起源之间的路由不形成负载分担；

3）来源不同的BGP路由之间不形成负载分担；

4）标签路由与非标签路由之间不形成负载分担，标签路由是指遵循RFC3107的BGP公网带标签路由；

5）反射路由和非反射路由之间不形成负载分担；

6）下一跳相同的BGP路由不形成负载分担；

7）转发路由时，多条等价路由只随机选取一条路由并向外发送；

在保证如上规则后，还需要在BGP视图或者BGP VPN视图配置等价负载分担命令balance，默认不进行负载分担，H3C COMWAREV5平台目前支持最大等价路由数目为8条。

等价 BGP 路由下一跳设置

IBGP负载分担路由在配置反射的情况下向IBGP邻居转发等价路由时，不改变下一跳，下一跳为选中的等价路由初始下一跳；其他情况下，下一跳为形成负载分担的BGP本地地址。

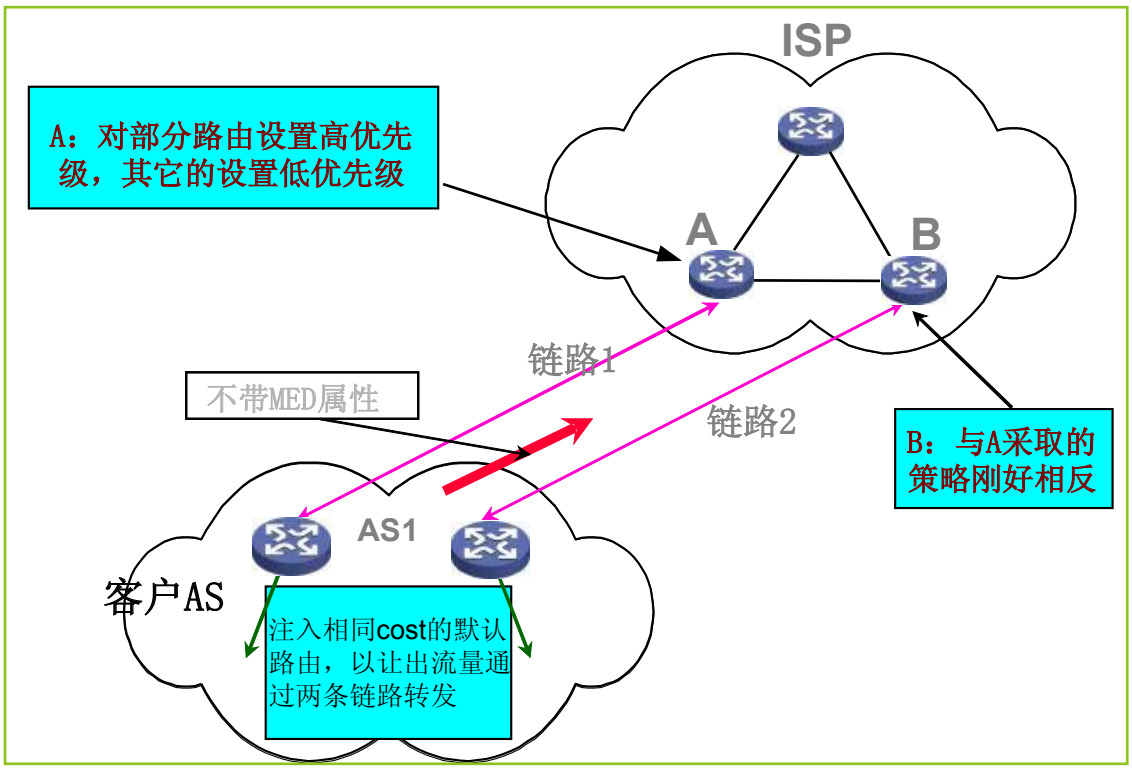


图5 BGP负载分担

如何通过 BGP 实现负载分担

BGP默认不形成等价路由。当存在等价路由后，在BGP或者BGP vpn视图下设置balance命令，可以使能等价路由功能。等价路由的形成具有很多限制，可以参考前面两节。

理论上对等体之间建立多个peer可以很容易形成等价路由，但是要注意这样带来的环路影响。同时通过引入IGP路由在自治域间形成等价路由也是比较常见的方式。比如在Multihomed AS拓扑中常会用到负载分担特性，当然这种简单的负载分担是不区分流量和业务，而是统一分配。

还有一种负载分担方法即根据不同业务和流量进行整体上的负载分担。如图5，针对不同业务X和Y的路由设置不同优先级，导致业务X的流量从link1通过，业务Y的流量从link2通过。

如何通过 BGP 实现链路备份

对于来自域间的路由，在进入本地AS系统后常会通过设置边界路由器的本地优先级，导致路由在进行选择的时候存在主备，而边界路由器之间存在备份，在全连接中经常用到这种备份方法。如图6所示，customer的聚合路由通过两条路径发送到ISP后在两台边缘路由器上会形成两条路由，但是由于优先级的不同在传递到最上面的ISP路由器后会进行优选导致有主备之分，这样两者之间就建立了备份链路。

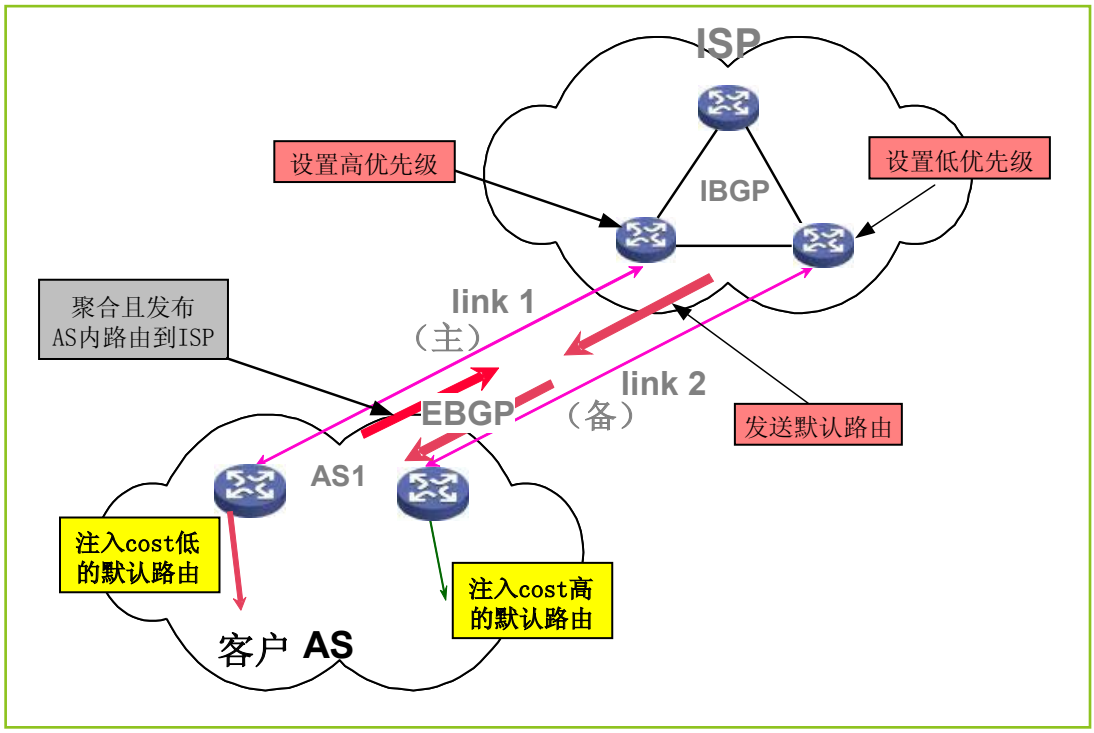


图6 BGP链路备份

团体属性

文/陈磊

引子

BGP协议将路由学习和路由策略很好的融合在了一起。团体属性的引入就是为了更好的使用路由策略功能。

团体属性的定义很简单，但是要想将之应用到真实的网络中，却是一个需要创造力和有着无限可能性的工作。本文是对于团体属性应用的一个简单介绍。

团体属性和扩展团体属性的定义

团体属性

团体属性可以添加在每一个路由前缀中，由RFC1997定义，是一个transitive optional属性。包含有团体属性的路由，表示该路由是一个路由团体中的一员，该路由团体具有某种或多种相同的特征。根据这些特征区分不同的路由，可以大大简化路由策略的配置工作，同时也增强路由策略的能力。

例如，一个ISP可以给自己所有的customer路由指定一个具体的团体属性，这样，学习到该路由的路由器要想给这些路由指定MED或者LOCAL_PREF等属性时，直接基于该团体属性进行操作，而不需要一条路由一条路由的去指定。

团体属性的Type code是8，32个bites长。可以解析为一个10进制的数，也可以解析为AA:NN的格式。RFC中规定，16个bites作为AS number，后16个bites由该AS自己使用。同时，这32个bites开头的部分0x00000000——0x0000FFFF和结尾的部分0xFFFF0000——0xFFFFFFFF被保留。

RFC1997还规定了几种公认的团体属性：

INTERNET：默认的团体属性，所有路由都属于该团体。

NO_EXPORT（4294967041，or 0xFFFFFF01）：含有该属性的路由不向任何联盟外的EBGP邻居发送，如果没有定义联盟，则认为该AS是一个独立的联盟。例如，大量的没有必要透传到internet的IP子网路由，可以标记该团体属性，以控制一些不需要的路由的规模。

NO_ADVERTISE（4294967042，or 0xFFFFFF02）：含有该属性的路由不向任何BGP邻居发送，包括EBGP和IBGP。

Local-AS（4294967043，or 0xFFFFFF03）：也称作NO-ADVERTISE-SUBCONFED：含有该属性的路由，不向任何EBGP邻居发送，包括联盟内的EBGP邻居。

设备收到带有这几个公认的团体属性的路由，是自动按照RFC规定来执行的，不需要再配置路由策略。

扩展团体属性

因为团体属性的使用越来越丰富，原有的32bite定义已经不能满足各种应用。应运而生的就是扩展团体属性。使用了新的Type code和格式，在RFC4360中定义。比起原来的团体属性，扩展团体属性提供了更长的取值范围，以减少冲突的可能；同时，还增加了一个Type字段，可以使得路由策略直接基于扩展团体属性的type字段进行操作。相当于将一些原来需要通过复杂的团体属性配置才能实现的功能，直接添加到了扩展团体属性的结构中。

扩展团体属性也是transitive optional，Type code是16，64个bites长，结构如下：

```
- Type Field: 1 or 2 octets
- Value Field: Remaining octets
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |  Type high    |  Type low(*)  |                               |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+          Value                |
        |                                                               |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type字段使得团体属性的应用更为灵活：

Type high字段的bit 0，表示该扩展团体属性是否是在IANA注册过的公认属性；

Type high字段的bit 1，表示该扩展团体属性的转发范围，0表示可以跨AS；1表示不能，只能在本地AS中使用。

扩展团体属性分为两种：regular type和extend type：regular type的type字段8bites长（只包含type high），extend type的type字段16bites长（type high和type low都包含）

RFC4360中给出了具体的扩展团体属性各字段的定义以及若干种应用模板，这里着重要注意的是已经得到了广泛应用的Route Target Community：在MPLS VPN应用中，RT团体属性来区分不同VRF的路由，路由器通过RT中的内容，判断该路由是否需要添加到相应的VRF中。

团体属性的应用

虽然RFC中规定了部分公认的团体属性，但是大部分情况下，团体属性都是由每个网络运营者自己定义规则和应用方法，然后供自己或者自己的客户使用。

一般情况下，团体属性承载了如下两方面的内容：

第一种是针对路由发送者，添加了一些路由的相关信息：例如路由是怎么学习到的，从哪里学习到的。这类内容可以给网络中的路由的使用者提供更多的信息进行路由选择；

第二种是针对路由接收者，通知接收者应该对该路由进行那些操作：例如接收者可以/不可以接收这些路由，接受者应该对这条路由的属性进行某些修改。

两方面的内容可以独立使用，也可以混合在一起。确定具体的承载内容和格式是一个很需要些创造力的工作，由网络运营商自行确定。

有时为了承载更多的不同类型的信息，会将多种含义融合到同一个团体属性中，匹配时会使用正则表达式。

下面是几个具体的应用举例。

AS 内部使用团体属性

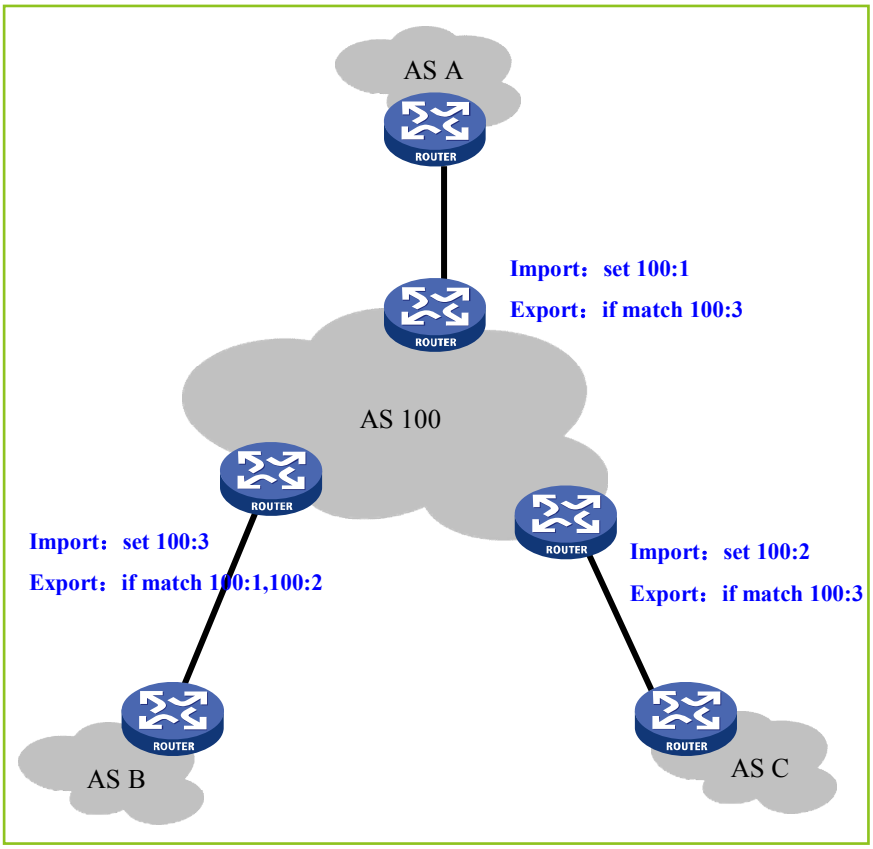


图1 AS内部使用团体属性示意图

首先，看一个典型的团体属性作为路由信息补充的例子：团体属性用来标识每条路由的来源，用于帮助路由接收者进行选路。

图1中，AS 100作为供应商，与三个客户AS A，B，C分别建立BGP连接关系。其中，三个客户基于自己的应用，对供应商提供给自己的路由提出了要求：AS C希望学习到AS A和B的所有路由；AS A和B都只希望学习到AS C的路由。

供应商AS100通过使用团体属性，可以很容易的实现这点：

首先，在AS100内，使用团体属性的个位数表示路由是来自那个AS：所有从AS A收到的路由增加一个团体属性100:1，所有从AS B收到的路由增加一个团体属性100:2，所有从AS C收到的路由增加一个团体属性100:3；

之后，按照上面的要求，在与AS C邻接的设备上配置允许发送团体属性为100:1，100:2的路由，在与AS A邻接的设备和与AS B邻接的设备上配置允许发送团体属性为100:3的路由。

将上面的例子扩展一下，如果对于整个internet上的不同节点根据地理位置编号，再添加到路由的团体属性中，就可以知道每条路由来自何方了。

同时，在进行路由引入操作时，我们也可以用团体属性标记该路由是从那个IGP引入的。例如，使用团体属性的百位数表示引入路由的IGP：100:100表示是从OSPF引入的路由，100:200表示是从RIP学习到的路由。这样，路由接收者可以方便的根据路由的来源，作为选路的一个标准。

更进一步，将上面的两种团体属性在规则上进行一下组合，我们可以用100:101表示从A学习到的由OSPF引入的路由。对于这种相对比较复杂的团体属性的匹配，我们都可以通过正则表达式来实现。

AS 间使用团体属性

在AS之间进行流量控制时，有多种方法，例如添加AS_path，使用MED，或者只是简单的发布掩码更长的路由。而团体属性，也可以很方便的告知邻接的AS，路由应该被如何如何处理。

多归属组网中作为DPA使用

这里我们看一个路由多归属网络的应用：用户同时在多个网络供应商处有出口，作为流量分担和备份。此时，为了实现用户流量的备份和流量分担，需要供应商和用户之间进行一系列的路由策略的交互。考虑一个供应商往往服务大量的用户，对于每个用户都需要若干路由策略的配置，这将是一个庞大复杂的工作，而且很容易引发问题。

为了更好地优化这个问题，RFC1998中定义了一种使用团体属性作为DPA（Destination Preference Attibute，目的优先级属性）的应用。也就是在团体属性中体现路由的优先级，然后路由器在给该路由分配Local_Pref时，依据这个DPA来分配。

RFC中的定义了如下的团体属性与Local_Pref的对应关系：

客户的路由	Community	Local_Pref
客户主用路由	供应商ASN:100	100
客户备份路由	供应商ASN:90	90
从其他ISP学习到的客户路由	供应商ASN:80	80
客户提供的其他客户的备份路由	供应商ASN:70	70

考虑如图2中的组网，AS99是用户，与两个供应商AS100和AS200相连。用户希望实现路由的分担和备份：1.1.1.0/24和2.2.2..0/24是用户的两个网段，AS100为1.1.1.0/24的主用路径，AS200为2.2.2.0/24的主用路径。

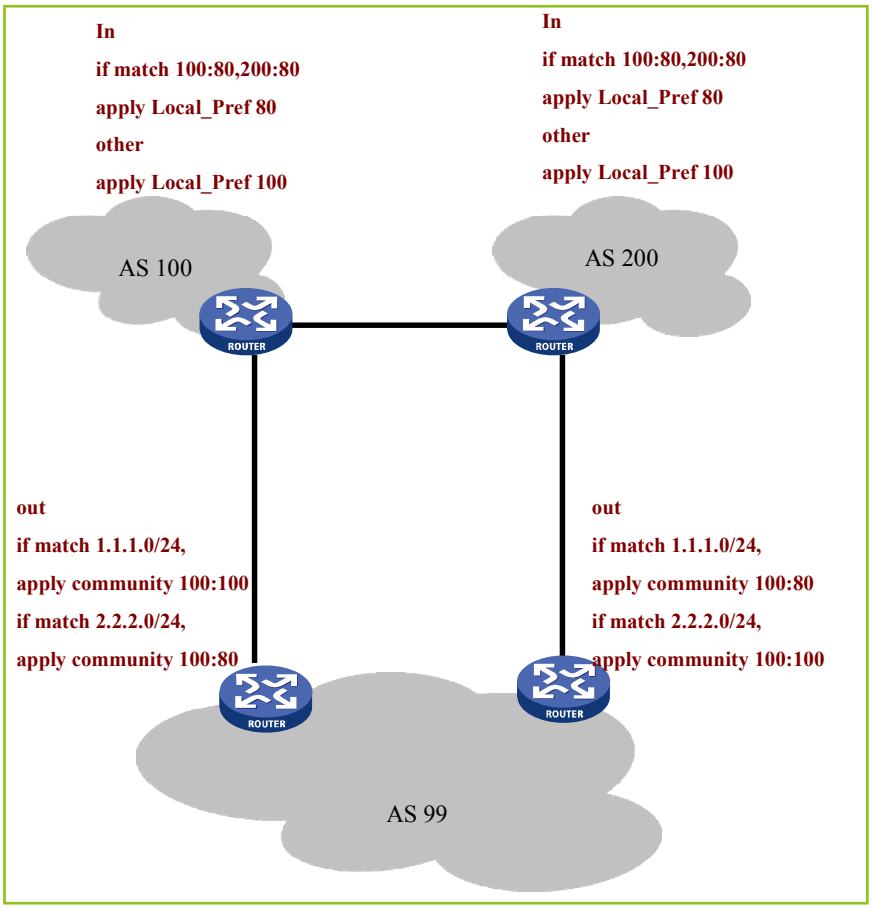


图2 多归属组网中作为DPA使用示意图

具体的实现方式，图2中已经标注：整个过程中，community作为一个路由的优先级使用，由客户主导。客户在向两个AS发送路由时，根据网络规划，分别给予1.1.1.0/24和2.2.2.0/24不同的community值。而两个供应商AS内，则根据不同的community值，指定该路由的Local_pref，从而实现流量分担和路由备份。

防DOS攻击应用

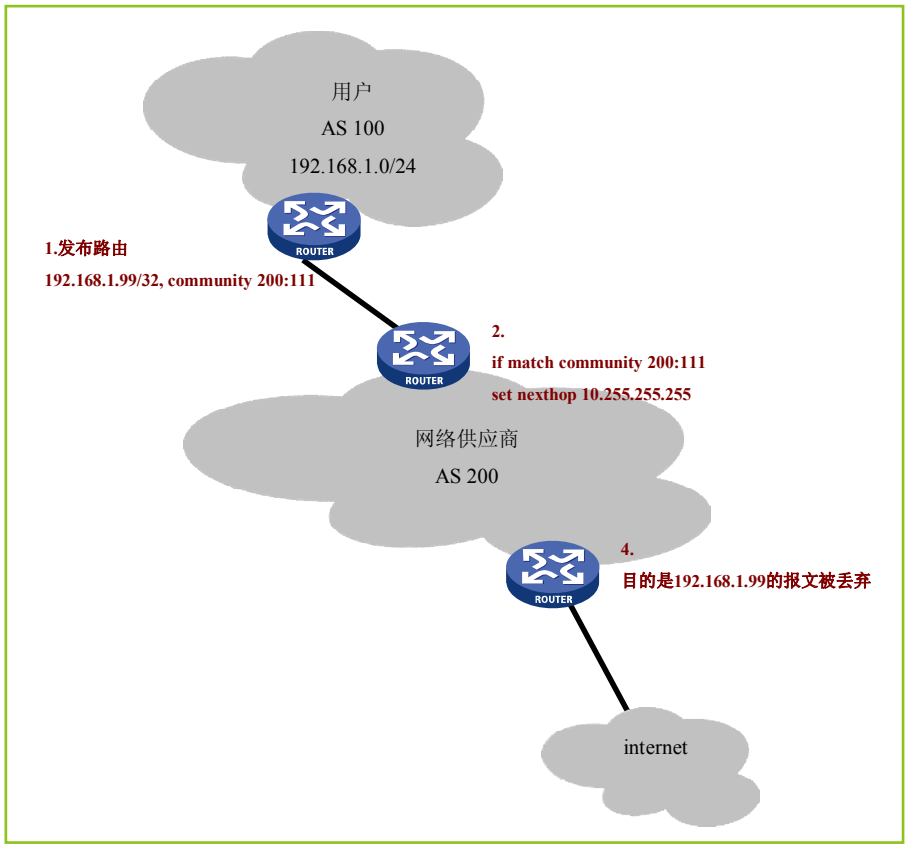


图3 防DOS攻击应用示意图

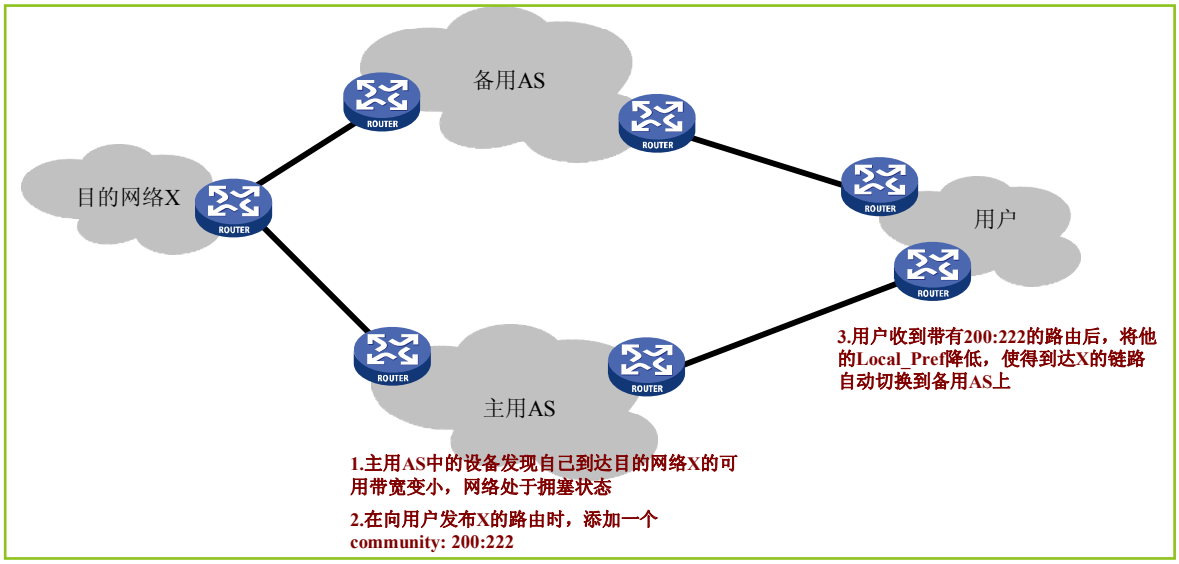


图4 传递网络拥塞信息/带宽信息的应用示意图

当用户的某一台主机（192.168.1.99）受到DOS攻击时，用户向供应商发布一条该受攻击设备的主机路由，community为与网络供应商事先协商好的值（200:111）。

供应商收到匹配该community的路由后，将路由的下一跳指向Null接口。

这样，供应商就可以在自己的边缘设备商直接丢弃所有指向192.168.1.99的报文。

传递网络拥塞信息/带宽信息的应用

图4中的应用中，用户有两条到达目的X的路径。当主用路径上的路由器发现自己到达X的链路可用带宽减小，或者处于拥塞状态时。他通过一个community值，将这个链路状态信息传递给用户。

用户收到带有该community值的路由后，自动降低其Local_pref，使得通往X的数据自动切换到备用AS上，从而实现了网络拥塞信息/带宽信息在设备之间的传递。

团体属性使用建议

团体属性极大地强化了BGP，它使得BGP除了路由功能以外，还添加了信息传递和策略指定的功能。如果可以合理地进行团体属性的部署，不仅可以有效地管理网络，还提供无限的可能性，来满足用户不同的需求。

这里，有几条团体属性使用的建议：

- 给网络选择一组内部使用的团体属性：可以合理地表现网络的拓扑和特征。因为网络供应商要么不提供团体属性，要么就是太简单，要么太繁琐，不适用于内部使用
- 证团体属性配置的简洁：过于复杂的团体属性结构，会要求在路由器上进行繁杂的路由策略配置。以至于很难进行问题定位
- 避免将从自己的邻居收到的不认识的团体属性转发给其他AS：它可能给你的网络流量带来不可控和不可知的潜在危险
- 所有内部和外部团体属性，都需要完整的文档记录

BGP路由聚合

文/孙丽

路由聚合的必要性

在大规模的网络中，BGP路由表变得十分庞大，存储路由表占有大量的路由器内存资源，传输和处理路由信息所必须的带宽和路由器传送与处理路由信息需要大量的资源。使用路由聚合（Routes Aggregation）可以大大减小路由表的规模。通过对路由的条目的聚合，隐藏一些具体的路由减少路由震荡对网络带来的影响。BGP路由聚合结合灵活的路由策略，从而使BGP更有效的传递和控制路由。

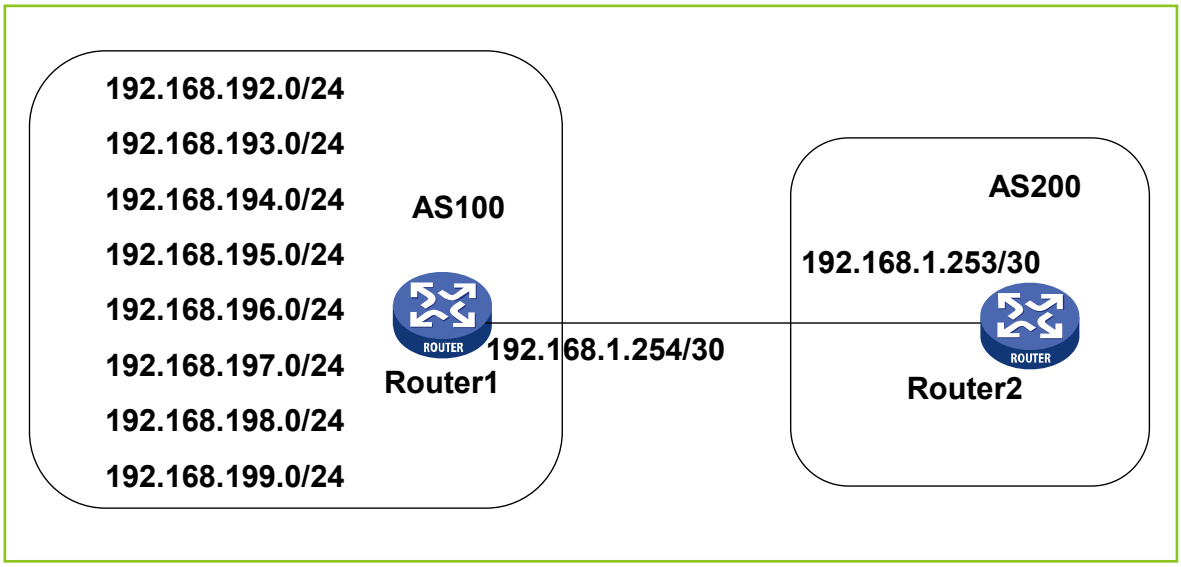


图1 AS100中所有的内部网络都可以聚合成单一的地址192.168.192.0/21

路由聚合的方法

聚合路由的方式：通过与静态路由组合进行路由聚合，自动聚合，手动聚合。

结合静态路由对具体路由条目进行聚合

为聚合路由建立一条静态路由，然后用network命令公布出去，

通过network命令公布的静态条目在BGP下生成一个聚合地址：

```
#
bgp 100
 network 192.168.192.0 255.255.248.0
 undo synchronization
 group ex external
 peer 192.168.1.253 group ex as-number 200
#
ip route-static 192.168.192.0 255.255.248.0 NULL 0
```

静态路由指向null0，因为聚合路由本身不是一个实际的目的地，在Router1中它只是用来代表更具体的地址。目的地址属于AS100中的C类地址的数据包与AS100外部一个路由器上的聚合地址相匹配，并且被转发到Router1。在Router1处，数据包匹配更具体的路由，并被转发到正确的下一跳路由器。如果Router1上的更具体的路由不存在，这些数据包将被丢弃。

自动聚合

自动聚合是按照自然网段进行聚合，而且只能对IGP引入的子网路由进行聚合，对从邻居学习来的路由和通过netwrok命令生成的BGP路由不起作用。命令为：

```
summary automatic
```

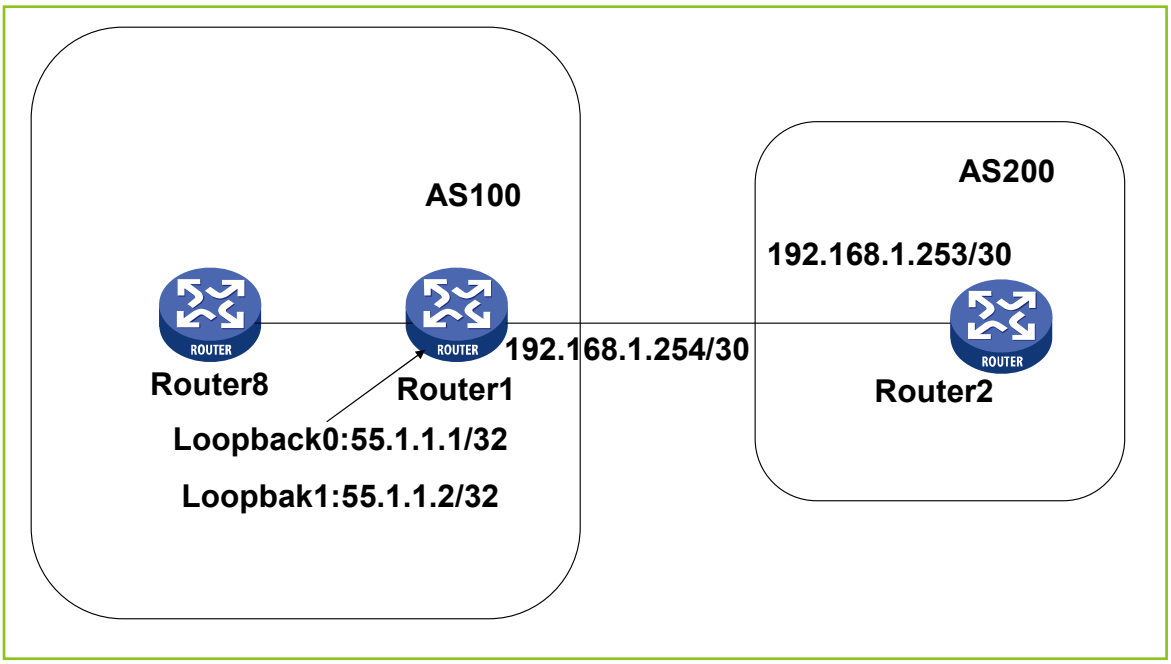


图2 自动聚合

如图2：Router1 和Router2建立起EBGP邻居，AS100内运行的IGP是rip。

Router1从Router8学习到的rip路由，引入到BGP中发布出去。

Router1上的loopback1和loopback2接口的所在的网段通过network发布出去.

```
[Router1] display  rip 1 route
 Route Flags: R - RIP, T - TRIP
              P - Permanent, A - Aging, S - Suppressed, G - Garbage-collect
 ------------------------------------------------------------------------------
 Peer 82.112.2.190  on Ethernet6/0
    Destination/Mask        Nexthop        Cost     Tag    Flags    Sec
    192.168.192.0/24      82.112.2.190       1        0     RA       26
    192.168.193.0/24      82.112.2.190       1        0     RA       26
    192.168.194.0/24      82.112.2.190       1        0     RA       26
    192.168.195.0/24      82.112.2.190       1        0     RA       26
    192.168.196.0/24      82.112.2.190       1        0     RA       26
    192.168.197.0/24      82.112.2.190       1        0     RA       26
    192.168.198.0/24      82.112.2.190       1        0     RA       26
    192.168.199.0/24      82.112.2.190       1        0     RA       26
        109.0.0.0/24      82.112.2.190       1        0     RA       26
        109.0.1.0/24      82.112.2.190       1        0     RA       26
        109.0.2.0/24      82.112.2.190       1        0     RA       26
```

Router1将rip路由引入BGP，BGP的相关配置如下：

```
[Router1-bgp]display  this
#
bgp 100
 network 55.1.1.1 255.255.255.255     \\loopback1接口的地址
 network 55.1.1.2 255.255.255.255     \\loopback2接口的地址
 import-route rip 1                   \\引入rip路由
 undo synchronization
 peer 192.168.1.253 as-number 200
 peer 82.112.2.190 as-number 900
```

我们先来看一下没有配置聚合前的Router1和Router2的BGP路由表:

Router1的BGP路由表

```
 [Router1]display bgp routing-table
 Total Number of Routes: 21
 BGP Local router ID is 213.168.133.93
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network            NextHop            MED        LocPrf     PrefVal Path/Ogn

 *>  55.1.1.1/32        0.0.0.0            0          0             i
 *>  55.1.1.2/32        0.0.0.0            0          0             i
 *>  83.100.1.0/30      82.112.2.190                  0           900?
 *>  83.100.1.4/30      82.112.2.190                  0           900?
 *>  83.100.1.8/30      82.112.2.190                  0           900?
 *>  83.100.1.12/30     82.112.2.190                  0           900?
 *>  83.100.1.16/30     82.112.2.190                  0           900?
 *>  83.100.1.20/30     82.112.2.190                  0           900?
```

（接下页）

（接上页）

```
 *>  83.100.1.24/30      82.112.2.190                0              900?
 *>  83.100.1.28/30      82.112.2.190                0              900?
 *>  109.0.0.0/24        0.0.0.0          1          0              ?
 *>  109.0.1.0/24        0.0.0.0          1          0              ?
 *>  109.0.2.0/24        0.0.0.0          1          0              ?
 *>  192.168.192.0       0.0.0.0          1          0              ?
 *>  192.168.193.0       0.0.0.0          1          0              ?
 *>  192.168.194.0       0.0.0.0          1          0              ?
 *>  192.168.195.0       0.0.0.0          1          0              ?
 *>  192.168.196.0       0.0.0.0          1          0              ?
 *>  192.168.197.0       0.0.0.0          1          0              ?
 *>  192.168.198.0       0.0.0.0          1          0              ?
 *>  192.168.199.0       0.0.0.0          1          0              ?
```

Router1配置summary之前，Router2的路由表 ：

```
[Router2]display bgp routing-table
 Total Number of Routes: 21
 BGP Local router ID is 10.35.251.29
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

 *>  55.1.1.1/32        192.168.1.254   0                     0       100i
 *>  55.1.1.2/32        192.168.1.254   0                     0       100i
 *>  83.100.1.0/30      192.168.1.254                         0       100 900?
 *>  83.100.1.4/30      192.168.1.254                         0       100 900?
 *>  83.100.1.8/30      192.168.1.254                         0       100 900?
 *>  83.100.1.12/30     192.168.1.254                         0       100 900?
 *>  83.100.1.16/30     192.168.1.254                         0       100 900?
 *>  83.100.1.20/30     192.168.1.254                         0       100 900?
 *>  83.100.1.24/30     192.168.1.254                         0       100 900?
 *>  83.100.1.28/30     192.168.1.254                         0       100 900?
 *>  109.0.0.0/24       192.168.1.254   1                     0       100?
 *>  109.0.1.0/24       192.168.1.254   1                     0       100?
 *>  109.0.2.0/24       192.168.1.254   1                     0       100?
 *>  192.168.192.0      192.168.1.254   1                     0       100?
 *>  192.168.193.0      192.168.1.254   1                     0       100?
 *>  192.168.194.0      192.168.1.254   1                     0       100?
 *>  192.168.195.0      192.168.1.254   1                     0       100?
 *>  192.168.196.0      192.168.1.254   1                     0       100?
 *>  192.168.197.0      192.168.1.254   1                     0       100?
 *>  192.168.198.0      192.168.1.254   1                     0       100?
 *>  192.168.199.0      192.168.1.254   1                     0       100?
```

在Router1上配置自动聚合：

即进行如下配置：

```
[Router1-bgp]summary automatic
```

查看Router2的路由表：

```
[Router2]display bgp routing-table
 Total Number of Routes: 19
 BGP Local router ID is 10.35.251.29
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn

 *>  55.1.1.1/32        192.168.1.254   0                     0       100i
 *>  55.1.1.2/32        192.168.1.254   0                     0       100i
 *>  83.100.1.0/30      192.168.1.254                         0       100 900?
 *>  83.100.1.4/30      192.168.1.254                         0       100 900?
 *>  83.100.1.8/30      192.168.1.254                         0       100 900?
 *>  83.100.1.12/30     192.168.1.254                         0       100 900?
 *>  83.100.1.16/30     192.168.1.254                         0       100 900?
 *>  83.100.1.20/30     192.168.1.254                         0       100 900?
 *>  83.100.1.24/30     192.168.1.254                         0       100 900?
 *>  83.100.1.28/30     192.168.1.254                         0       100 900?
 *>  109.0.0.0          192.168.1.254                         0       100?
 *>  192.168.192.0      192.168.1.254   1                     0       100?
 *>  192.168.193.0      192.168.1.254   1                     0       100?
 *>  192.168.194.0      192.168.1.254   1                     0       100?
 *>  192.168.195.0      192.168.1.254   1                     0       100?
 *>  192.168.196.0      192.168.1.254   1                     0       100?
 *>  192.168.197.0      192.168.1.254   1                     0       100?
 *>  192.168.198.0      192.168.1.254   1                     0       100?
 *>  192.168.199.0      192.168.1.254   1                     0       100?
```

通过比较不难发现Router2的路由表中没有了109.0.0.0/24， 109.0.1.0/24，109.0.2.0/24的路由，被聚合成了109.0.0.0/8的路由，而其他的路由没有发生变化。

由于自动聚合只聚合路由引入的路由：

83.100.1.0/30－83.100.1.28/30是从其他邻居学习到的路由，因此没有被聚合；

55.1.1.1/32和55.1.1.2是通过network 实现的BGP路由，也不能被自动聚合。

虽然自动聚合可以聚合自身引入的IGP路由，但192.168.192.0/24－192.168.199.0/24已经是自然网段，因此没有被聚合。

我们再来看被自动聚合的路由，可以看出路由被自动聚合后，路由器只向邻居发送聚合后路由，不再发送详细路由。

```
[Router2]dis bgp routing-table 109.0.0.0
 BGP local router ID : 10.35.251.29
 Local AS number : 200
 Paths:   1 available, 1 best
 BGP routing table entry information of 109.0.0.0/8:
 From            : 192.168.1.254 (213.168.133.93)
 Original nexthop: 192.168.1.254
 AS-path         : 100
 Origin          : incomplete
 Attribute value : pref-val 0, pre 255
 State           : valid, external, best,
 Aggregator      : AS 100, Aggregator ID: 213.168.133.93
 Not advertised to any peers yet
```

查看109.0.0.0这条路由不难发现，该条路由具有Aggregator属性，该属性标明路由是在哪里聚合的。针对109.0.0.0 ，产生这条聚合路由的自治系统为100，产生这条聚合路由的路由器的RouterID是213.168.133.93,即Router1的router ID。

注：V3的设备的命令是summary。

由于自动聚合只能对自身引入的路由按照自然网段进行聚合，不能满足各种组网需求。在许多场合会应用手动聚合。手动聚合Aggregate命令，有较多的选项，比较灵活，下面将重点说明。

手动聚合

通过手动聚合命令Aggregate对路由进行聚合，只发布聚合路由

要宣告有Aggergate 命令确定的聚合路由，首先属于聚合路由的更具体的路由已经在BGP的路由表中，这些具体路由可以是从邻居学习来的路由，也可以是引入的IGP的路由；或者是通过network命令生成的BGP路由。

通过detail-suppressed选项，控制路由器，只发送聚合路由，不发送详细路由。

图2中，在Router1上，用Aggregate命令对路由进行聚合：

```
[Router1-bgp]display this
bgp 100
 aggregate 192.168.192.0 255.255.248.0 detail-suppressed
 aggregate 55.1.1.0 255.255.255.252 detail-suppressed
 aggregate 83.100.1.0 255.255.255.224 detail-suppressed
 summary automatic
 network 55.1.1.1 255.255.255.255
 network 55.1.1.2 255.255.255.255
 import-route rip 1
 undo synchronization
 peer 192.168.1.253 as-number 200
 peer 82.112.2.190 as-number 900
```

查看Router1的路由表：

```
[Router1]display bgp routing-table
Total Number of Routes: 25
BGP Local router ID is 213.168.133.93
Status codes: * - valid, > - best, d - damped,
              h - history,  i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED      LocPrf      PrefVal Path/Ogn

*>  55.1.1.0/30        127.0.0.1                    0                   i
s>  55.1.1.1/32        0.0.0.0         0            0                   i
s>  55.1.1.2/32        0.0.0.0         0            0                   i
*>  83.100.1.0/27      127.0.0.1                    0                   ?
s>  83.100.1.0/30      82.112.2.190                 0                   900?
s>  83.100.1.4/30      82.112.2.190                 0                   900?
s>  83.100.1.8/30      82.112.2.190                 0                   900?
s>  83.100.1.12/30     82.112.2.190                 0                   900?
s>  83.100.1.16/30     82.112.2.190                 0                   900?
s>  83.100.1.20/30     82.112.2.190                 0                   900?
s>  83.100.1.24/30     82.112.2.190                 0                   900?
s>  83.100.1.28/30     82.112.2.190                 0                   900?
*>  109.0.0.0          127.0.0.1                    0                   ?
s>  109.0.0.0/24       0.0.0.0         1            0                   ?
s>  109.0.1.0/24       0.0.0.0         1            0                   ?
s>  109.0.2.0/24       0.0.0.0         1            0                   ?
*>  192.168.192.0/21   127.0.0.1                    0                   ?
s>  192.168.192.0      0.0.0.0         1            0                   ?
s>  192.168.193.0      0.0.0.0         1            0                   ?
s>  192.168.194.0      0.0.0.0         1            0                   ?
s>  192.168.195.0      0.0.0.0         1            0                   ?
s>  192.168.196.0      0.0.0.0         1            0                   ?
s>  192.168.197.0      0.0.0.0         1            0                   ?
s>  192.168.198.0      0.0.0.0         1            0                   ?
s>  192.168.199.0      0.0.0.0         1            0                   ?
```

注意到这里生成了聚合路由，83.100.1.0/27,AS-Path属性中没有AS900。

查看Router2的bgp路由：

```
[Router2]display bgp routing-table
Total Number of Routes: 4
BGP Local router ID is 10.35.251.29
Status codes: * - valid, > - best, d - damped,
              h - history,  i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
*>  55.1.1.0/30        192.168.1.254                         0       100i
*>  83.100.1.0/27      192.168.1.254                         0       100?
*>  109.0.0.0          192.168.1.254                         0       100?
*>  192.168.192.0/21   192.168.1.254                         0       100?
```

Router1只给Router2传送了聚合路由，因此看到Router2上只有聚合路由。

我们选其中一个聚合路由看一下其属性：

```
[Router2]display bgp routing-table 83.100.1.0
 BGP local router ID : 10.35.251.29
 Local AS number : 200
 Paths:   1 available, 1 best

 BGP routing table entry information of 83.100.1.0/27:
 From            : 192.168.1.254 (213.168.133.93)
 Original nexthop: 192.168.1.254
 AS-path         : 100
 Origin          : incomplete
 Attribute value : pref-val 0, pre 255
 State           : valid, external, best,
 This route is an atomic-aggregated route
 Aggregator      : AS 100, Aggregator ID: 213.168.133.93
 Advertised to such 1 peers:
    23.1.1.1
```

这条路由的AS-path为100，丢失了原来的路径AS 900。

可以看到这条路由有一个atomic-aggregated属性，当运行BGP的路由器将更详细的路由聚合为较少细节的聚合路由，且已经出现了路径信息的丢失，atomic-aggregate属性将附加到聚合路由中。

不使用detail-suppressed选项，聚合路由和具体路由都被传送。

聚合路由和具体路由都需要被传送

对于图2的简单拓扑，不需要公布具体路由给邻居。但在图3中，公布两种路由是理想的。AS100到AS200有多条路径，AS200需要根据来自AS100的所有路由来设置路由策略，但是它必须只将聚合路由公布给AS300。AS100的具体路由都携带一个NO_EXPORT的团体属性，携带该属性的路由不能公布给EBGP邻居。因此，AS200知道这些路由不会发布给AS300。

在Router1上的路由聚合不配置detail-suppressed选项：那么聚合路由和具体路由都会被公布给AS200；并配置邻居Router2发布路由的策略，使聚合路由不发布community属性，具体路由发布community属性，那么AS300就可以收到聚合路由了。

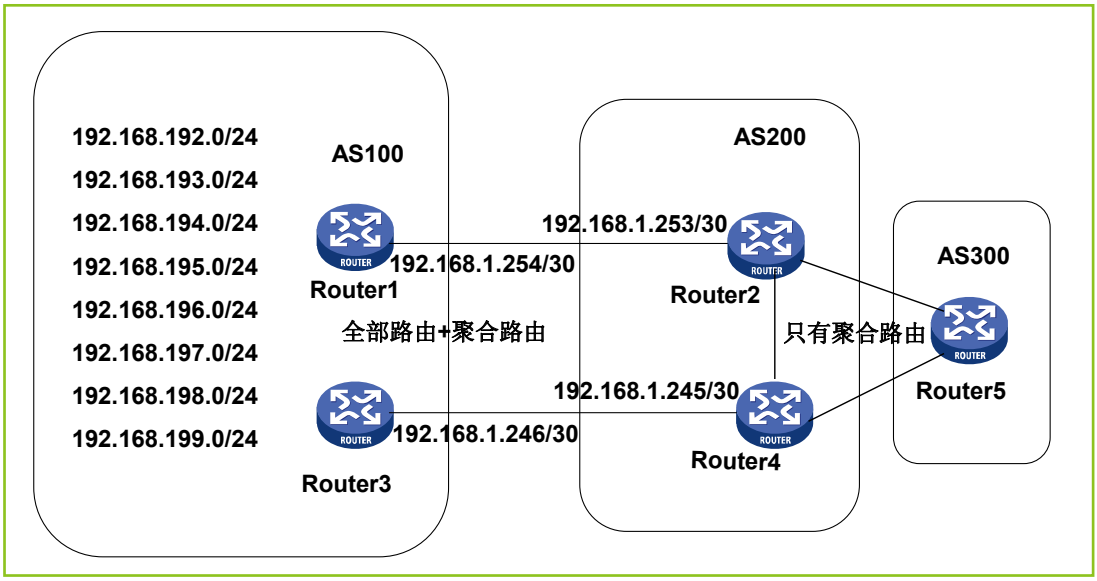


图3 AS100多宿主到AS200

宣告聚合路由及挑选出来的更具体的路由

利用Aggregate的suppress-policy参数和detail-suppressed结合；

在前一种方案中，将AS100的更具体的路由发送给了AS200，因此AS200能够执行路由策略。AS200这些路由来设置路由选项，从而使AS300能够向AS100发送业务量。

在图3中，AS100到AS200有两条链路，如果想从AS200到AS100的192.168.193.0/24业务量走Router3到Router4的链路，而Router1到Router2的链路作为备份链路。可以在Router1制定策略，使Router1上抑制192.168.193.0/24这条具体路由，发布聚合路由和其他更具体的路由。

在Router3上，制定策略，发布聚合路由和192.168.193.0/24这条聚合路由。

在生成聚合路由时如果使用了suppress-policy参数来抑制部分参与聚合的路由，即符合suppress-policy的路由被抑制；不匹配的部分不被抑制。那么在发布路由时除了发布聚合路由，还会发布与suppress-policy中不匹配的部分具体路由。

改变聚合的属性

Aggregate命令使用Attribute-policy选项，改变聚合路由的属性。使用此参数能轻易改变聚合路由的一些属性，比如团体属性、起源以及cost等等。

在图3中，AS100中的IGP为rip，将路由引入到BGP中，那么192.168.192.0到192.169.199.0BGP路由的Origin属性为incomplete .

假设AS200希望所有到AS100的业务量，将Router1到Router2的作为主链路，而把将Router3到Router4的链路作为备份链路。

那么就可以在Router1上进行路由聚合，使聚合路由的origin属性为IGP，而Router2上的聚合路由的属性不变，由于BGP会优选Origin为IGP的路由。当Router1到Router2的链路存在问题时，选择Router3到Router4的链路。

和聚合路由一起使用AS_SET

AS_PATH属性有两种类型：

- AS_SEQUENCE:这是AS号的一个有规则的列表
- AS_SET:到目的地的路径上AS号的一个无规则列表

AS_PATH的一个主要作用时防止路由环路。如果一个运行BGP的路由器从EBGP邻居收到一条路由含有它自己的AS号，知道出现了环路，将忽略此路由。

如图4所示，当执行路由聚合时，会丢失AS_PATH的一些细节，产生环路的潜在因素就增加了。 经过AS3113上的路由器聚合后，丢失了详细路由的路径信息。AS810和AS237，AS225的路径信息被丢弃。假设AS810到其他AS有可选连接（如图5）。来自AS3113的路由聚合公布给AS6571，然后AS6571又回到AS810。因为在聚合点号的AS号没有包括在AS_PATH中，AS810不会检测到潜在的环路。

假设AS810内的一个网络206.25.225.0/24出现了故障，在这个AS内的路由器会与来自AS6571的聚合路由相匹配，这样就出现了环路。

由上可知，AS_PATH防止环路的功能不要求AS号有顺序，重要的是接收路由器能够识别自己的AS号是否已经是AS_PATH的一部分了，此时就涉及到了AS_SET。

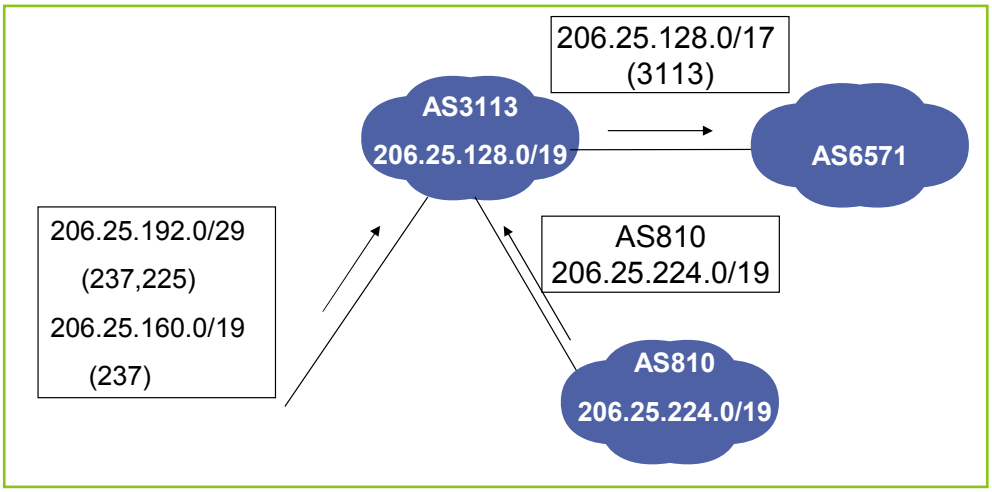


图4 聚合路由丢失一些具体路由路径信息

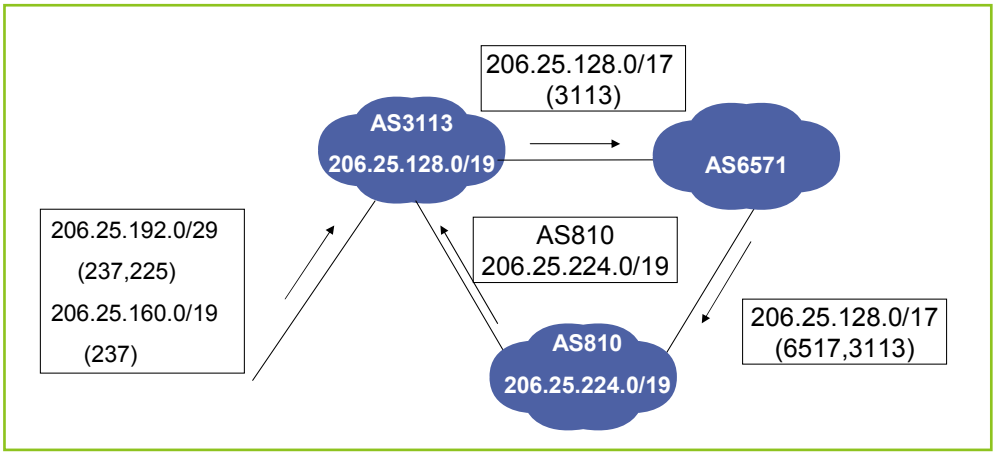


图5 聚合路由丢失一些具体路由路径信息形成环路

如图6，路由器生成聚合路由时，选择AS_SET选项，生成的聚合路由包括AS_PATH中所有的AS号并将它们作为一个AS_SET. 可以看到AS__SET是没有顺序的。从聚合路由器开始了一个AS_SEQUENCE。

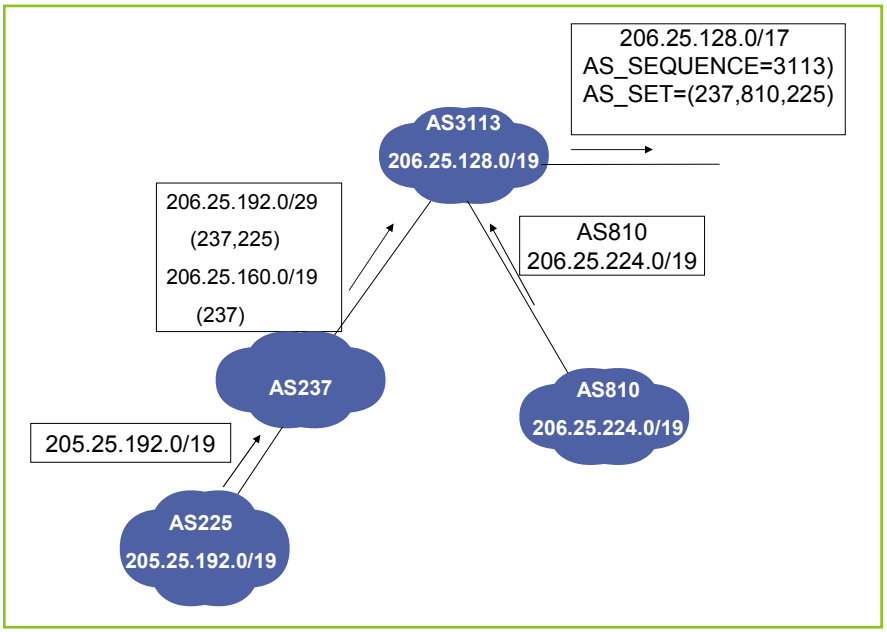


图6 AS_SET可以避免聚合路由形成环路

在进行路由聚合时，使用as-set参数后，BGP路由表中聚合路由的路径信息带有每条具体路由的路径信息，并随着被聚合路由的更新而变化。聚合路由重新进入as-set中列出的任何一个AS，BGP的环路检测机制检测到自己的AS号在聚合路由的as-set 属性列出的AS中，就会丢弃聚合路由，这样就避免了形成环路。利用AS_SET可以避免环路的同时，降低了网络的稳定性。如果AS225的链路出现故障，例如AS_SET发生了变化，那么会在聚合点以外的范围内公布此变化。

基于选中的更具体路由的聚合路由

Aggregate命令利用origin-policy选项来决定对哪些具体路由进行聚合，从而决定聚合路由携带什么样的属性。聚合路由只继承路由策略中指定的路由的属性，忽略了不匹配的路由的属性，从而达到了聚合路由控制的目的。如图6中，假设206.25.224.0/19具有NO_EXPORT属性，如果不排除这条路由进行聚合，聚合后的路由不会发送给其他的路由器。利用origin-policy选项进行聚合排除了这条路由，聚合后的路由将不继承NO_EXPORT属性。从而可以使聚合路由被传播。

总结

在现在的网络中，路由条目越来越多，路由聚合通过较少路由条目，减少了存储、传输和计算路由所需的网络资源的负担；通过隐藏一些具体路由，使聚合点之后的路由器，免受路由震荡带来的影响。路由聚合结合路由策略，可以实现链路的备份、负载分担等以满足日益丰富的组网的需求。同时，由于路由聚合隐藏了一些具体的路由，带来了形成路由环路的风险，需要设计者综合进行考虑使用。

BGP路由过滤

文/姜杏春

BGP路由是构成之Internet路由表的核心，目前规模已经达到十几万条。在实际应用中并不是所有的业务路由器都会需要全部的Internet路由。所以我们在很多时候需要对BGP路由进行过滤来控制路由的发送和接收。

BGP路由过滤的手段

我们知道路由过滤主要是以对路由所携带的信息作为匹配条件做过滤，BGP的属性众多，相较于其他路由所携带的路由信息就很多，所以对于BGP的路由过滤也要灵活的多。

ACL/IP 前缀列表

ACL：用户在定义ACL时可以指定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。

IP Prefix：IP Prefix的作用类似于ACL，但比它更为灵活，且更易于用户理解。使用IP Prefix过滤路由信息时，其匹配对象为前缀和掩码。

ACL、IP前缀列表主要是对BGP路由的前缀做过滤，可以实现对不同前缀地址做不同的过滤。

AS 路径过滤列表

AS路径过滤列表仅用于BGP。BGP的路由信息中，包含有自治系统路径域。as-path就是针对自治系统路径域指定匹配条件。

BGP可以直接使用AS路径过滤列表对路由做过滤，它可以以BGP路由的AS-PATH属性作为过滤条件，可以实现对来自不同AS的路由做过滤。

当想拒绝某一个AS始发的所有路由，用AS路径过滤列表显然要简单的多。

Route Policy

路由策略相较于前两种方法，提供了更丰富的手段。既可以使用ACL、IP前缀列表和AS路径列表对BGP路由做过滤，还可以使用其他匹配条件，比如：

团体属性列表（community-list）： BGP的路由信息包中，包含一个community属性域，用来标识一个团体。community-list就是针对团体属性域指定匹配条件。

扩展团体属性列表（extcommunity-list）：可用于VPN的Route-Target（路由目标）扩展extcommunity-list就是针对扩展团体属性指定匹配条件。

综上看，BGP路由常用来被过滤的条件主要是路由前缀、AS-PATH属性、Community属性，当然还有一些其他匹配条件（MED、next-hop、route-type、route-source）也可以被用来做过滤。

BGP路由过滤的实施点

BGP路由过滤的策略可以在本地对从邻居接收路由入方向、本地发布路由以及对邻居发送路由出方向处实施。

接收路由（Import Policy）

在收到BGP邻居的路由时，我们可以执行路由策略，过滤我们不需要的BGP路由。

路由策略的执行在BGP往路由表添加路由之前，所以路由一旦被过滤掉，这些路由不添加到在执行策略的设备的路由表中，在本地不负责转发。

路由策略可以对所有接收的路由作过滤，也可以只对特定的BGP邻居或邻居组作过滤。

本地发布路由

对于本地发布的路由，主要是指通过network、import方式本地发布的BGP路由，我们可以执行路由策略，过滤我们不需要发布的BGP路由。

路由策略的执行在在BGP往路由表添加路由之前，这些路由可以有选择的发布给所有BGP邻居。

发送路由（Export Policy）

在给BGP邻居发送路由的时候，我们也可以执行路由策略，过滤我们不想发布的BGP路由。

路由策略的执行在BGP往路由表添加路由之后，所以本地路由表中匹配deny策略的路由依然生效，在本地可以转发，只不过不向配置策略的邻居发送该BGP路由，让邻居无法从自己学习使用该路由。

路由策略可以对所有邻居做过滤，也可以只对特定的BGP邻居或邻居组作过滤。

实际案例

IP Prefix

IP Prefix是一种针对路由目的地址信息做过滤的工具，同样是对路由目的地址做过滤，既然有了ACL，为什么还需要IP Prefix呢？可以说，相对于ACL，IP Prefix用来做路由目的地址的过滤更专业。我们知道一条路由，不光有目的地址信息，还有掩码信息，ACL只能对目的地址信息做过滤，而IP Prefix可以做到对路由目的地址信息和掩码信息同时做过滤，这就是IP Prefix的优点。

举个例子，路由表里有2条这样的路由10.0.0.0/16和10.0.0.0/24，考虑路由表的容量，想将10.0.0.0/24这条路由过滤掉。

```
[H3C-bgp] display ip routing-table
Routing Tables: Public
         Destinations : 6        Routes : 6

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

10.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
10.0.0.0/24         BGP    255  0            10.1.1.2        GE0/1
10.1.1.0/24         Direct 0    0            10.1.1.1        GE0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

使用ACL做过滤，配置ACL 2000应用于BGP。

```
acl number 2000
 rule 0 deny source 10.0.0.0 0
 rule 1 permit
bgp 100
 filter-policy 2000 import
```

再次查看IP路由表，由于两条路由目的地址都为10.0.0.0，我们看到结果将两条路由都过滤掉。

```
[H3C-bgp]display ip routing-table
Routing Tables: Public
         Destinations : 4        Routes : 4

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

10.1.1.0/24         Direct 0    0            10.1.1.1        GE0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

这样ACL就无法满足我们的需求了，这时候我们可以使用我们的另一个工具IP Prefix，指定只deny 10.0.0.0/24，允许其他路由通过。同时提醒两点要注意：第一，无论是ACL还是IP Prefix过滤，缺省都是deny all的，所以我们在配置需要过滤的路由条目后，最后还要配置一条permit命令让其他路由通过。第二，设备如果不支持Route refresh能力，需要手动reset bgp邻居过滤策略才生效；设备如果支持Route refresh能力，邻居不支持Route refresh能力，需要和相应对等体配置peer x.x.x.x keep-all-routes。

```
ip ip-prefix 1 index 10 deny 10.0.0.0 24 greater-equal 24 less-equal 24
ip ip-prefix 1 index 20 permit 0.0.0.0 0 less-equal 32
bgp 100
 filter-policy ip-prefix 1 import
```

查看IP路由表，从路由表可以看出10.0.0.0/24路由已经从路由表中消失，10.0.0.0/16的路由依然在路由表中，就可以满足我们的要求，从这个例子，我们可以看出IP Prefix可以更精确的过滤路由，因为它可以匹配的路由信息更多，地址+掩码。

```
[H3C-bgp]display ip routing-table
Routing Tables: Public
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
10.1.1.0/24         Direct 0    0            10.1.1.1        GE0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

同时，IP Prefix也可以针对一段掩码范围做过滤。在原来的路由表中有多了很多11开头的路由，我们精简路由，要求只需要掩码是16的路由。

```
<H3C>display ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
10.1.1.0/24         Direct 0    0            10.1.1.1        GE0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
11.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
11.0.0.0/27         BGP    255  0            10.1.1.2        GE0/1
11.0.1.0/25         BGP    255  0            10.1.1.2        GE0/1
11.0.2.0/26         BGP    255  0            10.1.1.2        GE0/1
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

我们可以在原来的配置基础上再添加一条如下的命令即可。注意，IP Prefix匹配顺序是根据表项的index号匹配的，index号越小，越先匹配，本例中之前的两条表项的index分别是10和20，我们要使再配置的表项信息能在两者之间作匹配，只需配置的index在两者之间即可，例子中使用的index是11。

```
ip ip-prefix 1 index 11 deny 11.0.0.0 16 greater-equal 17 less-equal 32
```

查看IP路由表，路由表中除了11.0.0.0/16，其他的路由都没有了，满足了我们的要求，可以看出IP Prefix还是很好用的工具。

```
[H3C]dis ip routing-table
Routing Tables: Public
         Destinations : 6        Routes : 6

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
10.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
10.1.1.0/24         Direct 0    0            10.1.1.1        GE0/1
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
11.0.0.0/16         BGP    255  0            10.1.1.2        GE0/1
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

AS PATH

利用地址前缀去过滤BGP路由，在如此大规模的路由表时，一来有可能配置比较繁琐，二来且有新的路由加入不好维护，所以提出了BGP利用AS_PATH作过滤的办法。由于Internet核心AS的分布都是有记录的，所以利用AS的过滤更有针对性，例如用AS_PATH作过滤，过滤从某个AS_PATH始发的全部路由，只需一个AS_PATH列表即可。

下面让我们来看个小例子：

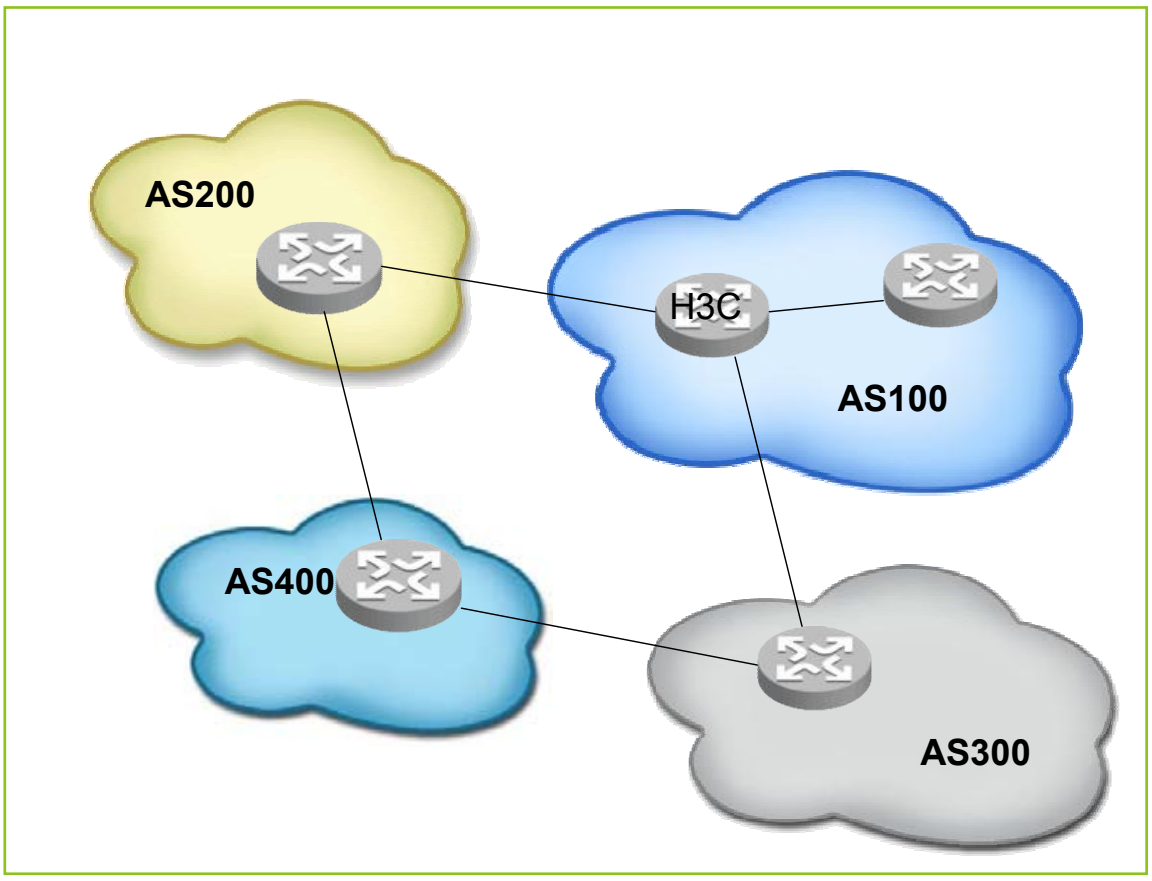


图1 AS PATH过滤应用

出于业务需要，H3C设备需要过滤掉来自AS400始发的路由。

可是查看H3C设备的BGP路由表，发现来自AS400的路由前缀和掩码几乎无规律可循，且从多个邻居可以学到，如果使用ACL或IP Prefix，需要配置表项很多，且如果有新的路由，还需要添加更多的表项，非常不好维护。

```
[H3C]dis bgp routing-table
 Total Number of Routes: 25
 BGP Local router ID is 10.1.1.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network            NextHop        MED     LocPrf       PrefVal Path/Ogn
 *>  1.0.0.0/12        10.1.1.2                 0              200 400?
 *>  1.16.0.0/12       10.1.1.2                 0              200 400?
 *>  2.0.0.0/13        10.1.1.2                 0              200 400?
 *>  2.8.0.0/13        10.1.1.2                 0              200 400?
 *>  3.0.0.0/14        10.1.1.2                 0              200 400?
 *>  4.0.0.0/15        10.1.1.2                 0              200 400?
 *>  5.0.0.0/16        10.1.1.2                 0              200 400?
 *>  6.0.0.0/17        10.1.1.2                 0              200 400?
 *>  7.0.0.0/18        10.1.1.2                 0              200 400?
 *>  8.0.0.0/19        10.1.1.2                 0              200 400?
 *>  9.0.0.0/20        10.1.1.2                 0              200 400?
 *>  10.0.0.0/21       10.1.1.2                 0              200 400?
 *>  11.0.0.0/22       10.1.1.2                 0              200 400?
 *>  12.0.0.0/23       10.1.1.2                 0             200 400?
 *>  14.0.0.0/24       10.1.1.2                 0             200 400?
 *>  16.0.0.0/20       10.1.1.3                 0             300 400?
 *>  17.0.0.0/21       10.1.1.3                 0             300 400?
 *>  18.0.0.0/22       10.1.1.3                 0              300 400?
 *>  19.0.0.0/23       10.1.1.3                 0             300 400?
 *>  20.0.0.0/24       10.1.1.3                 0             300 400?
 *>  21.0.0.0/22       10.1.1.3                 0             300?
 *>  21.0.4.0/22       10.1.1.3                 0             300?
 *>  22.0.0.0/23       10.1.1.3                 0             300?
 *>  22.0.2.0/23       10.1.1.3                 0             300?
 *>  23.0.0.0/24       10.1.1.3                 0             300?
```

这个时候我们可以使用针对BGP特有的工具AS PATH列表，让我们看看这个工具的强大之处。首先，配置过滤AS400始发的路由，允许其他路由通过的AS PATH列表；同时，配置一个BGP对等体组group 1，将所有要过滤的邻居加到这个对等体组；然后，对对等体组group 1做AS PATH列表过滤。

```
ip as-path 1 deny _400$
ip as-path 1 permit .*
bgp 100
 undo synchronization
 peer 10.1.1.2 as-number 200
 peer 10.1.1.3 as-number 300
 group 1 external
 peer 1 as-path-acl 1 import
 peer 1 keep-all-routes
 peer 10.1.1.2 group 1
 peer 10.1.1.3 group 1
```

配置完成之后，查看BGP路由表，发现AS400始发的路由不生效了，其它的路由依然生效。一个AS PATH列表就轻轻松松实现了我们的需求，并且以后再有AS400始发的路由，也会被过滤掉，不需要再添加任何命令去维护它。现在我们看到AS PATH列表的好处了，只要根据AS PATH过滤的需求，我们都可以使用AS PATH列表就把它过滤掉。

```
[H3C-bgp]dis bgp routing-table

Total Number of Routes: 25

BGP Local router ID is 10.1.1.1
Status codes: * - valid, > - best, d - damped,
              h - history,  i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete
     Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
     1.0.0.0/12         10.1.1.2                   0              200 400?
     1.16.0.0/12        10.1.1.2                   0              200 400?
     2.0.0.0/13         10.1.1.2                   0              200 400?
     2.8.0.0/13         10.1.1.2                   0              200 400?
     3.0.0.0/14         10.1.1.2                   0              200 400?
     4.0.0.0/15         10.1.1.2                   0              200 400?
     5.0.0.0/16         10.1.1.2                   0              200 400?
     6.0.0.0/17         10.1.1.2                   0              200 400?
     7.0.0.0/18         10.1.1.2                   0              200 400?
     8.0.0.0/19         10.1.1.2                   0              200 400?
     9.0.0.0/20         10.1.1.2                   0              200 400?
     10.0.0.0/21        10.1.1.2                   0              200 400?
     11.0.0.0/22        10.1.1.2                   0              200 400?
     12.0.0.0/23        10.1.1.2                   0              200 400?
     14.0.0.0/24        10.1.1.2                   0              200 400?
     16.0.0.0/20        10.1.1.3                   0              300 400?
     17.0.0.0/21        10.1.1.3                   0              300 400?
     18.0.0.0/22        10.1.1.3                   0              300 400?
     19.0.0.0/23        10.1.1.3                   0              300 400?
     20.0.0.0/24        10.1.1.3                   0              300 400?
*>   21.0.0.0/22        10.1.1.3                   0              300?
*>   21.0.4.0/22        10.1.1.3                   0              300?
*>   22.0.0.0/23        10.1.1.3                   0              300?
*>   22.0.2.0/23        10.1.1.3                   0              300?
*>   23.0.0.0/24        10.1.1.3                   0              300?
```

本文中，AS PATH列表过滤的例子是最简单的例子，因为AS PATH列表使用了正则表达式这把利剑，可以对AS PATH做各种简单的、复杂的匹配，非常灵活，具体的AS PATH列表使用方法和更深入的案例请见《常用BGP AS_PATH正则表达式应用》。

Community

我们知道，BGP的COMMUNITY属性是用来标识一组具有共同性质的路由。利用COMMUNITY属性我们可以把路由根据业务分成很多类，我们可以把境外路由部署同一组COMMUNITY属性，境内路由系统是部署另一组COMMUNITY属性，这个分组完全是我们人为决定和控制的，方便我们更轻松的识别和管理众多的路由。

实际应用中，我们在ISP管理时，部分境内路由可能不需要发布到境外，而境外路由需要发布到境外，这些路由前缀不同，可能来自不同AS，我们可以在ISP边缘给这些境内路由设置相同的COMMUNITY值为100:2，给境外路由设置另外一个COMMUNITY值100:1，这样的话我们就可以巧妙地运用COMMUNITY属性的特质去控制和过滤路由。

没有部署路由策略过滤之前，我们查看RTA的IP路由表，RTA可以学习到境内路由和境外路由，这不是我们所期望的。

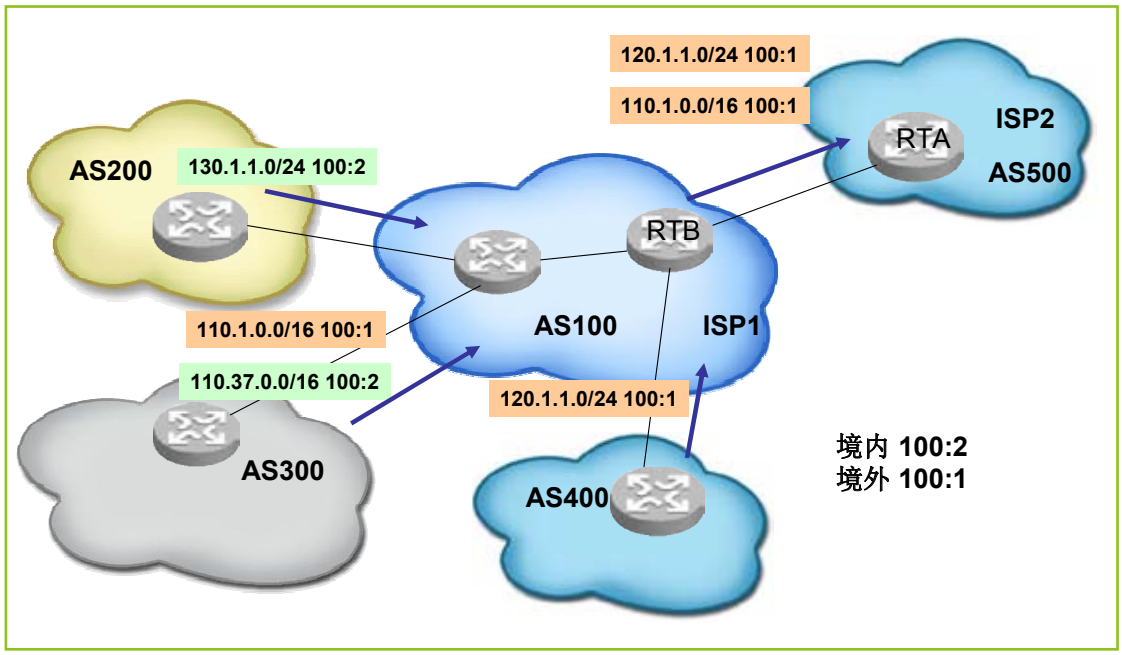


图2 COMMUNITY属性应用

```
<RTA>dis ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9

Destination/Mask    Proto   Pre  Cost      NextHop         Interface

8.8.8.8/32          Direct 0    0         127.0.0.1       InLoop0
110.1.0.0/16        BGP    255  0         200.1.1.1       GE2/1/1
110.37.0.0/16       BGP    255  0         200.1.1.1       GE2/1/1
120.1.0.0/16        BGP    255  0         200.1.1.1       GE2/1/1
127.0.0.0/8         Direct 0    0         127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0         127.0.0.1       InLoop0
130.1.1.0/24        BGP    255  0         200.1.1.1       GE2/1/1
200.1.1.0/24        Direct 0    0         200.1.1.2       GE2/1/1
200.1.1.2/32        Direct 0    0         127.0.0.1       InLoop0
```

现在我们利用境外路由都携带COMMUNITY属性100:1的特质，在RTB上配置过滤。对ISP2的邻居实施策略，只允许向ISP2邻居发布COMMUNITY属性为100:1的路由。

```
ip community-list 1 permit 100:1
ip community-list 1 deny
route-policy 1 permit node 0
 if-match community 1
#
bgp 100
 undo synchronization
 peer 40.1.1.2 as-number 400
 peer 30.1.1.2 as-number 100
 peer 200.1.1.2 as-number 500
 peer 40.1.1.2 keep-all-routes
 peer 30.1.1.2 keep-all-routes
 peer 200.1.1.2 route-policy 1 export
#
```

我们再看RTA的路由表，只剩下两条110.1.0.0/16和120.1.0.0/16两条境外路由，满足了我们的需求，再有新的境外路由，也会因为COMMUNITY为100:1，被匹配通过，所以我们无需再添加其他配置去维护。

```
[RTA]dis ip routing-table
Routing Tables: Public
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      NextHop         Interface

8.8.8.8/32          Direct 0    0           127.0.0.1       InLoop0
110.1.0.0/16        BGP    255  0           200.1.1.1       GE2/1/1
120.1.0.0/16        BGP    255  0           200.1.1.1       GE2/1/1
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
200.1.1.0/24        Direct 0    0           200.1.1.2       GE2/1/1
200.1.1.2/32        Direct 0    0           127.0.0.1       InLoop0
```

总结

BGP与其他路由协议对比，拥有很多自身独有的路由属性，我们可以灵活的根据BGP的多种属性去做路由过滤，我们可以通过不同的手段实现相同的目的，这完全取决于用户的需求和管理者的决策。由此看出，BGP这个协议，更类似一个管理协议，赋予网络工作者足够的权利去控制路由，能够将网络工作者的智慧充分体现，这正是该协议的独到之处。

RR、联盟及同步

文/高国义

简述

BGP越来越多的在规模比较大的企业或运营商得到部署应用， BGP丰富的路由属性能够轻松的进行路由选路。但是，受到BGP横向隔离规则规定的限制，BGP网络设备不会把它从一个IBGP邻居学习过来的路由传递给其他的IBGP邻居，导致在这些网络中需要在运行BGP的网络设备进行全连接的IBGP对等体的配置。那么在这些较大网络中需要配置的IBGP对等体为N*N个BGP邻居关系，这样就加大了实施的难度和后期网络分析的难度。由此BGP应运而生的两个特性反射器和联盟。

路由反射器（Route Reflector）

路由反射器（RR）的作用主要是为了简化IBGP邻居配置，使用反射器后允许反射器将来自IBGP邻居的路由信息发给另一个或一组IBGP邻居。BGP协议允许被配置为路由反射器的路由器向其他IBGP对等体传输由IBGP所学到的路由来修改BGP的横向隔离规则，也就避免了使用复杂的IBGP全连接的组网配置。

BGP 反射特性角色

✧ 路由反射器：是被配置为允许它把通过IBGP所学到的路由通告（或反射）到其他IBGP对等体的路由器；

✧ 客户：是和路由反射器有IBGP对等关系并配置成反射邻居关系的路由器；

✧ 非客户：不是路由反射器的客户的其他IBGP的对等体；

✧ originator（始发者）ID：是被路由反射器创建，这个属性带有本AS内部路由始发者的路由ID；

✧ 集群：路由反射器及其客户集合；

BGP 反射功能

✧ 路由反射器会依次在客户机之间反射信息。路由反射器和它的所有客户机构成一个群。一个群内允许有多个路由反射器，一个路由反射器可以把别的路由反射器配置成它的客户机或非客户机。

✧ 路由反射器在它的客户机和非客户机之间传送路由更新的规则：

—如果路由更新是从非客户机收到的，仅反射给客户机

—如果路由更新是从客户机收到的，反射给所有非客户机以及客户机，除了这个路由更新的始发者

—如果路由更新是从EBGP相邻体收到的，反射给所有的客户机和非客户机

路由反射器应用场合

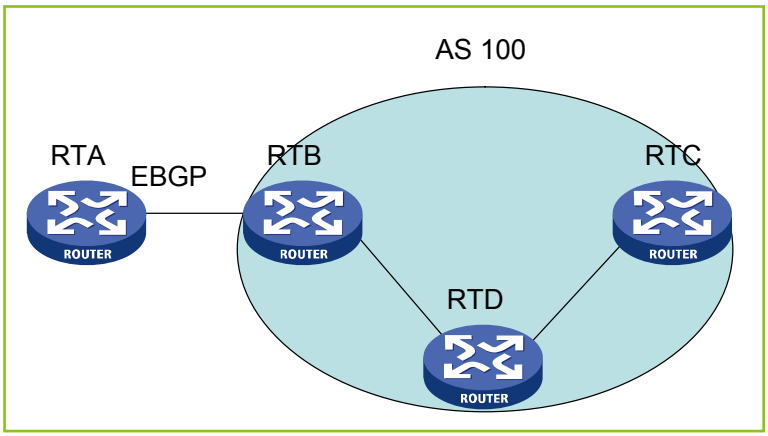


图1 反射器应用场合示意图

上图中如果没有配置反射器，由于有横向隔离原则，RTD收到IBGP邻居RTC的更新后不向别的IBGP邻居发送，结果导致RTB和RTA上就无法的到RTC的路由。那么如何解决这种问题呢？

方案一：

使AS域内的路由器进行物理上的全连接，在RTB和RTC之间增加一条物理链路，在RTC和RTB之间建一个IBGP邻居关系，使RTC能够直接把自己的路由传递给RTB路由器，由RTB路由器直接通过EBGP邻居关系，把路由传递给RTA。

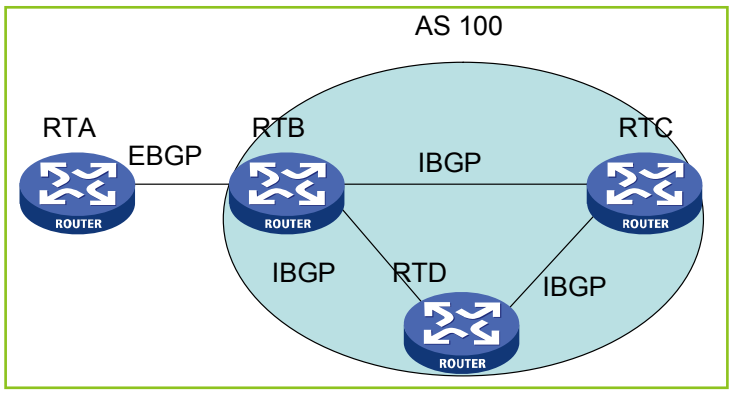


图2 增加物理链路的方案

这种全连接的网络部署方案，一方面组网成本高，另一方面BGP连接的数目增多相应的增加了系统的资源消耗，全连接的网络部署可扩展性差。

有没有简单一点的方法呢？

方案二：

使AS域内的路由器进行逻辑全连接，使RTB和RTC的IGP路由互相可达，在RTB和RTC之间建立一个IBGP邻居关系，使RTC能够直接把自己的路由传递给RTB路由器，由RTB路由器直接通过EBGP邻居关系，把路由传递给RTA。

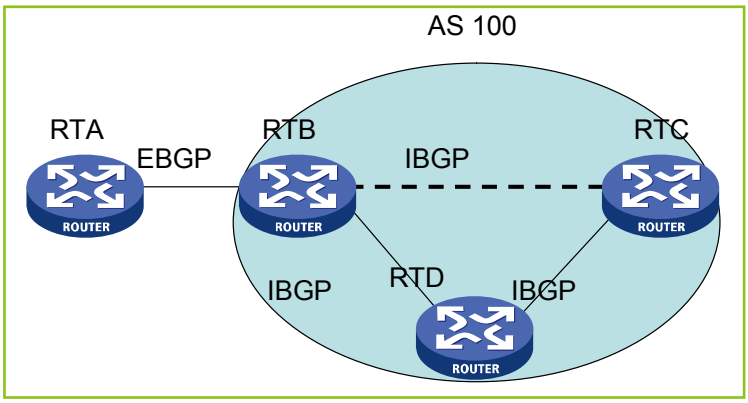


图3 逻辑全连接的方案

这种方案也需要增加了BGP邻居关系的配置。

还有没有更加简单的方法呢？

终极方案：

不用增加物理链路、不用新增IBGP邻居关系——配置BGP路由反射器。

把RTD配置路由反射器，同时RTB和RTC配置反射器RTD的客户，这样根据反射器的路由反射原理，RTC就可以轻松的把自己的路由反射给RTB，并由RTB通过EBGP邻居关系。

修改后的应用组网图：

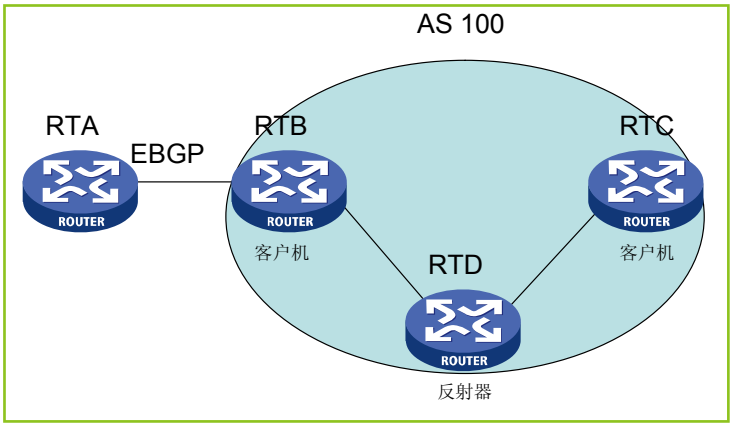


图4 使用反射器的方案

路由器反射过程

1.公网路由BGP路由传递过程

从RTC经过路由反射器传递一条路由给RTA路由器，RTA的router id为1.1.1.1、RTB的router id为2.2.2.2、RTD的router id为3.3.3.3、RTC的router id为4.4.4.4.

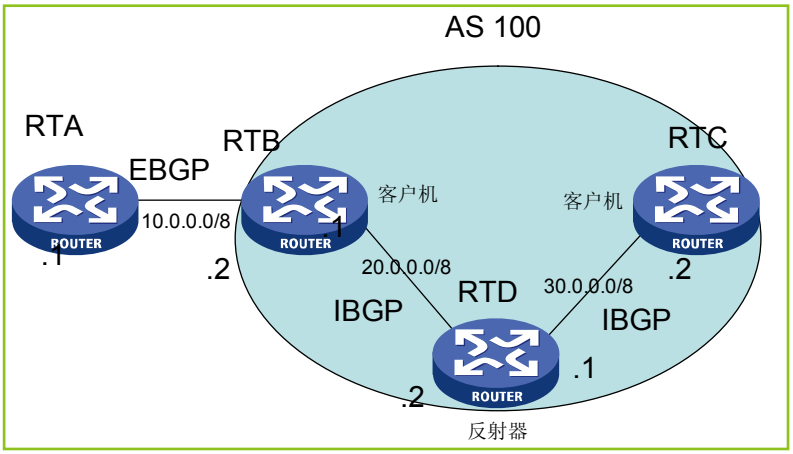


图5 公网路由传递过程示意图

RTC发布一条路由4.4.4.4/32给IBGP反射器。

反射器RTD学习到该条路由，并把这条路由4.4.4.4/32反射客户机RTB。

```
<RTD>dis bgp routing-table 4.4.4.4

BGP local router ID : 3.3.3.3
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 4.4.4.4/32
RR-client route
 From            : 30.0.0.2 (4.4.4.4)
 Relay Nexthop    : 0.0.0.0
Original nexthop  : 30.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Advertised to such 1 peers:
    20.0.0.1
```

客户机RTB学习到通过反射器反射的4.4.4.4/32的路由，该路由的Originator为4.4.4.4（即RTC路由器）、Cluster list为3.3.3.3（即该路由被反射器RTD路由器反射过来）、Original nexthop为30.0.0.2（该路由始发下一跳）、Relay Nexthop为20.0.0.2（该路由中继下一跳）。

```
<RTB>dis bgp routing-table 4.4.4.4

BGP local router ID : 2.2.2.2
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 4.4.4.4/32
RR-client route
 From            : 20.0.0.2 (3.3.3.3)
 Relay Nexthop    : 20.0.0.2
Original nexthop  : 30.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originator      : 4.4.4.4
 Cluster list     : 3.3.3.3
 Advertised to such 1 peers:
    10.0.0.1
```

RTA路由器通过EBGP邻居关系学习到4.4.4.4/32这条路由。

```
<RTA>dis bgp routing-table 4.4.4.4

BGP local router ID : 1.1.1.1
Local AS number : 200
Path: 1 available, 1 best

BGP routing table entry information of 4.4.4.4/32
From            : 10.0.0.2 (2.2.2.2)
Original nexthop  : 10.0.0.2
 AS-path         : 100
 Origin          : incomplete
 Attribute value   :pref-val 0, pre 255
 State           : valid, external, best,
Not advertised to any peers yet
```

这样路由反射器成功屏蔽BGP的横向隔离规则，把路由传递给RTA路由器。

2.公网路由BGP反射器嵌套路由传递过程

从RTE传递一条路由5.5.5.5/32给RTA路由器，RTA的router id为1.1.1.1、RTB的router id为2.2.2.2、RTC的router id为3.3.3.3、RTD的router id为4.4.4.4、RTE的router id为5.5.5.5；其中RTB和RTD为一级反射器、RTC为二级反射器。

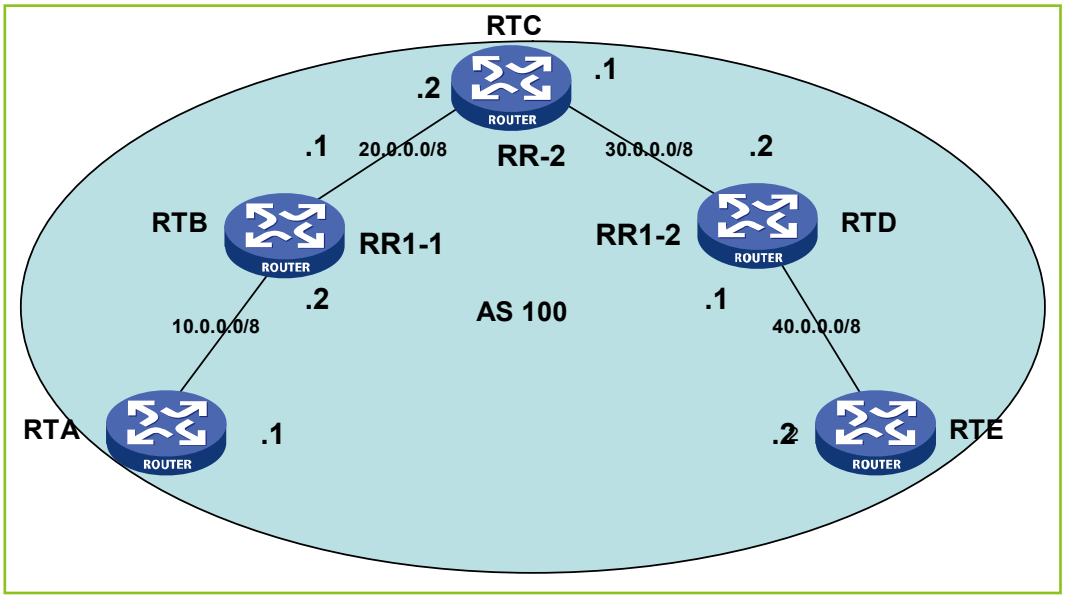


图6 嵌套路由传递过程示意图

在RTE上在BGP中发布一条路由5.5.5.5/32。

在反射器RTD查看5.5.5.5/32的路由，发现路由已经学习到该条路由，并把该条路由5.5.5.5/32反射给反射器RTC。

```
<RTD>dis bgp routing-table 5.5.5.5 32

BGP local router ID : 4.4.4.4
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 5.5.5.5/32
RR-client route
 From            : 40.0.0.2 (5.5.5.5)
 Relay Nexthop    : 0.0.0.0
 Original nexthop  : 40.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Advertised to such 1 peers:
    30.0.0.1
```

在RTC上查看5.5.5.5/32的路由，发现该路由已经被接受到。该路由的Originator为5.5.5.5、Cluster list为4.4.4.4（表明已经经过反射器4.4.4.4的反射）。

```
<RTC>dis bgp routing-table 5.5.5.5 32

BGP local router ID : 3.3.3.3
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 5.5.5.5/32
RR-client route
 From            : 30.0.0.2 (4.4.4.4)
 Relay Nexthop    : 30.0.0.2
 Original nexthop  : 40.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originatror      : 5.5.5.5
 Cluster list       : 4.4.4.4
 Advertised to such 1 peers:
    20.0.0.1
```

在RTB上查看5.5.5.5/32的路由，发现该路由已经被接受到。该路由的Originator为5.5.5.5、Cluster list为3.3.3.3,4.4.4.4（表明已经经过反射器4.4.4.4和反射器3.3.3.3的反射）。

```
<RTB>dis bgp routing-table 5.5.5.5 32

BGP local router ID : 2.2.2.2
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 5.5.5.5/32
From            : 20.0.0.2 (3.3.3.3)
 Relay Nexthop    :20.0.0.2
 Original nexthop  : 40.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originatror      : 5.5.5.5
 Cluster list       : 3.3.3.3, 4.4.4.4
 Advertised to such 1 peers:
    10.0.0.1
```

在RTA上查看5.5.5.5/32的路由，发现该路由已经被接受到。该路由的Originator为5.5.5.5、Cluster list为2.2.2.2,3.3.3.3,4.4.4.4（表明已经经过反射器4.4.4.4、反射器3.3.3.3和反射器2.2.2.2的反射）。

```
<RTA>dis bgp routing-table 5.5.5.5 32

BGP local router ID : 1.1.1.1
Local AS number : 100
Path: 1 available, 1 best

BGP routing table entry information of 5.5.5.5/32
From            : 10.0.0.2 (2.2.2.2)
 Relay Nexthop    :10.0.0.2
 Original nexthop  : 40.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originatror      : 5.5.5.5
 Cluster list       : 2.2.2.2, 3.3.3.3, 4.4.4.4
 Not advertised to any peers yet
```

在多反射器的存在或反射器嵌套的网络里，路由在被反射时，会添加反射器的ID，这样该路由经过的反射器就被有效的记录下来。这时当反射器收到一条路由的Cluster list中，包含自己的反射器ID时，反射器就不会反射该路由，这样就可以有效的防止路由环路。

3。私网路由BGP路由传递

RTB、RTD和RTC组网MPLS VPN网络，RTB和RTC为PE设备，使RTD配置成VPNv4

路由反射器。从RTC经过路由反射器传递一条VPN路由44.44.44.44/32给RTB路由器， RTB的router id为2.2.2.2、RTD的router id为3.3.3.3、RTC的router id为4.4.4.4。

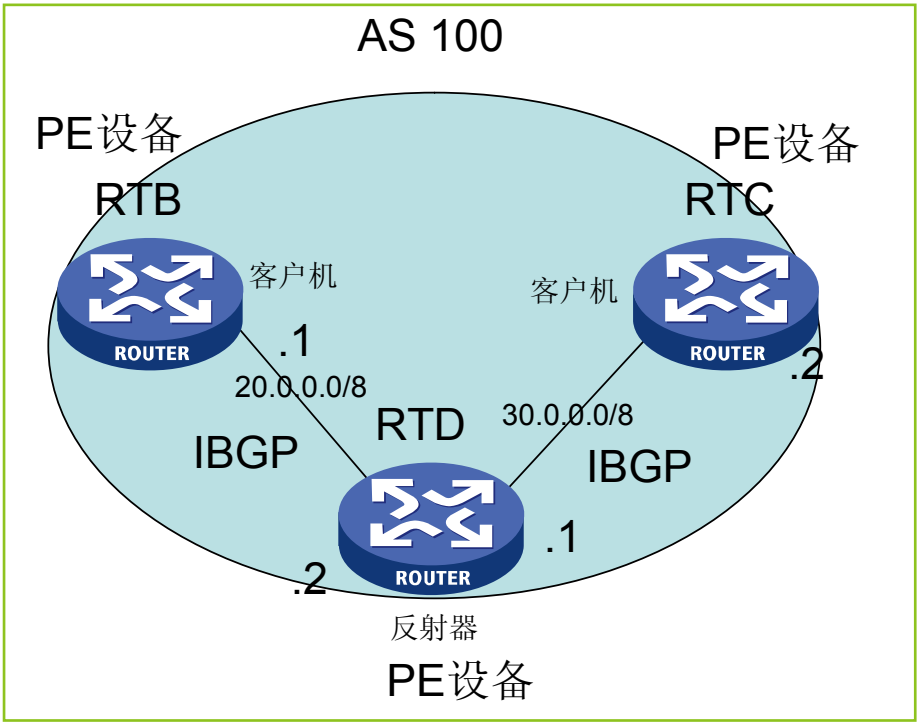


图7 私网传递示意图

RTC发布一条VPN路由44.44.44.44/32给MP-IBGP反射器。

反射器RTD学习到该条路由，并把这条路由44.44.44.44/32反射客户机RTB。

```
[RTD]dis bgp vpn vpn routing-table 44.44.44.44

 BGP local router ID : 3.3.3.3
 Local AS number : 100
 Paths:   1 available, 1 best

 BGP routing table entry information of 44.44.44.44/32:
 RR-client route
 From            : 4.4.4.4 (4.4.4.4)
 Relay Nexthop    : 0.0.0.0
 Original nexthop   : 4.4.4.4
 Ext-Community   :<RT: 100:1>
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Not advertised to any peers yet
```

客户机RTB学习到通过反射器反射的44.44.44.44/32的路由，该路由的Originator为4.4.4.4（即RTC路由器）、Cluster list为3.3.3.3（即该路由被反射器RTD路由器反射过来）、Original nexthop为4.4.4.4（该路由始发下一跳）、Relay Nexthop为0.0.0.0（该路由中继下一跳）（VPNv4路由为隧道迭代方式，故该项为全零）。

```
[RTD]dis bgp vpn vpn routing-table 44.44.44.44

 BGP local router ID : 2.2.2.2
 Local AS number : 100
 Paths:   1 available, 1 best

 BGP routing table entry information of 44.44.44.44/32:
From            : 3.3.3.3 (3.3.3.3)
 Relay Nexthop    : 0.0.0.0
 Original nexthop   : 4.4.4.4
 Ext-Community   :<RT: 100:1>
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originatror      : 4.4.4.4
 Cluster list       : 3.3.3.3
 Not advertised to any peers yet
```

这样路由反射器成功屏蔽BGP的横向隔离规则，把路由传递给RTA路由器。

路由器反射器的过滤

BGP另一个比较强的功能就是过滤功能了，那么这个功能是否因为配置BGP反射器后过滤功能不再强大了呢？可以肯定的告诉大家，反射器上的路由过滤功能依然强大。特别值得一提的就是反射器对VPNv4路由的过滤。公网BGP反射器上的路由过滤，一般情况下仅仅针对路由使用路由过滤策略即可完成。相对比较复杂的是VPNv4的路由过滤，这种过滤的在反射器上

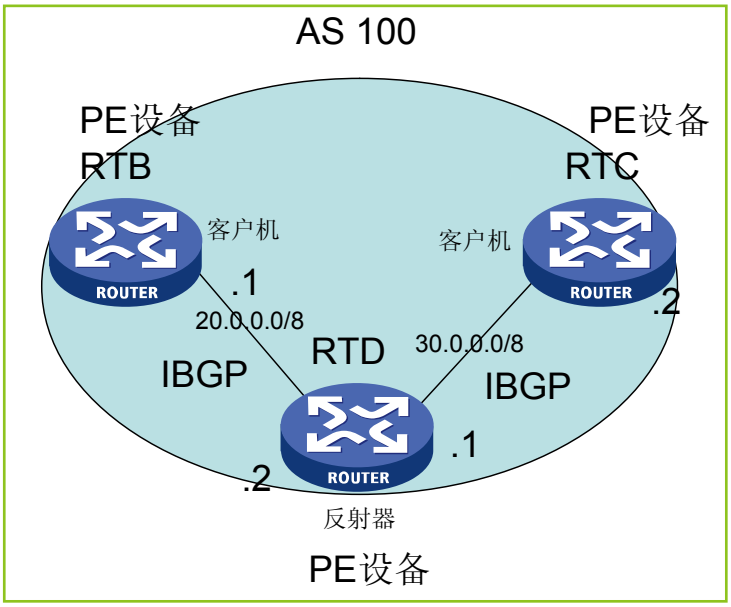


图8 反射器的过滤

的过滤需要使用的参数一般会包括Ext-Community即RT。VPNv4的路由过滤功能一般均把RT和具体路由进行结合的过滤方式。下面以一个MPLS VPN的组网举例进行讲解。

RTB、RTD和RTC组网MPLS VPN网络，RTB和RTC为PE设备，使RTD配置成VPNv4

路由反射器。RTC和RTB上均配置两个VPN，并设置RT分别为100：1和200：1。从RTC经过路由反射器同一个VPN内传递两条VPN路由44.44.44.44/32、55.55.55.55/32和66.66.66.66/32、77.77.77.77/32给RTB路由器， RTB的router id为2.2.2.2、RTD的router id为3.3.3.3、RTC的router id为4.4.4.4。

1. 过滤VPN的全部路由

针对VPN路由的过滤，很多情况均需要在反射器上过滤掉VPN的配置的RT属性直接即可。本例在反射器上配置过滤RT=100：1的路由即可。

```
#
bgp 100
 undo synchronization
 peer 4.4.4.4 as-number 100
 peer 2.2.2.2 as-number 100
 peer 4.4.4.4 connect-interface LoopBack1
 peer 2.2.2.2 connect-interface LoopBack1
 #
 ipv4-family vpnv4
  peer 2.2.2.2 enable
  peer 2.2.2.2 route-policy deny-vpn1 export
  peer 2.2.2.2 reflect-client
  peer 4.4.4.4 enable
  peer 4.4.4.4 reflect-client
 #
ospf 1
 area 0.0.0.0
  network 0.0.0.0 255.255.255.255
 #
route-policy deny-vpn1 deny node 10
 if-match extcommunity 1
route-policy deny-vpn1 permit node 20
#
 ip extcommunity-list 1 permit rt 100:1
#
```

在RTB上仅仅能够收到VPN2的66.66.66.66/32、77.77.77.77/32，VPN1的路由被过滤掉了。

```
[RTB] dis bgp vpnv4 all routing-table

 BGP Local router ID is 2.2.2.2
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete

Total Number of Routes from all PE: 2
Route Distinguisher:200:1
     Network                NextHop        In/Out Label  MED        LocPrf

 *>i  66.66.66.66/32         4.4.4.4          Null/1027     0            100
 *>i  77.77.77.77/32         4.4.4.4          Null/1027     0            100
Total Routes of vpn-instance vpn2: 2
     Network                NextHop        In/Out Label  MED        LocPrf

 *>i  66.66.66.66/32         4.4.4.4          Null/1027     0            100
 *>i  77.77.77.77/32         4.4.4.4          Null/1027     0            100
```

2. 过滤VPN的某些路由

如果需要过滤某VPN的某些路由，只需要在反射器上配置RT和明细路由结合的过滤规则即可。本例以过滤VPN1的44.44.44.44/32和VPN2的66.66.66.66/32的过滤为例。

```
#
route-policy deny-vpn-mingxi deny node 10
 if-match acl 2000
 if-match extcommunity 1
route-policy deny-vpn-mingxi deny node 20
 if-match acl 2001
 if-match extcommunity 2
route-policy deny-vpn-mingxi permit node 30
#
 ip extcommunity-list 1 permit rt 100:1
 ip extcommunity-list 2 permit rt 200:1
#


#
acl number 2000
 rule 0 permit source 44.44.44.44 0
acl number 2001
 rule 0 permit source 66.66.66.66 0
#
```

在RTB上仅仅能够收到VPN1的55.55.55.55/32和VPN2的77.77.77.77/32，其他路由被过滤掉了。

```
[RTB] dis bgp vpnv4 all routing-table

 BGP Local router ID is 2.2.2.2
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete

Total Number of Routes from all PE: 2
Route Distinguisher:100:1
     Network                 NextHop        In/Out Label  MED        LocPrf

 *>i  55.55.55.55/32          4.4.4.4         Null/1026     0            100

Route Distinguisher:200:1
     Network                 NextHop        In/Out Label  MED        LocPrf

 *>i  77.77.77.77/32          4.4.4.4         Null/1027     0            100
Total Routes of vpn-instance vpn1: 1
     Network                 NextHop        In/Out Label  MED        LocPrf

 *>i  55.55.55.55/32          4.4.4.4         Null/1026     0            100

Total Routes of vpn-instance vpn2: 2
     Network                 NextHop        In/Out Label  MED        LocPrf

 *>i  77.77.77.77/32          4.4.4.4         Null/1027     0            100
```

说明：RR在反射VPNV4路由时，如果没有配置对应的vpn-instance，则需要配置undo vpn-target policy。

反射器 AS 内部路由的环路避免

Originator ID属性：路由反射器的客户收到反射器发来的路由后自动携带Originator ID属性，在属性域指示出发布这条路由的始发者路由器ID，这样接收者就可以知道在反射器群内这条路由从哪个路由器始发，如果是自己始发则不接收，可以避免在路由反射器群内的环路。

Cluster ID List属性：路由反射器的客户收到反射器发来的路由后自动携带Cluster ID List属性，在属性域内指示出所经过的反射器群列表（反射器群ID缺省是反射器的路由器ID，可以配置），这样接收者就可以知道该路由在AS内部传播时经过了哪些反射器群，从而避免环路。具体实现方式请参考路由反射过程的范例2。

联盟(Confederation)

联盟也是为了解决大规模网络中IBGP全网连接的问题，是处理AS内部的IBGP网络连接激增的另一种方法。联盟是基于一个AS可以被分为多个子AS，子AS内使用IBGP全闭合网，子AS之间以及联盟本身与外部AS之间使用的EBGP连接的一种应用。虽然子AS之间的路由经EBGP交换，所有的IBGP规则仍然适用，因此对于AS外的路由器来看一个联盟就象一个单一的AS。联

盟和反射器应用时的区别是当BGP应用于更大的AS，即AS内有很多BGP Speaker 的情况，这时反射器配置也很复杂的话就可以考虑使用联盟，联盟的设计思想也是主要基于这一点。

联盟特性角色

✧ 联盟ID：在不属于联盟的BGP发言者看来，属于同一个联盟的多个子自治系统是一个整体，外界不需要了解内部的子自治系统情况，联盟ID就是标识联盟这一整体的自治系统号。在不属于联盟的BGP发言者看来，属于同一个联盟的多个子自治系统是一个整体，外界不需要了解内部的子自治系统情况，联盟ID就是标识联盟这一整体的自治系统号。

✧ 子系统：是联盟的组成元素，该子系统是一个内部IBGP全闭合的系统；该子系统对联盟外不可见。

✧ 子系统AS号：子系统的AS号，该AS号仅在联盟内可见；联盟内子自治系统可以使用私有的AS号，范围为：64512－65535，一个联盟最多可配置32个子自治系统。

联盟功能

1.联盟新增的两个属性

在RFC3065中新增加了两个为联盟定制的属性，即：

✧ AS_CONFED_SEQUENCE：有序的子系统号的序列集合。

✧ AS_CONFED_SET：无序的子系统号的序列集合。

二则主要区别是后者主要用于存在路由聚合等情况时导致路由属性丢失时使用；增加这两种属性是为了防止联盟内部的环路。

2.属性传递及处理过程

对于同一个联盟内部路由属性NEXT_HOP、MED和LOCAL_PREFERENCE的传递，联盟并没有特殊处理，故NEXT_HOP、MED和LOCAL_PREFERENCE仍然在联盟内传递。

PATH参数在联盟中进行传递，对于AS_CONFED_SEQUENCE和AS_CONFED_SET，

联盟内处理方式大致AS_SEQUENCE和AS_SET相同，同时：

1）当路由在联盟内子自治系统内传递时，不应修改AS_PATH属性。

2）当路由在联盟内子自治系统间传递时：

a）若第一个AS_PATH是AS_CONFED_SEQUENCE，BGP将自己的子自治系统AS号加在最左端

b）否则，创建一个AS_CONFED_SEQUENCE，包含自己的子自治系统AS号

3）当向联盟外EBGP传递路由时：

a）若第一个AS_PATH是AS_CONFED_SEQUENCE，将后续的AS_CONFED_SEQUENCE和AS_CONFED_SET删除，至b）

b）若第一个AS_PATH是AS_SEQUENCE，则将联盟AS加在最左端

c）若第一个AS_PATH是AS_SET，增加一个AS_SEQUENCE，将联盟AS加在最左端

4）对于本地初始路由的传播：

a）向本自治系统内IBGP发送，空的AS_PATH属性

b）向联盟内，本自治系统外EBGP发送，带有AS_CONFED_SEQUENCE属性

c）向联盟外EBGP发送，带有AS_SEQ属性

联盟应用场合

使用联盟之前：

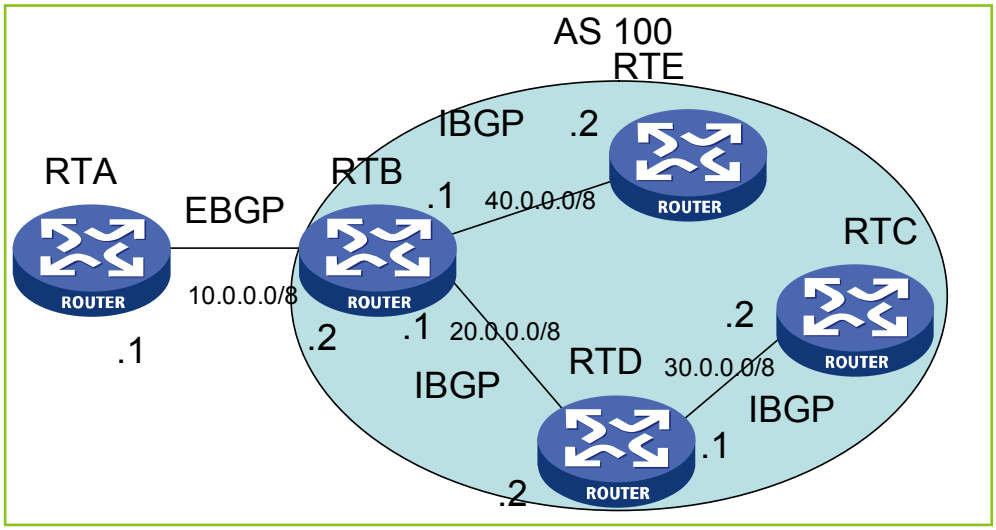


图9 使用联盟之前的情况

在上图中AS100域内台路由器都运行BGP，如果配置这4路由器的IBGP逻辑全互连的话会很麻烦，需要配置4×（4－1）/2=6个IBGP邻居关系。

使用联盟之后：

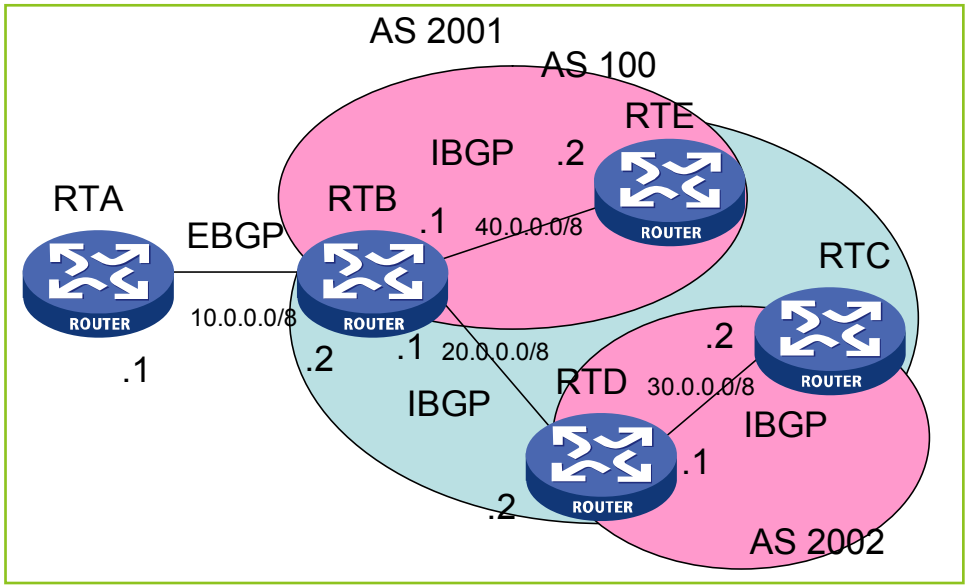


图10 使用联盟之后的情况

可以看到在使用联盟之前AS100内部配置IBGP邻居的工作将很麻烦，即使配置反射器也较复杂，在使用联盟后配置IBGP邻居的工作就相对简单了。配置联盟后AS100的EBGP邻居的配置没有变化，联盟对外仍然表现为一个AS，上图中的联盟ID为100，对外就表现为AS100，联盟外部的BGP Speaker 不了解联盟内子自治系统的情况。联盟内部有两个子自治系统分别为AS2001和AS2002，子自治系统之间为EBGP邻居，子自治系统内部和普通的自治系统相同。

联盟的路由传递过程

环境描述：应用图11组网网所示，RTA在AS 100内，RTB、RTC、RTD和RTE均在AS 200内，RTB和RTE在子系统2001内，RTC和RTD在子系统2002内。RTA的router id为1.1.1.1、RTB的router id为2.2.2.2、RTD的router id为3.3.3.3、RTC的router id为4.4.4.4和RTE的router id5.5.5.5。

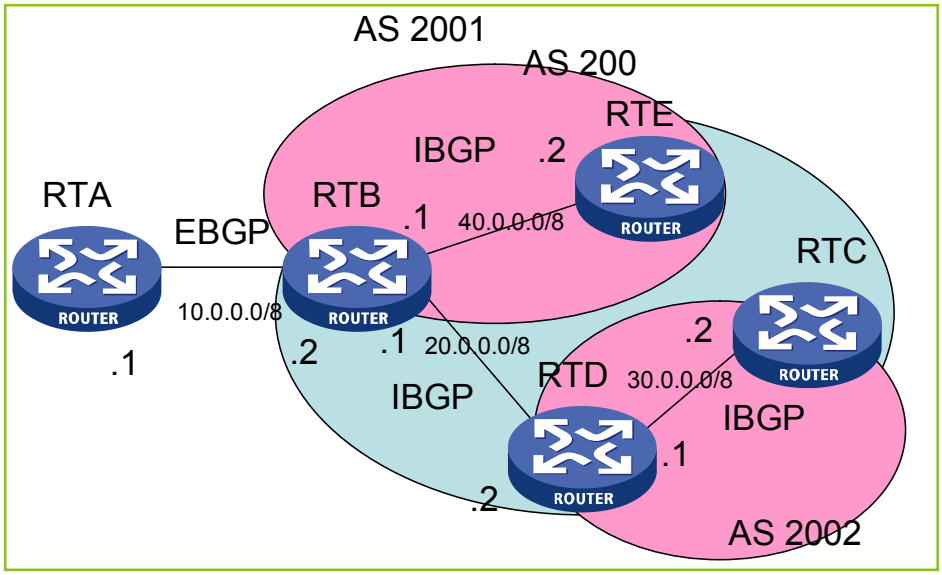


图11 联盟路由传递过程示意图

1.从RTC上发布一条4.4.4.4/32的主机路由，携带MED值为100。

```
[RTC]dis bgp routing-table 4.4.4.4

 BGP local router ID : 4.4.4.4
 Local AS number : 2002
 Paths:   1 available, 1 best

 BGP routing table entry information of 4.4.4.4/32:
 Imported route.
 From            : 0.0.0.0 (0.0.0.0)
 Original nexthop: 127.0.0.1
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 100, pref-val 0, pre 0
 State           : valid, local, best,
 Advertised to such 1 peers:
    30.0.0.1
```

在RTD上查看该路由为子系统内路由，但还路由还未出子系统，故该路由还未附上自己的子系统号；该路由的MDE值为100，Localpref为100。

```
[RTD]dis bgp routing-table 4.4.4.4

 BGP local router ID : 3.3.3.3
 Local AS number : 2002
 Paths:   1 available, 1 best

 BGP routing table entry information of 4.4.4.4/32:
 From            : 30.0.0.2 (4.4.4.4)
 Relay Nexthop    : 0.0.0.0
 Original nexthop  : 30.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 100, localpref 100, pref-val 0, pre 255
 State           : valid, internal-confed, best,
 Advertised to such 1 peers:
    20.0.0.1
```

在RTB上查看路由已经进入到了子系统2001内，故该路由为子系统外路由且携带2002的子系统号；该路由的MDE值为100，Localpref为100，并未发生变化。

```
[RTB]dis bgp routing-table 4.4.4.4

 BGP local router ID : 2.2.2.2
 Local AS number : 2001
 Paths:   1 available, 1 best

 BGP routing table entry information of 4.4.4.4/32:
 From            : 20.0.0.2 (3.3.3.3)
 Relay Nexthop    : 20.0.0.2
 Original nexthop  : 30.0.0.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 100, localpref 100, pref-val 0, pre 255
 State           : valid, external-confed, best,
 Advertised to such 2 peers:
    10.0.0.1
    40.0.0.2
```

在RTE上参看该路由为子系统内路由且携带2002的子系统号；该路由的MDE值为100，Localpref为100，并未发生变化。

```
[RTE]dis bgp routing-table 4.4.4.4

 BGP local router ID : 5.5.5.5
 Local AS number : 2001
 Paths:   1 available, 1 best

 BGP routing table entry information of 4.4.4.4/32:
 From            : 40.0.0.2 (2.2.2.2)
 Relay Nexthop    : 0.0.0.0
 Original nexthop  : 40.0.0.2
 AS-path         : (2002)
 Origin          : incomplete
 Attribute value : MED 100, localpref 100, pref-val 0, pre 255
 State           : valid, internal-confed, best,
Not advertised to any peers yet
```

在RTA上查看该路由为EBGP路由，并携带对段200的子系统号，2002的子系统号被删掉了，同时，MED和Localpef均已经被删掉。

```
[RTA]dis bgp routing-table 4.4.4.4

 BGP local router ID : 5.5.5.5
 Local AS number : 100
 Paths:   1 available, 1 best

 BGP routing table entry information of 4.4.4.4/32:
 From            : 10.0.0.2 (2.2.2.2)
 Original nexthop: 10.0.0.2
 AS-path         : 200
 Origin          : incomplete
 Attribute value : pref-val 0, pre 255
 State           : valid, external, best,
 Not advertised to any peers yet
```

2. 在RTA上发布一条1.1.1.1/32的主机路由，MED为200

```
[RTA]dis bgp routing-table 1.1.1.1

 BGP local router ID : 1.1.1.1
 Local AS number : 100
 Paths:   1 available, 1 best

 BGP routing table entry information of 1.1.1.1/32:
 From            : 0.0.0.0 (0.0.0.0)
 Original nexthop: 127.0.0.1
 AS-path         : (null)
Origin          : incomplete
 Attribute value : MED 200, pref-val 0, pre 0
 State           : valid, local, best,
  Advertised to such 1 peers:
    10.0.0.2
```

在RTB上查看该路由为EBGP路由，并携带100的AS号，MED为200，AS－path为100。

```
[RTB]dis bgp routing-table 1.1.1.1

 BGP local router ID : 1.1.1.1
 Local AS number : 2001
 Paths:   1 available, 1 best

 BGP routing table entry information of 1.1.1.1/32:
 From            : 10.0.0.1 (1.1.1.1)
 Original nexthop: 10.0.0.1
 AS-path         : 100
Origin          : incomplete
 Attribute value : MED 200, pref-val 0, pre 255
 State           : valid, external, best,
 Advertised to such 2 peers:
    20.0.0.2
    40.0.0.2
```

在RTE上查看该路由为子系统内路由，MED为200，Localpref为100，AS－path为100。

```
[RTE]dis bgp routing-table 1.1.1.1

 BGP local router ID : 5.5.5.5
 Local AS number : 2001
 Paths:   1 available, 1 best

 BGP routing table entry information of 1.1.1.1/32:
 From            : 40.0.0.1 (2.2.2.2)
Relay Nexthop    : 0.0.0.0
 Original nexthop: 40.0.0.1
 AS-path         : 100
Origin          : incomplete
 Attribute value : MED 200, localpref 100, pref-val 0, pre 255
 State           : valid, internal-confed, best,
Not advertised to any peers yet
```

在RTD上查看该路由为子系统外路由，路由MED为200、Localpref为100，并且携带始发AS号100和AS 200内子系统2001子系统号。

```
[RTD]dis bgp routing-table 1.1.1.1

 BGP local router ID : 3.3.3.3
 Local AS number : 2002
 Paths:   1 available, 1 best

 BGP routing table entry information of 1.1.1.1/32:
 From            : 20.0.0.1 (2.2.2.2)
Relay Nexthop    : 0.0.0.0
 Original nexthop: 20.0.0.1
 AS-path         : (2001)100
Origin          : incomplete
 Attribute value : MED 200, localpref 100, pref-val 0, pre 255
 State           : valid, external-confed, best,
Advertised to such 1 peers:
    30.0.0.2
```

联盟的环路避免

联盟可以很容易地检测到AS内的选路循环，因为子AS之间运行的是EBGP。AS路径列表用于检测离开一个子AS并想回到同一子AS的选路更新。这种想要回到它始发的子AS的选路更新被检测到是因为子AS会发现自己的子AS号码在这个更新的AS路径内。联盟的缺陷是：从非联盟向联盟方案转变时，要求路由器重新进行配置，逻辑拓扑基本上也要改变；而且，若没有手工设置的BGP策略，通过联盟的选路有可能选不到最佳的路径。

如果AS内部既没有配置全连接也没有配置RR/联盟时，就可能产生转发黑洞。那么如何来解决这样的问题呢？那我们就不得不提到BGP的同步功能了。

BGP的“同步”

所谓“同步”，是指IGP和BGP之间的同步，亦即“BGP一直要等到IGP在本AS中传播了同一条路由后，再给其他各AS通告过渡路由”。也就是说：在通告给其他AS一条路由时，先要保证本AS内部的路由器（无论是否运行BGP协议）都要知道去往该条路由的路径。同步的目的是避免出现误导外部AS路由器的现象发生，防止一个AS(不是所有的路由器都运行BGP)内部出现路由黑洞，即向外部通告了一个本AS不可达的虚假的路由。

BGP 同步功能

起用同步功能后在接收到IBGP邻居发过来的路由后都会查看该路由是否已经在IGP路由表中，如果IGP路由表中有这条路由，BGP路由表才会将这条路由置为有效；如果IGP路由表中没有该路由则BGP表中的该条路由是无效的。如果关闭同步功能，在收到IBGP邻居发来的路由更新后不检查IGP表是否有该路由，而直接将该路由置为有效。

但是还存在以下两种情况下可以安全地关闭同步：

1、本AS不是一个过渡的AS；

2、本AS内所有Transit路由器都运行BGP且全连接。

BGP 同步的应用场合

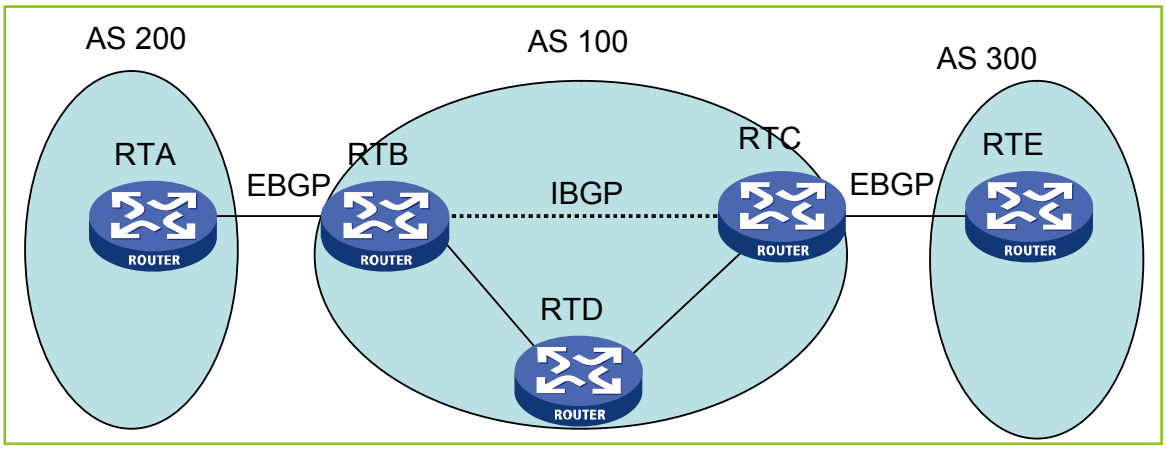


图12 同步的应用场合

在上图中，RTD没有运行BGP，RTC关闭了同步功能。从RTA始发路由1.1.1.1/32，向RTE传递。RTB和RTC收到该路由后将其置为有效， RTC把该路由向RTE进行传递，在RTE上该路由为生效路由。但是在RTC上并未检查该路由在IGP中是否存在。如有在RTE上存在到1.1.1.1的流量，RTE把改流量转发给RTC，RTC把该流量转发给RTD，但是RTD上不存在1.1.1.1/32的IGP路由，故流量丢失。如RTC开启BGP同步功能，当RTC检测不到1.1.1.1/32的IGP路由时，RTC便不会把该路由传递给RTE，这时，RTE可能会把该流量转给给其他流量，也避免流量丢失的情况。

如果要解决该问题，需要在RTB上把1.1.1.1/32路由引入到IGP中，并在IGP中进行传递，这样并开启同步功能，就不会出现AS内路由黑洞的问题。

BGP 同步的问题

如果一个AS内部存在非BGP路由器，那么就出现了BGP和IGP的边界，需要在边界路由器将BGP路由发布到IGP中，才能保证AS所通告到外部的BGP路由在AS内部是连通的。实际上是要求BGP路由和IGP路由的同步。如果将BGP路由发布到IGP中，由于BGP路由量一般较大，那么结果是IGP路由器要维护大量的外部路由，对路由器的CPU和内存以及AS内部的链路带宽的占用将带来巨大的开销；同时会带来收敛延时增加。通常BGP协议的运行需要关闭同步。

总结

联盟和反射两种技术都是为了解决大规模网络中IBGP必须全连接的问题，两种技术都有自己的特点，反射技术简单易理解，不需要更改现有的网络拓扑，兼容性好，反射器对于客户来说是透明的，群与群之间仍然需要全连接，适用于中到大规模网络；而联盟技术比较复杂，联盟内所有的路由器都需要支持联盟技术，但子自治系统之间是特殊的EBGP连接，因此不需要全连接，适用于大规模网络。二者需要根据实际需求进行部署。

BGP选路解析

文/贾欣武

BGP选路概述

解析BGP选路的意义

每个路由协议都有自己计算路由的方法，计算路由的方法称为路由算法，BGP选路方法就是BGP的路由算法，BGP运行路由算法的目的是计算出有效路由进而优选出最优路由，选路算法是BGP路由协议的核心算法之一。

BGP选路与常见IGP选路的区别

众所周知，链路状态算法的路由协议，其路由非通告所得，而是计算所得。在采用链路状态算法的路由协议如OSPF中，在其作用域内无法人为地干涉路由优选，即算法不可改变，在路由器的实现中在代码中固定，人为干涉的结果会导致路由无法计算或计算出错，在链路状态算法的作用域之间，有相对比较简单的计算规则，一般也没有必要人为地去干涉选路（如OSPF协议的区域间路由）。

以上原因导致OSPF的路由计算对网络管理员来说比较傻瓜化，大部分的选路工作由机器完成，管理员参与的部分极少。

基于距离矢量的IGP，如RIP由于路由协议中携带的信息量极少，可供选路决策的条件很少，所以讨论其协议内部的路由优选意义不大。

BGP选路是一个比较复杂的过程，需要深入讨论，原因是BGP的设计者将需要大部分由代码固化完成的工作分了一部分出来“允许”管理员参与完成，在协议中也包含了丰富的优选参数，可供选路时自动或人为地进行控制与决策。这也说明，关于BGP的主要工作内容由两部分：

- 在AS之间及AS内部传递路由——自动完成
- 控制、管理、优化路由——自动或由管理员手动完成

通过选路，我们可以看出设计者设计如此多属性的原因，与通用的IGP协议采用单一Metric计算路由相比，BGP的众多属性更细致地反映一条路由的“历史背景”，在选路过程中可以自动或手动地利用这些丰富的材料进行综合考虑，进而更为细腻地优选和控制路由。

BGP选路过程解析

选路规则

BGP IPv4选路规则如下：

- 下一跳（Next_Hop）不可达的路由及其他无效路由不参与优选；
- 优选协议优先级值低的路由；
- 标签路由（有LSP隧道）优于非标签路由；
- 若配置了Preferred-value值，优选值高的；
- 优选本地优先级（Local_Pref）最高的路由；
- 优选本路由器始发的路由；
- 优选AS路径（AS_Path）最短的路由；
- 依次选择Origin属性值为IGP、EGP、Incomplete的路由；
- 优选MED值最低的路由；
- EBGP路由优于联盟EBGP路由，联盟EBGP路由由于IBGP路由；
- 优选下一跳（Next_Hop）花费（Cost）值最低的路由；
- 优选Cluster_List长度最短的路由；
- 优选Originator_ID最小的路由；
- Router ID值小者优先；
- BGP会话地址小者优先。

以上优选规则是从前到后依次比较的，只有在前一个条件无法选出最优路由的情况下才考虑紧接的后一个条件。

实例解析选路过程

在本小节，通过实例的方式，按照以上选路规则解析路由器在选路过程中各步骤的决策条件。由于BGP属性丰富，实际使用环境千变万化，在实际使用中会出现各种应用场景，一个完整体现所有决策过程的拓扑会很复杂，为了简洁明了地展开，本文用简化的拓扑来解析每个决策步骤。

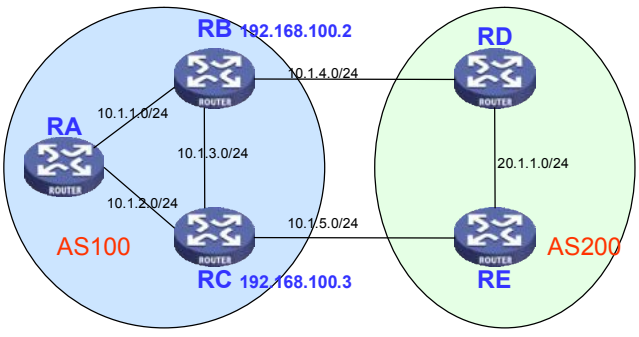


图 1 基本选路拓扑

图1说明：

RD、RE属于AS200，RD与RE建立IBGP连接，RB与RD建立EBGP连接，RC与RE建立EBGP连接

RD、RE属于AS200，建立Full mesh IBGP连接

RA环回地址：192.168.100.1

RB环回地址：192.168.100.2

RC环回地址：192.168.100.3

步骤1：NEXT_HOP不可达的路由及没有隧道的标签路由都是无效路由，不参与优选

解析：RA从RB和RC分别收到关于20.1.1.0/24的路由，但下一跳没有改变，下一跳属于AS外部路由，RA不可达，所以下面两条路由没有星号（*）标志，都是无效的路由。

在RA上的BGP表信息如下：

```
[RA]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop       MED       LocPrf     PrefVal   Path/Ogn
  i 20.1.1.0/24        10.1.4.2        0          100        0         200i
  i                    10.1.5.2        0          100        0         200i
```

步骤2：优选协议优先级低的路由

解析：在路由有效的情况下比较路由优先级。通过路由策略修改协议优先级，RA从RC收到的路由优先级修改为250，而RA从RB收到的路由优先级为缺省的255，所以在RA上优选的结果是从RC收到的路由更优。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network          NextHop          MED       LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24        192.168.100.3     0          100        0       200i
 * i                   192.168.100.2     0          150        0       200i
[RA-bgp]disp ip rout 20.1.1.1
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre  Cost          NextHop         Interface
20.1.1.0/24         BGP     250  0             192.168.100.3   GE0/1.112
```

步骤3：标签路由优于非标签路由

解析：取消上一步修改路由优先级的路由策略配置，RA从RB和RC收到的路由优先级相同，根据协议优先级无法选出最优路由。继续比较，首先考虑的条件是该路由是否是标签路由（是否存在LSP隧道）。重新配置路由策略，使RC向RA发送路由的同时分发标签，带有标签的路由优于非标签路由，所以优选从RC收到的路由。

```
[RA]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
   Network            NextHop         MED        LocPrf    PrefVal  Path/Ogn
 *>i 20.1.1.0/24        192.168.100.3     0          100        0       200i
 * i                    192.168.100.2     0          150        0       200i
[RA]disp ip rout 20.1.1.1
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto  Pre  Cost         NextHop         Interface
20.1.1.0/24         BGP    255  0            192.168.100.3   GE0/1.112
[RA]display mpls lsp include 20.1.1.1 24
------------------------------------------------------------------------------
                LSP Information: BGP LSP
------------------------------------------------------------------------------
FEC               In/Out Label  In/Out IF                       Vrf Name
20.1.1.0/24         NULL/1025     -/-
[RA-route-policy]display bgp routing-table 20.1.1.0
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 Label information (Received/Applied): 1025/NULL
 From            : 192.168.100.3 (192.168.50.29)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : 200
 Origin          : igp
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Not advertised to any peers yet
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.2 (135.1.1.1)
 Relay Nexthop   : 10.1.1.2
 Original nexthop: 192.168.100.2
 AS-path         : 200
 Origin          : igp
 Attribute value : MED 0, localpref 150, pref-val 0, pre 255
 State           : valid, internal,


  Not advertised to any peers yet
```

步骤4：Preferred-value值高的路由优先

解析：取消上一步分发标签的策略，RA从RB和RC收到的路由都没有LSP隧道，所有前述条件无法选出最优路由，系统将进一步比较Preferred-value值优选路由。配置路由策略，使RA从RC收到的路由Preferred-value值较高，RA优选从RC收到的路由。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24        192.168.100.3      0          100        30      200i
 * i                    192.168.100.2                 150        0       200i
[RA-bgp]display bgp routing-table 20.1.1.0
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.2 (135.1.1.1)
 Relay Nexthop   : 10.1.1.2
 Original nexthop: 192.168.100.2
 AS-path         : 200
 Origin          : igp
 Attribute value : localpref 150, pref-val 0, pre 255
 State           : valid, internal,
 Not advertised to any peers yet
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.3 (192.168.50.29)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : 200
 Origin          : igp
 Attribute value : MED 0, localpref 100, pref-val 30, pre 255
 State           : valid, internal, best,
 Not advertised to any peers yet
```

步骤5：Local_Preference值高的路由优先

解析：在Preferred-value值相同的情况，进一步比较Local_Preference值优选路由，取消上一步修改Preferred-value值的路由策略，RA从RB收到的路由Local_Pre值较高，所以优选从RB收到的路由。

```
[RA-route-policy]disp bgp ro
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24        192.168.100.2   0          150        0       200i
 * i                    192.168.100.3   0          100        0       200i
[RA-route-policy]display bgp routing-table 20.1.1.0
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.3 (192.168.50.29)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
```

（接下页）

（接上页）

```
    AS-path         : 200
    Origin          : igp
    Attribute value : MED 0, localpref 100, pref-val 0, pre 255
    State           : valid, internal,
    Not advertised to any peers yet
    BGP routing table entry information of 20.1.1.0/24:
    From            : 192.168.100.2 (135.1.1.1)
    Relay Nexthop   : 10.1.1.2
    Original nexthop: 192.168.100.2
    AS-path         : 200
    Origin          : igp
    Attribute value : MED 0, localpref 150, pref-val 0, pre 255
    State           : valid, internal, best,
     Not advertised to any peers yet
```

步骤6：本地路由优先

解析：在RA上用路由聚合命令生成本地路由，并且使本地生成路由、IBGP路由、EBGP路由的路由优先级都相等。这时其他先决条件都相等的情况下，本地生成的路由优先。

```
   [RA-route-policy]display bgp routing-table
    Total Number of Routes: 3
    BGP Local Router ID is 192.168.100.1
    Status codes: * - valid, > - best, d - damped,
                  h - history,  i - internal, s - suppressed, S - Stale
                  Origin : i - IGP, e - EGP, ? - incomplete
      Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
    *>  20.1.1.0/24        127.0.0.1                  100        0       i
    * i                    192.168.100.3              100        0       i
   [RA-route-policy]display bgp routing-table 20.1.1.0 24
    BGP local Router ID : 192.168.100.1
    Local AS number : 100
    Paths:   2 available, 1 best
    BGP routing table entry information of 20.1.1.0/24:
    From            : 192.168.100.3 (192.168.50.29)
    Relay Nexthop   : 10.1.2.2
    Original nexthop: 192.168.100.3
    AS-path         : (null)
    Origin          : igp
    Attribute value : localpref 100, pref-val 0, pre 150
    State           : valid, internal,
    This route is an atomic-aggregated route
    Aggregator      : AS 100, Aggregator ID: 192.168.50.29
    Not advertised to any peers yet
    BGP routing table entry information of 20.1.1.0/24:
    Imported route.
    From            : 0.0.0.0 (0.0.0.0)
    Original nexthop: 127.0.0.1
    AS-path         : (null)
    Origin          : igp
    Attribute value : localpref 100, pref-val 0, pre 150
    State           : valid, local, best,
    This route is an aggregated route
    Aggregator      : AS 100, Aggregator ID: 192.168.100.1
    Advertised to such 1 peers:
     192.168.100.3
```

步骤7：AS-Path较短的路由优先

解析：在前述条件都相同的情况下，将进一步比较AS-Path长度来优选路由。配置添加AS号的路由策略，RA从RC收到关于20.1.1.0/24的路由AS-path中包含两个AS号，从RB收到的路由中包含三个AS号，在其它先决条件都相同的情况下，RA从RC收到的路由AS-path列表相对更短，所以优选从RC收到的路由。AS_SEQ类型的AS-path长度为所包含的AS个数，AS_SET类型的AS-path长度为1，联盟AS类型不计入AS-path长度。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 4
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24         192.168.100.3   0          100        0       300 200i
 * i                     192.168.100.2              100        0       400 300
                                                                       200i
```

步骤8：依次选择Origin属性值为IGP、EGP、Incomplete的路由

解析1：在AS-Path长度相同的情况下将进一步比较Origin属性的值。

下表显示RA从RB收到的路由origin属性为incomplete，从RC收到的路由origin属性为egp，在其他先决条件都相等的情况下，origin属性egp优于incomplete，RA优选从RC收到的路由

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 4
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop           MED        LocPrf    PrefVal   Path/Ogn
 *>i 20.1.1.0/24         192.168.100.3     0          100       0         320 200e
 * i                     192.168.100.2               100       0         300 200?
```

解析2：下表显示RA从RB收到的路由origin属性被修改为igp，从RC收到的路由origin属性仍然为egp，在其他先决条件都相等的情况下，origin属性igp优于egp，所以最优路由发生变化，从RB收到的路由变成最优路由

```
[RA-route-policy]display bgp routing-table
 Total Number of Routes: 4
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf    PrefVal   Path/Ogn
 *>i 20.1.1.0/24         192.168.100.2              100       0         300 200i
 * i                     192.168.100.3   0          100       0         320 200e
```

总结：IGP优于EGP，EGP优于incomplete

步骤9：优选MED值最低的路由

解析1：取消上一步修改Origin属性的路由策略，在Origin属性及其他先决条件都无法比较出最优路由的情况下，MED属性值小的路由优先，下表RA从RC收到关于20.1.1.0/24的路由MED值更低所以更优。

```
[RA-route-policy]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf     PrefVal  Path/Ogn
 *>i 20.1.1.0/24       192.168.100.3   15         100        0        320 200e
 * i                   192.168.100.2   20         100        0        320 200e
```

步骤10：依次优选EBGP路由、联盟EBGP路由、IBGP路由

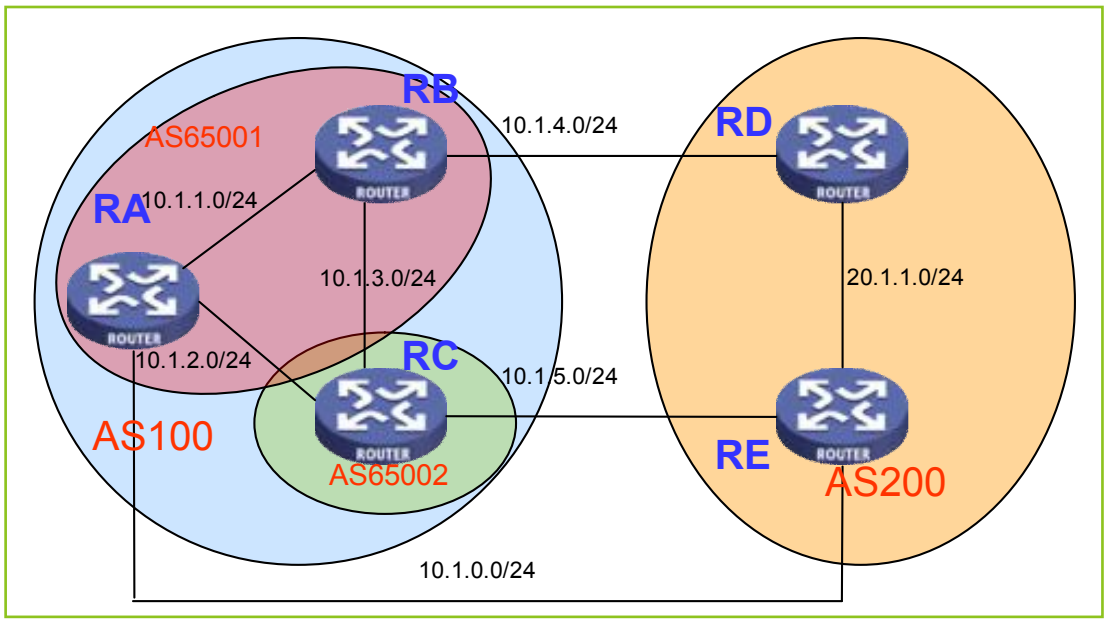


图2 联盟选路拓扑

图2说明：

RA和RB属于联盟子系统65001，RC属于联盟子系统65002；

RA与RB建立联盟子系统内IBGP邻居关系；

RA与RC建立联盟子系统间EBGP邻居关系；

RA与RE建立普通EBGP邻居关系。

解析1：在图2中，RA分别从RB、RC、RE收到路由，在本步骤前的其他先决条件都相同的情况下优选来自EBGP的路由，EBGP路由优于联盟EBGP和IBGP路由，所以RA优选来自RE的路由。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 3
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop        MED        LocPrf     PrefVal  Path/Ogn
 *>  20.1.1.0/24       10.1.0.2        5          160        0        200i
 * i                   192.168.100.3   5          160        0        (65002)
                                                                      200i
 * i                   192.168.100.2   5          160        0        200i
```

解析1：当EBGP路由被撤销后，重新优选，由于联盟EBGP路由优于IBGP路由，所以来自RC的路由优于来自RB的路由

```
[RA-GigabitEthernet0/1.110]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop          MED        LocPrf     PrefVal   Path/Ogn
 *>i 20.1.1.0/24       192.168.100.3     5          160        0         (65002)
                                                                        200i
 * i                   192.168.100.2     5          160        0         200i
```

步骤11：优选下一跳Cost值最低的路由

解析：在其他先决条件都相等的情况下将比较下一跳的Cost值，RA从RC收到的路由下一跳为RC的环回地址192.168.100.3，到达该地址的IGP cost为13；RA从RB收到的路由下一跳为RB的环回地址192.168.100.2，到达该地址的IGP cost为15。因此优选从RC收到的路由。

```
[RA-GigabitEthernet0/1.112]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Network            NextHop          MED        LocPrf     PrefVal   Path/Ogn
 *>i 20.1.1.0/24    192.168.100.3               100        0         200i
 * i                 192.168.100.2     0         100        0         200i

[RA-GigabitEthernet0/1.112]disp ip rout 192.168.100.2
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto   Pre  Cost          NextHop         Interface
192.168.100.2/32    OSPF    10   15            10.1.1.2        GE0/1.111
[RA-GigabitEthernet0/1.112]disp ip rout 192.168.100.3
Routing Table : Public
Summary Count : 1
Destination/Mask    Proto     Pre    Cost       NextHop         Interface

192.168.100.3/32    OSPF      10     13         10.1.2.2         GE0/1.112
```

步骤12：优选Cluster_List长度最短的路由

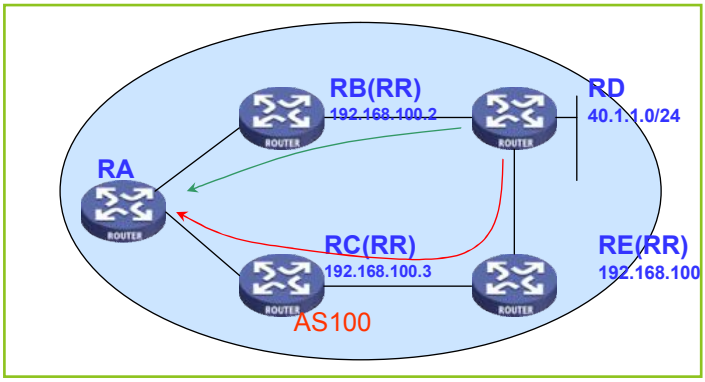


图3 根据Cluster_List选路

在图3中，BGP会话关系用黑色链路表示，绿色线表示RB向RA反射从RD收到的路由；红色线表示RD发出的路由经过两个反射器（RE和RC）反射到RA。

RA从RB收到的路由Cluster_List中只有一个条目；RA从RC收到的路由Cluster_List中有两个条目。所以从RB收到的路由更优。

```
[RA]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network             NextHop         MED      LocPrf     PrefVal Path/Ogn
 *>i 40.1.1.0/24        192.168.100.2              100         0       ?
 * i                    192.168.100.3              100         0       ?
[RA]display bgp routing-table 40.1.1.0
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 40.1.1.0/24:
 From            : 30.1.1.3 (30.1.1.3)
 Relay Nexthop   : 10.1.1.2

 Original nexthop: 192.168.100.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originator      : 192.168.100.4
 Cluster list    : 192.168.100.2
 Not advertised to any peers yet
 BGP routing table entry information of 40.1.1.0/24:
 From            : 30.1.1.2 (30.1.1.2)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal,
 Originator      : 192.168.100.4
 Cluster list    : 192.168.100.3, 192.168.100.5
 Not advertised to any peers yet
```

步骤13：优选Originator_ID最小的路由；

在前述步骤未选出最优路由的情况下，进一步比较Originator_ID。

在图4中，BGP会话关系用黑色链路表示，绿色线表示RB向RA反射从RD收到的路由；红色线表示RC向RA反射从RE收到的路由。

解析：RA从RB收到的路由中Originator_ID相对更小，所以从RB收到的路由更优。

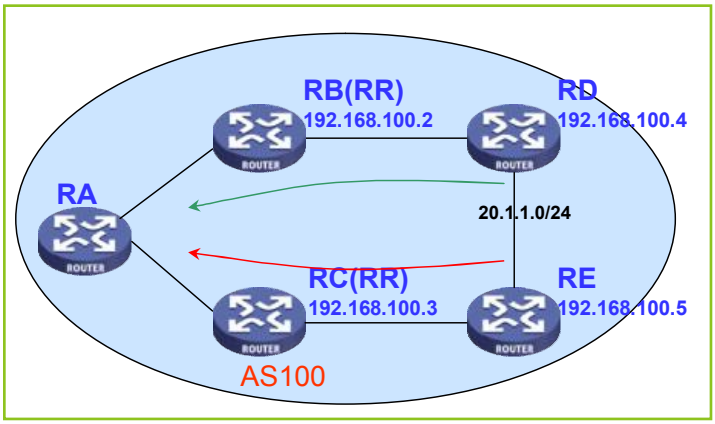


图4 根据Originator_ID选路

```
isplay bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network             NextHop         MED        LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24        192.168.100.2                100         0       ?
 * i                    192.168.100.3                100         0       ?

[RA]display bgp routing-table 20.1.1.1
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 From            : 30.1.1.2 (30.1.1.2)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal,
 Originator      : 192.168.100.5
 Cluster list    : 192.168.100.3
 Not advertised to any peers yet
 BGP routing table entry information of 20.1.1.0/24:
 From            : 30.1.1.3 (30.1.1.3)
 Relay Nexthop   : 10.1.1.2
 Original nexthop: 192.168.100.2
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Originator      : 192.168.100.4
 Cluster list    : 192.168.100.2
 Not advertised to any peers yet
```

步骤14：Router ID值小者优先

RA从RB和RC收到的路由在前述步骤都无法优选出路由的情况下需要进一步比较Router ID来选出最优路由，在此步骤中Router ID值小的路由优先。

解析：RB的BGP Router ID为6.6.6.6；RC的BGP Router ID为5.5.5.5。所以在RA的BGP路由表中，从RC收到的路由优先。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network            NextHop         MED        LocPrf    PrefVal  Path/Ogn
 *>i 20.1.1.0/24       192.168.100.3              100        0       200i
 * i                   192.168.100.2    0         100        0       200i
[RA-bgp]display bgp routing-table 20.1.1.1
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.3 (5.5.5.5)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : 200
 Origin          : igp
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Not advertised to any peers yet
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.2 (6.6.6.6)
 Relay Nexthop   : 10.1.1.2
 Original nexthop: 192.168.100.2
 AS-path         : 200
 Origin          : igp
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal,

 Not advertised to any peers yet
```

步骤15：建立BGP会话的地址值小者优先

RA从RB和RC收到的路由在前述步骤都无法优选出路由的情况下需要进一步比较建立BGP会话的地址值来选出最优路由，在此步骤中BGP会话地址值小的路由优先。

解析：在图1中RB的环回地址为192.168.100.2，RB与RA的BGP会话通过此地址建立；RC的环回地址为192.168.100.3，RC与RA的BGP会话通过此地址建立。因为RB与RA建立BGP会话的地址相对更小，所以在RA的BGP路由表中，从RB收到的路由优先。

```
[RA-bgp]display bgp routing-table
 Total Number of Routes: 2
 BGP Local Router ID is 192.168.100.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
    Network             NextHop        MED        LocPrf     PrefVal Path/Ogn
 *>i 20.1.1.0/24        192.168.100.2   0          100        0       200i
 * i                    192.168.100.3              100        0       200i
[RA-bgp]display bgp routing-table 20.1.1.1
 BGP local Router ID : 192.168.100.1
 Local AS number : 100
 Paths:   2 available, 1 best
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.2 (6.6.6.6)
 Relay Nexthop   : 10.1.1.2
 Original nexthop: 192.168.100.2
 AS-path         : 200
 Origin          : igp
 Attribute value : MED 0, localpref 100, pref-val 0, pre 255
 State           : valid, internal, best,
 Not advertised to any peers yet
 BGP routing table entry information of 20.1.1.0/24:
 From            : 192.168.100.3 (6.6.6.6)
 Relay Nexthop   : 10.1.2.2
 Original nexthop: 192.168.100.3
 AS-path         : 200
 Origin          : igp
 Attribute value : localpref 100, pref-val 0, pre 255
 State           : valid, internal,
 Not advertised to any peers yet
```

优选路由的策略

优选路由的策略方法有很多，简单举例如下：

影响某台路由器本地选路结果

建议配置路由策略在本地修改Preferred-value来影响选路过程；由于Preferred-value值不属于路由属性，修改后只在本地生效，不会随路由信息传播。但影响本地选路后，BGP只发送最优路由，对其他路由器也有一定的影响。

影响IBGP邻居的选路

如果希望影响本AS内部的路由器优选自己发出的路由，建议通过修改Local_Preference属性值来实现，在图1中，RB向RA发送的路由中Local_Preference属性值为150，RC向RA发送的路由中Local_Preference属性值为缺省值100，在RA上将优选从RB收到的路由，出AS的流量将通过RB发向AS200。

Local_Preference属性可以向所有IBGP邻居发送，可以对本AS内部所有的IBGP邻居选路产生影响。

影响EBGP邻居的选路

如果希望EBGP邻居优选自己发出的路由，建议通过修改MED属性值来实现，在图1中，RD向EBGP邻居RB发送的路由中MED属性值为20，RE希望AS100中的路由器优选自己发出的路由，所以向EBGP邻居RC发送的路由中MED属性值为15，低于RD向RB发送路由的MED值，因为MED值低的路由更优，在AS100中将优选来自RE的路由。

在使用MED属性时须注意，利用MED值进行优选是有一定条件的，来自不同AS的路由不能根据MED值优选路由。

优选过程总结

从以上选路方式可以看出，在选路过程中，人为干预的决策条件（如：Preferred-value）以及本自治系统内使用的管理属性（如：Local_Preference）等条件都放在选路决策的前几步，而随机条件（如：Router ID和会话地址的决策）都在最后，目的是使路由优选的管理和控制更加方便，这样可以在各自治系统内部按照统一的策略来管理和计算路由。

BGP Graceful Restart

文/陈磊

Graceful Restart简介

关于GR通用的基本概念，原理，以及作用，请参考《网络之路-OSPF专题讨论》中的文章《OSPF Graceful Restart》。

该文对于GR的来源，作用和通原理做了详细的介绍，本文将跳过这方面，直接进入BGP GR实现的介绍。

BGP Graceful Restart概述

GR中的角色

从GR中完成的任务来看，分为GR Restarter（协议重起的设备）和GR Helper（协助完成协议重起的设备）两个角色。在本文中，将BGP GR过程中的GR Restarter称为Restarting Speaker，GR Helper称为Receiving Speaker。

GR对BGP的要求

为了实现GR，BGP必须满足以下几点：

- 在邻居之间通告各自的GR能力；
- GR被触发后，通知邻居GR事件的发生；
- Receiving Speaker上，在与Restarting Speaker之间的BGP会话所依赖的TCP连接中断后或者重建过程中，需要

保留并使用从Restarting Speaker学来的路由，并进行相应的标记和维护；

- GR正常结束时机的确定，以及异常退出的条件；

为了做到以上几点，BGP引入了如下的变化：

Marker for End-of-RIB

定义：一个不包含任何NLRI，或withdrawn NLRI 的update被定义为Marker for End-of-RIB。

如：当一个BGP连接建立完成后，设备会向邻居发送自己的BGP路由。当所有路由发送完成后，会在最后一个Update后，再发送一个空的Update，这个空的Update就是Marker for End-of-RIB。被用作路由信息表结束的标志。

End-of-RIB可以帮助设备确认已经学习到了邻居所有的路由，是GR正常结束的条件之一。

Graceful Restart Capability

GR引入的新BGP capability类型。该能力放在BGP的Open报文中携带，用来在邻居之间通告各自的GR能力；同时，在发生RP（Route Processor）切换事件后，通知对端GR状态的开始。

该CAP的定义如下：

Capability code:: 64

Capability length: 可变长度

Capability value: 由Restart Flags，Restart Time，地址族编号组成。

可以按照不同的地址族，分别通告其GR能力

Restart Flags (4 bits)
Restart Time in seconds (12 bits)
Address Family Identifier (16 bits)
Subsequent Address Family Identifier (8 bits)
Flags for Address Family (8 bits)
...
Address Family Identifier (16 bits)
Subsequent Address Family Identifier (8 bits)
Flags for Address Family (8 bits)

- Restart Flags：该Flag只使用了最高位的一个bit。当Restarting Speaker发生RP切换后，新的RP重新触发BGP邻居的建立，发送的第一个OPEN报文中，该flag会置位。用来告知Receiving Speaker邻

居，本端的BGP开始GR。

- Restart time：由Restarting Speaker告知Receiving Speaker，GR过程中，邻居关系建立需要的时间上限。

- AFI，SAFI，Flags for Address Family：这三个字段组合，实现了GR能力的通告。AFI/SAFI是MP-BGP中用来区分不同网络层协议的地址族编号。每个地址族都对应了一个Flags for Address Family，表示相应的地址族对于GR的支持能力。该Flag也只使用了最高位的一个bit。当Restarting Speaker发生重启，如果能够保证在重启过程中，相应地址族报文的转发不受影响，其发送的OPEN报文中，会将相应的AFI的Flags for Address Family置位。

Graceful restart capability实现GR能力通告的一些补充

如果邻居的OPEN报文中含有Graceful restart capability，表示该邻居一定会在路由发送完成之后发送一个End-of-RIB marker。该标记可以帮助加速路由表的收敛。

无论设备是GR restarter还是GR helper，都要求设备在Open报文中包含Graceful restart capability。以标识自己对于GR的支持。同时，也表示自己支持End-of-RIB marker，邻居可以通过该标志来判断自己路由表的结束。

如果邻居的OPEN报文中的Graceful restart capability的capability value为空，表示邻居是一个GR aware设备：能够作为一个Receiving Speaker，可以识别Restarting Speaker的GR发起信令，配合在Restarting Speaker重起期间，保留从Restarting Speaker学习到的路由转发表项；但是，该设备自己，没有在协议重启过程中，保持路由转发表项的能力，不能做GR.。

BGP路由的Stale状态

Receiving Speaker设备在得知邻居进入GR后，会将从该邻居学来的BGP路由标记为Stale状态。标记为Stale状态的路由，在转发和选路方面，与其他路由没有区别。当GR结束后，Stale路由会根据新路由的学习情况被unmark stale状态或者被删除。

BGP Gracefu Restart过程的行为约束

GR的流程

RFC4724定义了BGP GR的详细流程，结合ComwareV5的GR实现，如图1：

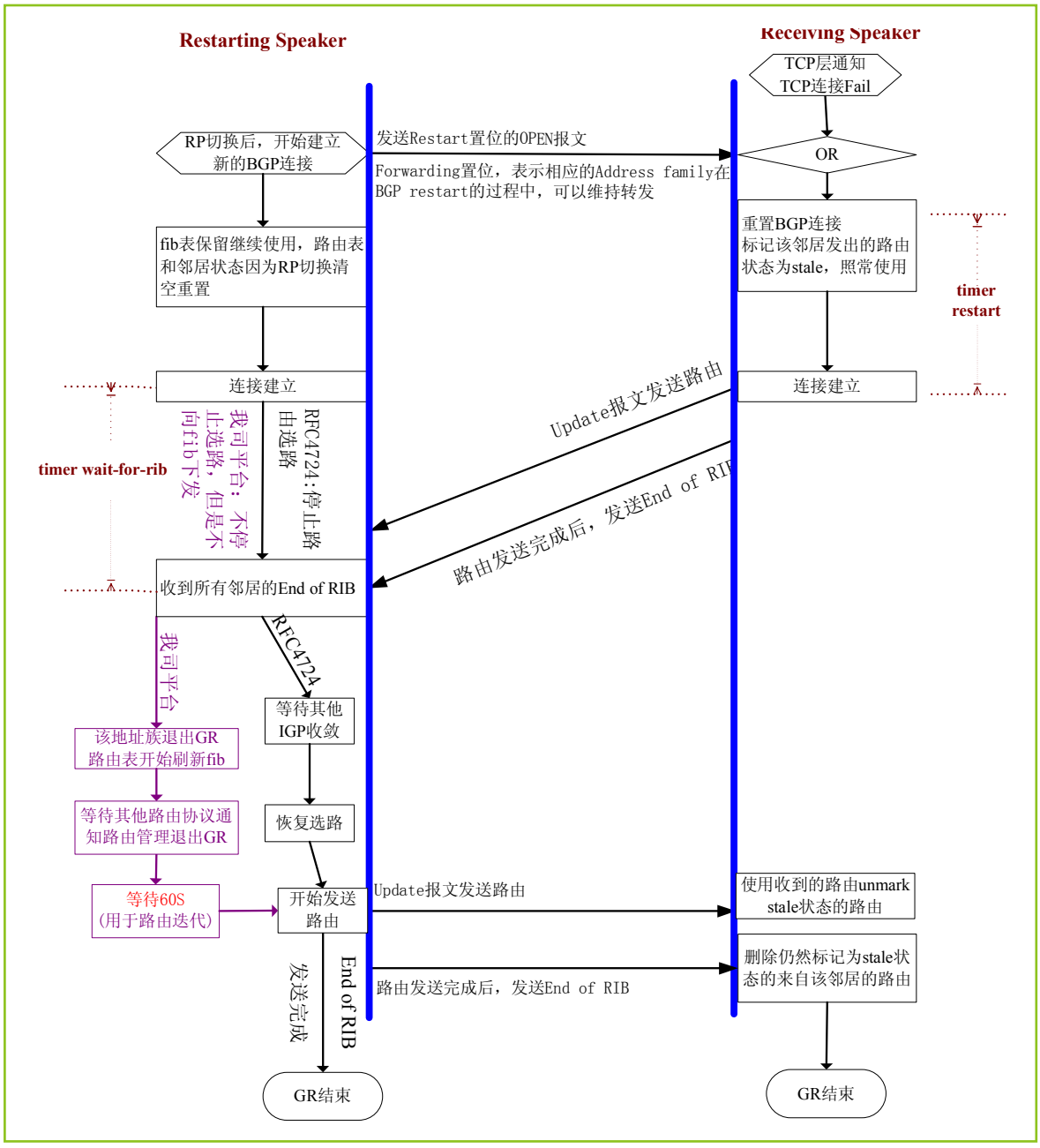


图1 BGP GR流程图

Restarting Speaker的行为约定

BGP邻居重新建立前

RP切换前：

- Restarting Speaker和Receiving Speaker之间在建立BGP邻居时，通过Open报文中的Graceful restart capability来交互两者对于GR的支持，同时，使用End-of-RIB marker来标识各自路由表的结束。

RP切换后，新的RP开始启动BGP连接：

- 保证GR相关地址族的转发表项不受影响，在重启过程中能够正常使用。——从Receiving Speaker转发过来的数据报文仍然能够正常转发。
- 重启BGP连接时，发送的OPEN报文中，restart位置位；GR过程中转发不受影响的地址族的Flags for Address Family置位。——通知Receiving Speaker GR开始，同时，告知其哪些地址族可以实现GR。

BGP邻居重新建立后（RFC4724）：

- 连接建立后，开始从邻居接收update信息，在此过程中，不进行BGP的路由选路，不向外发布路由。——防止在路由收敛完成前发送了错误的路由导致环路或者路由黑洞
- 等待Restarting Speaker收到了所有邻居的End-of-RIB marker（不包括restart位置位的设备和Open报文中没有Graceful restart capability的设备）。——这里的所有邻居中，不包括restart位置位的设备，是为了防止当两台相邻的设备同时重启进行GR时，都等待对方的End-of-RIB marker导致死锁。
- 再等待其他路由协议通知路由收敛完成，才恢复选路，刷新转发表，并向邻居发布路由。路由发布完成后，也需要发送End-of-RIB marker

BGP邻居重新建立后（我司V5平台实现）：

- 连接建立后，开始从邻居接收update信息，在此过程中，不向外发布路由，但是，路由选路仍然进行，但是不刷新FIB；
- Restarting Speaker等待收到所有邻居的End-of-RIB marker后，通知路由管理该地址族退出GR，此时，开始向FIB表刷新路由
- 等待路由管理通知所有的路由协议都已经退出GR后，开启一个60s的定时器；
- 这段时间用来等待路由管理完成路由的迭代，定时器超时后，开始向Receiving Speaker发送路由，路由发送完成后，需要发送End-of-RIB marker

BGP退出GR的条件：

- 按照上面的流程，正常结束退出GR
- 定时器wait-for-rib超时，仍然没有收到所有邻居的End-of-RIB marker，则退出GR。——该定时器用户可配，用来限制Restarting Speaker学习路由的时间上限

Receiving Speaker的行为约定

进入GR helper状态的条件：

- TCP层通知设备，与Restarting Speaker的TCP连接中断。——被动开始
- 收到Restarting Speaker发出的新的Restart置位的OPEN报文。——主动开始

侦测到GR事件后：

- 重置与Restarting Speaker的BGP连接，重置过程中不发送NOTIFICATION报文
- 标记从Restarting Speaker学到的相关地址族的路由为stale状态，当作正常的路由信息使用。——使用stale标记进行区分，方便之后的路由管理
- Receiving Speaker设备的Restart位不置位，除非Receiving Speaker设备自己也进行GR。——这里指两端同时进行GR的情况

连接建立后：

- 向Restarting Speaker发送路由，路由信息发送完成后，发送End-of-RIB marker。即使没有路由需要发送，也要发送End-of-RIB marker。——以告知对方路由表的完结。
- 收到Restarting Speaker发送的路由后，需要使用这些路由刷新stale状态的路由
- 当收到Restarting Speaker发送的End-of-RIB marker后，需要立即删除该邻居相关地址族的所有仍然处于stale状态的路由。

GR helper状态的退出条件：

- 收到Restarting Speaker发出的路由和End-of-RIB marker，刷新自己的stale路由，正常退出
- 定时器Restart timer超时，与Restarting Speaker的邻居关系仍然没有建立，则退出GR helper状态，删除所有标记位stale的路由——该定时器用户可配，用来控制GR过程的时间
- 与Restarting Speaker的邻居建立后，如果在其OPEN报文中没有Graceful restart capability，或者Graceful restart capability中没有某一个地址族，或者某个地址族的Flags for Address Family没有置位，则删除该邻居所有相关地址族的stale路由

GR导致的安全问题

支持GR的设备上，一个新BGP连接的建立，会导致旧连接的中断。这使得设备有可能受到DOS攻击。

解决方法是对BGP连接做认证。

常用BGP AS_PATH正则表达式应用

文/杨默寒 姜杏春

前言

BGP协议提供了非常丰富的路由策略，尤其是在对路由过滤与路由选择上有其它路由协议不可比拟的优势。这样给了使用者一个没有限制的舞台。BGP就像一道好的高考题一样，应用它的时候考察了我们的分析能力、应用能力以及发散思维能力。

BGP路由表，也可以称之Internet路由表，目前规模已经达到十几万。在面对庞大的Internet路由表时，我们不免需要进行路由过滤。利用地址前缀去过滤BGP路由，在如此大规模的路由表时，一来有可能配置比较繁琐，二来有新的路由加入不好维护，所以提出了BGP利用AS_PATH作过滤的办法。由于Internet核心AS的分布都是有记录的，所以利用AS的过滤更有针对性，例如可以使用AS_PATH作过滤，解决过滤从某个AS_PATH始发的全部路由，只需一个AS_PATH列表即可，当然利用AS_PATH过滤可以解决的问题远不仅如此，这还需要我们在下文中慢慢体会。

AS_PATH与正则表达式介绍

AS_PATH 格式

首先让我们来认识一下BGP的AS_PATH属性。

AS_PATH，公认必遵属性。这个属性在传递UPDATE报文中标识了到达一个目的地所经过的AS信息。

AS_PATH有4种类型：

- AS_SEQUENCE（用于路由AS路径记录）
- AS_SET（用于聚合路由的明细路由AS集合）
- AS_CONFED_SEQUENCE（用于联盟路由AS路径记录）
- AS_CONFED_SET（用于联盟聚合路由）

让我们看看AS_PATH在BGP路由表中的显示格式，如图1：

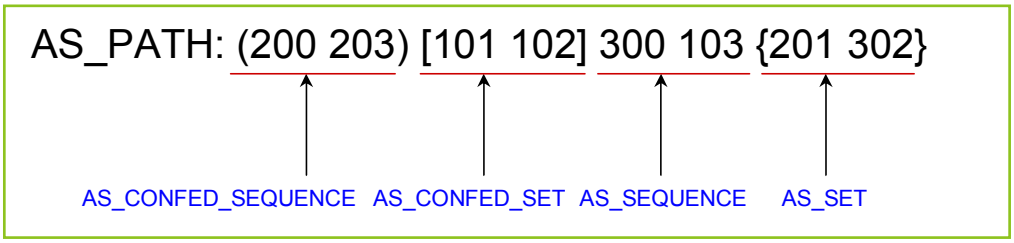


图1 AS_PATH显示格式

从AS_PATH的显示格式看出，AS_PATH可以看成一个由数字0-9，“（ ）”，“[ ]”，“{}”和空格组成的字符串。当然例子中是最复杂的格式，实际应用中并不是所有字段都有填充的。

正则表达式常用操作符

当我们想利用匹配AS_PATH做路由过滤的时候，那么怎么实现对AS_PATH匹配呢？之前分析过AS_PATH可以看作是字符串的这一特点，我们找到了处理字符串的强大工具正则表达式这把利剑来帮我们实现匹配，这样正则表达式就在BGP的舞台登场了。

正则表达式介绍：

首先让我们来认识一下正则表达式，正则表达式是按照一定的模板来匹配字符串的公式。在BGP中，可以对BGP路由的AS_PATH属性做出响应的判断（接收或者拒绝），实际上可以认为它是一个AS_PATH的ACL。

正则表达式常用操作符：

表1 正则表达式常用操作符说明

符号	说明
。	匹配任意字符，包括特殊字符，空格。
*	匹配*前面字符中0次或多次。
+	匹配+前面字符中1次或多次。
?	匹配?前面字符中0次或1次。
^	匹配字符串的开始。
$	匹配字符串的结束。
_	匹配一个符号，如逗号、左大括号、右大括号、左括号、右括号和空格等符号，在表达式的开头或结尾时还可作起始符、结束符（同^ ，$）。
( )	匹配一个变化的字符或一个独立的匹配，通常和“|”一起使用。
|	逻辑或，交替匹配
[ ]	表示一组字符的集合，如果集合中第一个字符为“^”，则表示补集
-	连接符
\	转义操作符，去除特殊字符的特殊意义

实际应用

AS_PATH 常用表达式

下面就利用介绍的符号来对AS_PATH进行匹配，看看这些符号能够给我们带来多么奇妙的效果。

^$

表示匹配的字符串为空，即AS_PATH为空，表示只匹配本地路由。

.*

表示匹配任意字符串，即AS_PATH为任意，表示匹配所有路由。

^100

表示匹配字符串开始为100，即AS_PATH最左边AS前3位（最后一个AS）为100、1001、1002等，表示匹配AS100、1001、1002等邻居发送的路由。

^100_

表示匹配字符串开始为100后面为符号，即AS_PATH最左边AS（最后一个AS）为100，表示匹配AS100邻居发送的路由，比较前一个表达式，”_”的好处就体现出来了，它可以和用来帮助我们限制匹配单独的一个AS。

_100$

表示匹配字符串最后为100，即AS_PATH最右边AS（起始AS）为100，表示匹配AS100始发的路由。

_100_

表示字符串中间有100，即AS_PATH中有100，表示匹配经过AS100的路由。

\(65535_

表示匹配字符串为（65535后面为符号，即AS_CONFED_SEQUENCE最左边AS（最后一个AS）为65535，表示匹配联盟AS65535邻居发送的路由，我们知道，AS_CONFED_SEQUENCE是用”（”、“)”表示的，”（” 、“)在正则中是特殊字符，有特殊用处，所以对于这种特殊字符，可以使用”\”来去除其特殊意义进行匹配，同理AS_CONFED_SET使用的”[“、”]” ，AS_SET使用的”{”、”}”都可以使用”\”符号来去除这些特殊符号的特殊意义，举例\[65533_，\{202_。

\(.*_205_.*\)

表示字符串AS_CONFED_SEQUENCE中间有205，即AS_CONFED_SEQUENCE中有205，表示匹配经过联盟AS205的路由。

_207\)

表示匹配字符串最后为207），即AS_PATH最右边AS_CONFED_SEQUENCE（起始AS）为207，表示匹配联盟AS207始发的路由。

应用场景

实例一

场景：

A国政府近年来和B国战争不断，而且许多B国网页上有A国的反动言论，所以A国政府希望本国人民无法访问B国的网页，A国经过调查发现B国的AS号为70。

部署：

于是A国政府在本国路由器上配置：

```
ip as-path 2 deny 70$     （拒绝从AS70始发的路由）
ip as-path 2 permit .*    （允许其他AS的路由）
```

实例二

场景：

但是A国有些政府官员要和B国保持联系，协商如何促进两国和平等事情。于是A国又收购了一个AS30，要求可以接受AS70始发的路由，但是一定要经过AS30检查过滤。

部署：

于是A国路由器的配置变成：

```
ip as-path 2 permit _30 .+ 70$    （接受从AS70始发的路由但是要经过 AS30）
ip as-path 2 permit _30 70$     （有可能AS30与AS70直接相连）
ip as-path 2 deny 70$     （拒绝从AS70始发的路由）
ip as-path 2 permit .*    （允许其他AS的路由）
```

实例三

场景：

B国恐怖组织后来发现自己在A国的分部无法访问AS70内的网站，于是联盟了多年饱受A国欺负的其他国家，在AS70-140里大肆放置A国反动网站。A国几番周折终于掌握了这些内部消息，但是想到自己的国家政府还要与这些国家政府保持联系(所以还是要经过AS30来过滤的)。

部署：

于是A国路由器的配置变成：

```
ip as-path 2 permit _30 .+ (7[0-9]|8[0-9]|9[0-9]|1[0-3][0-9]|140)$     （接受从AS70-140始发的路由但是要经过AS30）
ip as-path 2 permit _30 (7[0-9]|8[0-9]|9[0-9]|1[0-3][0-9]|140)$     （有可能AS30与AS70-140直接相连）
ip as-path 2 deny (7[0-9]|8[0-9]|9[0-9]|1[0-3][0-9]|140)$     （拒绝从AS70-140始发的路由）
ip as-path 2 permit .*    （允许其他AS的路由）
```

实例四

场景：

A国政府年老的网络管理员退休了，于是换了一位年轻的管理员，上任后发现本地AS内的路

由器保存着许多本地始发的BGP路由，同时路由表里装载的是该路由为IGP路由(只是前缀一样)，想了许久给出了一个解决办法，首先定义一个内部组，把所有IBGP邻居归纳整合到这个组，根据组来进行as-path过滤。

部署：

```
bgp xxxx
group 1 internal
   peer 1 as-path-acl 100 export
```

正则表达式如下：

```
   ip as-path 100 deny ^$      (拒绝发布本地始发路由)
ip as-path 100 permit .*    (允许发布其他AS的路由)
```

实例五

场景：

在以后的日子新任管理员逐渐熟悉了各个路由器的配置后，他发现以前的配置比较繁琐，比如”ip as-path-acl 2 permit _30 .+ (7[0-9]|8[0-9]|9[0-9]|1[0-3][0-9]|140)$”这条命令，看起来复杂而且理解起来又很晦涩，于是新任管理员重新修改了配置。

部署：

修改后的正则表达式如下：

```
ip as-path 2 permit _30 .+ (7.|8.|9.|1[0-3].|140)$  (作用与以前的一样)
```

总结

以上只是对一些常用应用的总结。对于正则表达式“[]”许多厂家实现的不一样，我们和CISCO是一样的在方括号里只能填写数字0到9，比如要是限制范围为10到20那么只能写成（1[0-9]|2[0-9]），如果是40000-65000那么配置任务还是很巨大的啊。有一些个别的厂商实现了该功能的扩展比如要限定范围为4000到5000，那么配置[4000-5000]就可以了。

最后想说的是正则表达式是一个非常灵活的东西，我们可以用不同形式表达相同的目的，当然这样也就有了简单复杂、好与不好的区分。例如实例二中提到的配置，

```
ip as-path 2 permit _30 .+ 70$
ip as-path 2 permit _30 70$
```

我们完全可以简化为一个命令ip as-path 2 permit _30 .+ 70$|_30 70$ 。所以这也是一个仁者见仁，智者见智的表现。

MBGP扩展

文/许亮

概述

BGP协议使用Update报文对路由信息进行更新和撤销。在最新的BGP标准RFC4271中，对Update携带的路由信息格式定义如下：

Total Path Attribute Length (2 octets)
Path Attributes (variable)
Network Layer Reachability Information (variable)

对于要撤销的路由，BGP只向邻居发布该地址的掩码长度和前缀。

Withdrawn Routes Length (2 octets)
Withdrawn Routes (variable)

图2 路由撤销格式

Length (1 octet)
Prefix (variable)

图3 Withdrawn Routes格式

Length (1 octet)
Prefix (variable)

图4 NLRI格式

对于更新的路由，BGP把属性完全相同的合并在一起发布，属性信息放在前面，后面紧跟属性完全相同的一个/组前缀信息。一个Update报文中只能发布一组这样的路由。

随着BGP广泛应用于MPLS VPN、组播以及非IPv4地址族，这种固定的结构不能完全满足应用需求。

为了解决BGP对多种网络层协议的支持，IETF（Internet Engineering Task Force，因特网工程任务组）对BGP-4进行了地址族能力扩展，形成MP-BGP（Multi-Protocol BGP，多协议BGP），使BGP能够为多种应用提供路由信息。在RFC4760（Multiprotocol Extensions for BGP-4）中，定义了2个新的可选非传递属性，BGP的多种协议扩展都用到了这两个属性：

- 扩展协议可达NLRI(MP_REACH_NLRI，属性类型14)
- 扩展协议不可达NLRI(MP_UNREACH_NLRI，属性类型15)

这两种扩展属性适用于所有的BGP协议扩展，为了对不同的扩展类型进行区分，在这两种属性中都携带了BGP地址族（Address Family Information）和子地址族（Sub-Address Family Information）信息。AFI 1分配给IPv4，2分配给IPv6。SAFI分配原则在RFC4760中定义如下：

- SAFI values 1 and 2 are assigned in this document.
- SAFI value 3 is reserved. It was assigned by RFC 2858 for a use that was never fully implemented, so it is deprecated by this document.
- SAFI values 5 through 63 are to be assigned by IANA using either the Standards Action process, defined in [RFC2434], or the Early IANA Allocation process, defined in [RFC4020].
- SAFI values 67 through 127 are to be assigned by IANA, using the "First Come First Served" policy, defined in RFC 2434.
- SAFI values 0 and 255 are reserved.
- SAFI values 128 through 240 are part of the previous "private use" range. At the time of approval of this document, the unused values were provided to IANA by the Routing Area Director. These unused values, namely, 130, 131, 135 through 139, and 141 through 240, are considered reserved in order to avoid conflicts.
-SAFI values 241 through 254 are for "private use", and values in this range are not to be assigned by IANA.

MP_REACH_NLRI (Attribute code: 14)

表1 常用AFI和SAFI列表

BGP扩展	CODE	AFI	SAFI
IPv4 Unicast	Multiprotocol (1)	11	1
IPv4 Multicast	Multiprotocol (1)	1	2
IPv4 Lable	Multiprotocol (1)	1	4
IPv4 VPNV4	Multiprotocol (1)	1	128
Ipv6 Unicast	Multiprotocol (1)	2	1
IPv4 MDT	Multiprotocol (1)	1	66
IPv6 Multicast	Multiprotocol (1)	2	2
L2vpn	Multiprotocol (1)	196	128
VPLS (rfc4761)	Multiprotocol (1)	25	65

扩展协议可达属性用来‘Carry the set of reachable destinations together with the next-hop information to be used for forwarding to these destinations’。一个完整的MP_REACH_NLRI属性结构包含如下内容：

- 地址族信息：包括AFI和SAFI。其中AFI携带和网络地址相关的网络层协议标识；SAFI携带相关属性中网络层可达信息的附加信息；

- 下一跳信息：包括网络地址长度（Length of Next Hop Network Address）和下一跳网络地址（Network Address of Next Hop）。在BGP IPv4定义种，下一跳是作为地址的属性进行传递的，属性类型3。BGP进行地址扩展后，下一跳地址也需要进行扩展，因此放在MP_REACH_NLRI属性中传递。

前缀信息：放在该属性的NLRI字段传递，不同的地址组能力扩展格式不同。

Address Family Identifier (2 octets)
Subsequent Address Family Identifier (1 octet)
Length of Next Hop Network Address (1 octet)
Network Address of Next Hop (variable)
Reserved (1 octet)
Network Layer Reachability Information (variable)

图5 MP_ REACH_NLRI格式

此外，定义了一个字节的保留未用，必须置为全0。

MP_UNREACH_NLRI (Attribute code: 15)

扩展地址不可达属性通告不可达路由，一个含有MP_UNREACH_NLRI的update报文不需要携带MP_UNREACH_NLRI属性以外的任何其他路径属性。其格式如下：

Address Family Identifier (2 octets)
Subsequent Address Family Identifier (1 octet)
Withdrawn Routes (variable)

图6 MP_UNREACH_NLRI格式

扩展地址族划分

从应用场景分，常用的BGP地址族扩展可以分为三大类：

- 与MPLS技术组合，用以分配公网标签、跨公网传播传递必要的协议信息，如：BGP VPNv4扩展、L2VPN扩展、VPLS扩展；
- 为组播应用携带信息，如：IPv4组播扩展和IPv6组播扩展；此外，在私网跨越公网运行组播业务的场景，BGP定义了组播VPN MDT扩展，简称组播VPN扩展。
- 对IPv4之外的地址族的支持，如IPv6扩展、6PE扩展；

MPLS相关扩展

MPLS（Multiprotocol Label Switching，多协议标签交换）起源于IPv4（Internet Protocol version 4，因特网协议版本4），在链路报文头和IP报文头之间插入一个标签头，标签处于2.5层。由于MPLS结合了IP网络强大的三层路由功能和传统二层网络高效的转发机制，在转发平面采用面向连接方式，与现有二层网络转发方式非常相似，这些特点使得MPLS能够很容易地实现IP与ATM、帧中继等二层网络的无缝融合，并为QoS（Quality of Service，服务质量）、TE、VPN等应用提供更好的解决方案。

基于MPLS的VPN

传统的VPN一般是通过GRE、L2TP、PPTP等隧道协议来实现私有网络间数据流在公网上的传送，而LSP本身就是公网上的隧道，用MPLS来实现VPN有天然的优势。

基于MPLS的VPN就是通过LSP将私有网络的不同分支连接起来，形成一个统一的网络。基于MPLS的VPN支持对不同VPN间的互通控制。

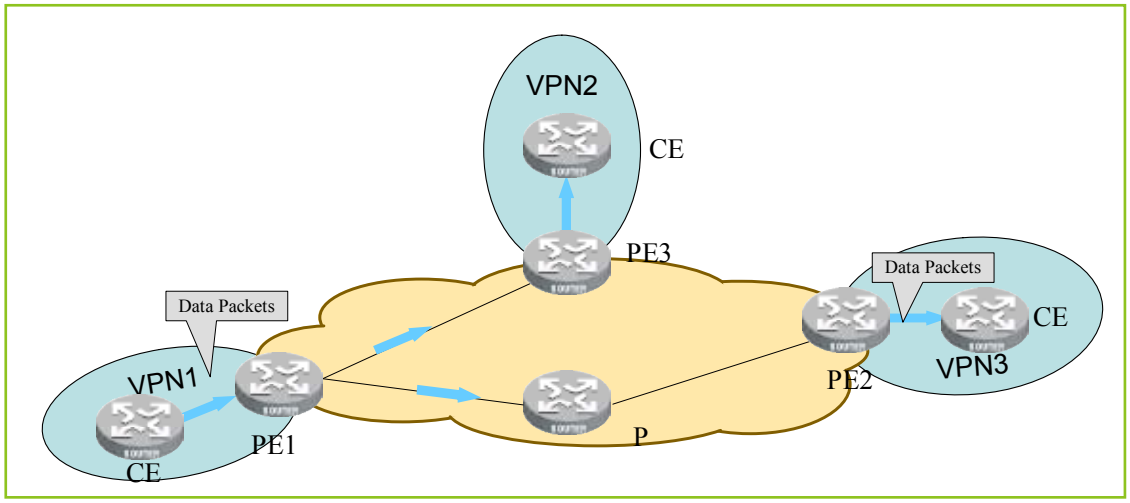


图7 MPLS VPN典型组网

在MPLS/VPN架构中设备有如下角色：

- CE（Customer Edge）设备：用户网络边缘设备，有接口直接与SP（Service Provider，服务提供商）相连。CE可以是路由器或交换机，也可以是一台主机。CE“感知”不到VPN的存在，也不需要支持MPLS。
- PE（Provider Edge）路由器：服务提供商边缘路由器，是服务提供商网络的边缘设备，与用户的CE直接相连。在MPLS网络中，对VPN的所有处理都发生在PE上。
- P（Provider）路由器：服务提供商网络中的骨干路由器，不与CE直接相连。P设备只需要具备基本MPLS转发能力。

PE为CE提供二层接入服务的称为L2VPN，提供三层接入服务的称为L3VPN。对MPLS的详细讨论参见网络之路第三期MPLS专题。

MPLS L3VPN

在MPLS/L3VPN组网中，PE和CE之间运行路由协议，PE与CE连接的接口运行路由协议多实例。CE把本站点的VPN路由发布给PE，并从PE学到远端VPN的路由。CE与PE之间使用BGP/IGP交换路由信息，也可以使用静态路由。在CE侧，完全感知不到公网的存在，认为PE是私网内的一台路由器。

在PE上，多个CE接入。PE把不同接入方划分到独立的VPN中。对私网用户而言，他们使用的是自己的私有地址空间，不同的VPN中可能存在相同的IPv4地址。而BGP假设其承载的每个IPv4地址都是全局唯一的，因此必须通过对地址族进行扩展将非唯一的IPv4地址转换位全局唯一的

地址。转换方法是将VPN的唯一标识RD（Route Distinguisher）附加在IPv4路由前缀之上：RD用来表示不同VPN；相同的机构在不同地区的接入站点应设置为相同；不同机构必须不同。此外对同一机构不同站点之间的路由传递使用RT（Route Target）进行控制。RT分为import和Export。Export携带在BGP路由中进行传递，PE收到后只保留与本地Import RT匹配的路由。

MPLS L2VPN

MPLS L2VPN就是在MPLS网络上透明传递用户的二层数据。从用户的角度来看，这个MPLS网络就是一个二层的交换网络，通过这个网络，可以在不同站点之间建立二层的连接。

MPLS L2VPN包括VLL和VPLS两种：

- VPWS（Virtual Provider Wire Service）：虚拟私有线路服务，在公用网络中提供的一种点到点的L2VPN业务。客户的二层设备跨过MPLS/IP核心网络相连，就像通过一根二层线路直连一样。它不能直接在服务提供者处进行多点间的交换。
- VPLS（Virtual Private LAN Service）：虚拟专用局域网服务，在公用网络中提供的一种点到多点的L2VPN业务。VPLS使地域上隔离的用户站点能通过MAN/WAN相连，并且使各个站点间的连接效果像在一个LAN中一样。
- VSI（Virtual Switch Instance）：虚拟交换实例，通过VSI，可以将VPLS的实际接入链路映射到各条虚链接上。
- PW（Pseudo Wire）：虚链路，在两个VSI之间的一条双向的虚拟连接，它由一对单向的MPLS VC（Virtual Circuit，虚电路）构成。
- Tunnel：隧道，用于承载PW，一条隧道上可以承载多条PW，一般情况下为MPLS隧道。隧道是一条本地PE与对端PE之间的直连通道，完成PE之间的数据透明传输。
- Encapsulation：封装，PW上传输的报文使用标准的PW封装格式和技术。PW上的VPLS报文封装有两种模式：Raw和Tagged模式。
- PW Signaling：PW信令协议，VPLS实现的基础，用于创建和维护PW。PW信令协议还可用于自动发现VSI的对端PE设备。目前，PW信令协议主要有LDP和BGP。

MBGP扩展：for VPNv4

因此，在BGP VPNv4扩展中，MP_REACH_NLRI扩展如下：PE从CE学到CE本地的VPN路由信息后，在路由信息中增加RD和RT，再通过BGP VPNv4邻居关系与其他PE交换VPN路由信息。不同VPN的路由在公网中通过RD进行区分，不同CE之间的路由使用RT进行引入控制。PE路由器只维护与它直接相连的VPN的路由信息（收到BGP VPNv4发来的路由后，只保留与本地import RT匹配的路由信息），不维护服务提供商网络中的所有VPN路由。

它的格式如下所示：

<table>
<tr><td rowspan="2">Route Distinguisher(8 octets)</td><td colspan="2">Type Field(2 octets)</td></tr>
<tr><td rowspan="2">Value Field(6 octets)</td><td>Administrator Subfield</td></tr>
<tr><td>Assigned Number Subfield</td></tr>
<tr><td colspan="3">IPv4 Address Prefix (4 octets IPv4 Address)</td></tr>
</table>

图8 RD格式示意图

VPNv4地址前作为地址的一部分存在。

在BGP路由中引入RD和RT只解决了路由传递的问题：在PE本地的路由冲突和路由网络传播过程中的冲突。但RD和RT不参与报文转发，在数据转发时如果接收端PE的两个本地VRF中同时存在10.0.0.0/24的路由，当它接收到一个目的地址为10.0.0.1的报文时，它如何知道该把这个报文发给与哪个VRF相连的CE？这就需要在PE上由BGP给私网路由分配标签来解决。改造后的MP-IBGP进行NLRI信息交换时会附加RD、标签等各种信息。格式如下：

Address Family Identifier (2 octets)	
Subsequent Address Family Identifier (1 octet)	
Length of Next Hop Network Address (1 octet)	
Network Address of Next Hop (PE路由器自己建立Peer使用的地址)	
Reserved (1 octet)	
Network Layer Reachability Information (variable)	label(3 octet，与MPLS标签一样，但没有TTL)
	RD(8 octet)+IP前缀

图9 MP-IBGP NLRI报文格式

在这之后是RT信息，如下图所示：

Extended community(RT1)
Extended community(RT2)
Extended community(RT3)

图10 MP-IBGP NLRI报文RT列表

MBGP扩展：for L2VPN

MPLS/L2VPN有多种实现方式，其中Kompella方式使用BGP作为交换信令。与MPLS L3VPN类似，各个PE之间通过建立BGP会话自动发现L2 VPN的各个节点，并传递VPN信息，使用VPN-target来区分不同的VPN。在不同VPN内，CE ID可以相同。BGP为不同的CE间建立的VC（Virtual Circuit）分配内层标签，扩展BGP报文格式如下：

新定义了一个属性用于描述二层报文封装的必要信息：

Length(2 octets)
Route Distinguisher(8 octets)
CE ID(2 octets)
Label-block Offset(2 octets)
Label-Base(3 octets)
Variable TLVs(0 to N octets)

图11 L2VPN扩展NLRI格式

MBGP扩展：For VPLS

Extended community type(2 octets)
Encaps Type(1 octets)
Cntrl Flags(1 octets)
Layer-2 MTU(2 octets)
Reserved(2 octets)

图12 L2VPN 扩展属性：layer2-info extended community

简介

VPLS也称TLS（Transparent LAN Service，透明局域网服务）或Virtual Private Switched Network Service（虚拟专有交换网络服务），是在公用网络中提供的一种点到多点的L2VPN业务。VPLS使地域上隔离的用户站点能通过MAN（Metropolitan Area Network，城域网）或WAN（Wide Area Network，广域网）相连，并且使各个站点间的连接效果像在一个LAN中一样。

VPLS提供二层VPN服务。在VPLS中，用户是由多点网络连接起来，不同于传统VPN提供的P2P（Point to Point，点到点）的连接服务。

典型组网

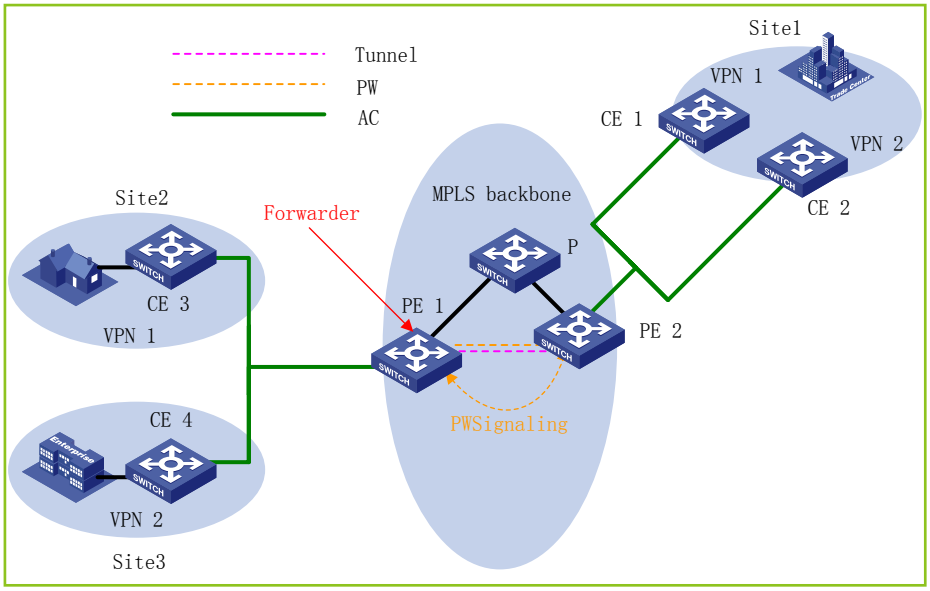


图13 VPLS典型组网

BGP扩展的作用

PW隧道的建立常用有两种信令 LDP（rfc4762）和MP-BGP（rfc4761）。

采用BGP作信令时，利用BGP的多协议扩展VPLS 成员信息。其中MP-reach和MP-unreach属性传递vpls的标签信息，RD、VPN-TARGET和接口参数信息在扩展团体属性中传递。

下图是一个采用BGP方式作信令的PW建立与拆除的典型过程 当PE1配置了一个VSI(Virtual Switch Instance) 建立了到PE2的BGP session，并且在该session上使能VPLS地址族， BGP session建立后会分配标签并给PE2发送带MP-REACH属性的update消息。PE2收到update消息后检查：本地是否也配置了同样的VSI、VPN-TARGET匹配（与L3VPN的匹配含义相同）、接口参数一致；则PE2端的PW就建立起来了。PE1收到PE2的update消息后作同样的检查和处理。

当PW1不想再转发PE2的报文，则发送带MP-UNREACH属性的update消息给PE2，同时拆除PW、释放标签；PE2收到update消息后拆除PW。

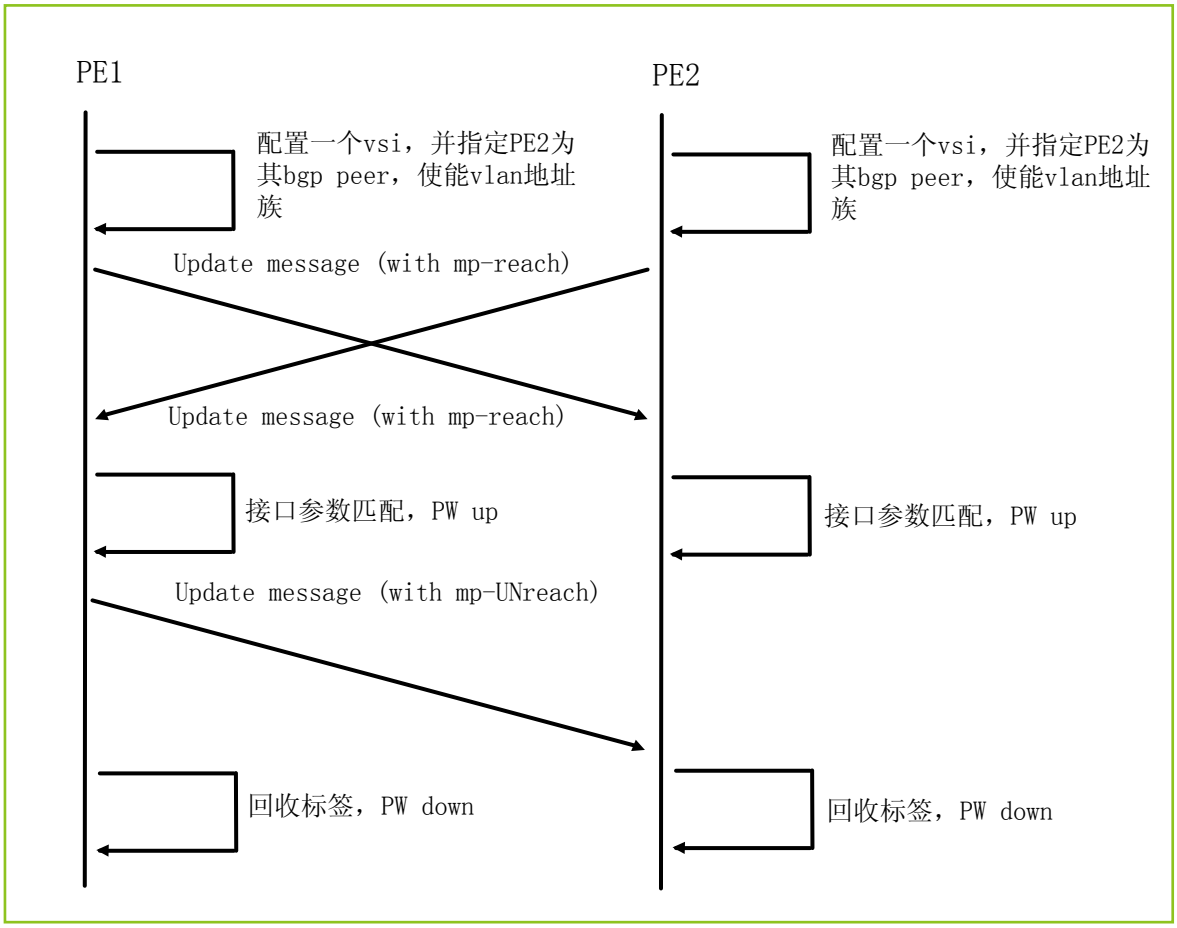


图14 用BGP作信令时PW的建立/拆除过程

组播相关扩展

MBGP扩展：for Multicast

组播BGP（Multicast BGP或BGP for IPv4 Multicast）扩展用于携带组播源信息。当使能组播扩展的BGP Router收到邻居发来的组播地址族NLRI后，经过优选加入RPF（Reverse Path Forwarding）路由表，不加入单播路由表。组播BGP扩展为跨域的组播应用提供了路由信息，还

可以实现单播和组播的拓扑分离。

IPv4 Multicast扩展使用地址族：AFI为1，SAFI为2。

应用场景

网络中存在多个自治系统，各域内部采用IGP进行互联。组播源属于域B内，接收者则分布在域D，E，G，F；要求在屏蔽其他域网络拓扑的情况下接收者可以视频点播。如下图所示：

MBGP扩展：for IPv6 Multicast

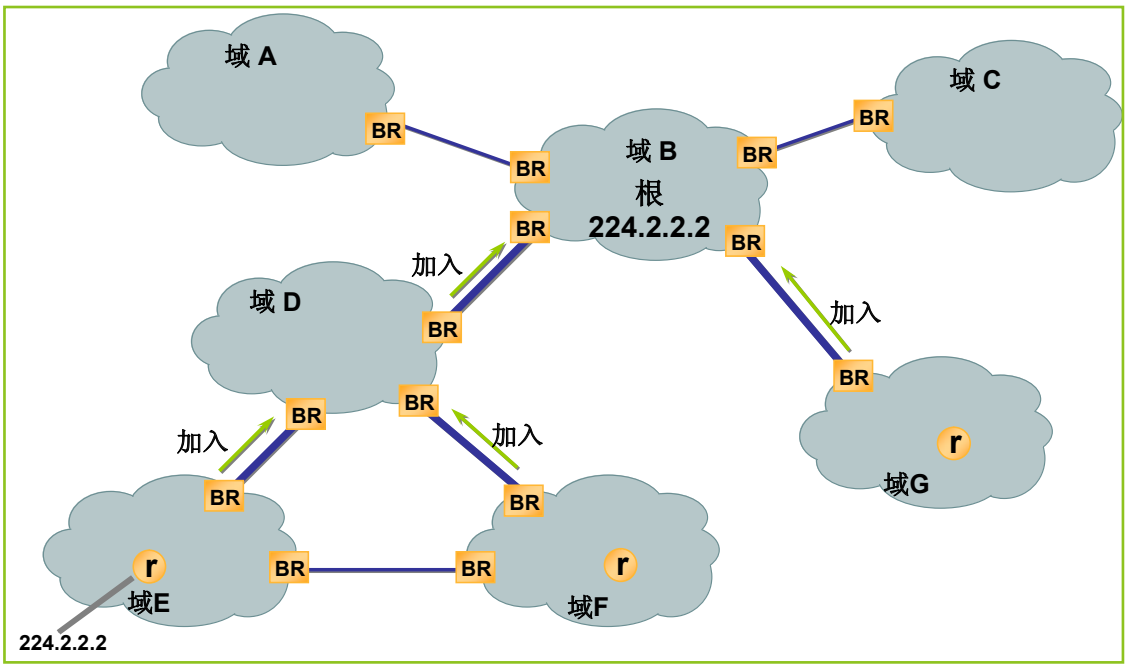


图15 Multicast BGP应用举例

IPv6组播BGP（IPv6 Multicast BGP或BGP for IPv6 Multicast）扩展用于携带IPv6组播源信息。原理和应用场景与MBGP for IPv4 Multicast扩展类似，只是地址族不同：当使能组播扩展的BGP Router收到邻居发来的组播地址族NLRI后，经过优选加入IPv6 RPF（Reverse Path Forwarding）路由表，不加入单播路由表。其目的是实现单播和组播的拓扑分离。

IPv4 Multicast扩展使用地址族：AFI为2，SAFI为2。

MBGP扩展：for 组播VPN

组播VPN

MPLS/BGP VPN结构中只支持单播应用，VPN客户的组播服务需求无法满足。但公网可以运行组播协议，私网内部也可以运行组播，只要解决两个问题就可以实现私网用户跨公网的组播应用：

- 私网组播报文在公网的转发
- 私网源组信息到公网源组信息的映射

1. 私网组播报文跨公网的转发

解决办法是在PE上采用隧道对私网组播报文进行封装，可以用GRE、IPinIP或MPLS隧道。目前业界普遍采用GRE隧道进行封装。

2. 私网源组映射到公网源组

解决办法是在PE上为每一个私网分配一个组地址，用于公网转发。这样每个VPN内的组播报文到达PE后经过隧道封装，最外层的IP报文还是组播报文的形式：报文组地址为该VPN分配的Share-Group地址。显然，Share-Group地址必须全局唯一。PE收到组地址在Share-Group范围的组播报文时，根据本地的私网/公网映射关系将报文外层IP头和隧道头剥去，根据Share-Group对应的VPN将报文送到制定私网中进行处理，从而实现跨公网的组播业务，即组播VPN MD协议。

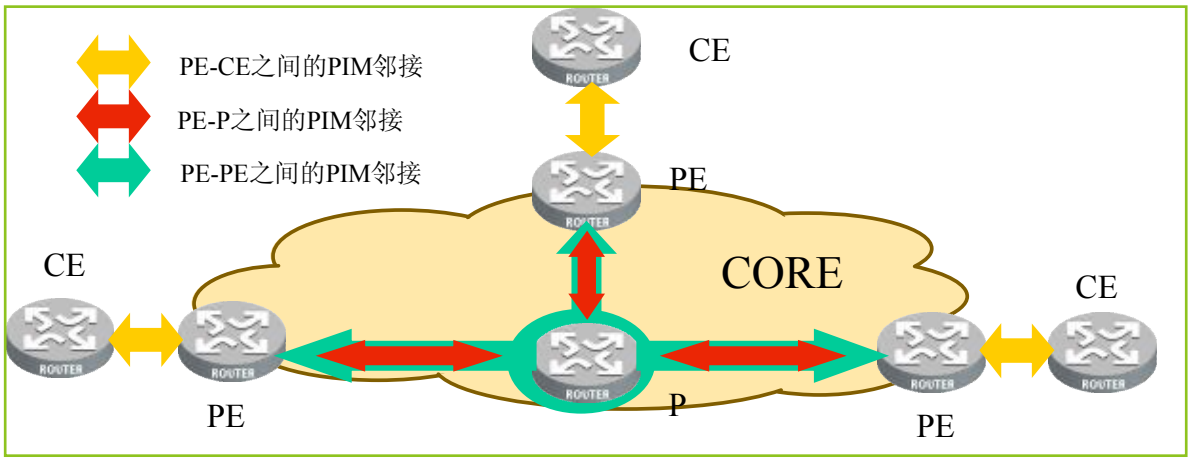


图16 组播VPN的邻居关系

应用模型如上图所示：公网运行普通的组播协议，私网运行普通的组播协议，在PE上完成地址映射和报文封装/解封装。

MD协议在骨干网上为每个VPN维护一棵组播转发树，称为Share-MDT，这棵树在MD配置完成后就自动建立了，不管VPN中有没有组播业务，也不管骨干网上有没有组播业务。来自于VPN的任何一个站点的组播报文（包括协议报文和数据报文）都会被封装成为公网组播数据报文，沿着这个棵Share-MDT树转发到所有属于该MD的PE。如果该PE连接有该组播组的接收者，则往CE转发，否则丢弃组播报文。

BGP MDT扩展

实际应用中还可能碰到这样的问题：Share-Group是PE上配置，其他PE并不知道其他PE配置的Share-Group是多少。而要在公网部署PIM SSM必须知道源地址。BGP MD扩展解决了这个问题：将PE上Share-Group的源组信息通过PE之间的BGP会话进行传递。为此，BGP定义了一类新的地址族：IPv4 MDT地址族。AFI为1，SAFI为66。

MDT SAFI NLRI具体格式如下：

Route Type (1 octet)
Length (1 octet)
Route Type specific (variable)

图17 MDT SAFI NLRI

其中，Route Type定义如下：

1 - Intra-AS I-PMSI auto-discovery route (or just auto-discovery route);
2 - Inter-AS I-PMSI auto-discovery route (or just inter-AS auto-discovery route);
3 - S-PMSI auto-discovery route;
4 - Intra-AS segment leaf auto-discovery route (or just leaf auto-discovery route).
5 - Source Active auto-discovery route.
6 - Shared Tree Join route;
7 - Source Tree Join route;

RD:IPv4-address (12 octets)
MDT Group-address (4 octets)

RD：MD配置所在VRF的RD，长度为8字节。

IPv4-address：MTI的源IP地址，为IPv4地址，长度为4字节。

MDT Group-address：MVRF绑定的组播组地址，长度为4字节。

图18 Route Type Specific（携带Share-Group信息）

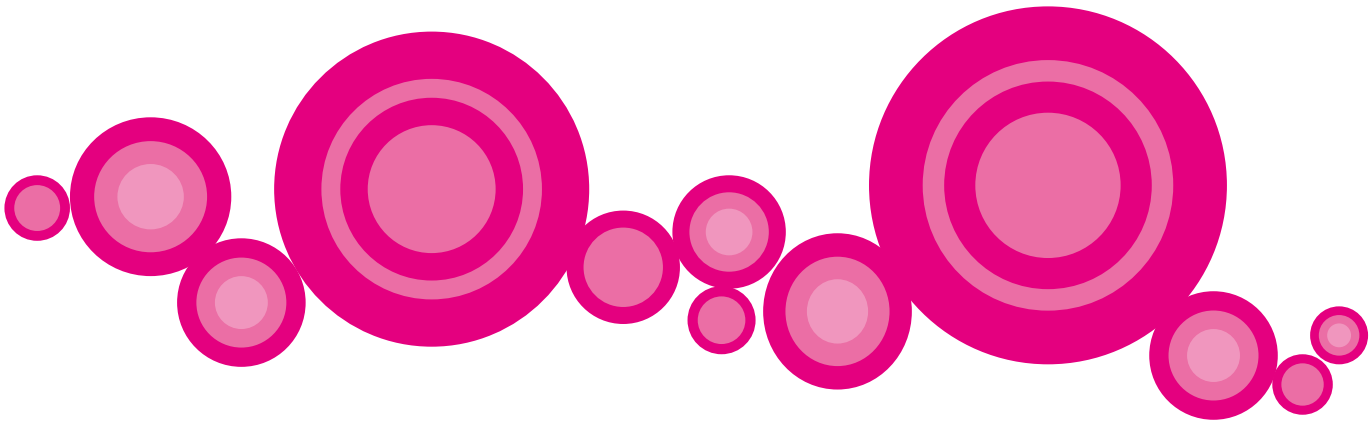
地址族扩展

随着IPv6的发展，几乎所有路由协议都开发除了适配IPv6的版本。BGP for IPv6扩展有两个：

- IPv6地址族扩展，解决纯IPv6网络中传递IPv6路由的问题；
- 6PE扩展：解决IPv6孤岛跨越公网的问题。

IPv6扩展在网络之路第六期IPv6专辑有详尽的介绍，请参见IPv6路由技术章节和IPv6 over MPLS章节。

[网络应用]



BGP网络性能优化浅析

文/杨默寒

前言

在构建实际运营的BGP网络中，由于需要考虑到性能参数对网络效率的影响，往往需要对BGP的一些参数进行仔细的斟酌，本文主要以对一些与网络性能相关的参数与特性的设置进行讲解，供读者在后续设计及优化BGP网络时参考。

BGP邻居PMTU检测

BGP协议是运行在TCP之上的，所以TCP的参数设置会影响BGP的性能。在路由数目比较少的情况下TCP的参数调整可能对BGP性能影响不大，但是当路由数目比较巨大的时候调整TCP参数可以起到优化性能的作用。下面我们开始分析具体的优化方法。

首先我们来了解一下BGP协议包发送的方法，请看下图1：

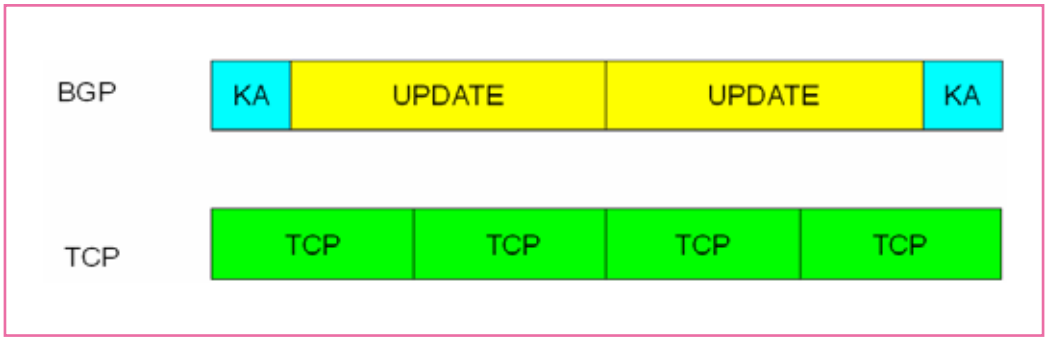


图1 BGP与TCP数据传输格式

BGP首先需要把自己需要发送的数据告诉TCP，然后TCP根据数据的长度进行分段，分段大小由TCP协商的MSS值的大小决定，每个TCP分段对应着一个发出去的IP包。所以MSS参数的设置对于BGP数据传输的性能起着关键作用，如果设置过大可能会造成中间某台设备的IP层分片，BGP

协议报文的传输其实是一个端到端的传输过程，如果数据被分片了那么必然还需要重新组合恢复回来，这样会给接收者的CPU带来一定的负担，组包的过程降低了处理效率；如果MSS值设置过小，那么又会使网络的有效利用率很低，发送端和接收端对能够一次处理的报文进行多次处理，降低了效率。

通过BGP邻居PMTU检测可以解决前面提到的问题，在建立BGP邻居之前，路由器会自动发送一个PMTU报文来检测路径上的最大MTU，当得到这个值后TCP协议可以根据这个值来设定MSS的大小，发送报文时按照PMTU探测的结果，进而达到性能的最优。

BGP路由更新定时器

在BGP的RFC4271上定义了BGP的路由更新的定时器，该定时器只能对同一地址族的相同前缀的路由起作用，其主要作用是防止网络中的某条路由震荡过于频繁，同时也是对CPU的一种保护。用文字对该特性进行描述可能过于晦涩，所以我们用图2来做简单的介绍：

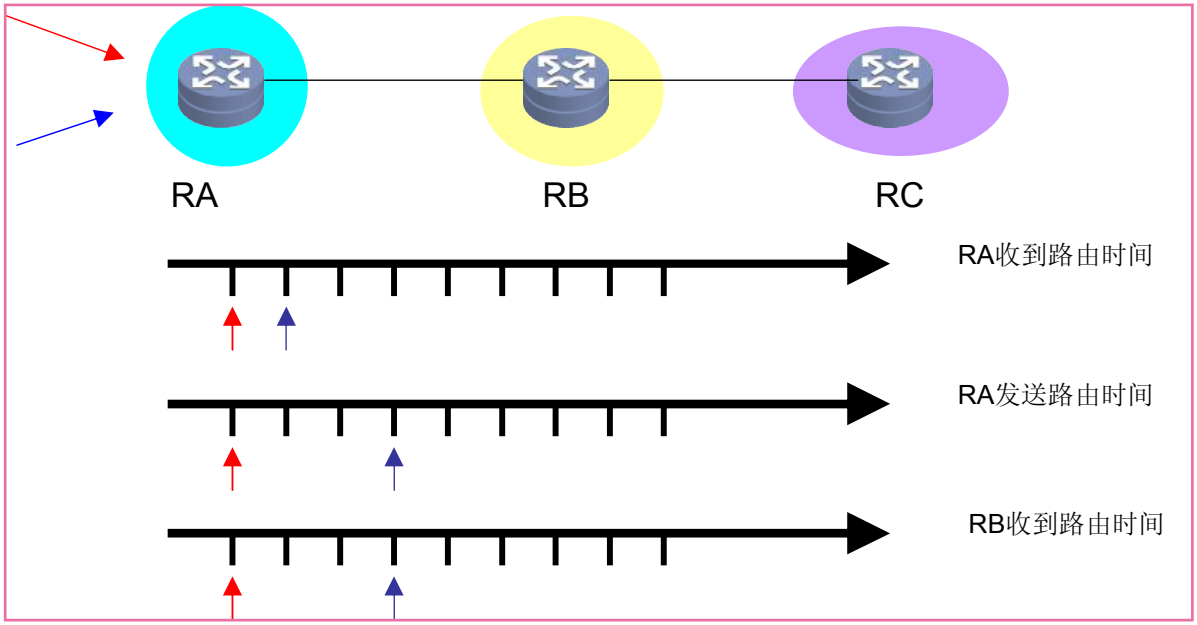


图2 路由更新定时器说明

如图2红色与蓝色的箭头代表相同前缀的路由，但是从不同邻居学习而来，而且蓝色的路由优于红色的；黑色的坐标轴代表路由发送与接收的时间，每个单位距离为10秒。我们假设RA上配置的更新定时器时间为30秒。那么RB路由收敛过程如下：

- RA接收到红色路由后立刻发送给RB，同时RA上启动更新定时器（30秒）；

- 10秒以后RA接收到更优的蓝色路由，由于定时器没有超时暂时不发送给RB，但是更新本地路由表，在第10秒RA完成路由收敛；
- 第30秒RA上更新定时器超时，所以发送蓝色路由给RB并且更新掉红色路由，RB在第30秒完成收敛。

从上面的分析我们可以看出RB的收敛时间比RA会慢上20秒左右，由于BGP是距离矢量路由协议这种延迟可以可能对于整个网络的BGP路由器都会有一定影响，所以在设计BGP网络中对该参数的设计需要有一定考虑，如果对自己设备的路由处理能力有足够信心的话可以把该定时器的值设置为最小。

这里需要说明一下，路由惩罚（Dampening）也有类似的情况，如果希望网络发生路由震荡后可以尽快的收敛，那么完全可以不设置Dampening参数。

与BFD协议联动

上一节所介绍的功能只能使路由的收敛时间限制在秒级，但是对于一个运营商（SP）的网络来说，往往需要更快地感知路由的变化或者BGP邻居的状态变化。但是IBGP邻居状态的感知往往由于邻居非直连的原因，需要依靠IGP的收敛或者BGP自身的KEEPALIVE报文来感知邻居的状态，这样最多可能会需要180秒来完成收敛。

Bidirectional Forwarding Detection（BFD）是一个简单的“Hello”协议，和路由协议的邻居检测部分相似。一对系统在它们之间所建立会话的通道上周期性的发送检测报文，如果某个系统在足够长的时间内未收到对端的检测报文，则认为在这条到相邻系统的双向通道的某个部分发生了故障。BFD目前存在两个版本：VER 0和VER 1，并且两个版本不能互相兼容。

BGP的BFD就是利用了这个特性，配置BGP与BFD关联后，一旦BGP邻居建立后，BFD自动和BGP邻居关系进行关联，并在每单位时间发送探测数据，这个单位时间一般为几十毫秒，当超过3倍的时间没有收到探测报文BFD会通知BGP断开邻居关系，这样可以迅速的完成路由收敛。

BGP流量负载分担规划

文/张宇弟

BGP流量负载分担概述

如何优化的利用网络带宽资源，是流量负载分担的关注重点。BGP（Border Gateway Protocol，边界网关协议）选择单条最优路径的这一特征往往会出现流量负载不均衡的流量模型， BGP流量负载均衡从两个角度出发解决这个问题：通过BGP强大的策略控制流量的负载均衡；通过多路径选路实现负载分担。本文就要从这两个角度来展开分析BGP在流量负载分担方面的技术应用。

负载均衡

在实际网络中进行负载均衡需要综合考虑链路和设备节点的负载情况，在满足业务的实际需求前提下，可以通过BGP的策略工具对流量进行均衡的规划和调整。对于一个AS来说，流量的方向分为入境和出境两个方向，这种区分对应到实际的网络有不同的规划，所以我们在此通过不同的场景进行介绍。

入方向流量负载均衡

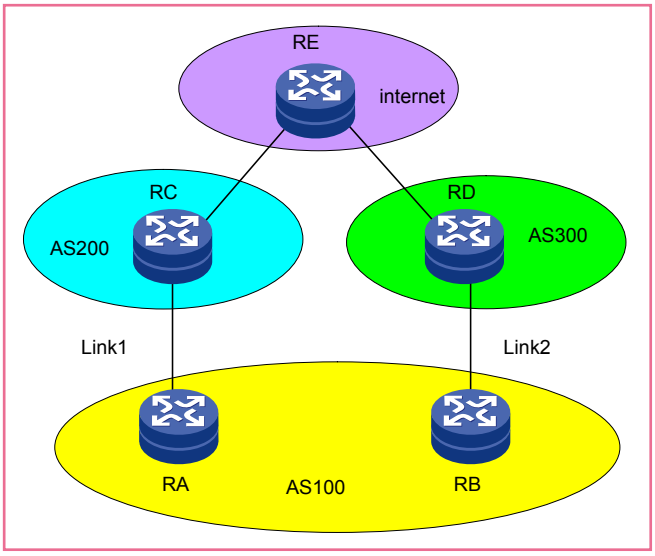


图1 多宿主到不同的上游的负载均衡

我们先分析一下图1的场景，AS100希望流量能够在AS200和AS300间进行负载均衡，也就是说根据业务分别映射到Link1和Link2上。

在规划中我们有如下思路：

1. AS100可以在RA和RB上通过策略只向各自的对等体通告部分路由前缀，这样可以起到不同的业务对应由不同的AS承载。如172.168.1.0/25通过RA通告给RC，172.168.1.128/25通过RB通告给RD。这种规划能够满足流量分担的效果，但是一旦出现链路或节点的失效，就会导致部分流量无法切换，业务中断。

2. 通过步骤1我们可以看到简单的通过路由过滤无法很好的实现需求。我们可以通过对不同的前缀进行策略区分。接着步骤1的思路，AS100希望172.168.1.0/25优先通过AS200进入，希望172.168.128.0/25优选通过AS300进入。可以在RA上通过策略将172.168.128.0/128通告的AS-PATH加一个AS-Number，如：1000 100。RB上通过策略将172.168.1.0/25通告给RD的AS-PATH加一个AS-Number，如：2000 100。

RE上关于172.168.1.0/25的前缀从RC通告过来的AS-PATH是： 200 100，通过RD通告过来的AS-PATH是：300 2000 100，因此优选走AS200。关于172.168.128.0/25同理会优选AS300。该规划可以在满足需求的同时解决路由备份的问题。但是我们考虑下面一个场景，当RC和RD建立BGP连接，RD上关于172.168.1.0/25的前缀从RC通告过来的AS-PATH是100 200，从RB上通告过来

的AS-PATH是2000 100，也就是说RD上关于172.168.1.0/25无法很好的进行路由选路控制。

3.团体属性是进行本地进行路由控制的重要属性，但是团体属性需要BGP对等双方有属性处理的共识。在步骤2的基础上，RB在通告172.168.1.0/25时，可以将团体属性值修改为100：120，在RD上将团体属性的前缀预定义Local-preference为120，这样对于RD来说在收到RC和RB的前缀AS-PATH相同长度的情况下，RB通告的前缀Local-preference高，优选RB。

分析完图1所示的场景，我们可以继续看看多宿主相同上游的场景，如图2：

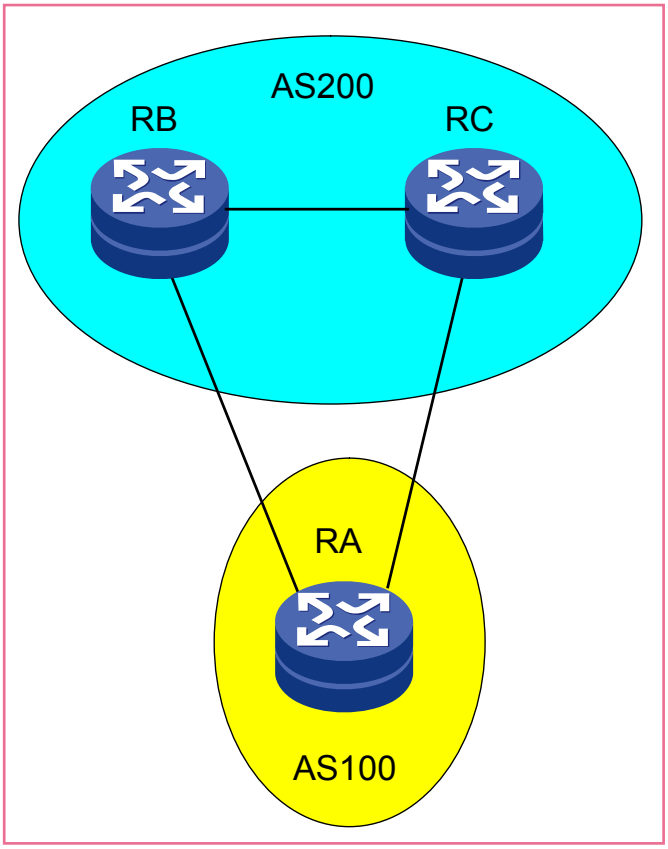


图2 多宿主相同AS的负载均衡

对于相同的上游AS，我们更多的需要考虑上游AS内部的选路，对于跨过上游AS的远端AS的选路我们很难控制，因此本文讨论的重点就在本地和上游AS的选路策略部署。先分析需求，本地AS希望172.168.1.0/25通过RB进入，172.168.128.0/25通过RC进入。存在下述三种方式：

1.借鉴上文中多宿主不同AS的做法，将172.168.128.0/25通告给RB时，将AS-PATH设置为1000 100，将172.168.1.0/25通告给RC时，将AS-PATH设置为2000 100。由于RB和RC之间是IBGP邻居，因此对于RB来说，172.168.1.0/25从RA学习到的AS-PATH是100，从RC上学习到的AS-PATH是2000 100，优选RA的路由。对于172.168.128.0/24来说，从RA学习的路由AS-PATH为1000 100，从RC学习的路由AS-PATH为100，优选RC。这满足了需求。

2.同样，在AS间协商进行策略部署的前提下，可以在RA上对不同的业务前缀进行属性赋值的区分，在RB/RC对不同的属性设置不同的本地优先级进行业务的分担均衡。

3.还是针对图2，对于AS200进行入方向的流量负载分担部署，最直接的方法就是通过修改MED值的方法，使得RA通过MED值直接进行路由的优选。

出方向负载均衡

我们还是先分析图1的场景，AS100希望出方向的业务流量能够在RA和RB间进行合理的负载均衡：

1. 可以通过在RA和RB上进行入境路由前缀过滤，通过前缀在不同的出口路由器的通告分担来实现业务流量出方向的负载均衡。

2. 入境的过滤适用于对端AS的业务负载分担，但是对于远端Internet的业务，无法通过入境路由前缀过滤的方法实现，否则会出现单点故障导致业务中断的情况。对于Internet业务就需要进行入境路由前缀策略控制，例如通过添加AS-PATH或者对特定前缀设置不同的本地优先级等。

3. 对于单点故障导致业务中断的考虑还可以通过出口路由器发布缺省路由的方式作为路径的备份，一旦出现某个出口路由器故障，路径可以通过缺省路由切换至其他出口路由器，起到备份的效果。

4. 同样可以和上游AS协商，通过通告来的前缀携带不同的团体属性进行相应的策略控制。

对于图2的场景，单出口路由器上进行负载分担可以借助路由策略的方法更为灵活：

1. 针对不同对等体进行入境路由前缀过滤，使业务自然分担到不同的出口链路上。同时配置缺省路由指向对等体，防止单点故障业务中断；

2. 通过针对特定前缀设置多种属性，如Local-preference、Origin、MED等，在本地进行路由优选。

AS内部负载均衡

AS内部的负载均衡相对容易部署，通过策略对业务进行区分，对不同的业务使用不同的BGP属性进行控制。如图3，RB和RC同时向RA通告172.168.1.0/25和172.168.128.0/25，RA希望172.168.1.0/25业务从RB走，172.168.128.0/128从RC走。满足这种需求，可以直接在RA上对业务进行区分，对RB通告的172.168.1.0/25的Local preference设置为120，将RC通告的172.168.128.0/25的Local preference设置为120，本地优选结果能够满足需求。

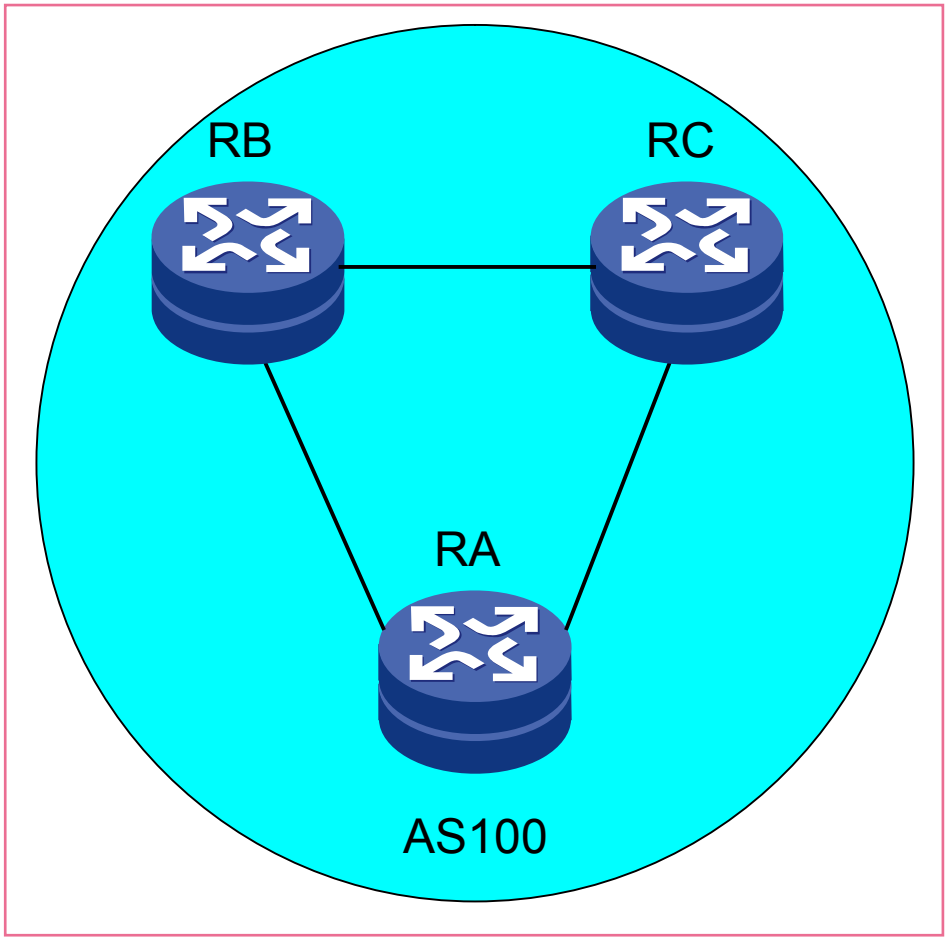


图3 AS内部负载均衡

等价负载分担

上文介绍的是多宿主情况下的负载均衡规划思路，本章节需要关注的是通过等价路由在路由器之间进行负载分担的部署方式。

EBGP多跳负载分担

图4的场景是两台出口路由器之间通过环回口建立EBGP邻接，我们知道对于EBGP邻接超过1跳建立邻接需要通过命令peer x.x.x.x ebgp-max-hop <Maximum hop>，其中Maximum hop设置大于1。对每一个链路接口对应配置静态路由，指向对端环回口地址，路由下一跳为链路对端接口地址。这种方法通过路由下一跳地址的迭代，将流量负载分担到不同的链路上，实现多跳的EBGP对等体间的多链路负载分担。

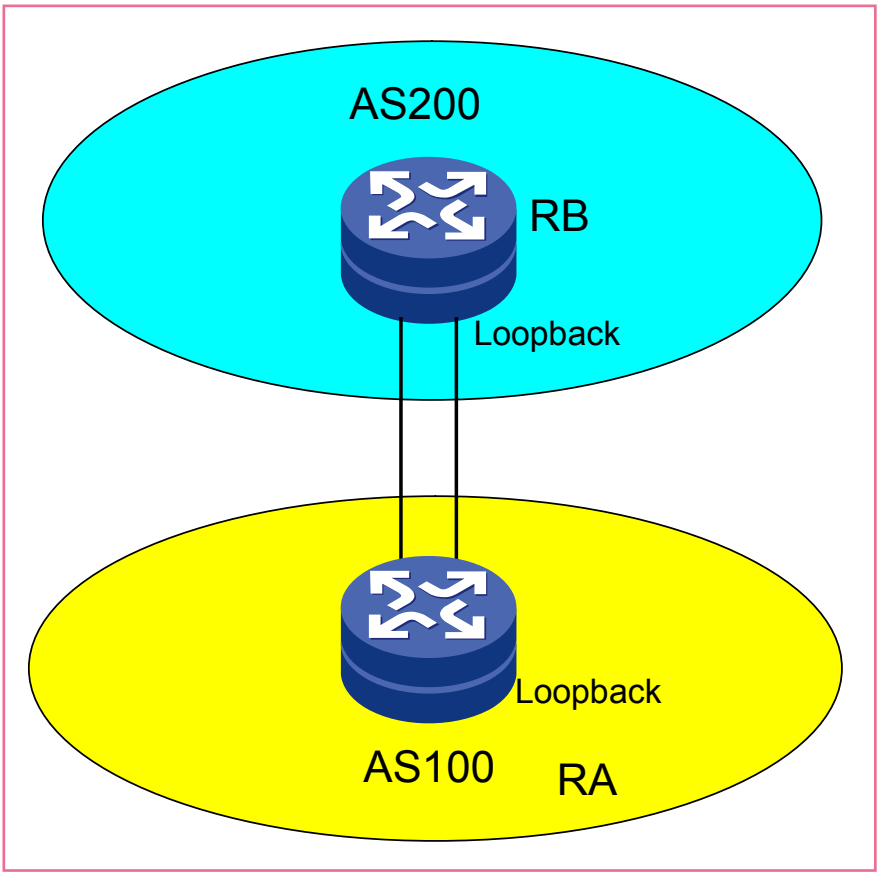


图4 多链路多跳EBGP对等体负载分担

EBGP多路径负载分担

图5的场景和图4类似，区别在于两个出口路由器通过链路接口地址建立了多个EBGP会话，每一个会话对应于特定的链路。

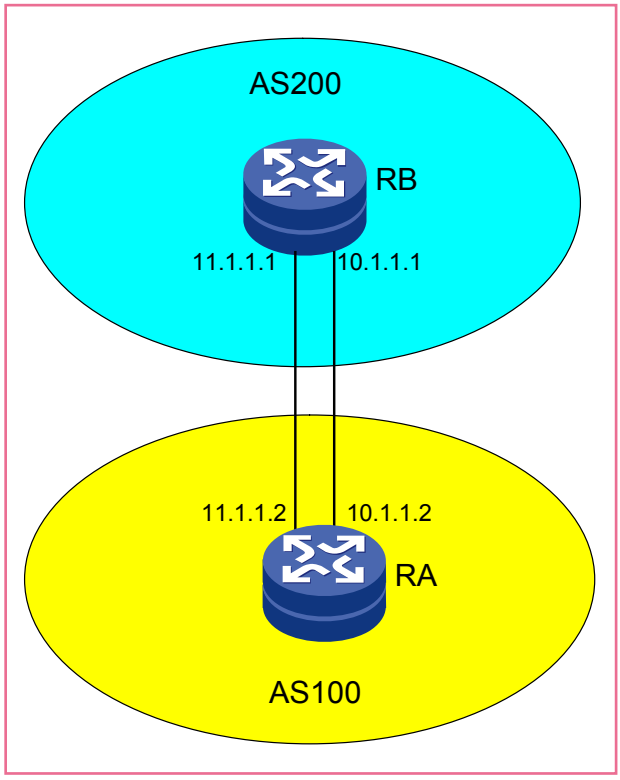


图5 多链路多EBGP对等体负载分担

RA和RB通过两条链路的物理接口地址建立两个单跳EBGP邻接关系，以RA举例，从两个EBGP会话均收到关于172.16.1.0/24的路由信息，在其他属性都相同的前提下，两个EBGP对等体通告的路由形成等价。

比较上述两种负载分担方式，第一种方式，建立了一个EBGP会话，会话直接绑定在两个出口路由器的环回地址，通过直连路由迭代下一跳的方式在链路间实现负载分担。这种方式EBGP会话属于多跳会话，需要两个AS间进行环回地址的路由部署，适用性有一定的限制。第二种方式，需要建立多个会话，对资源有一定的消耗，同时对于EBGP等价路由的配置，是针对本设备所有BGP对等体适用，无法区分对等体，因此，缺少一定的灵活性。

AS内部负载分担

AS内部的负载分担规划同样可以采用下一跳迭代的方式在同一个IBGP会话间进行多链路的负载分担。对于多IBGP对等体通告的路由在优选属性相同的前提下能够形成负载分担。对于图6中的场景，RB和RC向RA通告172.168.1.0/24的路由前缀，在ORIGIN，LOCAL-PREFERENCE以及AS-PATH路径属性均相同的前提下能够形成负载分担。

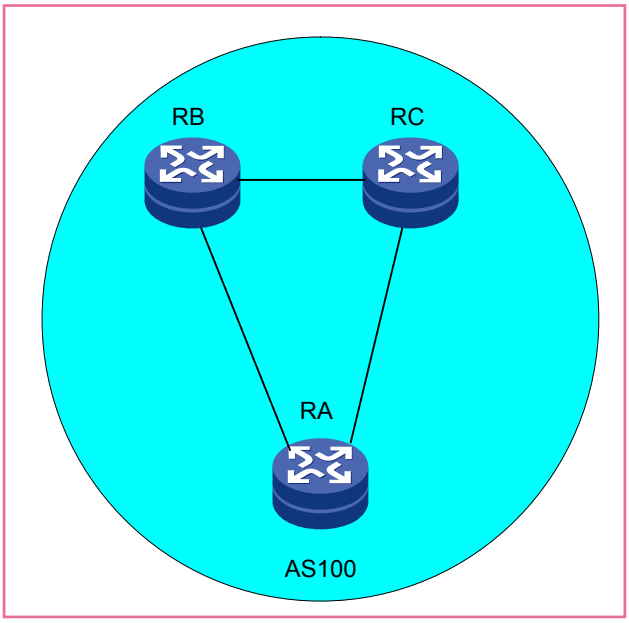


图6 AS内部负载分担

多链路的负载分担还是可以通过下一跳迭代到等价IGP路由或者默认路由来实现，例如图7所示场景：

RD和RA建立IBGP邻接关系，RD向RA通告路由172.168.1.0/24，RA上关于172.168.1.0/24的路由下一跳是RD的环回口地址，在RA上针对RD环回口地址有两条等价的IGP路由：RA-RB-RD和RA-RC-RD，通过下一跳迭代，在RA上针对172.168.1.0/24的BGP路由也形成等价。

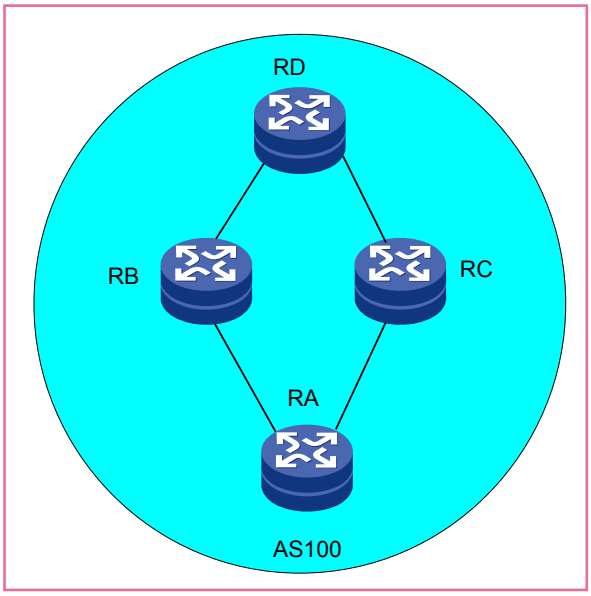


图7 IGP路由的下一跳迭代

负载分担规划原则

ORIGIN，LOCAL-PREFERENCE，MED以及AS-PATH路径属性均相同是形成负载分担的前提，在此前提下还要遵循一定的原则进行负载分担规划。以图8举例，

RB/RC均与RA通过广播网络建立单跳EBGP对等体，RA向RB/RC通告路由前缀172.168.1.0/24，RB和RC同时向RD通告，RE上收到RB和RC的路由不会形成负载分担，为什么？

缺省情况下，边界路由其器向IBGP对等体通告EBGP路由时，不会修改下一跳，因此RB和RC向RE通告的路由下一跳均为RA的接口地址，在RE上由于下一跳相同，因此不会形成负载分担。

可解决的办法是：

- 在RB和RC上针对IBGP对等体通过配置修改为：peer x.x.x.x next-hop-local，将下一跳修改为本地地址，这样RE上收到的路由下一跳不同，可以形成负载分担。
- 同样利用图8举例，RD是RR，RB和RE是其RR Client，同时RB和RE间建立IBGP邻接关系，RB通告路由172.168.1.0/24，RD反射至RE。RR在反射路由时缺省不会修改下一跳，因此在RE上关于172.168.1.0/24的路由下一跳均为RB的环回地址，可以利用前文的路由迭代，将RE-RB和RE-RD-RB的IGP路由Metric设置相同，此时能否形成负载分担？此时无法形成负载分担，原因是因为非反射路由和反射路由间无法形成负载分担。在图8中，可以将RC/RD均配置为RR，RB/RE为RR Client，RC和RD反射RB的路由172.168.1.0/24至RE，这样RE可以在两个反射路由间形成负载分担。

- 同样利用图8举例，RD是RR，RB和RE是其RR Client，同时RB和RE间建立IBGP邻接关系，RB通告路由172.168.1.0/24，RD反射至RE。RR在反射路由时缺省不会修改下一跳，因此在RE上关于172.168.1.0/24的路由下一跳均为RB的环回地址，可以利用前文的路由迭代，将RE-RB和RE-RD-RB的IGP路由Metric设置相同，此时能否形成负载分担？此时无法形成负载分担，原因是因为非反射路由和反射路由间无法形成负载分担。在图8中，可以将RC/RD均配置为RR，RB/RE为RR Client，RC和RD反射RB的路由172.168.1.0/24至RE，这样RE可以在两个反射路由间形成负载分担。

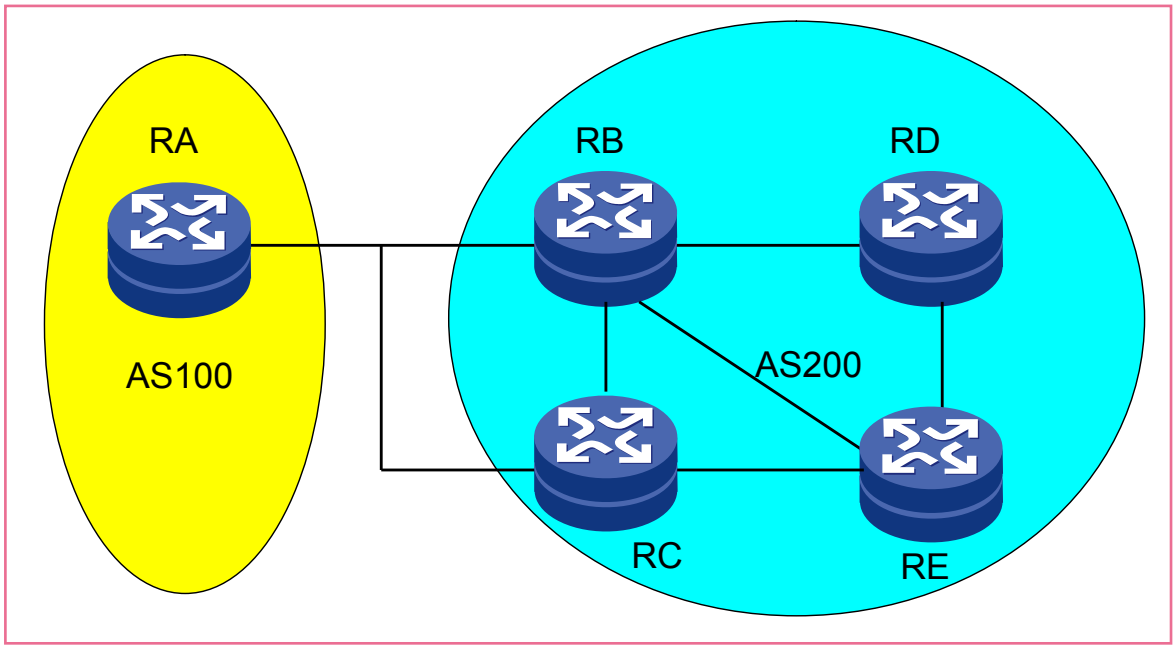
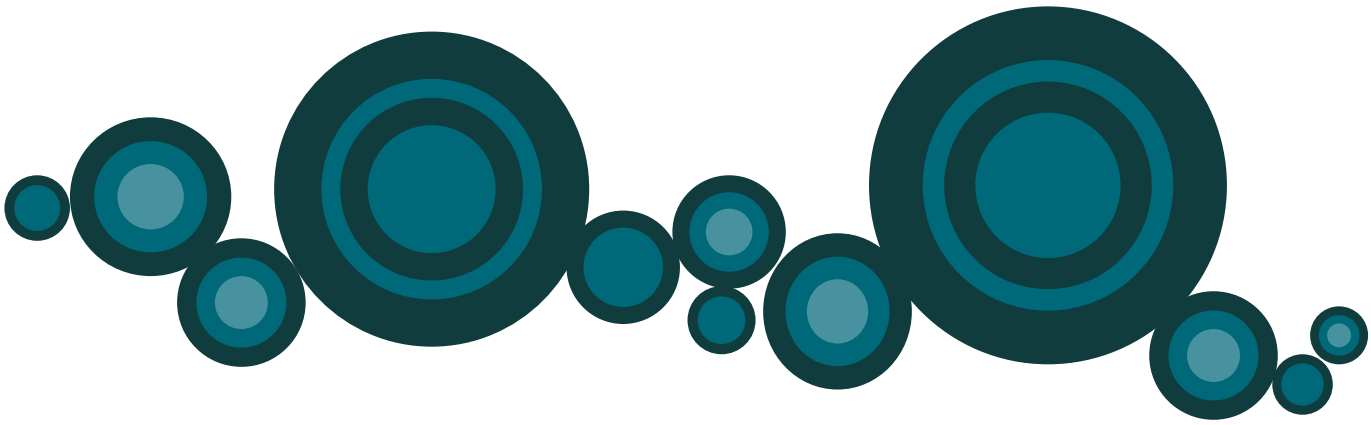


图8 负载分担规划原则

总结

负载均衡和负载分担均是BGP通过策略控制达到流量在网络中不同节点、不同链路间的平衡，起到合理利用网络资源的效果。本文介绍的是对具体网络负载均衡或分担的需求进行规划的思路，实际中满足需求的手段多种多样，需要根据实际情况做相应的规划调整，因此在实际的网络规划时不要拘泥于本文提及的方法，更多的是思路上的借鉴。同时，负载均衡和分担往往需要经过多次尝试才能尽可能接近均衡的，过于精确的均衡往往会带来策略上的复杂度，同时在考虑均衡的同时还要兼顾路径的备份。

[测试方法]



BGP测试工具及测试仪器介绍

文/许亮

引言

BGP的邻居关系是基于TCP Session的，手工构造邻居关系和路由更新比较困难。在测试BGP邻居规模，路由规模，路由收敛等性能指标时，需要借助测试仪器。此外一些独立的小软件，如我司开发的BGP Tester也非常实用，可以虚拟邻居、构造路由，还可以对路由的各个字段单独进行设置。

BGP Tester

BGP Tester是我司开发的BGP协议测试工具，用来在主机上模拟一台运行BGP协议的路由器，它具有以下几种功能：

1)四种BGP报文的构造：OPEN、KEEPALIVE、UPDATE、NOTIFY

2)与被测设备进行能力协商和维持邻居关系；

3)构造错误报文和异常报文；

4)大量路由的更新和振荡。

测试邻居建立

在BGP Tester工具栏中有一项为connection ，用于配置BGP邻居关系：

1.配置连接关系：通过选择网卡（1）、设置本端和对端地址（2）完成设置。建立BGP邻居的地址不必是网卡地址。

2.测试BGP状态机：默认情况下是Auto Open和Auto KeepAlive的，在连接建立过程中BGP Tester会自动发送open报文和keepAlive报文，使协议状态机进入Established状态。如果要对

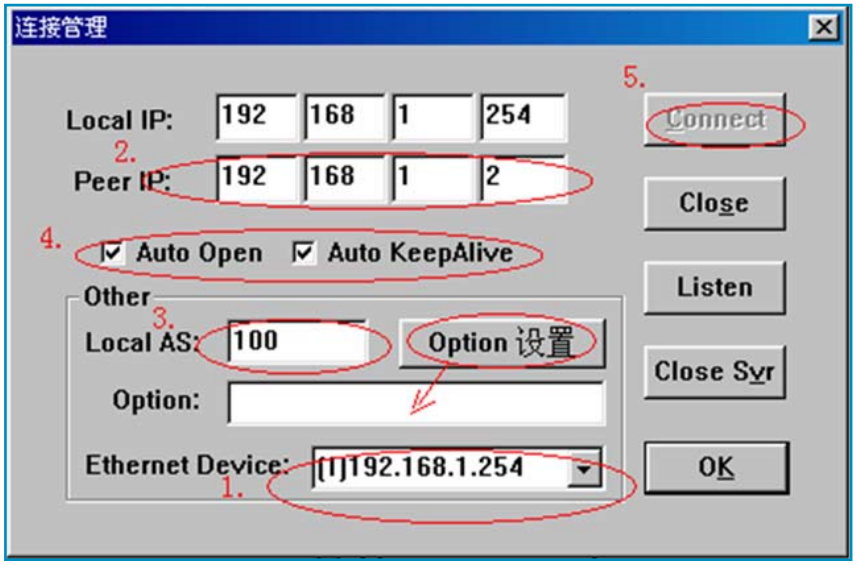


图1 BGP Tester连接管理界面

邻居状态机进行测试，不要选这两项，让BGP邻居关系停留在TCP连接建立，然后单独发送Open报文和KeepAlive报文。

3.其他连接设置：Connect表示主动发起TCP连接，Listsen表示等待被测设备发起TCP连接。

在选择主动发起TCP邻居建立和自动发送open以及keepalive的情况下，BGP邻居建立大约要一分钟。

发送大量路由

BGP Tester的一个很重要的功能就大路由表的发送，几万条路由几分钟就可以灌输进去，不仅可以作为BGP路由的大容量测试，也可以被其他协议引入，而间接的作为其他协议的大容量测试。

使用方法

选取菜单大路由发送菜单

BGP Tester提供两种前缀长度模型用于发送大路由表：24位固定长度；长度可变。

图中：

“前缀起点”值指24位长度的IP地址的第一段值；

“前缀散布”指IP地址的字段是随机数，不是有规律的；

“包之间时间”指两个UPDATE报文之间的发送时间间隔，为浮点数；

“打包条数”指每个UPDATE包中装多少条路由。

图2 大路由菜单设置

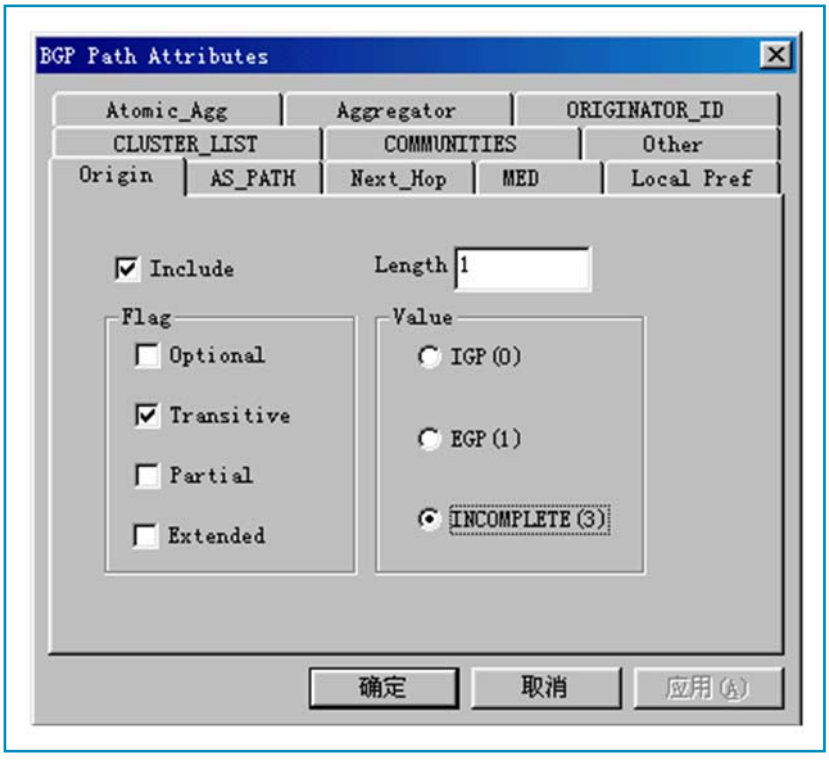
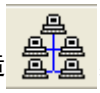


图3 可达路径属性设置

构造特殊路由

选取菜单中的update选项进行报文构造  ，点击路径属性数据的“设置”，弹出对话框，基本所有的路径属性都可以进行设置。注意：三个必遵属性必须添加；可变长属性的设置在输入后要点击Add进行追加。

Agilent测试仪器——N2X

N2X提供FE，GE，POS，ATM，FR网络接口，并且支持多种路由协议，而且可以制造数据流（2层---7层）。可以测试BGP多种功能：

- 支持测试BGP邻居类型（IBGP，EBGP，BGP多跳，MBGP）
- 支持对BGP四种报文的测试
- 支持模拟大量BGP路由
- 可以模拟BGP路由振荡
- 可以根据模拟的路由动态生成响应的数据流

连接N2X和DUT

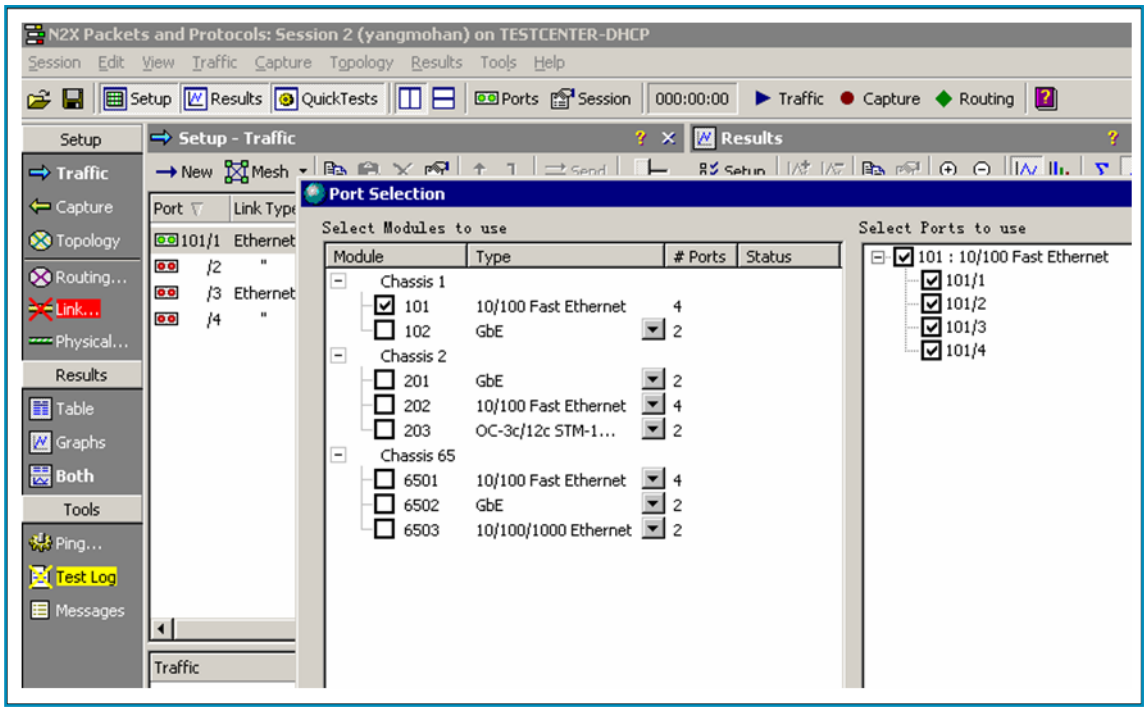


图4 N2X端口选择

N2X的硬件设备叫RouterTester。Router Tester需要一台PC作为controller与其直连进行地址分配和管理。Controller会运行DHCP Server给Router Tester上的每一块板卡分配10.0.0.0/8的地址。控制Router Tester可以在本机安装N2X Controller或者登陆到作为controller的PC进行远程控制。因此，在使用N2X时，是不用选择测试仪器地址的，直接选择机框和板卡即可。

启动N2X后会自动让你选择所想应用的端口Ports，如图中port Selection所示。Chassis选择机框；module选择板卡，板卡编号在板卡的面板上有显示，编号顺序为由左至右，从上到下；完成module选择后，该板卡的所有端口会显示在右侧的select ports to use中。

图4选择了101的四个端口101/1-101/4：

选择端口后，建立配置与被测设备的连接，配置物理层参数，点击配置链路层参数。

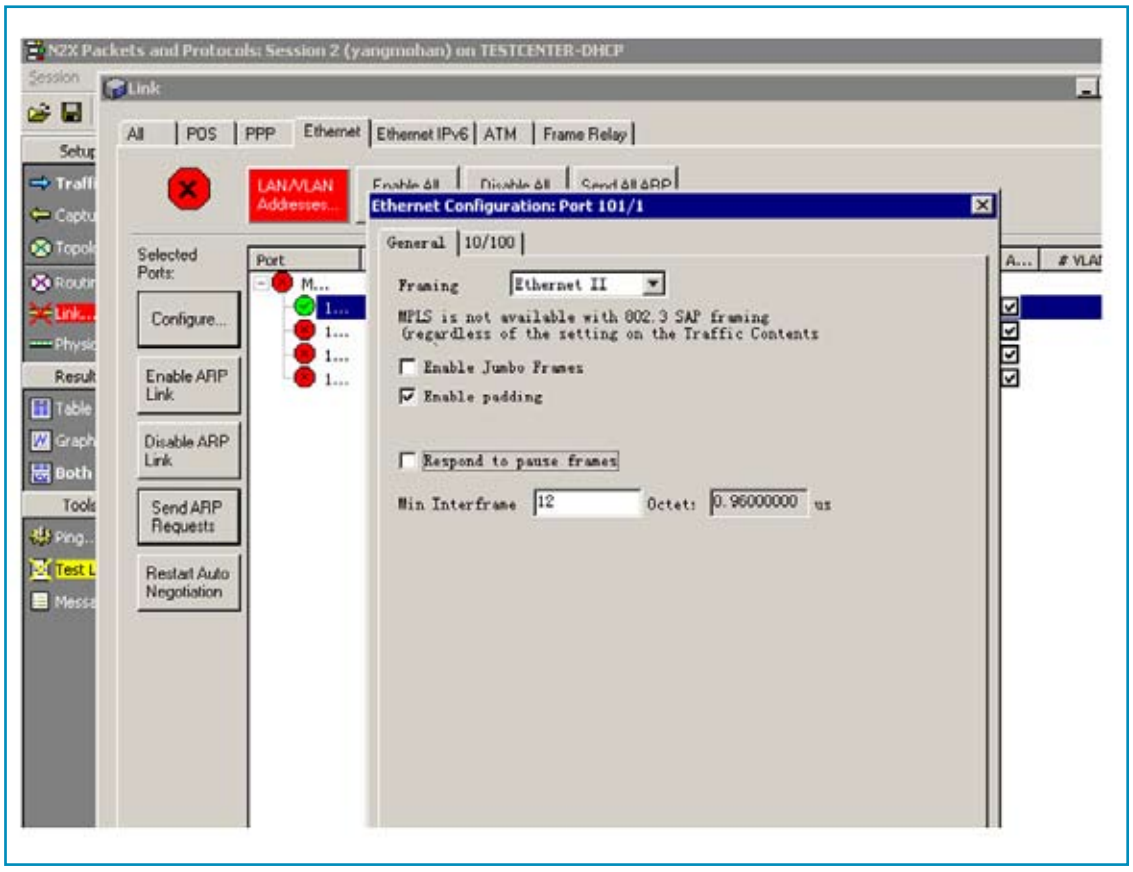


图5 连接设置

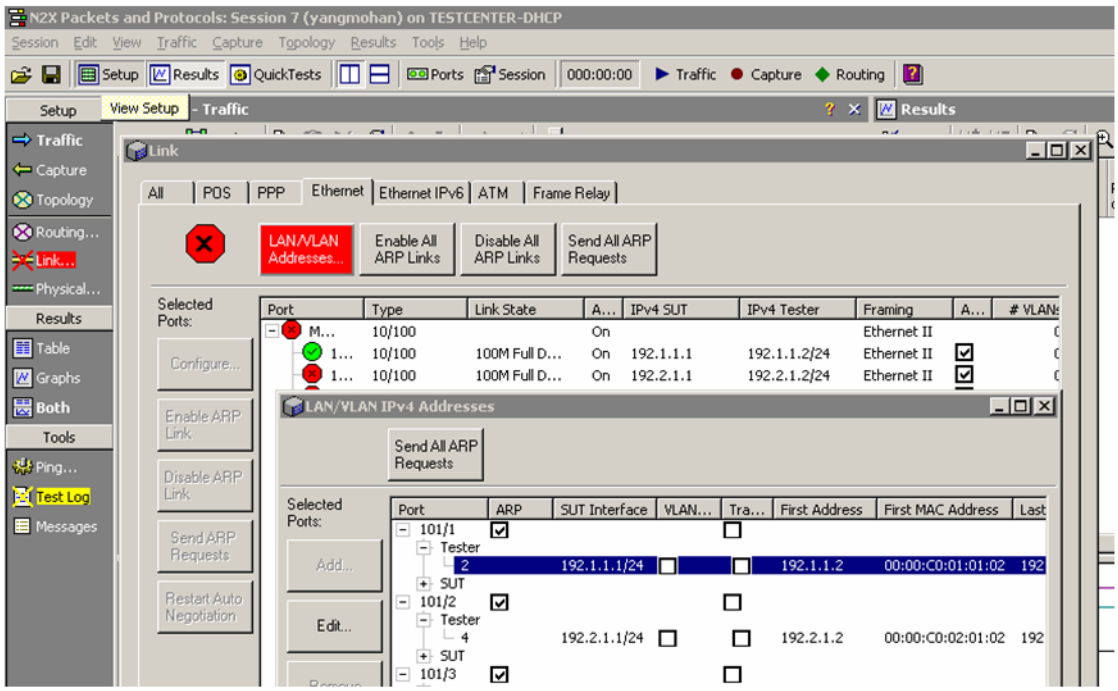


图6 配置IP地址

点击左侧Setup菜单的 Link... ，配置链路层参数。图4链路层协议是Ethernet的一个例子。这里你可以配置链路层的一些属性比如：10/100的强制，全双工的强制，帧间隙，超大帧等。

配置完链路层参数后，通过LAN/WAN Address选项配置IP与ARP，如图6所示。

发送ARP请求后，端口状态图标由 会变成 ，这个时候说明这个端口可用。

创建一个BGP session

接下来就是配置BGP的一些参数，模拟路由，流量，点击左侧Setup菜单的 Routing... ，显示Routing对话框。选择一个我们需要添加session的测试端口，下图中选择101/1端口，然后再点击Routing对话框左侧的 Add BGP-4 Session... ，添加一个session，这时会显示一个session对话框。

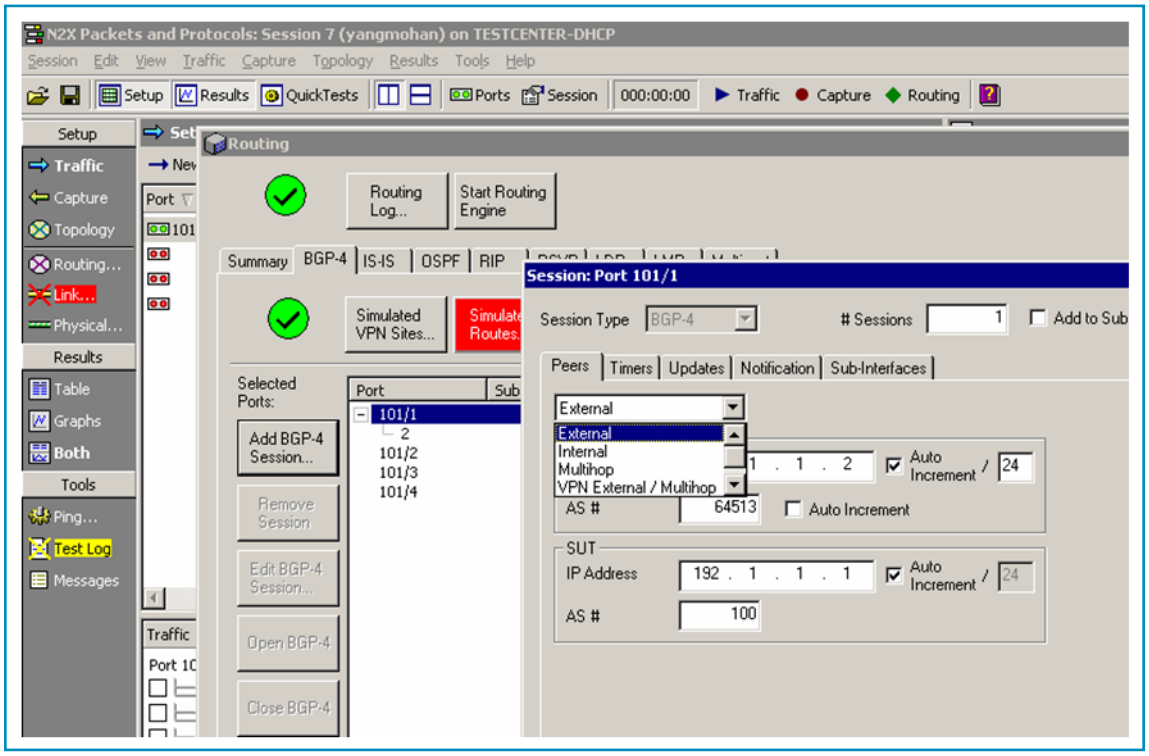


图7 配置BGP Session

模拟BGP路由

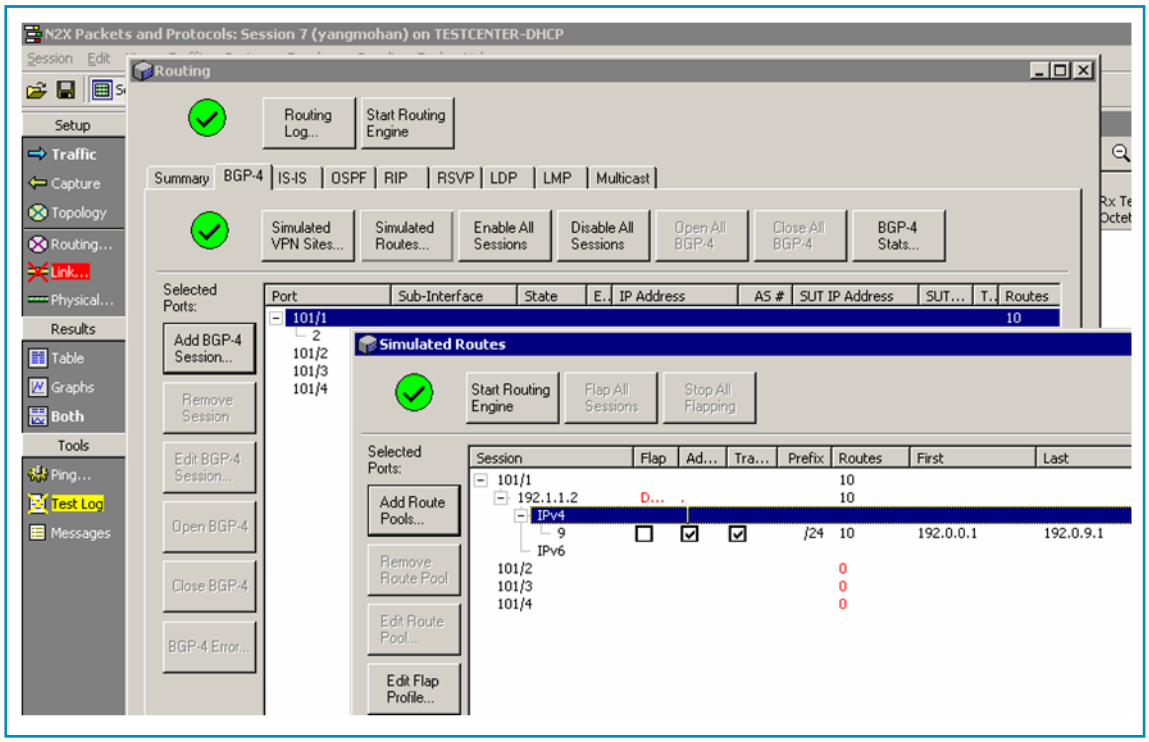


图9 模拟路由

点击Routing对话框上侧的 Simulated Routes... ，显示BGP模拟路由窗口，如下图所示。

在路由模拟窗口里你会刚才你已经建立的邻居，你要先选中它然后再点击 Add Route Pools... 。

在BGP路由路由模拟窗口下，如图7。你可以根据实际情况先输入想要给SUT的路由，在 Mandatory | Optional | Labeling 这3个选择框下可以修改BGP的属性，比如：AS，AS-SET，起源属性，本地优先级别，MED属性等。

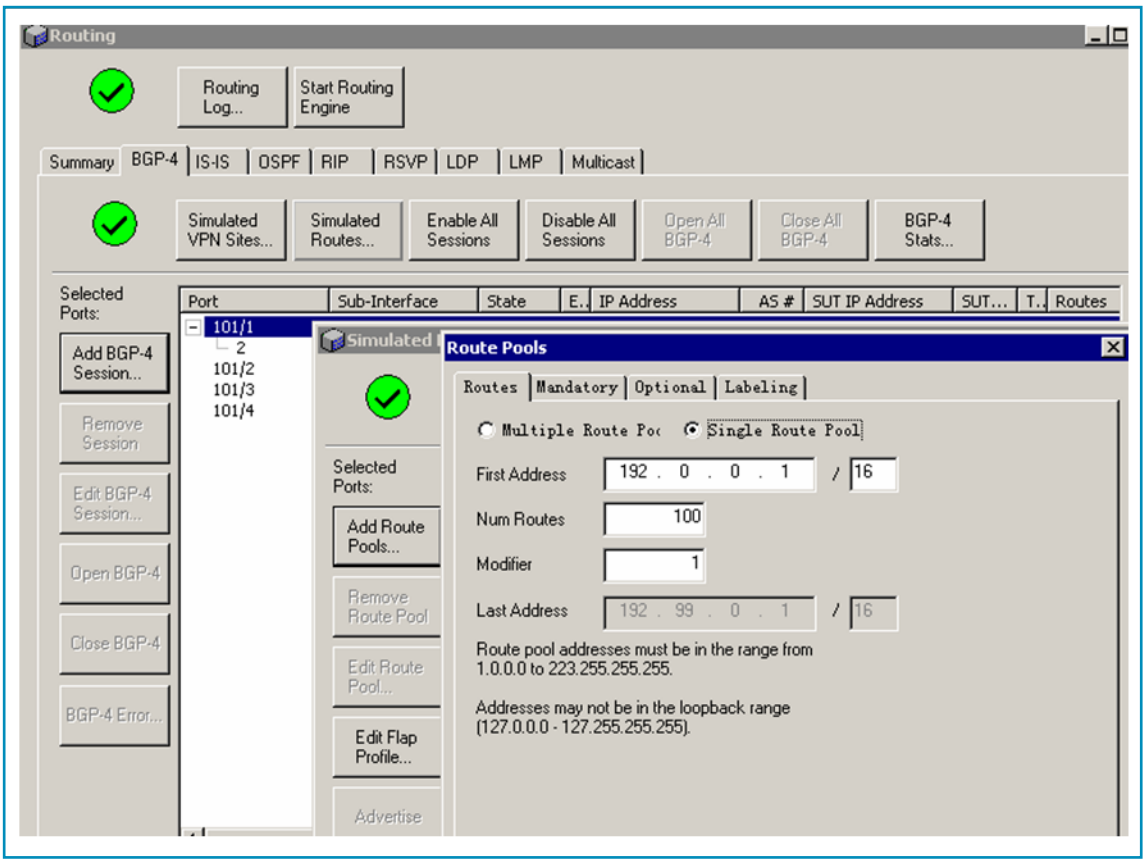


图9 修改BGP路由属性

完成以上步骤当你点击 Start Routing Engine 的时候，SUT会和你刚才的模拟的路由器端口建立邻居，并且接受到模拟的路由。在我司设备上通过DISPLAY命令可以看到相关信息，尤其可以看看测试中修改的属性。

模拟BGP路由振荡

我们可以把与FLAP相关的勾打上，然后点击 Start Route Flap ，我们就可以在SUT的路由表里看到路由抖动的效果。

点击 Edit Flap Profile... 可以设置一些和抖动相关的时间参数。

IXIA测试软件——IxExplorer

IxExplorer与N2X类似，也可以进行BGP性能测试。下面就IxExplorer特殊的地方做一些说明。

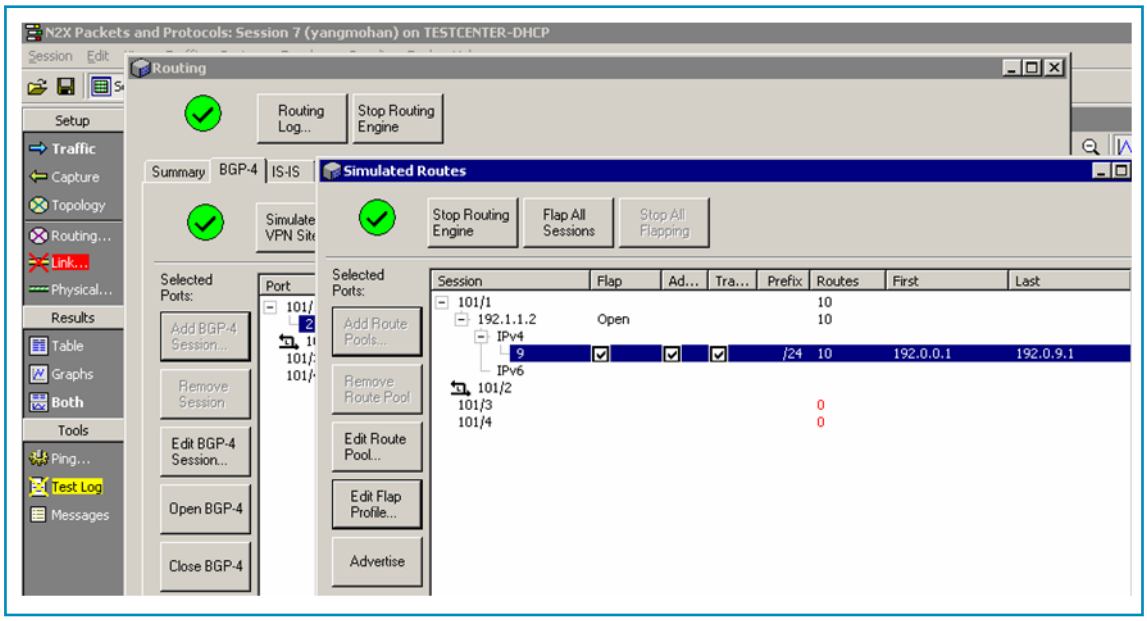


图10 设置路由振荡

连接连接IXIA和DUT

IxExplorer允许多用户操作，多个用户可以分别占用同一块板卡上的不同端口。因此登陆IxExplorer后先要选择Multiuser菜单进行login操作，才能占用端口。

测试BGP

选择路由测试图标 ，进入路由相关测试界面。

可以通过Configuration Wizards进行快捷设置：选择BGP，打开“Run Wizard”，并选中之前占用的板卡，然后跟进提示一步步设置IXIA虚拟的BGP邻居以及路由。

或者通过Protocol Management进行设置：先使能BGP再进入BGP菜单进行设置。

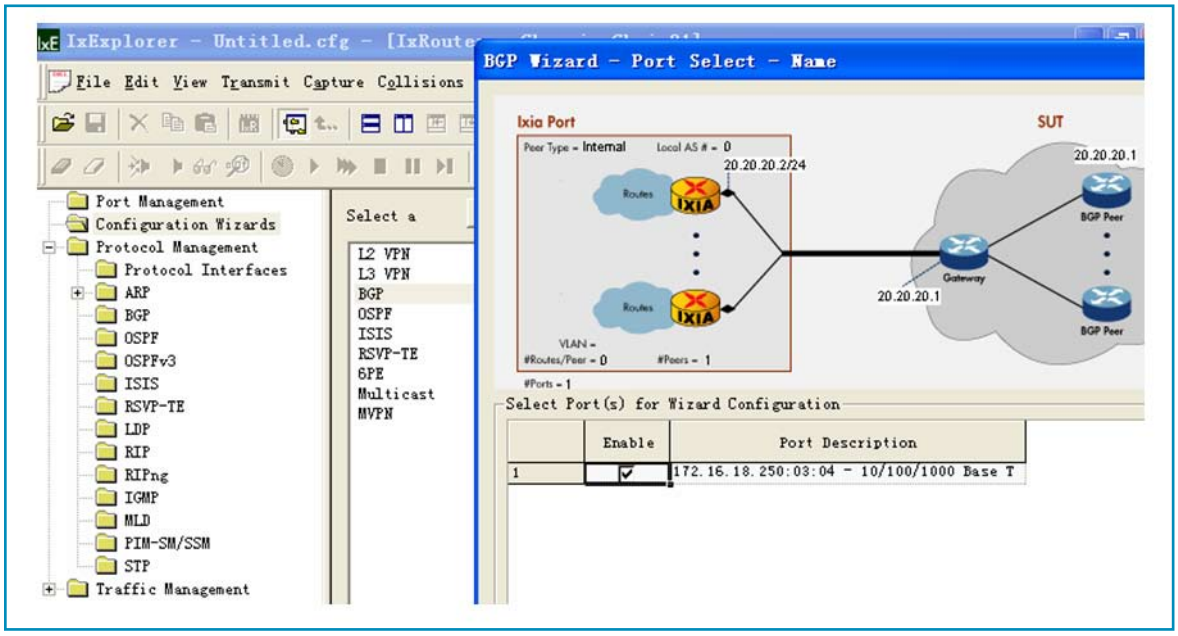


图11 IxRouter基础设置

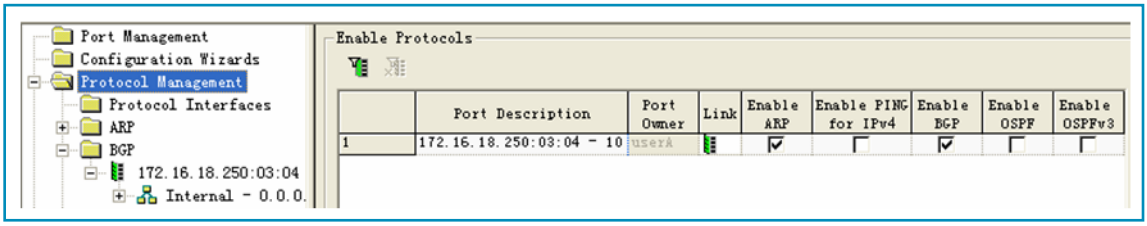


图12 IxRouter Protocol Management

BGP性能测试方法

文/张宇弟

BGP测试概述

BGP（Border Gateway Protocol）是一种自治系统间的动态路由发现协议，是网络部署的基础性协议，在ISP间得到非常广泛的应用。在网络选型中BGP作为主要的路由协议之一，其测试的重要性备受关注。

从测试角度来说，BGP测试对应的需要从如下几个方面考虑：

协议一致性，多数主流测试仪厂商提供了完善的协议一致性测试套，同时很多入网测试标准对BGP的协议一致性测试项目有明确规定，因此本文不作赘述。

稳定性，这是BGP协议测试的重点，同时本文也作为重点介绍一些主要的功能稳定性测试项目及测试方法。

性能规格，不同网络位置，不同网络规模，对于路由设备的性能规格要求不同，这也决定了不同级别的路由设备的BGP性能规格的差异，但是基本的测试思路和方法是一致的，本文也将重点进行阐述。

稳定性测试

Internet网的核心路由交换设备交换的路由数量庞大，任何网络拓扑的变化都会直接影响路由交换设备的性能，因此路由交换设备的路由表必需能够对网络拓扑变化及时处理，才能保证路由数据报文转发的正确性。对于BGP而言，作为主流的EGP路由协议，尤为关注路由维护的

稳定性，因此BGP的稳定性测试是测试的重点。

稳定性测试主要通过网络拓扑变化和路由持续变化的模拟，来测试路由交换设备处理动态路由环境的能力，本文主要关注路由收敛和抗路由抖动能力这两个方面。

路由收敛测试

路由收敛测试从两个方面可以考虑，一是路由学习收敛能力，另一个是路由切换收敛能力。针对图1的测试组网分别进行阐述。

路由学习收敛能力

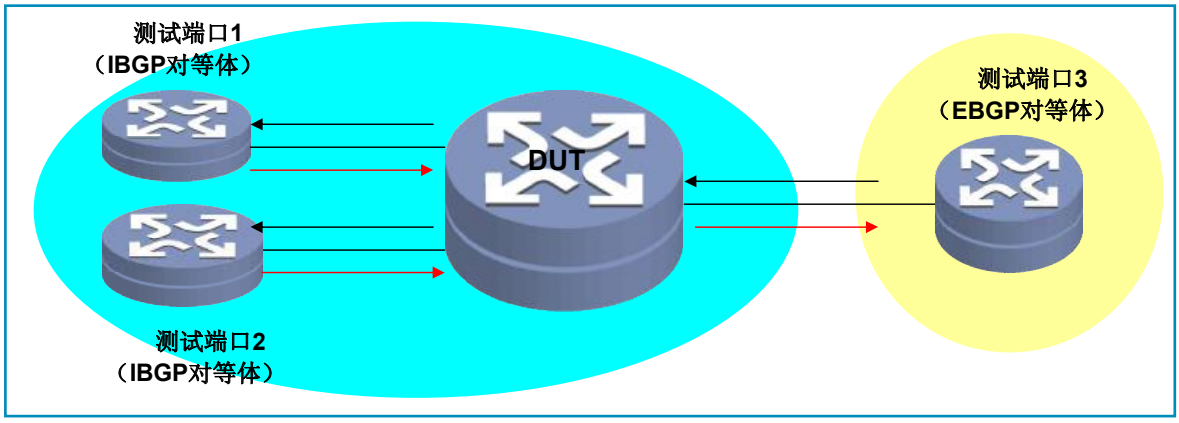


图1 路由收敛能力测试组网

1）场景设置

在路由收敛学习能力测试中，待测设备和测试端口1建立IBGP邻接关系，和测试端口3建立EBGP邻接关系。需要注明的是，IBGP和EBGP的选用可以根据测试需要进行调整，没有强制要求。

2）测试步骤

a 在待测设备上观察，等待BGP对等体连接建立

b 测试端口1构造BGP路由，路由数目建议取待测设备实测的BGP路由规格的50％

c 测试端口2构造流量，目的是步骤b通告的路由条目，流量大小取吞吐量的30-50%

d 启动测试端口的路由发布和流量发布

3）测试结果分析

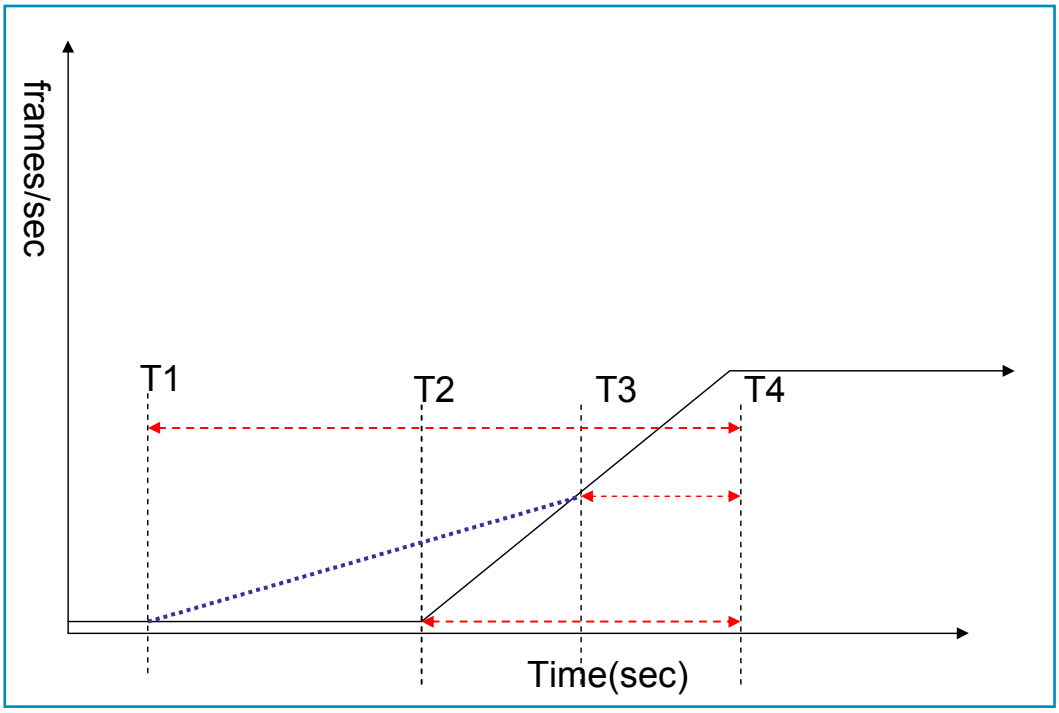


图2 路由学习收敛能力评估

路由学习收敛能力评估难点在于收敛时间起始点的确定，如图2所示，T1代表测试端口开始发送第一个update报文的时间点，T2代表待测设备第一条BGP路由计算成功时间，T3代表测试端口发送完所有Update的时间点。首先我们看看选用T2这个值，往往在图形界面的结果统计中，T2的选取相对直观。但是选择T2其实忽略了待测设备接受路由时间和路由计算的时间，这个时间的忽略对收敛能力的评估不能非常实际的反应实际情况，毕竟在路由接收和路由计算算法上在T2之前会因为实现的差别存在着能力的差异。再看看选用T3的情况，图2的蓝色虚线代表的就是测试端口发完所有BGP路由，T3是这个事件和路由收敛过程的交汇点。这个时间点应该是滞后于待测设备第一条路由计算成功的时间，因此在T2-T3区间，部分路由已经计算收敛，因此该时间段是待测设备在，选择T3的考虑是能够抛开测试端口发送BGP路由的时间。

T1时间点的选用，我们先看T1-T2的时间段，这个时间段包含了测试端口发送第一个update到第一条路由计算收敛完成，路由转发生效的时间，T2-T4是所有BGP路由计算收敛和路由转发生效的时间。因此T1-T4能够将路由接收、路由同步、路由计算及下发等等因素都纳入计算时间，所以T1的选择更合理一些。但是需要说明的是，无论哪个时间点的选择，收敛时间严格比较的前提是测试端口发送时间相同及忽略链路延迟的因素。

路由路径切换收敛能力

1）场景设置

待测设备和测试端口1、2通过链路端口地址建立IBGP邻接关系

2）测试步骤

a 查看待测设备和测试端口1、2的IBGP邻接关系是否建立成功

b 测试端口1、2同时向待测设备通告相同的路由，路由条目取规格数目的50%

c 通过通告路由属性的设置例如将测试端口1的路由origin属性设置为IGP，将测试端口2的路由origin属性设置为incomplete，让路由优选端口1。或者在待测设备上通过路由策略，例如将测试端口1的local-preference设置比测试端口2大，优选端口1

d 测试端口3构造流量，目的是步骤c通告的路由前缀，流量大小为实测吞吐量的30－50%

3）测试结果分析

路径切换收敛是发生路由变化或者网络拓扑变化，可以模拟测试端口1发送针对步骤c的路由撤销，也可以直接将测试端口1进行shutdown。这两种事件对于待测设备进行路由处理行为稍有不同，对于发送路由撤销，测试端口会发送Update，待测设备收到撤销报文后进行路由撤销并重新优选路由；路由属性发生变化，待测设备进行路由优选；测试端口1链路中断，由于待测设备和测试端口1是通过链路端口建立BGP邻接，因此会快速检测到邻接关系DOWN，并删除对应的路由条目进行路由重选。下面我们就分别对这三种方法进行分析。

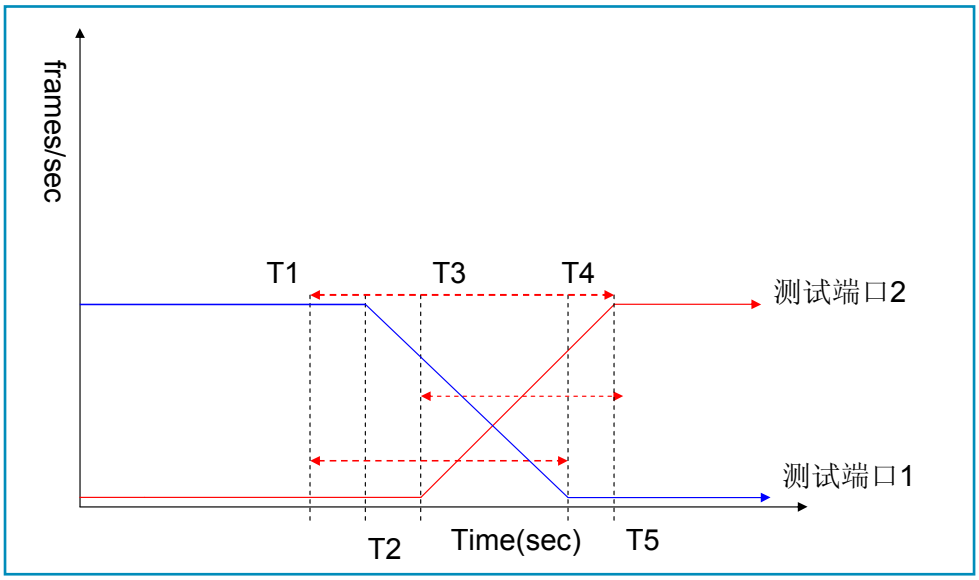


图3 路径切换路由收敛能力评估1

图3是中蓝色代表测试端口1收到的测试流量，红色代表测试端口2收到的测试流量。T1是测试端口1发送第一个撤销报文的时间点，T2是待测设备路由删除起始时间点，T3是第一条测试端口2通告路由优选成功的时间点，T4是测试端口1所有路由删除的时间点，T5是所有测试端口2通告路由优选的时间点。从这几个时间点可以得到几个收敛值，T1-T4可以得到待测设备收到路由撤销到全部撤销的时间值，T3-T5可以得到路由起始重选到全部重选的时间值，T1-T5是计算整个路由收敛的时间值。这一点和路由学习收敛能力计算的第三种方法相对应。T1-T2在忽略发送接受延迟的前提下，可以大概得出路由接受到删除前的处理时间。

在路径切换收敛能力测试中，还需要注意的是路径切换过程中发生的报文丢失，可以通过报文丢失率作为评估收敛能力的一个指标。

前文提到的测试端口1发送撤销报文或者属性修改的路由更新，其收敛特征曲线基本都符合图3，但是在实际测试中，两种不同的方法得出的结果会因为处理流程的不同得出不同的结果，因此建议在测试中可以进行测试对比。

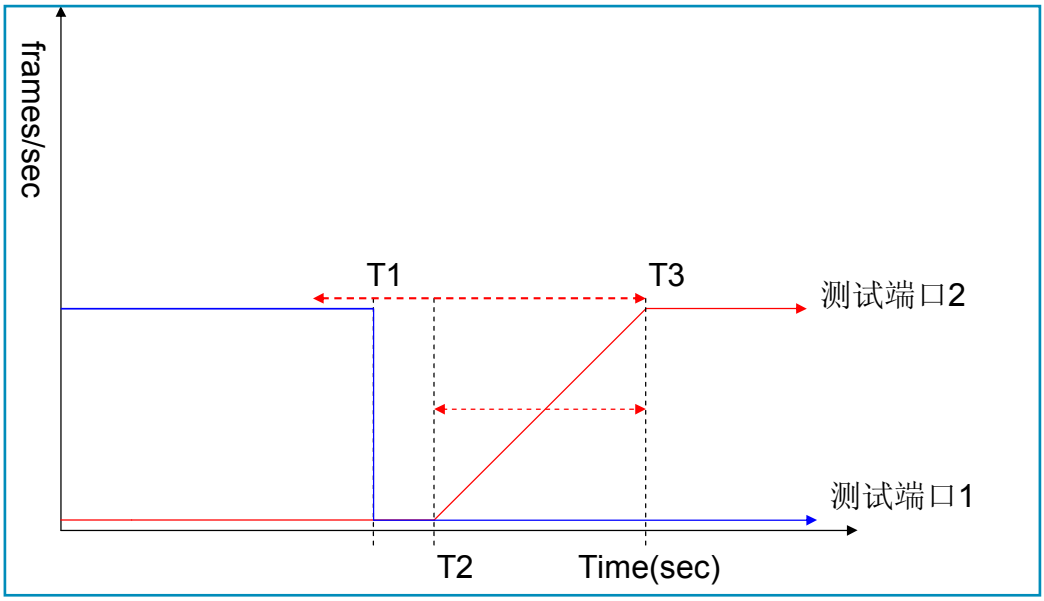


图4 路径切换路由收敛能力评估2

图4显示的是测试端口1发生接口DOWN时间的流量特征，T1是测试端口1发生DOWN的时间点，在这个时间测试端口1的流量中断，在T1-T2过程中待测设备进行路由更新，T2开始重选出第一条路由，T3所有的路由重选成功。相对来说，这种测试方法相对简单，但是无法测试收到路由撤销进行路由删除的过程，但是可以避免测试仪器发送路由撤销快慢的干扰。

等价路由收敛

在考虑上述两种情况后，我们还需要考虑存在等价路径的情况，在具体分析前我们先确定，等价路由生效后，路由转发是平均分担在几个等价路径上进行转发这样的前提。

等价路由负载分担的情况下的收敛测试，考虑两种场景：断开其中一条链路或者添加一条链路。断开一条链路，BGP通过计算会删除一条路由，转发表会删除该路由出接口对应的条目。但是实际的流量转发理论上不会中断或丢包。

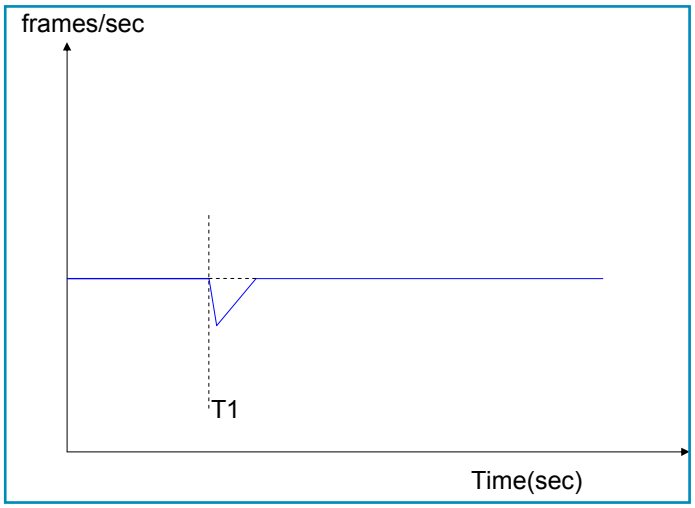


图5 等价路由收敛1

图5显示的是实际测试中流量特征，在路径切换时会有少量的丢包，在实际测试过程中丢包率是测试衡量的重要指标，在很多选型或入网测试中要求丢包数量限定在一个范围内。

当添加一条路径和已有路径形成等价时，由于BGP需要重新路由收敛，并同步刷新路由转发表项，因此会带来一定的转发时延，图6显示的是该测试过程中的延迟情况。

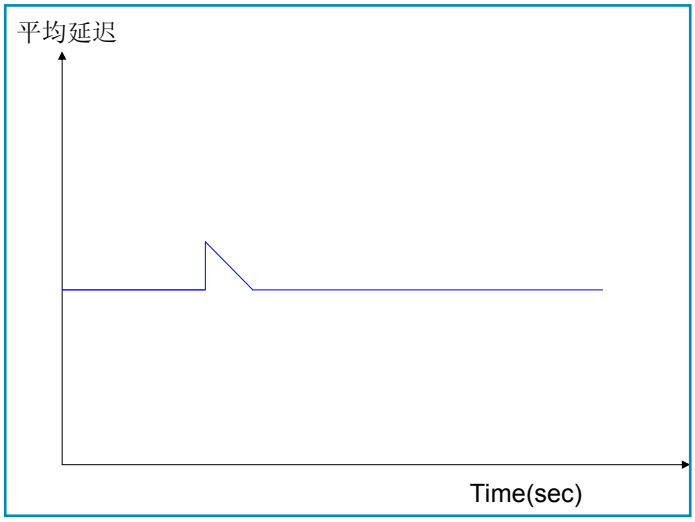


图6 等价路由收敛2

抗路由抖动能力测试

路由抖动是模拟在网络中路由的反复更新/撤销时，路由交换设备处理路由更新的能力。这种能力关注两个方面，一是在路由更新处理的同时，路由转发的抗干扰能力，另一个是在转发路径上，路由的更新和转发恢复能力。

路由转发的抗路由抖动干扰能力

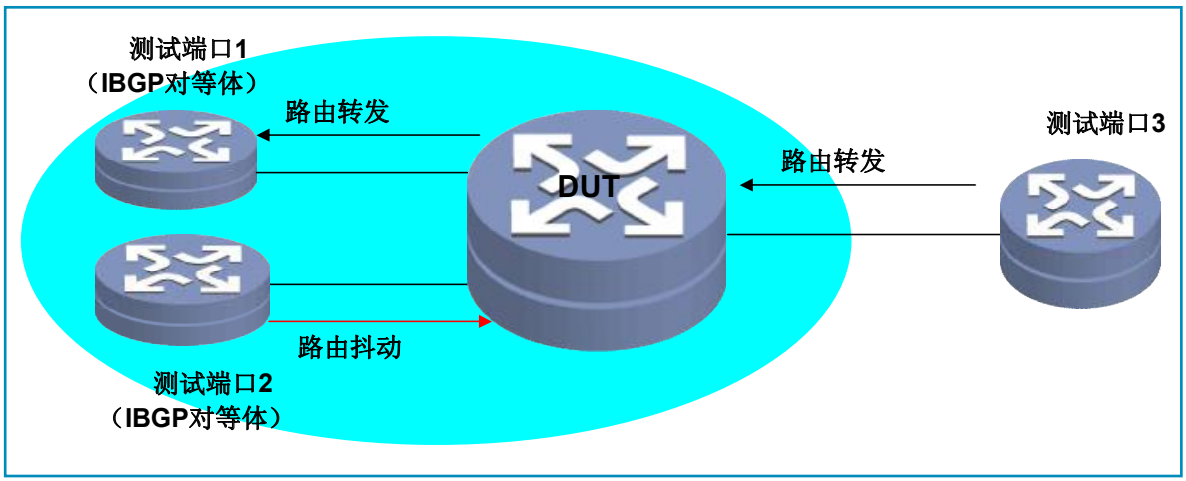


图7 路由抗抖动能力1

1）场景设置

图5是路由转发抗路由抖动干扰能力的组网图，测试端口1、2和待测设备建立IBGP邻接关系，测试端口1和测试端口3作为路由转发测试端口，测试端口2作为路由抖动测试端口。实际在测试的复杂度可以根据需要进行扩展，比如提高路由抖动测试端口的数量，流量进行full-mesh设计。

2）测试步骤

a 测试端口1向待测设备发送一定数量的路由，建议路由数量取30％规格

b 测试端口3发送流量，目的地是测试端口1通告的路由前缀

c 测试端口2向待测设备发布路由，数量取路由规格的30%

d 测试端口2每隔一个固定周期进行路由的撤销/重新通告

3）测试结果分析

在测试步骤d进行过程中，待测设备会被动的反复进行路由更新处理，其路由处理和同步转发表项的压力都会骤增，这种压力的情况下测试端口3和测试端口1之间的路由转发受干扰的程

度是本测试关注的重点。可以观测测试端口3到测试端口1之间的转发的传输延迟和传输速率平均值和路由抖动前的百分比。

路由的更新和转发恢复能力

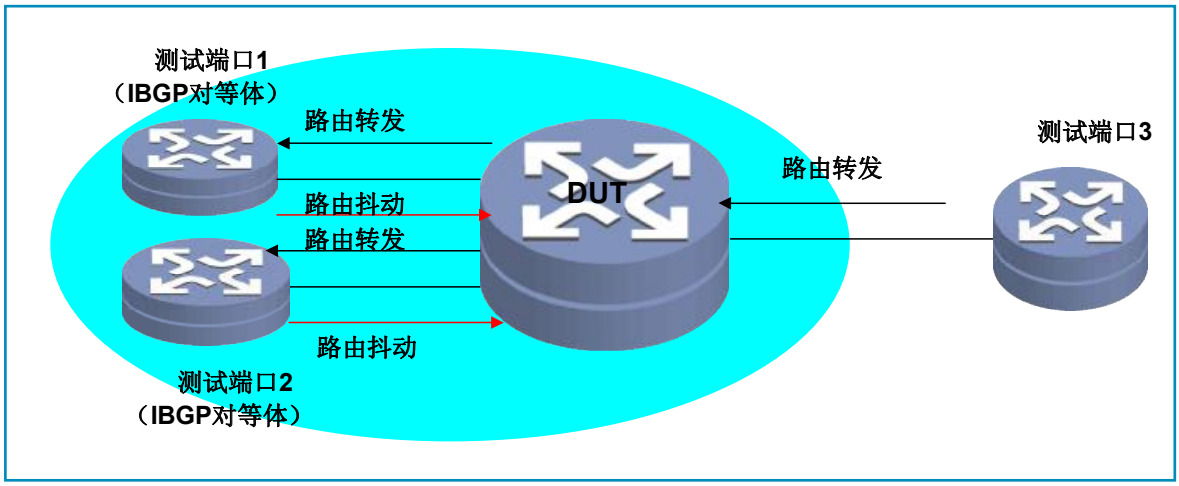


图8 路由抗抖动能力2

1）场景设置

路由抗抖动能力测试组网，路由抖动和路由转发的测试端口重合，也就是说路由抖动是发生在路由转发路径上。

测试端口1、2和待测设备建立IBGP邻接关系，测试端口1、2作为路由抖动测试端口，测试端口1、2、3作为路由转发测试端口。

2）测试步骤

a 测试端口1、2向待测设备通告路由，数量取路由规格的50%

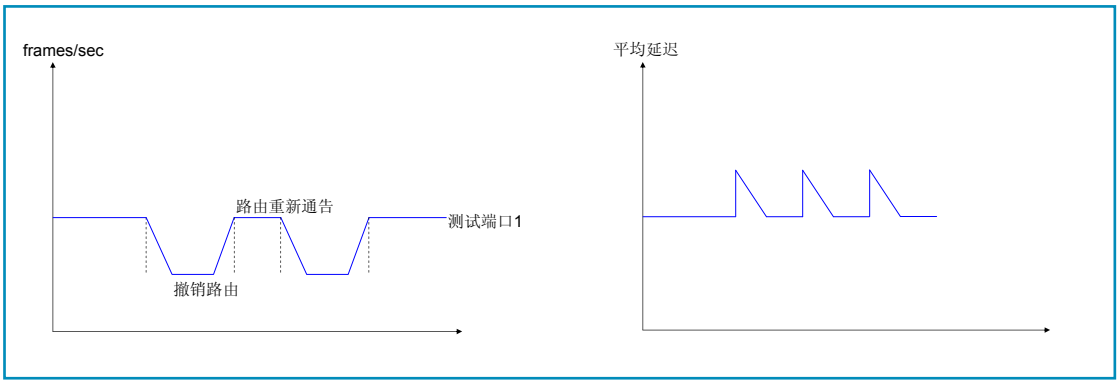
b 测试端口1、2周期进行50％已通告路由的撤销和重发布

c 测试端口3发布流量，目的为步骤a通告的路由前缀

3）测试结果分析

步骤2中的路由撤销，会导致该部分路由的删除，以及转发表的同步删除，流量对应丢失。重新发布后，路由会重新计算并同步转发表，转发恢复。同时，未抖动的路由应该保持路由稳定及持续转发。

图9显示的是流量特征和转发延迟分布特征，通过平均转发延迟的计算可以评估抗抖动能力：



性能规格测试

对于路由设备，尤其是核心路由设备，对其路由的性能规格是测试的重点。性能规格测试分为两种情况，一是在给定的规格数据前提下，进行验证测试；另一种是通过逐步添加进行规格数据的摸底测试。下面介绍几个典型的性能规格测试项，部分是针对摸底测试的测试方法，一部分是规格验证测试。

BGP对等体会话连接数

BGP的对等体是建立在TCP连接上，因此可以建立端到端的对等体连接。TCP连接能够保证BGP连接的稳定性和可靠性，但是对系统资源的消耗也相应增加。BGP对等体会话数量是BGP性能规格测试的一个重点。

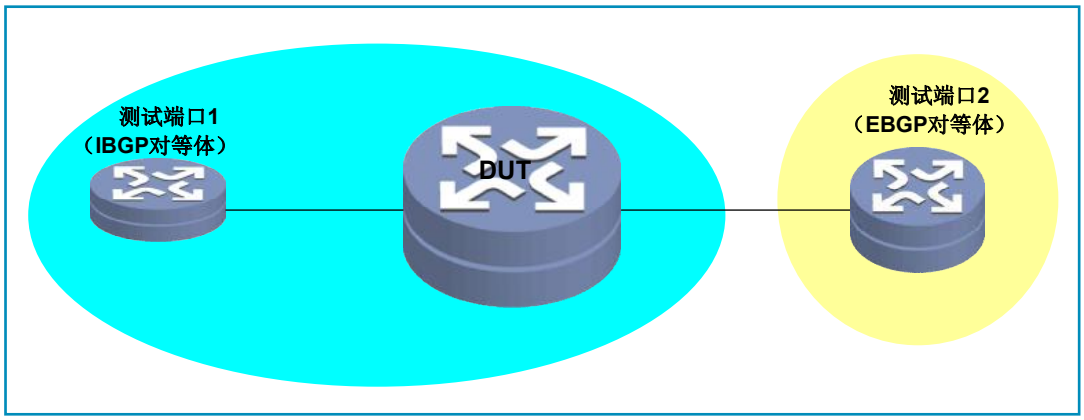


图10 BGP对等体会话连接数

1）场景设置

测试端口1和测试端口2的设置是为了测试待测设备的IBGP邻接和EBGP邻接，已IBGP为例。

2）测试步骤

a 测试端口1创建逻辑子接口，待测设备对应的建立逻辑子接口

b 待测设备和测试端口1通过子接口建立IBGP邻接

c 测试端口1通过建立的IBGP邻接通告路由，整体通告的路由数量建议是规格的50％

d 测试端口2打目的是步骤c通告的路由前缀，流量大小去实际吞吐量的30-50％

e 不断增加步骤a的子接口数量和步骤b的IBGP会话邻接数量，重复步骤c、d

3）测试结果分析

如何衡量连接数量的多少是会话连接规格测试的难点。很多测试资料中忽略了测试步骤c、d，不断增加连接数，直到无法建立新的BGP连接。这种方法的客观性较差，在没有路由通告和路由转发的压力下得出的数据无法贴近实际网络。通过观察路由学习、路由转发稳定性、CPU利用率几个方面综合考虑，持续BGP会话连接数的增加，系统的压力会逐步加大，对于路由学习的能力、路由转发的稳定性以及CPU利用率都会受到影响。可以通过观察保证路由学习、路由转发、CPU利用率正常的临界点，来确定BGP会话连接数量的规格。

可以选择测试EBGP的邻接，以及IBGP和EBGP的混合邻接数量，在测试资源允许的情况下可以多接入一些测试端口，和待测设备建立邻接关系。

BGP路由表容量

路由表容量测试是路由协议性能规格测试的重点之一，测试组网可参照图8。

1）场景设置

测试端口1和测试端口2的设置是为了测试待测设备的IBGP邻接和EBGP邻接，已IBGP为例。

2）测试步骤

a 测试端口1和待测设备建立IBGP邻接关系

b 测试端口1向待测设备发送一定数量的路由条目，假设为5万条

c 测试端口2发送流量，目的是步骤b发送的路由前缀

d 在步骤b的基础上按一定数量递增通告路由数量，同时步骤c的流量目的对应修改

3）测试结果分析

有两种方法来检测路由学习容量，一个是在测试端口2和待测设备建立EBGP邻接，在测试端口2观察收到待测设备发来的update报文，当无法收到新的Update报文时，检查当前有效通告的Update报文包括的路由数量。这种方法是目前使用较多的方法，但是在使用这种方法种都忽略了路由转发这个环节，实际的路由容量应该是以正常进行路由转发为前提，当选择无法发送Update这个时间点，可能在此之前路由转发已经无法正常。笔者认为，应该综合update发送、路由转发稳定性、CPU利用率几个方面来考虑，任何其中一项无法保持正常状态都应停止路由递增，结束测试。

BGP等价路由数目

往往在待测设备的软件实现中限定了等价路由的数目，这种测试往往在满等价路由规格的情况下测试待测设备的路由学习能力、路由转发稳定性及CPU利用率。

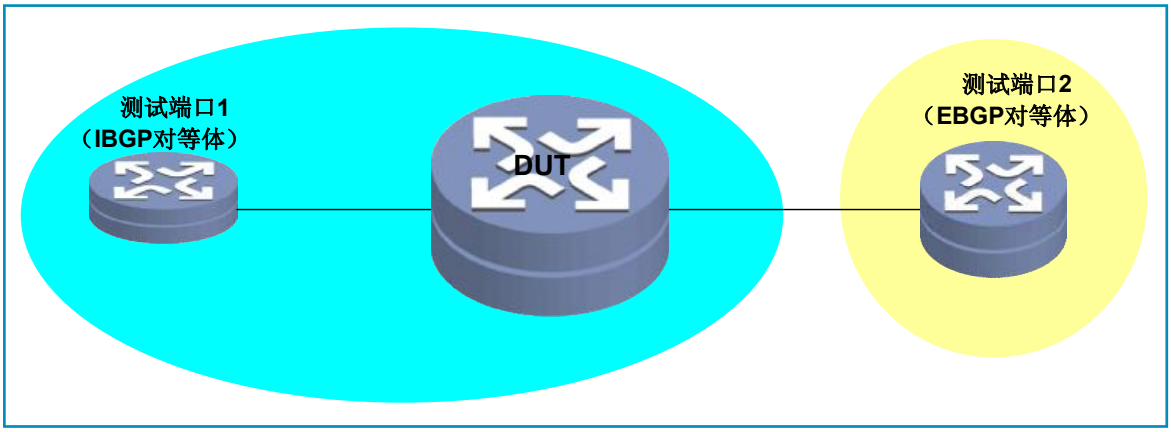


图11 BGP等价路由规格测试

1）场景设置

图11显示了最简化的等价路由测试环境，待测设备和测试端口1对应创建N（等价路由规格数）个逻辑子接口，通过子接口建立IBGP邻接关系。

2）测试步骤

a. 测试端口1的N个子接口向待测设备通告相同的路由前缀，路由前缀数量乘以N等于路由容量的50%

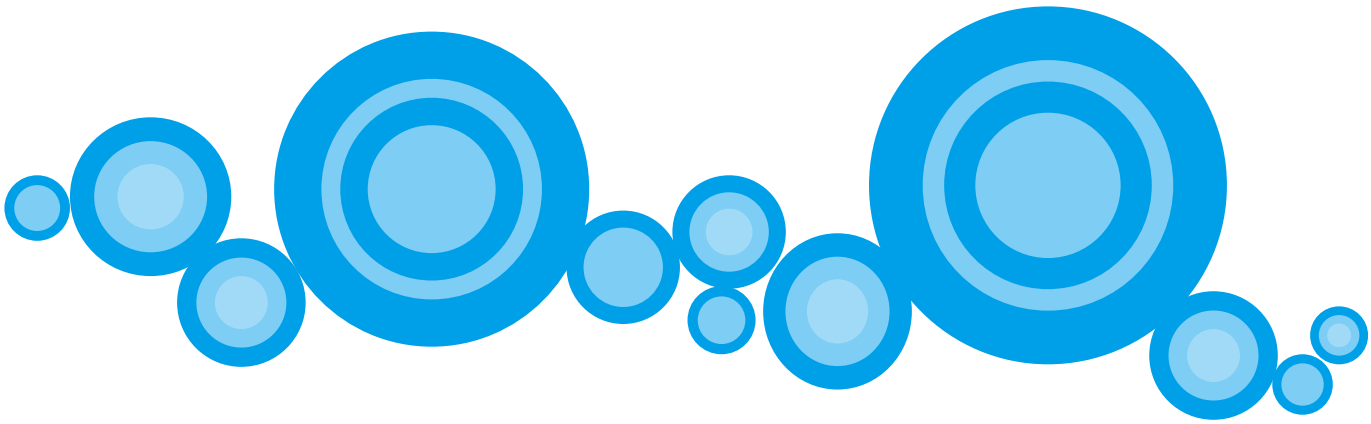
b. 测试端口2构造流量，目的是测试步骤a通告的路由前缀，流量大小是实测吞吐量的30-50%

3）测试结果分析

首先我们需要观察测试步骤a中通告路由的学习情况以及测试步骤b的转发稳定性，以及此时的CPU利用率。在上述三个方面都正常时，说明等价路由规格能够满足。

很多厂家可能会提供其他的规格项，如AS联盟内子AS的规格数目、团体属性的团体个数、network引入的路由数目，这些规格项多数是验证测试，思路可以延承等价路由的测试思路，本文不再具体描述。

[最新进展]



BGP最新发展

文/程锋章

前言

在过去的几年中，人们对边界网关协议（BGP）进行了大量的研究和思考。但是在目前的互联网中甚至是在任何运行BGP协议的私有网络中大规模更换BGP几乎是不可能的。与其更换BGP，不如通过引入新的功能来增强它并满足新的需求。近年来，BGP在如下方面取得了不少的进展，主要表现在性能方面（如提高收敛速度）、可靠性（如动态能力协商、GR）、安全性（TTL检测、防攻击）以及扩展性（VPNv6，4字节AS）等。

性能优化

性能优化的研究方向很多，比如路由决策、路由更新以及路由收敛等，各个厂商都有其内部的一些实现来更好的满足实际需要，不便一一展开详述。比如在H3C设备上针对BGP提供的特性keep-all-routes，该特性用来保存所有来自对等体/对等体组的原始路由信息，即使这些路由没有通过已配置的入口策略。这样在策略控制更改后能够迅速进行新的路由决策和更新。当然目前的研究更多的是集中如何提高路由的收敛性能上。

BFD for BGP

BFD提供了一个通用的、标准化的、介质无关、协议无关的快速故障检测机制，可以为各上层协议如路由协议、MPLS等统一地快速检测两台路由器间双向转发路径的故障。BFD在两台路由器上建立会话，用来监测两台路由器间的双向转发路径，为上层协议服务。BFD本身并没有发现机制，而是靠被服务的上层协议通知其该与谁建立会话，会话建立后如果在检测时间内没有收到对端的BFD控制报文则认为发生故障，通知被服务的上层协议，上层协议进行相应的处理。

BGP协议的keepalive时间间隔缺省为60秒，最小可以配置为1秒，这样holdtime缺省为180秒，最小为3秒，邻居关系的检测比较慢，对于报文收发速度快的接口会导致大量报文丢失。通过BFD进行快速故障检测，可以实现邻居关系的快速检测，加快协议收敛，有点类似ospf的快速hello特性。

路由更新

路由更新早在BGP提出的时候已经发展成熟，属于路由协议中必不可少的一部分，通过定时的更新路由报文来满足实际需要，并且针对IBGP和EBGP默认设置不同的更新周期，比如H3C设备IBGP默认更新周期为15S，EBGP则为30S。

实际上这种实现也有其缺点，比如可能从一定程度上降低了收敛的速度、部分路由的震荡可能影响其他路由的收敛，实现过于保守。在RFC4271第9节中提到：“The parameter MinRouteAdvertisementIntervalTimer（MRAI） determines the minimum amount of time that must elapse between an advertisement and/or withdrawal of routes to a particular destination by a BGP speaker to a peer”。

协议要求指定对等体在发送或者撤销路由过程中必须间隔MRAI，比如对等体更新完路由后启用MRAI定时器，在这个时间内不会发送新的更新或者撤销消息。当定时器超时后开始发送更新或者撤销报文（如果有），这个更新过程完成后再次启用MRAI定时器重复之前步骤。通过该方法能够避免持续路由震荡对设备影响，提高设备稳定性，同时一定程度上能够提高收敛性能。

比如在实际应用中考虑到AS内部要求快速收敛，所以针对IBGP设置的MRAI要比EBGP小。由于该方法并不能控制路由决策，所以在MRAI超时的时候，最近一次被优选出来路由将被发送出去。也就是说对端收到的路由可能暂时不是最优的路由，针对这种情况可以调小MRAI，尤其是IBGP或者PE-CE的EBGP邻居。

可靠性

Graceful Restart Mechanism for BGP with MPLS

在RFC 4724（Graceful Restart Mechanism for BGP）中讲述了一种帮助减少BGP重起对路由的负面影响机制，而在最新的 RFC 4781（Jan 2007 Graceful Restart Mechanism for BGP with MPLS）继续扩展该机制以便在BGP携带MPLS标签时，减少在BGP重起时对MPLS转发的负面影响。该机制对于BGP NLRI中携带的地址类型是不可见的，因此它可以在BGP中携带的任何地址族中工作。

在LSR能够在其控制平面重起（尤其是BGP重起）时能够保留其MPLS转发状态，那么在此过程中最好能够不对经过该LSR的LSP（尤其是由BGP路由）造成干扰。该文档中讲述的机制和Graceful Restart Mechanism for BGP一起来实现该目标。

支持该机制的LSR使用Graceful Restart Capability向对等体进行能力通告，在能力通告中的SAFI应该表示NLRI不仅仅包括地址前缀还应该包括相应的标签。在LSR控制平面重起后遵循“Graceful Restart Mechanism for BGP”中的处理过程。另外，在此过程中如果LSR能够保留MPLS转发状态，LSR通过将对所有AFI/SAFI的Graceful Restart Capability的设置适当的Flag域。为了叙述简便这里所说的MPLS转发状态是指入标签到出标签、下一跳或者地址前缀到出标签、下一跳的映射（见RFC4781）。转发状态就是指MPLS转发状态在重起过程中不需要保留IP转发状态。一旦重起的LSR完成了路由选择除了完成正常的GR重起过程还要进行如下操作：

第一种情况：

a）路由器选择的最优路由是和标签一起接收的；

b）该标签不为空；

c）LSR将自己作为路由的下一跳；

重起LSR在MPLS转发状态中搜索<出标签下一跳>和收到的路由中一致的表项。如果找到该表项LSR，不再将该表项标注为stale，而且如果找到的表项是（入标签，<出标签下一跳>）而不是（前缀，<出标签下一跳>）时，LSR在向邻居通告时使用入标签。如果找到的表项中没有入标签或者没有这样的表项，LSR在向邻居通告路由时将选择一个没有使用的标签。

第二种情况：

a）重起的LSR选择的最优路由时要么没有标签要么是一个空标签或者该路由是由重起路由

器自己产生的；

b）LSR将自己通告为该路由的下一跳；

c）LSR必须为该路由分配一个非空标签；

此时LSR在MPLS转发状态中查找需要进行标签弹出操作并且下一跳相同的表项。如果找到了该表项，LSR在向邻居通告路由时使用该表项中的入标签；如果找不到该表项，那么在向其他邻居通告路由时将选择一个没有使用的标签。

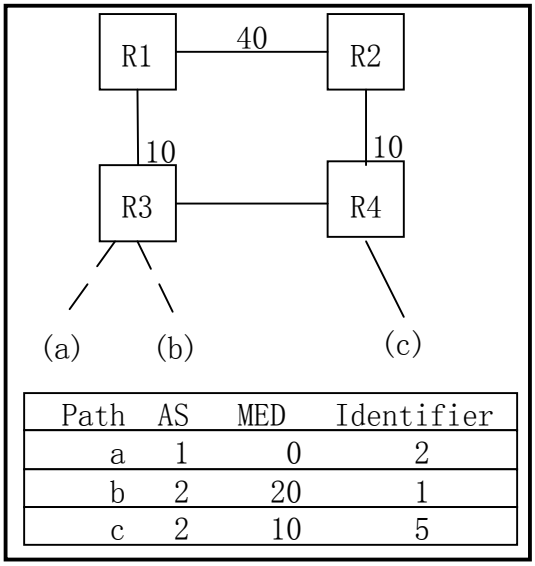
上段描述默认重起的LSR对于相同下一跳分配相同的标签。如果实际不是这种情况而且重起的LSR将为每条有相同下一跳的路由分配一个不同的标签，那么LSR在重起过程中不仅需要保留<入标签，（出标签，下一跳）>的映射，还要保留和该映射相关联的地址前缀。此时LSR将在MPLS转发状态中查找(a)标签弹出；（b）路由下一跳相同；(c)具有相同前缀的表项，如果找到了该表项LSR在向邻居通告时使用表项中的入标签；如果没有该表项，那么在通告路由时将选择一个没有使用的标签。

第三种情况，这种情况适用于重起的LSR没有将自己设置为BGP下一跳的情况。此时重起的LSR在通告特定NLRI的最佳路由时使用和路由一起接收的标签。如果没有标签和路由一起接收那么LSR在通告时也不带标签。

避免最佳路径迁移(Avoid BGP Best Path Transitions)

BGP有一个很重要的特性即是防止环路，但是在某些情况下环路的确存在而且无法避免，导致路由不断更新和变化始终无法收敛，具体可以参看《BGP MED Churn》。避免最佳路径迁移的RFC刚形成不久，其目的是：“The proposed extension to the BGP route selection rules avoids unnecessary best-path transitions between external paths under certain conditions. Clearly, the extension would help reduce routing and forwarding changes in a network, thus helping the overall network stability.More importantly, as shown in the following example, the proposed extension can be used to eliminate certain BGP route oscillations in which more than one external path from one BGP speaker contributes to the churn. Note however, that there are permanent BGP route oscillation scenarios [RFC3345] that the mechanism described in this document does not eliminate.”

实际上通过5001提出的方法可以防止环路的出现。具体方法见下图：



Path	AS	MED	Identifier
a	1	0	2
b	2	20	1
c	2	10	5

图1 避免最佳路径迁移示意图

假设路由c不存在，在R3上面应该是选择B（因为默认EBGP路由不比MED值，所以选择BGP ID小的即B）；当C存在的时候在R3上面应该是选A（因为C比B优，但是A又比C优），所以C撤销的话触发A切换到B，引起最佳路径切换。一旦R3上面比较两个EBGP路由选定A为最佳路由，则该路由被设定为最佳路径而且不再改变，这样即使C撤销或者反复震荡也不会影响R3设备AS域间路由的变化，提高了网络设备稳定性。

AS稳定迁移

H3C设备提供了AS稳定迁移功能。在实际应用中，运行BGP的路由器一般情况下是只能属于一个AS，但是可能由于某种原因该AS可能需要迁移或者和其他AS进行合并，H3C的BGP命令FAKE-AS就是用来解决AS迁移过程所遇到的麻烦。该特性具体介绍可以参看章节《BGP新特性》。

动态更新邻居能力

传统BGP在变更相关能力配置的时候，需要断掉邻居关系重新建立。举个例子：一台正在转发数据的BGP路由器，由于需要提供VPLS能力，所以需要配置上相关的能力地址族。这样必然导致BGP邻居重新建立，从而引发数据丢失、整网路由震荡、路由重新学习等问题。动态更新邻居能力这个特性，可以在配置新的能力地址族的时候发送新的OPEN报文，同时邻居动态地把新

增加的能力记录下来。这样可以保证在邻居关系不会重新建立前提下，提供了更多其他业务服务，提高了转发稳定性。

这个特性对保证网络稳定性很有效果，在《draft-ietf-idr-dynamic-cap-05.txt》中提到：“This document defines a new BGP capability termed "Dynamic Capability", which would allow the dynamic update of capabilities over an established BGP session. This capability would facilitate non-disruptive capability changes by BGP speakers.”为此定义了一种新型BGP报文，type为6，其内容如下：

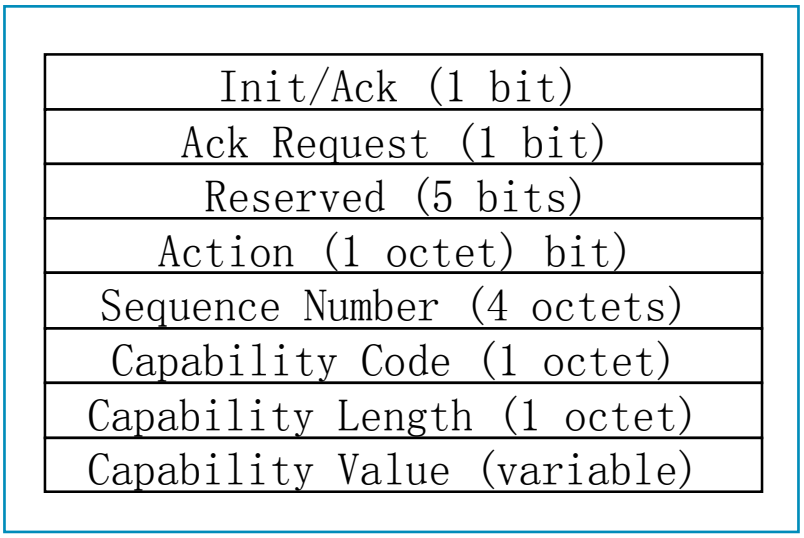


图2 动态能力字段编码格式

安全性

先看一个小故事，某年2月24日，Google You Tube视频服务意外中断了两个小时。原因起始于巴基斯坦，该国的ISP（互联网服务供应商）决定切断对You Tube的访问，在网络上发布“错误”的BGP路由，即开始发布自己是You Tube 所属208.65.15.3.0网络空间256个地址的正确BGP路径。

由于这样的BGP路由远比正宗You Tube网站发布的还要详细，根据即时网络流量监控公司Renesys的时间表，亚太地区的网络服务商在15秒内就开始将You Tube的访问导向这家巴基斯坦的ISP，而其他地区的路由器也在45秒后就开始跟进。很快，这些数据同时也通过网络传送给了

中国香港的ISP服务商PCCW。后者又通过Internet把这些数据传播给了其他ISP。一场大范围的You Tube访问中断就此出现。

那么，为何全球的路由器都会“误入歧途”？直接原因是PCCW没有检验来自他们客户的BGP数据，而最值得人们关注的则是，像Google在防止这种问题上仍然无能为力。Arbor Networks公司首席研究官Danny McPherson说：“他们不能阻止Internet上的某人发布他们的地址空间，这是个巨大的安全漏洞。”这是一起典型的非法路由注入事件。

目前为止，各类互联网服务商只能事先“相信”其他服务商不会刻意捣乱或者去拦截他人的互联网地址。而一旦有类似的情况发生，则可以采用人工介入加以修正，比如这次事件中，You Tube就通过随后加入更精确的广播，从而给路由器一个正确的导向。但是考虑到这种错误广播的出现次数可能还会增多，研究人员已经开始建议设立一种拦截警示程序，当互联网地址的虚拟地址变更时，网络供应商可自动获得通知。此外，这类事件也有望让人们更加重视Secure BGP这类技术，它采用加密方式来确认哪些网络供应商拥有网络地址并有权广播。问题是，这一技术虽然早在1998年提出，但被认为复杂度太高，要真正采用可能还需要做很多工作。

事实针对BGP弱点的攻击形式很多，在《draft-convery-bgpattack-00.txt》中有一些介绍，比如针对MD5 (RFC 2385) Attacks、建立未经授权的BGP连接、发布和注入未经授权的BGP路由、发送欺骗性质或者非法的BGP消息、影响BGP连接建立的TCP报文攻击等。还有不少潜在的风险，比如路由震荡、聚合路由、BGP团体属性等都有可能成为被攻击的地方。下面简单介绍一些新的安全方面的进展：

TTL安全检测机制

TTL安全检测机制主要是用来防止多跳攻击。我们知道BGP存在2种邻居关系：内部邻居关系（IBGP），外部邻居关系（EBGP）。建立这两种邻居关系的时候IBGP是不检测TTL的，EBGP在缺省是发送个TTL等于1的协议报文。如果RA与RB开启了TTL安全检测功能，设定从RB转发给RA的BGP协议报文的TTL必须为255，同时RA拒绝接受任何TTL比254小的BGP报文。那么即使存在攻击者其报文TTL最大为255，转发给RA的时候TTL必然减2，这样即使RA接受到一个TTL=253的协议报文也会自动丢弃，从而达到防攻击的效果。

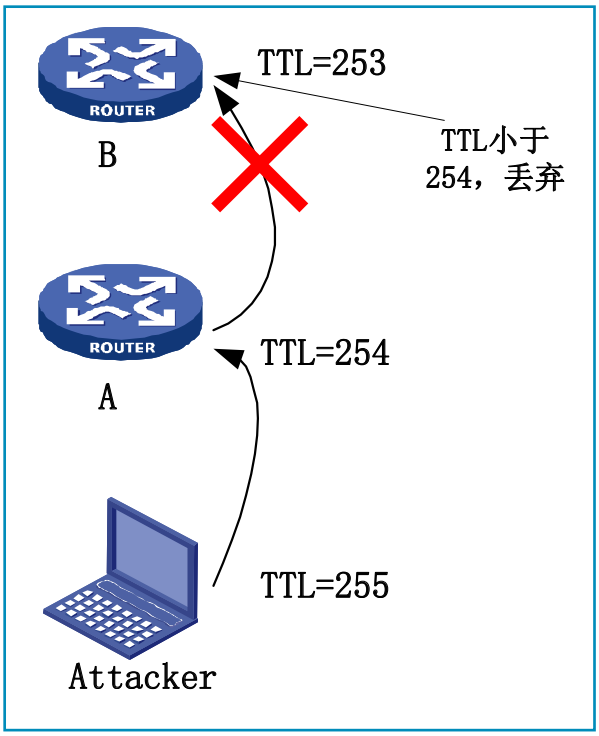


图3 TTL安全检测

当网络存在安全威胁的时候，通过TTL安全检测可以防止潜在的安全隐患。RFC3682（The Generalized TTL Security Mechanism (GTSM)）详细描述了这种机制。

攻击流量分析隧道技术

《draft-turk-ertb-00.txt》文档描述一种使用BGP团体属性进行远程触发特定目的网络的黑洞路由技术，黑洞路由可以在选择的一些BGP路由器中应用，而不需要在网络中所有的BGP路由器中应用；同时还提出了利用BGP团体属性的“攻击流量分析隧道”（“sinkhole tunnel”，本义是污水池隧道技术），该技术可以将网络中的流量牵引到一个指定路由器中进行分析。

当前的远程黑洞技术依赖于将曾经经历过某种异常（包含攻击）的一些目的网络地址在BGP中通告，该通告是由BGP域中的一台路由器来完成，在通告中将这些网络地址的下一跳修改并将它指向在RFC1918中指定的私有地址范围内的一个地址（10.0.0.0~10.255.255.255，172.16.0.0~172.31.255.255，192.168.0.0~192.168.255.255）而在Internet中的大多数路由器尤其是边界路由器都会配置将上述地址的下一跳指向接口为null0的静态路由。BGP speaker在BGP speaker在收到上述通告后在自己的路由表中将安装该目的网段的路由并将下一跳指向私有地址

范围内的一个地址，路由器在执行路由查找时将决定以这些私有地址为目的地址的数据包转发到什么接口，由于在路由器中有一条静态路由将这些地址为目的的路由指向null接口，因此这些流量将被丢弃因此攻击者对所通告的网络不可达从而不能攻击。

这项技术对网络基础设施减轻了攻击流量的负担，另一方面，上述网段在BGP运行的区域是不能运行的即使某个BGP speaker没有将RFC1918地址指向null接口，修改下一跳将导致其合法的目的地不可达，当然大部分ISP不会将这些黑洞在所有时间内存在，他们仅在一段时间内将黑洞打开将进入网络的所有路由都丢弃，依靠大部分路由器都有对丢弃的流量报文给源地址发送ICMP不可达报文的配置，可以将这些ICMP报文发到某一个特定地址上来搜集这些ICMP不可达报文。从这些报文中将可以找出上述被丢弃流量是从那些边界路由器进入网络的，然后运营者将选择停止从那些路由器进来的流量。

ISP有几种方法来减少攻击对网络的冲击，以上述将受攻击目标网络引入黑洞并将相应的ICMP不可达报文引出来从而确定攻击流量的入口等措施开始，然后隔离相应接口和对等体网络，最后安装ACL，限速策略或将这些流量转发到null接口。其他的技术可以用一种可以用一种技术识别DOS攻击并识别攻击流量从那些端口进入然后利用Netflow，这样比较省时省力。这些技术都依赖在特定路由器上手工制止攻击流量。

本文档提出一种远程触发可选择的一些路由器将以受攻击网络为目的地址的流量转发到null接口或者将这些流量转发到“攻击流量分析隧道”。该技术不使用关于攻击流量的ACL或限速策略，也不将攻击流量的下一跳地址修改为RFC1918地址。它仅仅通过边界路由器的BGP的团体属性改变路由选择。

首先ISP需要为每个可能成为网络攻击流量入口的边缘路由器分配一个唯一的团体值。以一个包含两个边界路由器R1和R2的小ISP网络为例，假定AS为65001，ISP可以为R1分配团体值为65001:1，为R2分配团体值为65001:2，为R1和R2全体分配团体值65001:666，然后在边界路由器上进行如下操作：

1. 在R1 R2上配置将RFC1918地址指向null接口的静态路由；
2. 配置匹配BGP本地产生的网络前缀的AS-Path访问列表；
3. 配置匹配ISP为本路由器分配的BGP团体访问列表(比如65001:1 for R1)；

4. 配置匹配ISP为所有路由器分配的BGP团体访问列表(比如65001:666 for R1and R2)；

5. 在BGP进程下IBGP输入路由策略将应用于如下逻辑操作如下步骤按照逻辑与的顺序：

a..允许通过如下匹配的路由；

I. 与特定路由器的团体值匹配（比如65001:1, for R1）；

II.与本地产生的BGP通告路由的AS-PATH匹配；

III.将BGP路由的下一跳设置成RFC 1918地址；

IV.将BGP的团体属性修改成no-advertise；

b. 允许通过如下匹配的路由

I. 与所有路由器的团体值匹配（比如65001:666, for R1和R2）；

II.与本地产生的BGP通告路由的AS-PATH匹配；

III.将BGP路由的下一跳设置成RFC 1918地址；

IV.将BGP的团体属性修改成no-advertise；

这些策略在R1和R2上配置后，ISP在受到攻击的情况下，在BGP中通告受攻击网络同时带上引入攻击的路由器的团体值，并保留其实际下一跳，IBGP将该路由通告到AS内所有路由器，除了与这个团体值匹配的路由器外，其他路由器将忽略该团体值并在路由表中安装带有合法下一跳地址的该路由，而与该团体值匹配的路由器将安装该路由并将其下一跳修改为RFC1918地址，然后将他转到null接口，匹配本地通告的路由是保证EBGP用户不会错误地使用该团体值，从而将该网段的数据包引入null接口。

该技术在标识为攻击流量转发的路由器上停止转发到合法目的网段的流量，因此网络中的其他到达合法目的网络地址的流量将不受影响。

“攻击流量分析隧道”进一步发展该增强的远端触发黑洞路由技术，有必要观察这些攻击路由以便将来分析。该需求增加了复杂性，通常在广播接口在遍历端口（spanned port）通过安装网络监听软件将流量输出分析；另一种方法是发送一个包含攻击主机地址的网络地址到BGP域，将下一跳地址改变为攻击流量分析设备。进行记录并且分析。

当需要记录攻击网络地址并进行数据包级别的分析时，攻击流量隧道的概念应运而生。这个概念的思想是当流量从隧道的一端进入后将会从另一端出来，这个概念在流量转发时下一跳地址没有改变时有实际意义。

首先攻击流量分析路由器sinkhole router和网络监听分析sniffers工具连接在一起，这些将所有的可能从其他AS引入数据包的边界路由器到sinkhole router之间的隧道可以通过比如MPLS TE来建立。这样允许利用团体值的技术来将目标网段的下一跳改变成一系列的/30子网（该子网的两个地址连接隧道的两端），换言之，边界路由器将下一跳变成隧道另一端的sinkhole router，AS内的其他路由器对于通告中预先设定的团体值忽略；由于改变路由匹配在其他地方不存在，如果合法的流量从网络的其他部分进入AS，其下一跳不会被改变攻击流量在sinkhole被终结。如果需求不要求中断流量而是在分析后重新送回到流量的目的地，那么流量将被重新送回原来网络，路由协议要保证将数据包重新送到原来的地址中。

目前相关技术有RFC文档3882（Configuring BGP to Block Denial-of-Service Attacks），通过配置BGP来防止DOS攻击，作者是同一人，感兴趣可以看看。

BGP over IPSEC

IPSec（IP Security）是IETF制定的三层隧道加密协议，它为Internet上传输的数据提供了高质量的、可互操作的、基于密码学的安全保证。特定的通信方之间在IP层通过加密与数据源认证等方式，提供了以下的安全服务：数据机密性（Confidentiality）、数据完整性（Data Integrity）、数据来源认证（Data Authentication）、防重放（Anti-Replay）。可以通过IKE（Internet Key Exchange，因特网密钥交换协议）为IPSec提供自动协商交换密钥、建立和维护安全联盟的服务，以简化IPSec的使用和管理。IKE协商并不是必须的，IPSec所使用的策略和算法等也可以手工协商。

实际上IPSEC的应用已经非常成熟，目前解决BGP安全问题的一个方法就是利用IPSEC加密BGP报文，保证数据的机密性等。实际上这种技术不算什么新的技术，虽然能从一定程度上防范未经过认证BGP报文的攻击，但是BGP攻击的方式很多，比如破坏TCP连接的TCP RST报文、带有欺骗性质的路由更新报文等，BGP over IPSEC对于欺骗性质报文的防范存在很多的困难。

入口和出口策略过滤

根据draft-ietf-idr-route-filter-06.txt（Cooperative Route Filtering Capability for BGP-4）中描述到：在目前的BGP实现中一般由BGP speaker接收路由然后根据本地的路由策略将一些不需要的路由过滤掉。考虑到发送方路由的产生发送和更新以及接收方处理路由更新均需要消耗资源。文档定义了一种基于BGP的机制允许BGP speaker给对等体发送一系列输出路

由过滤Outbound Route Filters (ORF)。对等体将应用这些过滤条件和对等体自己配置的过滤条件共同过滤要发送的路由，同时加强了安全控制。

ORF定义

ORF项目由<AFI/SAFI, ORF-Type, Action, Match, ORF-value> 一个ORF可以由一个或多个具有相同的<AFI/SAFI, ORF-Type>的ORF项目组成；Action控制对等体处理ORF Request的动作可以是ADD, REMOVE,REMOVE-ALL；Match为PERMIT或者DENY；

团体ORF-Type

团体ORF-Type允许用BGP团体属性来表示ORF，也就是说团体ORF-Type提供基于团体属性的路由过滤。团体ORF-Type由<Scope, Communities>组成。Scope表示对等体对于给定ORF request必须考虑的路由范围可以是EXACT或者NORMAL。EXACT表示让对等体仅仅考虑考虑路由的团体属性和ORF列表中给出的团体属性相同的部分；NORMAL表示让对等体可以考虑ORF列表中的团体属性列表的子集部分；

扩展团体ORF-Type

扩展团体ORF-Type允许用BGP扩展团体属性来表示ORF 也就是说扩展团体ORF-Type提供基于扩展团体属性的路由过滤扩展团体ORF-Type由<Scope, Communities>组成，Scope表示对等体对于给定ORF request必须考虑的路由范围可以是EXACT或者NORMAL；EXACT表示让对等体仅仅考虑考虑路由的扩展团体属性和ORF列表中给出的扩展团体属性相同的部分；NORMAL表示让对等体可以考虑ORF列表中的扩展团体属性列表的子集部分；

在BGP中携带ORF项

ORF项目是在BGP ROUTE-REFRESH消息中携带BGP speaker能够通过消息头带的长度中确定BGP ROUTE-REFRESH消息中是否携带了ORF项目。一个BGPROUTE-REFRESH消息可以携带多个ORF项目，只要这些项目的AFI/SAFI是相同的。从编码角度讲ORF包括公共部分和类型相关部分：公共部分由<AFI/SAFI, ORF-Type, Action, Match>组成

团体ORF-Type 类型相关部分

团体ORF-Type的ORF-Type值为2 由<Scope, Communities>组成，scope包括EXACT和NORMAL；

扩展团体ORF-Type 类型相关部分扩展团体ORF-Type的ORF-Type值为2，由<Scope, Extended Communities>组成scope包括EXACT和NORMAL。

ORF操作

能够从对等体接收ORF或者向对等体发送ORF的BGP speaker需要用BGP能力通告（RFC 2842）进行能力协商，需要实现ORF的BGP speaker需要支持BGP ROUTE-REFRESH消息（在RFC2918中定义）。BGP speaker在向对方通告ORF能力时不一定需要向对方通告BGP Route Refresh的能力。

BGP speaker向对等体通告ORF能力表示可以接受<AFI, SAFI, ORF-Type>，并且从对等体接收ORF能力表示对方希望给自己发送<AFI, SAFI, ORF-Type>。如果对于一个给定的<AFI, SAFI>两者的交叉部分不为空，BGP speaker在从对等体接收到关于该<AFI, SAFI>的任何ROUTE-REFRESH消息前，不需要向对等体通告关于该<AFI, SAFI>的路由。这些ROUTE-REFRESH可以不包含任何ORF项，或者带有一项或多项ORF When-to-refresh域设置成IMMEDIATE。如果两者的交叉部分为空时则遵循普通BGP过程。

AS路径和NLRI信息认证

在BGP draft< draft-ietf-rpsec-bgpsecrec-09.txt>文档中提出一系列的安全性要求，对目前的应用有着较好的指导作用，比如AS路径和NLRI信息认证等。使用该机制来验证接收的路由信息中承载的自治系统路径的有效性。目前在该领域的研究较多，而且该文中提到的其他方法也在研究之中。

针对特定前缀的路径信息的保护大致可以包括一下四方面的内容：

A）授权AS发布路由：路由前缀的所有者授权指定AS产生和发布其指定路由；

B）检查AS：要求从某对等体收到的Update报文中路由信息的AS-path属性中的第一个元素匹配该对等体设置的AS号；

C）检查AS路径可行性：AS-PATH列表符合as中规定策略中的有效列表；

D）路由更新信息传输检测：比C更严格的检查项，主要是检查通过该AS的路由更新消息能够满足某一特定的安全要求，避免路由非法注入等攻击。

扩展性

BGP这个协议本身已经不会有大的变动，由于其报文格式采用TLV编码，非常方便组合和修改。针对BGP的扩展应用很多，自从RFC2858（Multiprotocol Extensions for BGP-4，BGP-4多协议扩展，obsoletes RFC2283)发布以来就可以使BGP不仅仅具备IPv4能力，还能支持其他能力比如BGP4+、MPLS、Multicast等。同时一些新的能力协商也跟随着BGP的扩展应用不断加入进来。

4字节AS

自治域系统号(Autonomous System number，下文简称AS)是拥有同一选路策略，在同一技术管理部门下运行的一组路由器的集合。BGP的RFC里留给AS的范围是2个字节，所以AS的范围为1-65535，其中64512以上的为私有AS。但是鉴于IPv4地址空间不够这个前车之鉴，在RFC4893里记录了一个BGP的新功能——4字节AS（BGP Support for Four-octet AS Number一般用M.N来描述）。

由于BGP在邻居协商以及路由发送接受的时候都需要利用AS这个属性，所以RFC4893里也对相应的属性的扩展变化做出了解释。为了便于读者理解，下面列出了RFC4893定义的相关新的属性以及说明。由于该特性最大的变化是AS的变化，所以所有的属性扩展都是基于AS的相关属性，只是属性的TYPE值的变化，具体介绍可以参看章节《BGP新特性》。

MPLS L3VPN for IPv6

早在<draft-ietf-L3vpn-bgp-ipv6>已经提出IPv6 VPN的概念，目前该draft已经形成正式的RFC4659。在理论上IPv6 VPN和IPv4 VPN没有什么区别。与IPv4相似的是，MP-BGP是MPLS VPNv6的核心部件。它被用于在SP骨干网上分发IPv6路由，具有相同的重叠地址、再分发策略和扩展问题的处理方式。先介绍一些简单术语：

1）6VPE路由器——在基于IPv4的MPLS核心网络上提供BGP-MPLS IPv6 VPN服务的PE路由器。其基础是一个VPNv6 PE和双栈（IPv4+IPv6），双栈实现了面向核心接口的6PE概念。

2）VPNv6地址——一个VPNv6地址是一个24字节的标识符，以8个字节的路由区别符号（RD）开始并以16字节的IPv6地址结束。有时它被称为VPN IPv6地址。

3）VPNv6地址族——地址族标识符（AFI）定义了一个特殊网络层协议和子AFI（SAFI）提供的附加信息。AFI IPv6即SAFI VPN（AFI=2，SAFI=128）被认为是VPNv6地址族；

在IPv6 VPN中，尽管不希望在IPv6出现地址重叠问题，但地址仍旧使用RD来规划。同时定义一个新的网络层可达信息（NLRI）的三元组格式<长度，IPv6前缀，标签>，其目的是使用MP-BGP分发路由。一个VPNv6地址是一个16字节的IPv6地址，预先带有8个字节的RD，构成了24个字节的地址、如同IPv4一样，VPNv6前缀只在MP-BGP内有意义。

VRF概念对于熟悉MPLS的人来说是一个常用的概念，一个VRF定义为一个虚拟路由选择的转发表项，它绑定了一个私有的路由选择和转发表，即常提到的私网路由表。有人可能认为，既然IPv6带有自己的路由选择和转发表，那么应该有不同的IPv4和IPv6 VRF。实际上，尽管IPv4和IPv6路由选择表的确不同，但是从部署角度来看，共享一个VRF是非常方便的。目前可以看到的实现即是如此，在全局的VRF视图下增加IPv6地址族视图，这样一个VRF下面可以同时配置IPv4和IPv6的RD和route-target信息。总体来说，VPNv6的实现和IPv4较为类似，许多BGP特性比如反射、路由刷新、路由过滤、聚合等支持并无二样。

MPLS TE

当前IGP协议对于MPLS TE的支持比较完备，比如isis和ospf都已经有相关的成熟实现和RFC支撑。MPLS TE是使用MPLS技术来实现TE，首先收集TE关心的链路信息，然后使用IGP算法满足某条流的特定约束条件计算出一条路径，再通过其信令协议RSVP-TE或者CR-LDP建立lsp隧道。

针对EGP协议的应用暂时没有，实际相关研究也不多。目前BGP也没有正式的RFC提出BGP支持MPLS TE，不过笔者在IETF上面发现一篇draft（draft-ietf-softwire-bgp-te-attribute-00.txt），是基于GMPLS信令协议的。拿过来简单分析一下，有兴趣可以关注一下。该draft定义一种新的BGP属性（Traffic Engineering），该属性是可选非过渡属性。

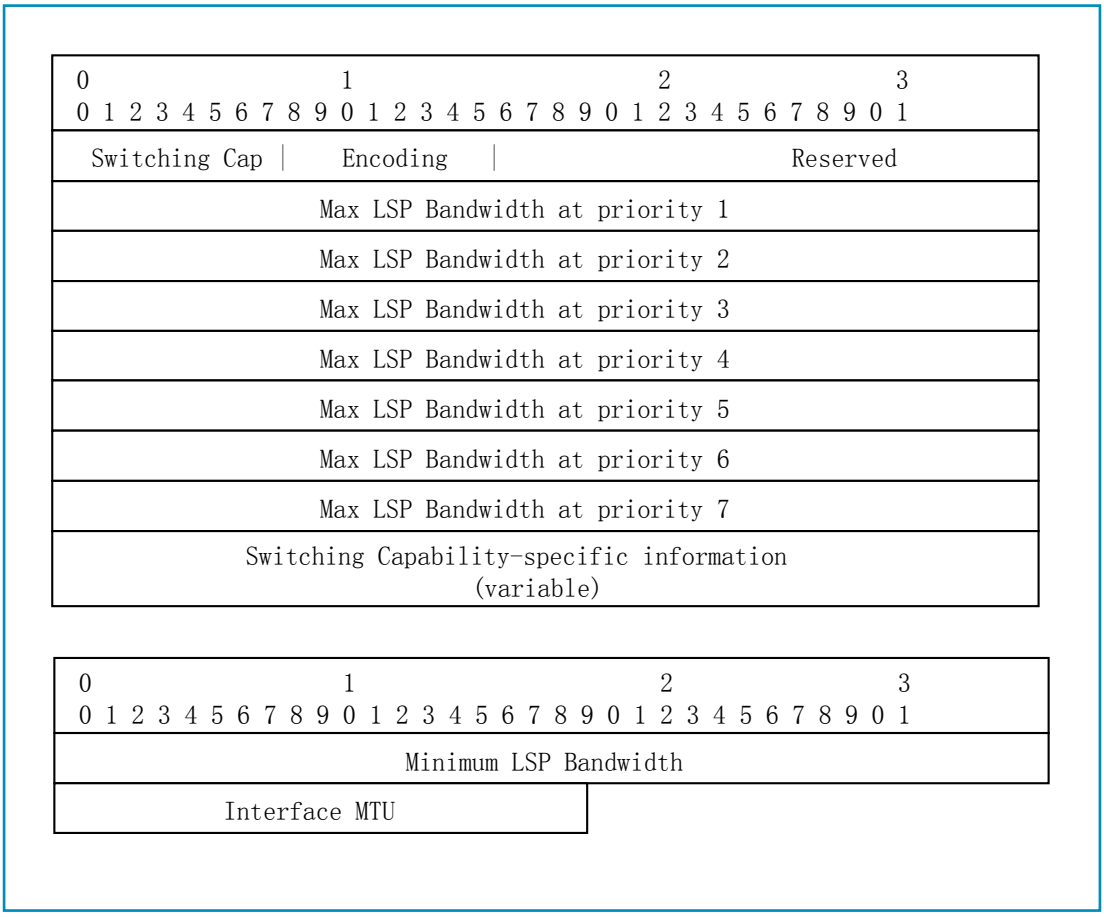


图4 TE属性字段编码格式

Switching Capability(Switching Cap)和Encoding字段的定义等同于(RFC3471,Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description) 中Section 3.1.1所描述的定义。Reserved字段为保留字段在传输过程中必须设置为，在接收端被忽略。

Switching Capability specific information字段的内容跟Switching Capability字段有关，当Switching Capability是PSC-1, PSC-2, PSC-3, or PSC-4时候，它由最小lsp带宽和接口mtu组成，见图5：

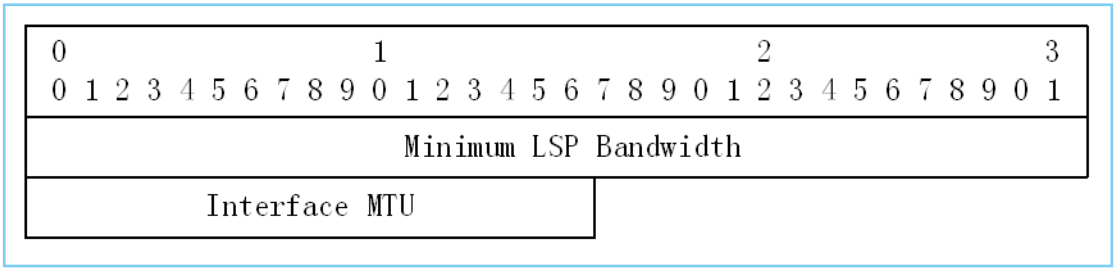


图5 Switching Capability属性字段编码格式

当Switching Capability是L2SC时候，则不带该字段。当Switching Capability是TDM时候，它由最小lsp带宽和接口标识（Indication）组成，Indication字段表示该接口是支持标准还是私有的SONET/SDH（1表示标准，0表示私有）。

总体来说，在BGP扩展性上面TE的研究不是很多而且暂时看不到巨大的需求。在BGP扩展的研究中，还有许多比较有新意的东东，比如draft-boucadair-qos-bgp-spec-01.txt、draft-jacquenet-qos-nlri-04.txt中对于QOS的研究等，不过目前进展不大，这里不深入讨论。

针对BGP的缺点和应用，目前提出的改进方法和研究很多，以上只是从四方面描述一下目前的发展。其中收敛速度、策略和安全是BGP亟待解决的问题，为BGP发展提供了发展方向参考。然而不断对BGP进行逐步改进也促使我们进行思考：是对BGP进行小的改变使它越来越复杂？还是承受所有部署问题而更换BGP协议？

BGP新特性

文/杨默寒

BGP（Border Gateway Protocol，边界网关协议）是目前因特网骨干网络使用的核心路由协议，也是部署最广泛的路由协议。在过去的几十年里，Internet的发展速度令人震惊，伴随着发展的同时，ISP或者企业网对自身网络有了新的要求，它们希望自身网络可以提供更良好的扩展性与服务质量。这些需求大部分是通过BGP协议的相关新特性来满足，所以本文着重介绍了这些新特性供读者参考。

FAKE-AS

运行BGP的路由器一般情况下是只能属于一个AS的，但是可能由于某种原因该AS可能需要迁移或者和其他AS进行合并，FAKE-AS就是用来解决AS迁移过程所遇到的麻烦。请看图1：

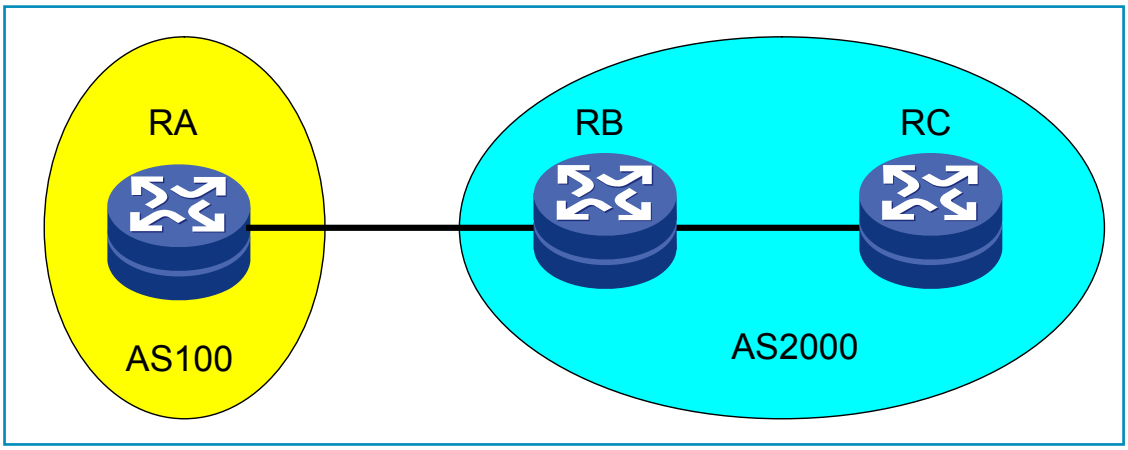


图1 FAKE-AS迁移前拓扑

如图1 RA属于AS100，RB属于AS200他们之间建立EBGP邻居。由于某些原因，可能AS200内的路由器要进行AS迁移，也就是说要修改自己的BGP配置使网络过渡成图2的状态：

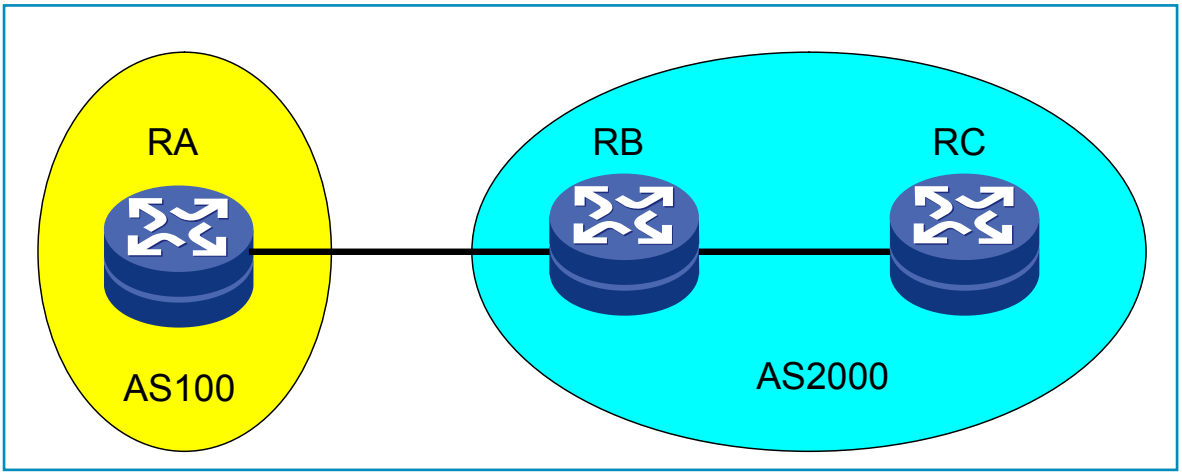


图2 FAKE-AS迁移后拓扑

AS200迁移为AS2000后如果RA也根据RB AS的变化做出相应的配置修改，那么没有问题。但是可能由于某种原因RA所在的AS组织并不想修改自己的配置，所以遇到这样的问题就需要FAKE-AS来解决。

由于RA不修改自己的配置所以至少需要解决以下2个问题：

- BGP在建立邻居的过程中会互相发送OPEN报文，要让RA认为RB发送过来的OPEN报文里面关于AS的记录是200，而不是2000；
- RB在发送给RA的路由，在AS PATH属性里面记录的第一个AS必须是200，而不是2000。

我们通过在RB上配置FAKE-AS功能可以完美的解决上述的2个问题，达到RB在迁移AS的时候RA完全不需要修改任何BGP相关配置。

条件通告

在实际环境中一台 BGP 路由器经常会保存多条相同的前缀，这样可以起到备份的作用，如图 3:

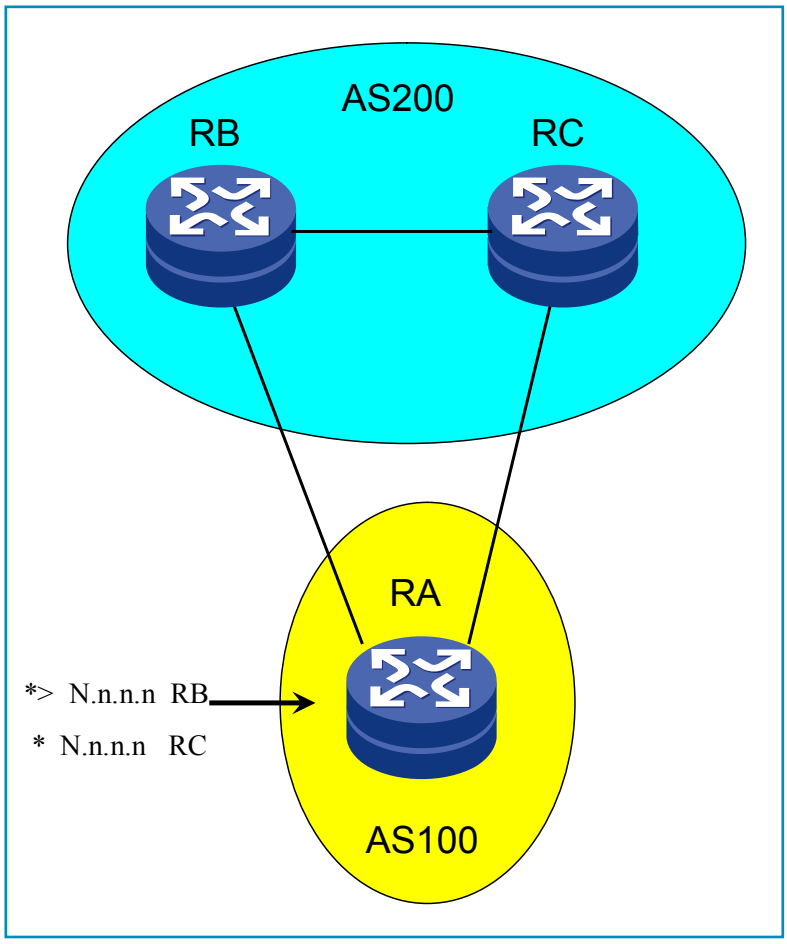


图3 条件通告功能描述拓扑图

RA通过AS200去访问目的为N.n.n.n的网络，由于有备份的需求RA与RB，RC都建立BGP邻居关系，并且优选了从RB来的路由，这样当RB出现故障或者RA与RB之间的链路出现故障，RA可以通过从RC学习到的路由访问网络N.n.n.n。

实际网络中RA需要访问的目的地址可能很多，按照图3所用的方法当RA需要访问10000个目的网段，那么需要存储20000个BGP路由，这样对路由器的内存是有一定要求的。当内存不足时可能会出现部分网络无法访问，或者BGP邻居关系重新建立。

条件通告这个功能就是解决上述问题的。下面具体介绍一下这个功能的处理流程：

- 首先在RA上配置一个LOOKBACK接口（假设为1.1.1.1/32），把该地址发布到BGP，并且只发送给RB；

- RC会通过与RB的IBGP邻居关系学习到该路由；
- 在RC上启用条件通告，判断条件为：是否存在1.1.1.1/32的BGP路由，如果存在则不发送任何路由给RA；如果不存在，发送路由给RA；
- 这样当网络正常的时候RA只有从RB学习到路由，当RB或者RA与RB之间的链路出现故障那么RA会只有从RC学习到的路由。

动态RT

最早在RFC2547里定义了MPLS-L3VPN的网络模型，该RFC具体描述了在MPLS-L3VPN里设备的角色以及数据封装传输的过程，以及BGP需要的相应扩展。

BGP为了支持MPLS-L3VPN，除了增加了VPNv4的能力之外，还在发送VPNv4路由的时候在路径属性里增加了扩展团体属性Route-Target(RT)。该属性的TYPE值为16属于可传递属性，该属性的作用是根据团体属性的值来决定是否接受某条VPNv4路由。请看图4：

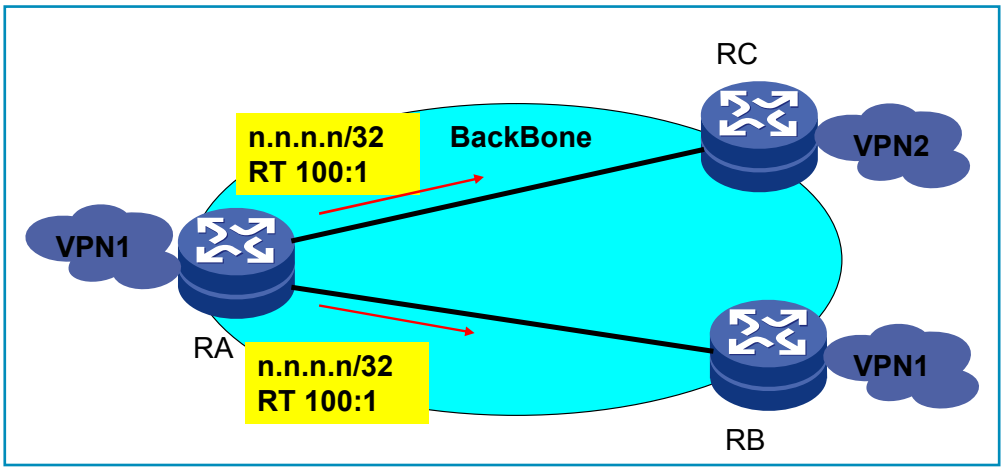


图4 未配置动态RT前拓扑

RA从VPN1学习到的路由以VPNV4的形式发送给RB，RC同时携带RT属性。在RB与RC上有相关VPN的配置，可能在开始部署的时候VPN2并不想接受VPN1的路由所以当路由N.n.n.n/32到达RC的时候，因为RT的值和本地的配置值不匹配，该路由被丢弃掉；当路由N.n.n.n/32到达RB的时候，因为RT的值和本地的配置值匹配，路由被接收。

可能会存在这么一个需求：VPN1虽然与VPN2是隔离的，但是VPN2需要访问部分VPN1的网络资源，那么解决办法有以下三个：

- VPN1重新配置相关的RT规则，缺点：所有路由的RT属性都被修改，可能都被VPN2接收

- VPN2与VPN1融合，缺点:无法完成隔离的需求，需要用ACL来隔离而且修改配置可能会引起路由震荡

- 利用动态RT的特性

我们主要讲解一下如何用动态RT这个特性来解决该问题，主要分为一下几个步骤：

- RA发送路由的时候，可以利用动态RT的特性来区分发送的路由，不同的路由附加上不同的RT组，如图5（假设VPN2需要访问VPN1内部的M.m.m.m/32服务）；

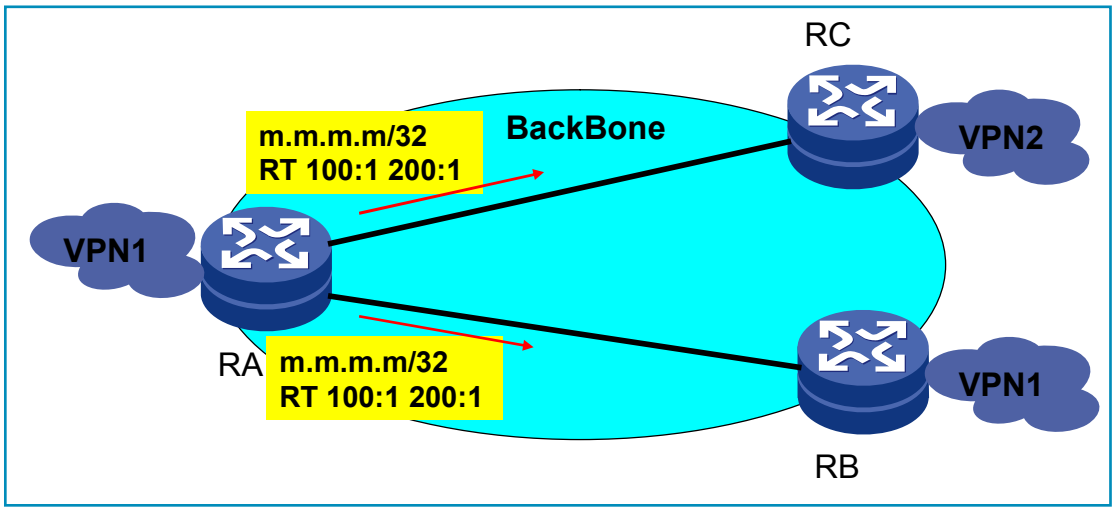


图5 配置动态RT后拓扑

- 动态RT对M.m.m.m/32这条路由附加了两个RT属性：100:1 200:1由于VPN2默认配置是可以接受200:1的路由的，所以这条路由会被收到VPN2的路由表。对于其他路由由于动态RT对其不感兴趣，所以不做修改依然只有100:1这一个RT值；

- 在RB上的处理和以前一样，只要路由的RT列表里面存在100:1就可以被接受；

邻居动态建立

熟悉BGP配置的读者可能知道在配置BGP的时候需要指定邻居的AS和邻居的具体地址，这主要是为了保证邻居建立的可靠性和安全性。但是随着BGP的广泛部署，可能有时候不能确定邻居的具体地址，或者需要建立多少个邻居，因为这些邻居可能是通过ADSL或者DHCP等动态方式动态获得的IP地址，尤其在Hub&Spoke组网这种问题更为常见。

邻居动态建立特性就是为了解决该问题，该特性解决的问题以及注意事项如下：

- 至少需要有一个邻居设备有固定的IP地址和AS号码，我们称它为RA；

- RA上需要配置一个邻居的地址池，来确定邻居的IP地址范围；
- RA上需要配置一个AS的号码池，来确定邻居的AS范围；
- 当建立邻居的时候RA需要检测邻居的IP地址与AS号码是否在相应的范围内；
- 邻居的安全可以通过TCP的MD5保障；
- 实现上需要把RA在BGP的状态机停留在ACTIVE比较合理。

TTL安全检测

BGP存在2种邻居关系：内部邻居关系（IBGP），外部邻居关系（EBGP）。建立这两种邻居关系的时候IBGP是不检测TTL的，EBGP在缺省是发送个TTL等于1的协议报文。但是当网络存在安全威胁的时候，通过TTL安全检测可以防止潜在的安全隐患。请看图6：

图6 TTL安全检测功能描述拓扑

RA和RB建立EBGP邻居关系，默认RB发送给RA的TTL为1。如果存在攻击者模拟RB发送的数据包，可能会影响RA的性能甚至RA的路由选路。如果RA与RB开启了TTL安全检测功能，那么RA可以设定RB发过来的BGP协议报文的TTL必须为255。那么即使存在攻击者，当攻击数据被RB转发给RA的时候TTL必然减1，这样RA接受到一个TTL非255的协议报文会自动丢弃，从而达到防攻击的效果。

4字节AS

AS （Autonomous System number，自治域系统号）是拥有同一选路策略，在同一技术管理部门下运行的一组路由器的集合。BGP的RFC里留给AS的范围是2个字节，所以AS的范围为1-65535，其中64512以上的为私有AS。但是鉴于IPv4地址空间不够这个前车之鉴，在RFC4893里记录了一个BGP的新功能——4字节AS（BGP Support for Four-octet AS Number，一般用M.N来描述）。

由于BGP在邻居协商以及路由发送接受的时候都需要利用AS这个属性，所以RFC4893里也对

相应的属性的扩展变化做出了解释。为了便于读者理解，下面列出了RFC4893定义的相关新的属性以及说明。由于该特性最大的变化是AS的变化，所以所有的属性扩展都是基于AS的相关属性，只是属性的TYPE值的变化。

- AS4_SEQUENCE：记录了该路由传递过程中所经过的AS；
- AS4_SET：当出现聚合的时候，记录了聚合路由所合并的AS；
- AS4_CONFED_SEQUENCE：在联盟中使用，作用和AS4_SEQUENCE类似；
- AS4_CONFED_SET：在联盟中使用，作用和AS4_SET类似；
- AS4_AGGREGATOR：记录了聚合者的AS号码；

细心的读者可以发现上面所描述的属性与2字节的AS_PATH相关属性具有相同的作用，所以对于熟悉BGP的使用者来说很好理解。接下来我们会讨论一下在网络迁移的过程中存在部分路由器只支持2字节AS的时候的过渡方法，请看图7：

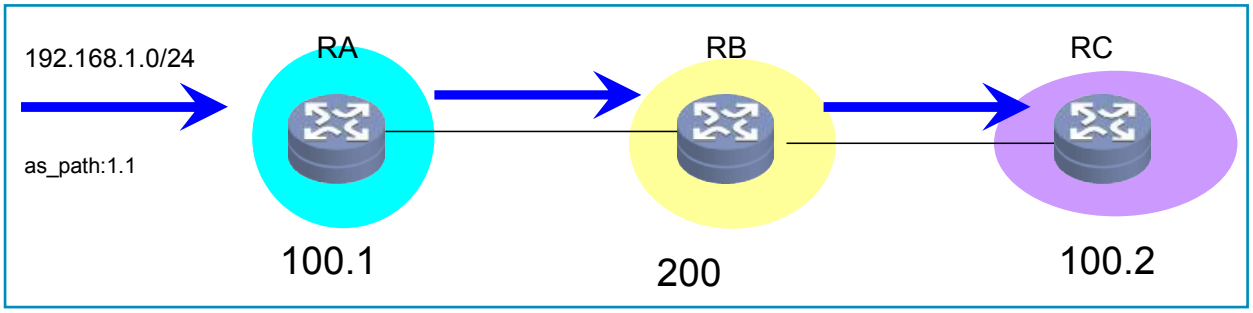


图7 迁移组网拓扑

RA与RC为支持4字节的AS功能的路由器，RB为只支持2字节的AS功能的路由器。如果出现上图这种情况，我们需要解决以下这些问题：

- RFC4893给了一种建议，里面定义了一个公用2字节AS号码AS_TRANS,也就是说需要一个单独的2字节AS为专门用于衔接4字节BGP路由器与2字节BGP路由器，并且AS_TRANS不能被其他路由器或者组织使用；
- 如上图RA收到一条四字节AS的路由,AS号码为1.1；
- RA与RB建立邻居，需要令RB认为RA的AS号为AS_TRANS；
- RA发送路由给RB的时候把AS_TRANS记录在AS_SEQUENCE里面，把1.1与自己的AS号码100.1按照BGP要求的顺序记录在AS4_SEQUENCE；
- RB对于不识别的属性AS4_SEQUENCE不作处理依然保留，它只按照BGP的规则来发送路由

给RC。当然它认为RC的AS号码也是AS_TRANS，这样路由发送的过程如图8（我们假设AS_TRANS为23456）：

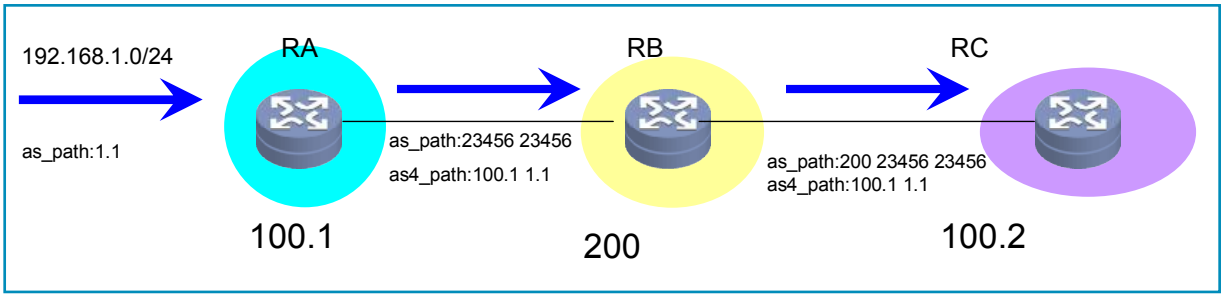


图8 路由发布说明

- 这样当RC收到从RB来的路由会把AS_TRANS按照顺序来替换为AS4_SEQUENCE里所记录的相应的地址，在RC上把AS4_PATH属性还原为0.200 100.1 1.1。

有些更复杂的情况是关于聚合的时候4字节AS与2字节AS相关的聚合属性如何处理，本文就不作过多描述了，有兴趣的读者可以参看RFC4893里面的相关章节。

正则表达式过滤团体属性

团体属性是一个可选传递的属性，具体定义在RFC1997。该属性为一个32BIT的字段可以记录在BGP路由的路径属性当中，按照RFC的定义从0x0000000到0x0000FFFF和从0xFFFF0000到0xFFFFFFFF是被预留出来的，其他部分都可以被网络规划者所使用。

目前，网络规划者和路由器厂商都习惯把该属性分为2部分：前16位一部分，后16位一部分。因为我们知道AS号的范围是2个字节，所以可以用前16位来代表AS，后16位一般位用户自定义的数值。比如一个常见的团体属性表达方法可能是1111:1。团体属性在过滤路由的时候有着自己独有优势，同时也可以根据用户的规划使管理者可以迅速的知道某些路由的特殊用途。用户可以定义一些业务与团体属性的后16位对应，比如在一个SP网络里定义101代表WWW服务器路由，102代表MAIL服务器路由，103代表VOIP设备路由。那么在AS100发布这些路由的时候，可能就需要对相应的路由设置上相应的团体属性（100:101,100:102,100:103）。当然，对于其他AS来说，可能会不需要VOIP的服务，所以在AS边界可以通过对团体属性的过滤完成（deny 100:103）。

上述方法在AS数量比较少的时候是可行的。当存在大量AS并且都按照相同的规则发布路由，如果依然根据刚才的方法过滤路由，对于配置者来说可能是一个挑战。所以可以利用正则表达式过滤团体属性这个功能来完成。我们假设，要过滤所有的MAIL服务的路由，那么只需要配置deny .*:102即可。关于正则表达式的具体用法，可以参考本刊的相关章节。

保存多条相同前缀的标签路由

“保存多条相同前缀的标签路由”这个特性从文字上理解不是很好理解，我们还是从实际网络中遇到的问题来具体说明这个特性的作用，请看图9：

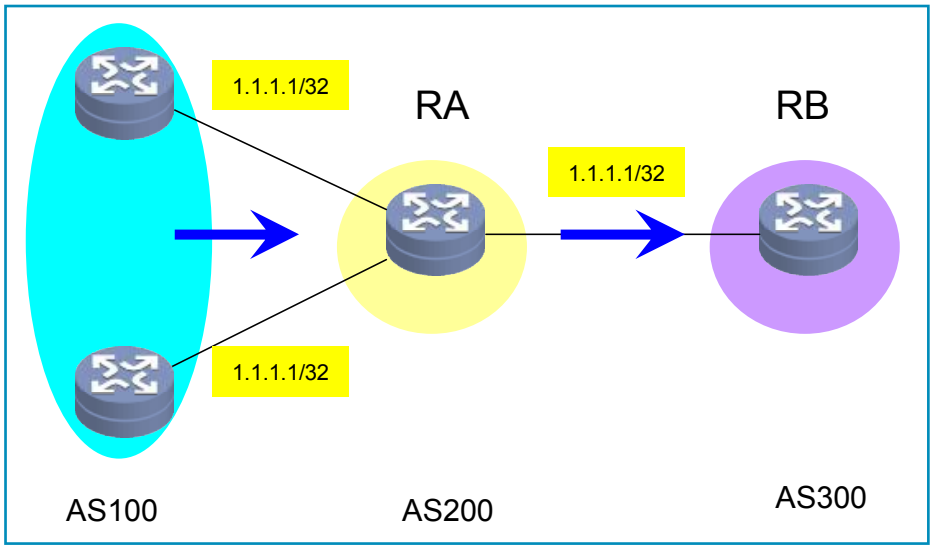


图9 路由发布的方向

RA从AS100的两个EBGP邻居收到了到1.1.1.1/32的路由。如果RA上配置了负载分担功能，那么RA访问1.1.1.1/32的时候可以同时利用两条路径。当RA把路由发送给RB的时候，虽然RA的BGP路由表里有两条最优的BGP路由，但是RA会认为他们是同一条路由。所以RA只会发送一条路由给RB。当RB需要访问1.1.1.1/32的时候，会先把数据发给RA，再由RA负载分担。这样依然可以到达有效利用带宽的效果。但是，当网络需要进行MPLS转发的时候，问题可能就不这么简单了。我们以标签路由为例（该标签路由可以是IPv4路由，也可以是VPNv4路由），请看图10：

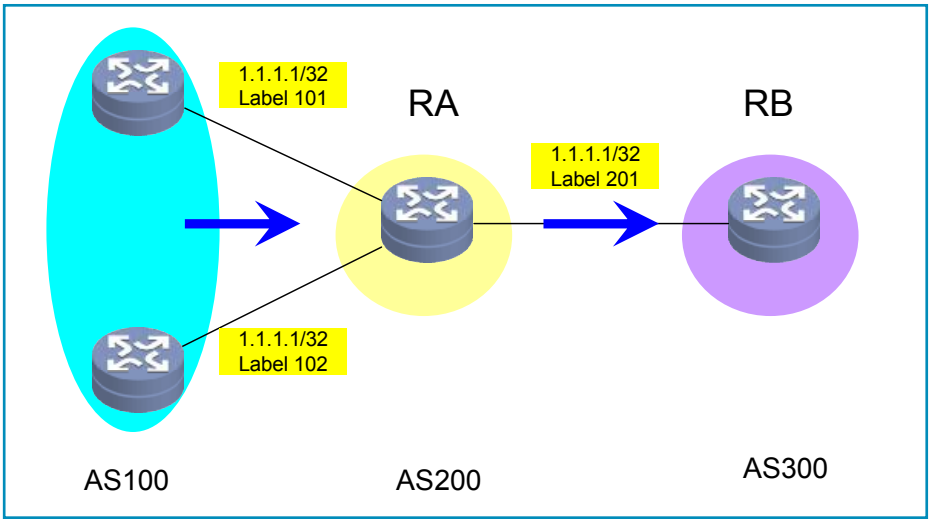


图10 未支持保存多条相同前缀的标签路由功能前路由发布说明

RA依然从AS100的两个EBGP邻居学习到了1.1.1.1/32的路由，但是有一点小变化，这条路由是按照RFC3107所定义的标签路由。这样当RA访问1.1.1.1/32的网络时候，不会再走IP转发，而是走MPLS转发。如果RA上配置了负载分担功能，依然可以实现负载分担，这个和IP转发的情况一样。但是RB可能会有点小麻烦，因为RA仍然只会发送一条路由给RB。这样当RB访问1.1.1.1/32的时候也是先走MPLS转发。当数据到RA的时候，RA根据入标签查找出标签，也就是MPLS转发表（注意不再是IP转发表！）。然而很遗憾在RA上的MPLS转发表只有两种可能，分别是：

表1 标签映射1

OUT LABEL	IN LABEL
101	201

表2 标签映射2

OUT LABEL	IN LABEL
102	201

或者

这样当RB访问1.1.1.1/32的时候，在RA上无法实现负载分担。

“保存多条相同前缀的标签路由”就是为了解决这种问题，该特性记录在RFC3107后半部分，属于BGP的一种新能力，代码为4。RFC上原文的描述如下：“A BGP speaker that is capable of handling multiple routes to a destination (as described above) should use the Capabilities Optional Parameter, as defined in [BGP-CAP], to inform its peers about this capability. The value of this capability is 4.” [1]

该特性要求，BGP在区分路由的时候，不能只根据前缀判断还要根据标签值。我们可以重新回顾以下刚才的那个问题,见图11.

当RA与RB都支持该特性，RA在发送路由给RB时认为1.1.1.1/32 LABEL 101与1.1.1.1/32 LABEL 102是两条不同的路由，所以都会发送给RB。同时RB也认为这两条路由虽然前缀相同但标

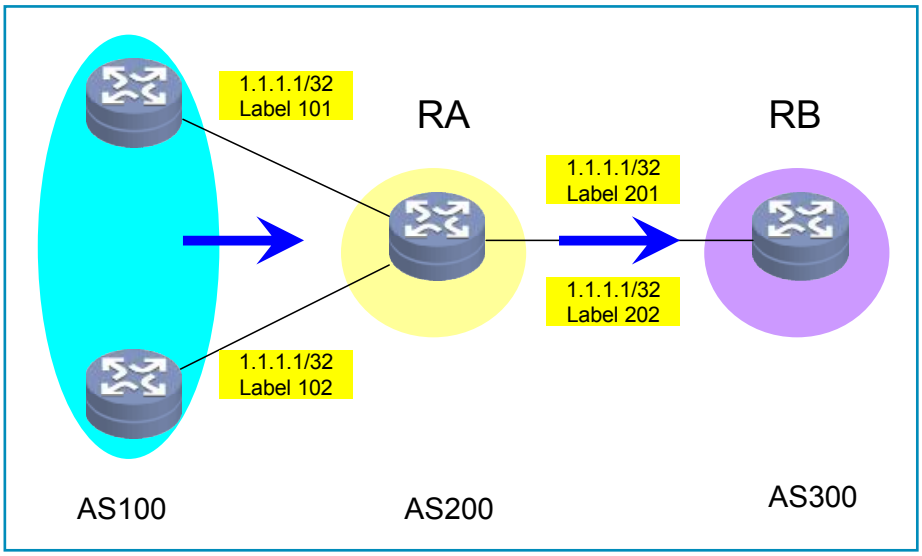


图11 支持保存多条相同前缀的标签路由功能前路由发布说明

签值不一样，所以不是同一条路由。当RB启动负载分担的功能后，访问1.1.1.1/32时可以封装上不同的标签（201和202）然后发送数据给RA。RA接受到数据时，根据入标签查找出标签。这个时候RA上的MPLS转发表如下：

表3 标签映射3

OUT LABEL	IN LABEL
101	201
102	202

通过“保存多条相同前缀的标签路由”特性，可以使得网络带宽更有效的被利用。最后需要说明一点，BGP在发送路由撤销的时候是不携带任何属性的。但是启用这个特性后，需要在撤销路由的时候明确标签值，来告诉邻居具体要撤销掉哪条路由。

动态更新邻居能力

传统BGP在变更相关能力配置的时候，需要断掉邻居关系重新建立。举个例子：一台正在转发数据的BGP路由器，由于需要提供VPLS能力，所以需要配置上相关的能力地址族。这样导致了和其他BGP邻居关系的重新建立，这样必然会导致转发数据的丢失、整网路由的震荡等、路由的重新学习等问题。动态更新邻居能力这个特性，可以在配置新的能力地址族的时候发送新的OPEN报文，同时邻居动态地把新增加的能力记录下来。这样可以保证在邻居关系不会重新建立前提下，提供了更多其他业务服务。这个特性对保证网络稳定性很有效果，但是目前该特性还无具体的RFC。

[缩略语]

缩写	英文原文	中文释义
AS	Autonomous System	自治系统
BGP	Border Gateway Protocol	边界网关协议
EBGP	Exterior Border Gateway Protocol	外部边界网关协议
EGP	Exterior Gateway Protocol	外部网关协议
GR	Graceful Restart	优雅重启动
IBGP	Interior Border Gateway Protocol	内部边界网关协议
IGP	Interior Gateway Protocol	内部网关协议
L3VPN	Layer 3 VPN	三层VPN
MBGP	Multiprotocol extension Border Gateway Protocol	多协议扩展边界网关协议
MED	MULTI_EXIT_DISC	多出口鉴别
MPLS	Multiple Protocol Label Switch	多协议标签交换
NLRI	Network Layer Reachability Information	网络层可达性信息
RR	Route Reflector	路由反射器
VPN	Virtual Private Network	虚拟私有网

H3C
IToIP Solutions Expert

测试中心
TESTCENTER

主办单位：H3C测试中心
策　　划：刘　宇　　陈旭盛　　杜祥宇
主　　编：陆宇翔
编　　委：张宇弟　　贾欣武　　程锋章　　姜杏春　　陈　磊
　　朱　皓　　杨默寒　　许　亮　　叶　翀　　孙　丽
　　高国义　　智晓彦　　曹　霞　　杜一鸣

吾生也有涯 而知也无涯
以有涯随无涯 殆已

——《庄子·养生主》

ROUTE TO NETWORK　ROUTE TO NETWORK　ROUTE TO NETWORK　ROUTE TO NETWORK

网络之路