测试中心
TESTCENTER

Route to Network

网络之路

OSPF专题讨论

华为3Com
你身边的好网络

2005年第二季 总第1期

技术之路不只是梦想

人生际遇风云变幻，也许唯有技术之路你自己可以把握。

世界上有很多人热爱技术，不喜欢或不擅长管理。沉浸于精妙的技术时，那样的感觉是非常平和忘我而又充实的，技术的不断更新和挑战更让人跃跃欲试。

曾经我和每一个进入团队的新同事谈话时都说过同样的一句话：希望你们在这里能够努力，有信心成为自己负责领域的国内一流的技术专家。华为无疑是国内最好的做数据通讯的公司，在华为公司，你成为一流，自然也就是国内一流的专家，有了一流的专家，我们才会造就一流的产品和公司。

虽然我不再如此谈话，但我相信对每个人来说这不只是梦想。

曾经和Cisco中国号称数一数二的高手聊过，他告诉我，在Cisco的七年多的时间里，除了工作，他几乎在抽烟的时候，等车的时候，睡觉前，甚至是上洗手间的时候，都在看Tech Documents，所以他比一般人要知道的多很多，只是因为热爱和坚持，并没有什么别的诀窍。我告诉他，其实和我前几年一样，非常热爱和投入，享受着工作和技术的乐趣。

在华为3COM涉及到的网络技术与量子物理、纯粹数学比起来远不如其复杂，我想每个人，只要有心和用心，都可以做的很好。华为3COM已经为我们提供了非常好的技术成长的条件和机会。

很多的时候，即使是看着3COM和我们同样职务的工程师几乎都比我们大了10来岁，我们都在感慨，为什么外国人这么愿意一直做着技术？记得在和3COM合作前，IPR专利之战初次打响的时候，白发的Ives对我说，因为在我们那里做技术和做管理一样，可以得到很好的尊重和待遇。

似乎每个人都在说中国人做技术很浮躁，谁都想做“Manager”。身在华为，我们肩负着公司的希望，甚至作为一些人眼里“中国做技术最好的地方”，肩负着民族的希望。我们看到了现实上的差距。公司也在努力为大家提供真正意义上的技术、管理两条发展路线。但过程总是痛苦的，甚至源于中国几千年的观念。但我相信，总有一天，中国人也会塌实的去做技术，去做一辈子优秀的工程师。

热爱，塌实，勤奋，创新，希望梦想从我们每个人自己开始。

OSPF毫无疑问是全球网络最流行和最基本的路由协议，也几乎是公司最关注或最多人有些了解的协议了，似乎每次开大小会议无论是讨论网上问题、质量进度、测试策略涉及到协议模块大家都喜欢用OSPF来举例。

事实上OSPF是个非常复杂的协议，作为基础协议，非常广泛的在我们公司设备和网络上应用，出过几乎无数看起来和OSPF相关的问题，涉及到几乎每一处细节，真正的精通确实需要“板凳要坐十年冷”的精神去钻研。好在我们已经做了多年的积累，给予了相当的关注。

《网络之路》从OSPF开始，路由我们的梦想。

------ 鉴定测试中心　刘宇

目　录

序言

初学乍练

循序渐进学习OSPF ····· 朱皓
- 入门之前
- 了解OSPF
- 熟悉OSPF
- 掌握OSPF
- 精通OSPF

OSPF基础试题 ····· 李劲松 郭逵 朱皓
- 选择题
- 判断题

深入探讨

OSPF FAQ ····· 刘宇
- 写在前面
- OSPF FAQ

OSPF中的最短路径算法 ····· 陈旭盛
- Dijkstra算法介绍
- Dijkstra算法的证明
- OSPF协议中对Dijkstra算法的使用

OSPF Graceful Restart ····· 朱皓
- 关于Graceful Restart
- OSPF Graceful Restart
- RFC 3623：Graceful OSPF Restart
- Cisco Nonstop Forwarding
- VRP & IOS的相关配置
- Graceful Restart两种主流实现方式的比较

OSPF多实例 ····· 李劲松 杨默寒
- 前言
- OSPF多实例
- 配置案例
- 留给读者
- 附录

OSPF附录E及相关问题与规避措施 ····· 周迪
- 不支持附录E导致的组网缺陷
- OSPF附录E的解决方案
- 其他相关的问题以及规避方法
- 后记

OSPF DNA LSA ····· 张宇弟
- 背景知识
- DNA LSA详解
- 总结

OSPF缺省路由 ····· 姜杏春
- OSPF区域类型
- 缺省路由的产生
- 配置实例
- FAQ
- 小结

OSPF协议新特性指南 ····· 刘宇
- OSPF协议/模块简介
- OSPF新特性指南
- VRP3 vs. IOS OSPF实现差异
- 总结

OSPF中5类LSA的Forwarding Address ····· 朱皓
- RFC中对于FA的描述
- FA对路由计算的影响
- 总结

路由协议选择OSPF vs. EIGRP ····· 刘宇
- 路由协议规划选择原则

OSPF vs. EIGRP路由协议特性比较
IPv6 OSPFv3协议详解 ····· 许亮
概述
与Version2的差别
实现细节
OSPF MTR ····· 朱皓
关于MTR
OSPF MTR
MTR与相关技术的比较
我司OSPF链路状态数据库分析 ····· 石松
前言
OSPF概述
OSPF路由计算过程
路由计算例析
关于OSFP不同区域负载分担的问题 ····· 陈志刚
问题的提出
问题的分析
问题的解决
解决办法的分析
思考过程的补充
配置例

测试方法

如何测试OSPF ····· 许亮 程新娥 李敏 李玉 陶豆 金炜
测试方法描述
RouterTester之测试OSPF初步 ····· 姜杏春
OSPF概述
RouterTester测试OSPF
使用TRT测试OSPF ····· 张敬军
软件概述
路由协议测试基本思想
TRT基本操作
端口设置
OSPF路由测试
流量设置
路由震荡设置
测试结果
范例
结束语

友商专栏

OSPF链路状态数据库解析 ····· 蔡金龙
路由器之间通过UNNUMBERED串口连接的情况
路由器之间通过分配地址的串口P TO P连接的情况
路由器之间通过P TO MP网络类型连接的情况
路由器之间通过多路访问网络类型连接的情况
路由器之间通过非广播型多路访问网络类型连接的情况
OSPF的区域间路由
OSPF如何将外部路由引入到多个区域
OSPF的虚连接
OSPF NSSA区域类型7及类型5的LSA解析
OSPF如何向普通区域引入默认路由
OSPF如何向STUB及完全STUB区域引入默认路由
OSPF如何向NSSA区域引入默认路由

典型应用

OSPF应用案例 ····· 杜祥宇
概述
网络详细设计
设备清单

网上问题

OSPF向STUB区域产生默认路由的问题 ····· 郜中华
问题的提出
问题研究

最新进展

OSPF最新进展 ····· 潘凝 钱雪彪 郜忠华
性能优化
可靠性
流量工程
OSPFv3
其它

循序渐进学习OSPF

鉴定测试中心 朱皓

循 序 渐 进 学 习 OSPF

OSPF是目前使用最广泛的IGP协议，也是一个数通领域工作者的必修科目。虽然学习的人很多，但是因为OSPF协议的复杂，学习起来非常吃力。而且经常有老虎吃刺猬，无从下口的感觉。本文试图对OSPF的学习过程进行一个整理，希望能够对学习理解这一重要协议有所帮助。

RFC2328是OSPFv2最权威的资料，这篇RFC有244页之长，在IETF的几千篇文档中也是属于超长超大的，可见其复杂程度。笔者认为对于OSPF的学习还是要循序渐进，在不同的阶段选择不同的关注点。当然对于逻辑思维很强、有很深厚数理基础的同学可能直接看RFC效率更快，每个人还是要根据自己的情况选择合适的学习方法。本文是面对如笔者一般普通的学习人群。

对于大多数人而言，学习OSPF总是经过这样几个过程：了解、熟悉、掌握、精通。而对于数通领域的初涉者，我认为还需要在了解OSPF之前再做一些准备。

❶ 入门之前

OSPF是一种动态路由协议，所以需要首先知道什么叫路由，什么叫路由协议，什么又是动态路由协议，设备是如何通过路由来执行数据转发过程的。这些东西在看OSPF之前是首先要搞清楚的问题。我个人强烈建议在学习OSPF前，首先学习RIP。通过对RIP协议的学习可以明白路由的构成要素、路由的匹配、路由如何指导转发的过程、包括路由协议的相关概念（比如路由的传递、路由计算和路由选择、路由收敛等）。

RIP是最传统的IGP协议，上手比较快，也容易理解。学习了RIP后可以明白RIP的缺点在哪里，这样学习OSPF的时候就更容易理解，一些机制的提出实际是为了解决相关的一些问题，比如环路、距离的度量等等。

入门之前的准备工作很重要，否则在学习OSPF的时候，很容易在一些基本问题上耗费很多时间而不知其所以然。学习OSPF并不是一个死记硬背的过程，如果那样的话，会越学越糊涂。

学习基本技能的几种途径包括，其它同事写的理解报告，公司精品课程中的《IP路由基础》，当然还少不了《TCP/IP路由：卷一》的相关章节。公司的技术网站上你能找到从入门到精深各个方面的文档，地址是http://tech.huawei-3com.com。

❷ 了解OSPF

做好准备就可以开始阅读一些基本的OSPF教材了。刚开始的时候会发现有大量的新概念搞得人头很大，看起来也很容易就犯困（记得我当年就是这个样子）。OSPF经典的教材很多，但往往经典的东西都试图把所有的问题都讲到，其实这不一定是个好主意。对于初学OSPF的人而言，找个合适的课程听一堂课作为学习OSPF的第一步会轻松的多。好在公司里经常会有OSPF的培训课程，课前粗粗的看一遍胶片，然后就去听吧。很多同学因为这样那样的事连一堂完整的OSPF课程都没听过，这很可惜。要知道在外面想系统的听人给你讲讲OSPF是一件多么困难的事！第一次听OSPF不需要全部听懂，搞清楚一些问题就行了。什么是OSPF？和RIP有什么不同？OSPF有什么好处？OSPF的基本原理是什么？

OSPF协议并不传递路由条目，而是传递链路状态信息，并根据该信息来计算路由。这样一个最最基本的OSPF问题当年可是困扰了我很久。一条5类LSA不就是携带一条外部路由么，怎么又说LSA并不是路由呢。相信直到今天同样的问题也会困扰很多OSPF的初学者。在了解OSPF的这个阶段，不用去关心OSPF到底是如何计算出路由的，也用不着去关心OSPF都有那几类LSA，每类LSA的Link ID是什么意思。这个阶段关注的应该是OSPF大面上的东西，比如OSPF是要建邻居的，通过邻居交换链路信息达到链路信息数据库的同步、划分区域的目的、OSPF对于不同链路的开销进进行比较的基准等等。

这个阶段要开始在设备上配置OSPF，其实OSPF的配置并不复杂，只要花几天时间做几个小实验，就能弄清楚前面所学习到的那些理论知识到底是怎么回事了。

到这里的时候一切都还是很简单的，现在可以跟那些对OSPF完全不懂的朋友们侃侃关于链路信息的概念了！不过心里要很清楚，后面的路还很远。

这个时期建议把公司里几个版本的OSPF胶片都找出来看一遍，试着跟随胶片的思路去了解OSPF吧。

❸ 熟悉OSPF

熟悉OSPF的过程，是需要对OSPF协议的整个过程进行细致一点的研究了。从OSPF的网络类型开始看起，网络类型的不同决定了hello发送方法和周期的不同，这些当然就影响到邻接关系的建立。邻接关系是如何建立起来的？需要去研究邻接的状态机了。邻接的建立目的是为了交换LSA，那么如何交换，如何保证交付。对于OSPF的区域问题，要知道所有类型的区域划分目的是什么，比如NSSA区域既允许区域间LSA也允许外部LSA的泛洪，那么NSSA的意义到底是什么这类的问题。既然说到区域当然要知道虚链路，搞清楚虚链路和IP隧道的区别在哪里。你需要知道OSPF的认证、OSPF的聚合、OSPF中的路由引入等等，总之操作手册中关于OSPF部分的所有命令都要去试一遍，看看到底有什么用，是干什么的。

在这个阶段，前进的每一步都需要通过做实验来加深加快理解，很多问题不是躺在床上靠脑袋想能够想明白的。做实验的时候打开debug信息是一个好主意，虽然这东西有时候看的人眼晕，不过你可以在屏幕快速翻滚的时候借机喝口水啥的，也算是休息一下头脑。建议从这个时候起要养成做笔记的好习惯，笔记这个东西是可以温故而知新的，过些日子回头再看，保证能看出其中的问题来，或者看出新的疑问来；另一方面，查找笔记总是能最快的找到你要的答案。

还好你是在H3C，在这个阶段你可以再去听一堂OSPF的培训。相信这次你会听的津津有味，天啊，你终于知道那个在台上念念叨叨的家伙在说些啥了。这次培训的效果是最好的，基本上讲课全部的内容都对你有用，这时候以前积累的问题就狂问吧，即使你所有的问题都已经扔给思想导师了，还是可以再问一遍，有的时候问东的结果会引出西来，这样也许又能多学点东西。

熟悉OSPF要花多久？因人而异，关键还看你用了多少时间，做了多少实验。保守的估计三个月怎么也够了。在这个阶段，公司里经典的胶片和讲义，都要找出来看过；Jeff Doyle的《TCP/IP路由：卷一》里关于OSPF的部分也起码要看过一遍，其中有些部分还是可以跳过，什么时候都别忘了你的重点在哪里。

❹ 掌握OSPF

掌握的意思是终于可以少犯或者不犯错误了，起码说关于OSPF的话题，很少或者不会有什么太明显的错误。对于开发或者测试OSPF的技术工程师，熟悉OSPF是远远不够的，你必须要走的更远些。

是时候去看那些报文的结构了，相信这个时候再去看各种报文的结构会容易很多。随便画一个简单的拓扑，你必须要知道每个路由器会产生那些LSA，它的LSDB是怎样的。看LSA结构的时候要结合SPF路由计算的过程。这个过程是OSPF的核心问题，终于要揭开关底BOSS的神秘面纱了。当然了，前提是你要做好打BOSS的心理准备，反正我觉得这个过程虽然充满乐趣，但却很容易头晕脑涨。结合实例来看会比较好些，你可以直接用WVRP建一个简单的OSPF网络，来看看各个设备上的LSDB是否能看明白。有很多好的文档也能指导你更快的理解。对于LSA和LSDB，英文文档可以看看Cisco的OSPF Database Explanation Guide，中文文档季刊里就有好几篇，那篇关于SPF计算的文档更是要读通读懂。

这个阶段需要去了解一些比较细节的东西了，比如各种LSA的属性及标志性字段啦，LSA的更新老化机制啦等等，是时候看RFC了。好好的看一遍RFC2328是必要的，而且在今后的日子里，你可能会经常的翻看这篇文档。

RFC虽然非常的详细，但RFC毕竟只是一个规程，在具体的实现上，各个厂商还是有很多不同的。而且很多支持的新特性，不一定会在RFC中提及，所以产品新特性的介绍是另一个最重要的文档。光知道一个厂商的实现还不行，最起码要知道主要厂商的OSPF实现。对于市场主流厂商的研究是一个慢慢积累的过程，但是作为一个研发人员，不看外面的世界显然是不可能的。新特性会牵扯到很多问题，OSPF作为一个成熟使用多年的协议，特性也非常多，要花不少时间。

什么叫做掌握OSPF，没有一个定义。在这个阶段中可能有很多人，有的人掌握的好些，有些人掌握的差些。这个阶段其实已经可以说是学习OSPF的一个结尾了，更深层次的研究有时候需要的不单是理论上的东西，还包括经验和经常性的实践。我自己感觉也是停留在这个层面上，尤其是因为工作的关系，并不常接触到协议中的问题，对实际使用案例接触的也少了，所以经常会碰到一些OSPF的问题会很含糊。

❺ 精通OSPF

精通OSPF就是大牛了，我觉得起码要几个方面都做到吧。一个是对OSPF的协议非常清楚，包括几乎所有的细节。一个是在OSPF的疑难解答方面，不说来一个答一个，起码立刻能找到分析问题的方法，找到并正确定位问题。第三就是对协议的发展要心中有数。网络技术的不断发展，对路由协议也提出了各种各样新的要求，并且对协议进行着不断扩展。作为一个精通OSPF协议的人，需要关注这些新技术的发展方向这是理所当然的。

精通OSPF并没有指南，不是看过哪些书，过多少年就一定能达到了。甚至是不是精通了OSPF，只是自己给自己的一个定义。不过，我想，如果一直向精通OSPF这个目标不断迈进，永不停歇的人，一定会成为一个真正的高手。

OSPF基础笔试试题

技术支持中心 李劲松 / 鉴定测试中心 郭逵 朱皓

❶ 选择题

1. 对于路由表，下列说法中正确的是：
A、在缺省的情况下，如果一台路由器同时运行了RIP和OSPF两种动态路由协议，则在路由表中只会显示RIP发现的路由，因为RIP协议的优先级更高。
B、在缺省的情况下，如果一台路由器同时运行了RIP和OSPF两种动态路由协议，则在路由表中只会显示OSPF发现的路由，因为OSPF协议的优先级更高。
C在缺省的情况下，如果一台路由器同时运行了RIP和OSPF两种动态路由协议，则在路由表中只会显示RIP发现的路由，因为RIP协议的花费值（metric）更小。
D在缺省的情况下，如果一台路由器同时运行了RIP和OSPF两种动态路由协议，则在路由表中只会显示OSPF发现的路由，因为OSPF 协议的花费值（metric）更小。

2. 下列动态路由协议中，属于CISCO公司私有协议的有：
A、RIP
B、IGRP
C、EIGRP
D、IS-IS
E、OSPF
F、BGP

3. OSPF 协议生成的路由分为四类，按照优先级从高到低排列，正确的顺序为：
A、区域内路由、区域间路由、第一类外部路由、第二类外部路由
B. 第一类外部路由、区域内路由、区域间路由、第二类外部路由
C. 区域内路由、第一类外部路由、区域间路由、第二类外部路由
D. 第二类外部路由、区域内路由、区域间路由、第一类外部路由

4. 对于运行OSPF协议的路由器来说，Router ID是路由器的唯一标识，所以协议规定：必须保证Router ID在如下范围之内唯一：
A. 网段内　　B. 区域内　　C. 自治系统内　　D. 整个因特网

5. 一台运行OSPF的路由器，它的两个正常工作的接口一个属于区域0，另一个属于区域9，并且引入了5条静态路由，则该路由器一定会生成哪类LSA：
A、Router LSA（Type = 1）
B、Netwrok LSA（Type = 2）
C、Network Summary LSA（Type = 3）
D、ASBR Summary LSA（Type = 4）
E、AS External LSA（Type = 5）

6. OSPF选举DR、BDR时会使用如下的那些报文？
A、HELLO报文（Hello Packet）；
B、DD报文（Database Description Packet）；
C、LSR报文（Link State Request Packet）；
D、LSU报文（Link State Update Packet）；
E、LSAck报文（Link State Acknowledgmen Packet）

7. 下列那些OSPF报文中会出现完整的LSA信息？
A、HELLO报文（Hello Packet）；
B、DD报文（Database Description Packet）；
C、LSR报文（Link State Request Packet）；
D、LSU报文（Link State Update Packet）；
E、LSAck报文（Link State Acknowledgmen Packet）

8. 下列那些OSPF报文中只会出现LSA的摘要（LSA HEAD）信息？
A、HELLO报文（Hello Packet）；
B、DD报文（Database Description Packet）；
C、LSR报文（Link State Request Packet）；
D、LSU报文（Link State Update Packet）；
E、LSAck报文（Link State Acknowledgmen Packet）

9. 两台运行OSPF协议的路由器通过点对点的链路相连，在两者链路状态数据库达到同步（邻居状态机为FULL）之后的2个小时之内，在两台路由器之间的链路上，一定有下列的那些报文在传递？
A、HELLO报文（Hello Packet）；
B、DD报文（Database Description Packet）；
C、LSR报文（Link State Request Packet）；
D、LSU报文（Link State Update Packet）；
E、LSAck报文（Link State Acknowledgmen Packet）；

10. 关于OSPF协议中DR（Designated Router —指定路由器）的产生，下列说法中正确的是：
A、DR是系统管理员通过配置命令人工指定的。
B、DR是同一网段内的路由器通过一种算法选举出来的。
C、成为DR的路由器一定是网段内优先级（priority）最高且Router ID最大的。
D、可以通过配置命令使一台路由器不可能成为DR。

11. 关于OSPF协议中的DR、BDR，下列说法中正确的是：
A、在广播型的网络中，如果没有DR，则协议无法正确运行，但如果没有BDR，协议仍然可以正确运行；
B、在一个广播型的网络中，即使只有一台路由器，仍旧需要选举DR；
C、DR和BDR与本网段内的所有运行OSPF协议的路由器建立邻接（adjacency ）关系，但DR和BDR之间不再建立邻接关系；
D、DR会生成本网段内的Netwrok LSA（Type = 2），但BDR不会。

12. 根据OSPF协议规定，在下列类型的网络中需要选举DR、BDR：
A、Broadcast；

B、NBMA；
C、Point-to-Multipoint；
D、Point-to-Point。

13. 根据OSPF协议规定，下列哪些LSA只在区域内传播：
A、Router LSA（Type = 1）；
B、Netwrok LSA（Type = 2）；
C、Network Summary LSA（Type = 3）；
D、ASBR Summary LSA（Type = 4）；
E、AS External LSA（Type = 5）。

14. 根据OSPF协议规定，对于一个运行在STUB区域的区域内路由器，它的LSDB中不会出现下列哪些LSA：
A、Router LSA（Type = 1）；
B、Netwrok LSA（Type = 2）；
C、Network Summary LSA（Type = 3）；
D、ASBR Summary LSA（Type = 4）；
E、AS External LSA（Type = 5）。

15. 关于OSPF协议的LSA，下列说法中正确的是：
A、每一条Router LSA（Type = 1）中只包含该路由器一个接口的相关信息；
B、每一条Netwrok LSA（Type = 2）中只包含该路由器一个接口所属的网段中的相关信息；
C、每一条Network Summary LSA（Type = 3）中只包含一条路由信息；
D、每一条ASBR Summary LSA（Type = 4）中只包含一个ASBR的相关信息；
E、每一条AS External LSA（Type = 5）只包含一条AS外部路由信息。

16. 关于OSPF协议中AREA（区域）的概念，下列说法中错误的是：
A、每个AREA中有自己的LSDB，不同AREA的LSDB是不相同的；
B、为了标识出自己所属的区域，Router LSA（Type = 1）中包含了区域信息。
C、每个区域都用一个32位的整数——AREA ID来标识，且必须保证 AREA ID在自治系统内唯一。
D、区域的标识——AREA ID必须向相关的国际组织申请，不可自行指定。

17. 关于OSPF协议中的虚连接（Virtual Link），下列说法中正确的是：
A、虚连接不能穿过STUB区域；
B、虚连接不能穿过骨干区域；
C、虚连接不能穿过多个区域；
D、对于同一个区域来说，最多只能有一条虚连接将其作为转换区域（transit area）。

18. LSAck报文（Link State Acknowledgmen Packet）是对下列那些OSPF报文的确认？
A、HELLO报文（Hello Packet）；
B、DD报文（Database Description Packet）；
C、LSR报文（Link State Request Packet）；
D、LSU报文（Link State Update Packet）；
E、LSAck报文（Link State Acknowledgmen Packet）

❷ 判断题

1. 两台运行OSPF协议的路由器正在进行链路状态数据库的同步，从其中一台上查看邻居状态机，已经达到FULL状态，因此可以判断在另一台路由器查看邻居状态机，应该同样达到FULL状态。

2. OSPF协议中提出DR的概念是为了减少广播和NBMA网络中路由器之间进行链路状态数据库同步的次数，但如果网络中的路由器少于3台，这样做是没有意义的。所以如果在广播和以太网中只有两台路由器，则不选举DR。

3. 根据OSPF协议规定：在一个广播类型的网络中，由于有DR的存在，两台DROther路由器之间不再进行路由信息的交换，也不再互相发送HELLO报文。

4. 两台路由器建立起邻居关系的一个前提是：两台路由器至少各有一个接口属于同一个区域。

5. 虽然OSPF协议中规定，每台路由器的Router ID必须保证在自治系统内唯一，但由于Router LSA只在本区域内传播，所以即使有两台区域内路由器（分别属于不同的区域）的Router ID相同，也不会引起路由计算的错误。

参考答案

选择题
1. B　2. BC　3. A　4. C　5. ACE　6. A　7. D　8. BCE　9. ADE　10. BD

11. ABD　12. AB　13. AB　14. DE　15. BCDE　16. BD　17. AC　18. D

判断题
1. F　2. F　3. F　4. T　5. F

OSPF FAQ

鉴定测试中心 刘宇

写在前面

OSPFv2（RFC 2328）算是RFC 0001-3000最重要最复杂的协议之一了，其中的细节描述非常之多而细微繁杂。如果你有兴趣去对比RFC的文件大小，可能会发现只有一个关于新千年的非技术文档比它要大点了。在公司经常有各部门的同事询问各种OSPF的细节的问题，很多是在网络运营和测试中的实际问题，几乎验证了再细小的细节都有可能出现网上问题的论断。

前几天有开发人员在我转发的对RFC 2328 Appendix E的讨论中说，这几乎成为测试OSPF的每个员工必提（不知道是必提的问题单，还是必询问的问题）的50个问题之一，可惜没有时间来整理下。我想整理下FAQ的时间绝对是远少于公司前赴后继的开发、测试、市场、技术支援询问讨论这些FAQ的时间。正好今天下午有同事问我在MPLS/BGP VPN环境中，PE-CE运行OSPF，multi-VRF的时候为什么收不到路由。其实仔细检查下，发现multi-VRF用的是Frame Relay做的两个子接口，而在OSPF中没有配置neighbor造成连邻居都不能形成，何况路由。于是决定，这个事情我自己来做——至少是我自己开个头——虽然我也有很多事情要做很多东西要写，但至少不是日理万机，加一两天的班能完成的。

希望公司测试，开发OSPF的同事都能熟悉文中所有的内容，相信这对于一个精通OSPF的技术专家来说，是必须了解的，阅读它能让你迅速增加对OSPF常见细节的了解。对于文中的FAQ的一些更详细的解释，请参考RFC 2328和tech.huawei-3com.com中相关的总结文档，文档URL可能会改变，可以按文档标题搜索。

OSPF FAQ

OSPF是否很难懂？

A：前面提到过，OSPFv2（RFC 2328）算是RFC 0001-3000最重要最复杂的协议之一了，其中的细节描述非常之多而细微繁杂。我曾经对比过RFC的文件大小，可能只有一个关于新千年的非技术文档比它要大点了。相对起来，OSPF确实比较复杂。但绝对不是非常难以掌握。如同其它Internet核心技术一样，它是非常有趣和有挑战性的；同样，一般的了解和配置都是易于掌握的，也就是说如果你不开发或测试OSPF，那么你所需要达到的程度还是很容易实现的。但作为一个开发或测试OSPF模块的人员，你必须对其中所有的细节都非常精通，而这确实需要些时间的积累。好在目前的资料和积累都非常多了，深入了解OSPF已经是件相对5年前容易多的事情了。至少，如果你能不去参考文档能回答本文几乎所有的FAQ，那么你确实对OSPF算的上相当的精通了。

为什么新路由器启动后我不能配置OSPF？

A：很简单，一上来是不能配置OSPF的，因为OSPF运行是需要一个确定的Router ID作为路由器标志的，这个东西对于OSPF来说非常重要，在OSPF的各类协议报文中都要用到。所以，要么你在OSPF中明确指定Router ID，要么配置一个以上的三层接口让OSPF模块计算出一个可以供OSPF协议使用的Router ID，这样才能启动OSPF进程。

OSPF的router id如何配置，缺省是什么？

A：VRP3.0和IOS一样，如果没有显示配置OSPF Router ID（很遗憾，VRP3.0的OSPF没有命令配置，但在MPLS/BGP VPN环境下，对于OSPF多进程是起进程的时候就同时一条命令配置Router ID），简单的说我们采用如下策略：1.如果有loopback接口配置了，就选IP地址数值最大的loopback地址；2.如果没有配置loopback接口地址，就选IP地址数值最大的物理接口地址。

需要注意的是，如果当前OSPF进程正在运行，Router ID即使是重新手工配置或计算都不会马上生效，而需要OSPF进程重新启动才会生效。这个要求是合理的，因为Router ID对OSPF协议来说太重要，不可能在OSPF保持邻居不断的情况下更新。

为什么引入的两条静态路由不能同时发布？

A：这个是VRP3.0的历史遗留问题，不支持RFC 2328 Appendix E，一直没有很好解决。实际上一般情况下静态路由都可以正确发布的，但对于类似于RFC 2328 Appendix E情况，由于OSPF type 5 LSA的ID是以这些网段的地址来标识的，每一类型的LSA是以产生该LSA的路由器ID、LSA的序列号以及所承载的网段的网络地址这三个信息的组合作为其关键值进行区分的。所以同一台路由器产生的或者引入外部路由而生成的这些LSA，对于那些网段地址一样、而掩码信息不一样的路由，将不能进行区分。例如对于静态路由20.20.0.0/16和20.20.0.0/24的引入将只会生成一个LSA，其中必有一条路由因不能引入而被丢弃。OSPF的RFC 2328附录E推荐了一种改进的方法，很可惜VRP3.0并不支持，在多个项目中也暴露出这个问题。

更详细的总结请参考测试中心周迪整理的文档：OSPF附录E及相关问题与规避措施，文档URL如下：http://tech.huawei-3com.com/article.php/1003。

为什么我配置两个区域，从这两个区域学到的路由总的开销相等，为什么不能形成ECMP（等价路由）？

A：按协议标准，在RFC 2328 Section 16.8. Equal-cost multi path中规定OSPF要形成等价路由必须有如下的条件：1.涉及的是同一区域；2.优先级和总的开销一致。很多情况下大家忽略了第一个条件。对于Intra或Inter Area的路由，协议并没有非常明确的规定，在多区域各种比较条件相同的情况下到底选择哪一条，VRP3.0的选择条件也并非选择后面发布的一条。但对于type 5的外部路由，RFC 2328 Section 16.4. Calculating AS external routes中明确规定，when there are multiple least cost routing table entries the entry whose associated area has the largest OSPF Area ID (when considered as an unsigned 32-bit integer) is chosen。也就是最终选择经过的Area ID大的那条路由作为优先。不过需要注意的是，很多实现都没有严格按照这个来做的。对于Cisco IOS来说（以IOS 12.3（10）T为例），无论是区域间路由，还是AS外部路由，都会形成等价路由，这应该是与协议不符的。

更详细的总结可以参考tech网站的相关文档：关于OSFP不同区域负载分担的问题，URL在：http://tech.huawei-3com.com/article.php/913。

为什么有的Stub区的ABR没有正确产生确省路由？

A：在某些特定组网下（主要是Area 0只

采用loopback接口，没有形成邻居），STUB区的ABR没有产生缺省路由。看来这样的组网方式还是有些市场，我对你发现的问题感到很遗憾，这其实是出了网上问题几方讨论后的一个妥协结果。实际上我个人认为，目前的处理并非很有道理。更详细的总结可以参考tech网站的相关文档：为什么STUB区的ABR没有产生确省路由？，URL在：http://tech.huawei-3com.com/article.php/822。

ABR确切定义是什么？是否在路由器上配置了两个以上的区域就是ABR？

A：非常遗憾，RFC 2328给出的ABR定义如下：

Area border routers：

A router that attaches to multiple areas. Area border routers run multiple copies of the basic algorithm, one copy for each attached area. Area border routers condense the topological information of their attached areas for distribution to the backbone. The backbone in turn distributes the information to the other areas.

这里并没有给出ABR非常严格的定义。导致一些厂商的实现不一致，而ABR的功能又如此重要。但在RFC 2328中 3.1. The backbone of the Autonomous System 中有描述：The OSPF backbone always contains all area border routers. 反过来，ABR必须属于骨干区。如果你一个路由器只配置了Area 2，Area 3而没有接口属于骨干区，肯定就不能算是ABR了。对于ABR的比较严格或比较有效的定义，可以参考：RFC 3509 ：Alternative Implementations of OSPF Area Border Routers。，这个文档正式了RFC 2328对ABR定义不严格导致的一些问题，提出了更严格的两种定义方法。目前的VRP3.0对ABR定义如下：1.配置多个区域；2.不但要求要有骨干区的接口，而且还要有邻居在骨干区域形成。

OSPF没有路由，甚至邻居都不能形成Full关系，最常见的原因是什么？

A：原因很多。但最常见的是如下几种情况：

1.OSPF网络类型是NBMA的，但你忘记配置邻居了；

2.OSPF网络类型是NBMA的，你配置了邻居，但在诸如Frame relay的map语句中忘记加broadcast关键字了，导致组播报文不能到达对方；

3.OSPF邻居的hello及dead interval值不一致；

4.在Stub或NSSA区域，有些路由器没有配置成Stub或NSSA；

5.OSPF验证配置错误；

6.OSPF Router ID有问题，可能和某个其它路由器一样了；

7.OSPF链路两端的网络类型不一致；

8.OSPF链路两端的MTU相差比较大，尤其注意和比较老的IOS实现互通时；

9.该网络根本就没有启动OSPF；

10.区域号不一致；链路的网络地址不一致，注意检查两边的mask；

11.ip unnumbered的地址借用接口一般也要注意对端也需要配置为ip unnumbered；

OSPF的排错不是几句话就能写的很全面的，建议参考专门的Troubleshooting OSPF，Cisco Networker 2004的最新文档在http://tech.huawei-3com.com/article.php/162。

有什么好的办法知道OSPF出了什么问题？

A：其实很简单，也是必须知道的。调试开关是需要打开的，其中最有效，最常用的就是debugging ospf event（IOS对应命令为debug ip ospf event）！它能让你对OSPF的大部分问题看的一目了然。当然它也不是万能的，它是在正确接收OSPF报文的基础上才能有相应的错误事件。如果没有看到任何动静，建议打开OSPF的所有报文调试开关，看看报文的收发是否正常。

我配置了Frame Relay或X.25，为什么OSPF不能形成邻居呀？

A：这个是大家经常犯的错误，甚至对于一些比较熟悉OSPF的工程师而言。请注意，一般来说，在跑Frame Relay，X.25，ATM等封装类型的链路上，OSPF缺省的网络类型是NBMA的，这时候要注意两个事情：一是别忘记在OSPF配置模式下配置OSPF邻居；二是别忘记了在映射IP地址的时候别忘记加入broadcast选项，以便让广播或组播报文正常收发。

OSPF如何自动计算接口cost的？

A：VRP和IOS一样，当链路接口没有明确配置OSPF cost的时候，cost按配置的基值除以接口带宽来计算。这个基值缺省为100M，例如10M的链路，cost缺省是100/10=10。显然当运行OSPF的路由器存在多个速率不同的1000M以上的高速接口时候，如果接口没有明确赋予OSPF cost，按缺省公式自动计算的cost将都为1，不能反映链路速率。这个时候IOS是有一个auto-cost reference-bandwidth的命令，来调节基准值的，但要注意，整个OSPF路由域都要对应调整。因此，最好的方法，还是在网络做好规划，手工对链路接口的cost赋值。

需要注意的是VRP规定，对于链路速率小于等于2K的，按2K计算；大于等于100M的，按100M计算。而对于loopback口，由于是虚拟的接口，VRP在某些版本（比如新的AR46）上cost缺省和IOS一样是1，更多的版本（比如老的NE系列）是1562，我记得很早前我就反馈过应该设置为1。而且对于100/10M自适应的以太网接口，在VRP3.3下，100M端口cost是1，但是如果将强制为10M （Speed 10），cost还是1，天啦，世界真奇妙。

OSPF的收敛速度到底是多少？如何能调整加快收敛？

A：大家都说OSPF快速收敛，但具体要给个数据，却一时找很多人要都没有给出，只能重新测试。按Spirent标准的OSPF Route Reconvergence测试方法，在为FT提供的数据，两个小组的测试数据比较接近。当网络拓扑规模为20个结点，100条路由的情况下，用Agilent的Router Tester在spf-schedule-interval为1s时候，收敛速度为2140.9 ms；spf-schedule-interval 为10s时候，收敛速度为7108.4 ms。这个网络规模比较小，数据仅供参考。

事实上，对OSPF来说网络变化影响路由收敛主要由如下几个因素组成：1.侦测对端链路down，邻居失效；2.新的LSA 产生并Flooding；3.LSDB同步更新，重新进行SPF计算新路由重新下发到FIB。个人认为在标准的OSPF Route Reconvergence测试方法中，如果STU是单个设备的话就会忽略了1，2的时间测试，收敛时间并不是非常准确。

为了加快OSPF的收敛速度，VRP3.0可调整的选项比较少：1.邻居的hello和dead间隔时间；2.spf-schedule-interval（缺省5秒），缩短两次SPF算法运行的间隔时间；3. 减少接口下配置ospf trans-delay，加快LSA的flooding速度。IOS提供了相对丰富的多的微调选项，尤其是i_SPF算法，能较大的改进收敛速度。

OSPF的点到NBMA网络类型，一定要求是Full mesh的吗？

A：可能受一些培训资料的影响，比如广为流传的李劲松OSPF讲座，大家都认为OSPF的NBMA网络类型从L3的角度看来必须是Full Mesh直接可达的，否则就不可以配置成NBMA。事实上不是如此，只要保证你的DR，BDR和所有的邻居L3直接可达就行了。虽然DR选举并不确定，但你可以通过设置其它路由器的优先级为0来取消它的选举权。但从健壮性的角度看来，我们建议，当你打算配置OSPF NBMA网络的时候，还是在Full Mesh的条件下配置。

OSPF的P2P网络类型，一定要求两端的IP地址在同一网段吗？

A： 很多人都认为，只有地址借用的P2P链路是不用检查是否在同一网段的。实际上VRP的实现中，在OSPF P2P网络类型的前提下，如果链路层协议封装为PPP，配置不同的网段地址，是可以形成Full并正确路由的；但如果链路层协议封装为HDLC或其它协议，则不能形成邻居。差别在于，PPP可以协商到对端的IP地址，不用额外的操作也有对端的路由。顺便所一句的是，OSPF广播网络，NBMA，P2MP等其它网络类型，是要求严格检查邻居的IP地址是否和自己接收接口在同一网段的。

OSPF链路两端配置不同的网络类型，能否形成Full关系？

A：看起来很奇怪的问题，其实比较有意思。很多人的第一感觉就是：两端的了链路网络类型都不一样，哪能形成邻居关系呢？其实不然。OSPF协议并没有规定，要去严格检查链路的网络类型，链路的网络类型最重要的描述也是在type 1 LSA中，形成邻居的关系条件检查并没有去检查它。仔细阅读协议并做实验，你会发现不少情况

下，比如两台路由器以太网连接，一端保持缺省的广播网络类型，一端配置成OSPF P2P网络类型，肯定是可以形成邻居，并交换LSDB达到Full状态的。但很奇怪的事情是：到达Full状态了，为什么学不到路由呢？其实答案很简单，OSPF路由器需要LSDB来构建SPT（Shortest Path Tree），由于LSDB的数据库是脱节有问题的（在我的Router LSA中，我认为你是个广播邻居；而在你的Router LSA中认为我应该是个P2P邻居），根本无法构建正确的SPT， SPF算法也无法计算出正确的路由。

OSPF的type 5的外部路由中的Forwarding Address有什么用？是如何填写的？

A：和RIPv2，EIGRP，BGPv4等其它比较“聪明”的路由协议一样，OSPF type 5 LSA中的Forwarding Address（以下简称FA）的作用是通告本路由域内部路由器如何能到达Type5 LSA描述的引入的AS外部网络的更快捷的下一跳，以免内部路由器在广播网络上以自己为下一跳路由到自己，自己再转发到同一广播网络上的外部路由域的路由器上，而产生“额外”的一跳。

ASBR引入外部路由产生的Type 5 LSA中，FA可能是0，也可能是非0的.

很遗憾，RFC 2328并没有明确规定FA如何填写，这个也是很多人不熟悉的地方。VRP3.0的实现是很早前在我的建议下参考Cisco IOS的实现修改大致类似的。

规则如下：

如果ASBR引入路由，但OSPF没有在这些路由的下一跳接口上启动（enable），FA设置为0.0.0.0。

而在满足如下所有条件的情况下FA设置为非0.0.0.0：

1. OSPF在ASBR的下一跳接口被启动；
2. ASBR的下一跳接口没有被设置为被动接口（大家都知道VRP叫silent interface，IOS就passive了）
3. ASBR的下一跳接口不是OSPF P2P或P2MP类型的；
4. ASBR的下一跳接口地址落在OSPF协议配置的network命令范围内

除此之外，其它情况FA都填为0.0.0.0。

必须一提的是：在RFC 2328 ，section 16.4 (Calculating AS external routes)中说明：

"If the forwarding address is non?zero, look up the forwarding address in the routing table. The matching routing table entry must specify an intra?area or inter?area path; if no such path exists, do nothing with the LSA and consider the next in the list."

很早期的VRP是有问题的，我建议过修改。如果没有太多的改变的话，目前的实现依然没有完全检查是否有“intra?area or inter?area path”，而是检查是否有OSPF路由（包括了OSPF ASE路由）。

关于OSPF FA，建议阅读tech网站上提供的 ：The Effects of the Forwarding Address on Type 5 LSA Path Selection 及Common Routing Problem with OSPF Forwarding Address。

OSPF路由聚合是否可以跨区域聚合？

A：先看一个网上问题，简单示意的OSPF网络拓扑，area 1——area0—area2，area1中三条路由：10.1.0.0/16，10.2.0.0/16，10.3.0.0/16，在area1和area0之间的ABR没有配置聚合（将上述三条聚合成10.0.0.0/8），但在area0和area2之间的ABR配置聚合却不生效。这就是跨区域的聚合问题，这个表现是否正确呢？

仔细看下RFC 2328 12.4.3 Summary-LSAs中的描述，我们可以知道ABR产生type 3 LSA时，如果是inter-area，就直接处理，产生相应的type 3 LSA，而不需要考虑配置的range，而在考虑intra-area路由的时候，才要去考虑配置的聚合！

所以，这个网上问题是用户的理解问题，不应该是问题。即使能这样聚合，这样做的结果是更多的LSA传到骨干区域，为网络的不稳定带来隐患

为什么我的路由在cost比较小的情况下没有优选通过Backbone的？

A：一般资料上都没有提到按RFC 2328实现的OSPF必须有个“RFC1583Compatibility”的选项（不过很遗憾，VRP3.0没有这个配置选项，让人怀疑是否是按RFC 2328实现的）。RFC 2328规定，如果“RFC1583Compatibility”是disable的话，路由如果可以经过普通区域和骨干区域同时学习到，而且总的开销相同，是会优选普通区域的。主要的考虑也是为了减轻骨干区的压力。这是个很细节的地方。

我可以在以太网上配置OSPF网络类型为P2P或NBMA等类型吗？

A：可以的。但除非你有必要并且清楚修改缺省网络类型的后果，不要轻易去改它。初学者可能会认为OSPF中的广播网络，点对点等网络类型，必须是链路层对应也是广播或点对点的。实际上那只是个参考模型而已，与物理介质和链路层协议无关。

OSPF网络规划时，一个区域到底最好是支持多少个路由器？

A：很多文档，包括RFC 2328作者J. Moy本人在其大作《OSPF：Anatomy of an Internet Routing Protocol》中都建议，最好每个区域不要多于50个路由器。其实我们可以看下，那些文档和这本书是在90s年代成文的。那时候路由器CPU处理速度，内存大小都与当今主流的设备能力不可同日而语。测试表明，200台路由器一个区域都可以非常快速的收敛。

个人认为，50台路由器的经验值应该修改了。否则OSPF vs. IS-IS的争论中，OSPF单区域只能支持相对少的多的路由器数目会成为受攻击的弱点。

我配置了区域需要MD5认证，是否本区域的每条链路都要配置认证？

A：很遗憾，VRP3.0目前是这么要求的，你必须保证该区域的所有链路都要配置MD5认证。而IOS早在12.08就支持每接口认证了。

OSPF如何过滤接收的路由呀？

A：首先要清楚OSPF是基于链路状态的路由协议，与D-V算法的路由协议不一样，它为了保证拓扑关系的一致性——这也是OSPF路由无环路的重要条件，没有那么轻易能过滤路由并不扩散。不过你可以在OSPF的配置下，利用distribute-list in来过滤路由。但要注意的是：千万不要以为你的OSPF数据库中没有该路由的LSA记录，实际上OSPF进程是算出了该路由，只是路由管理未将其加入到路由表中；还有很重要的一点是该路由对应的LSA依然会扩散，也就是说它的邻居路由器还是会有这条路由的。值得注意的是distribute-list out只能在引入发布配置的外部路由时过滤用。

IOS有：neighbor database-filter的命令，它不过是在冗余连接的情况下限制从某个接口向邻居发送，过滤掉产生的重复多余的LSAs，减少协议报文流量对网络的影响和路由器对协议报文的处理。类似IS-IS的mesh-group的处理。

OSPF还有一个新的特性，OSPF ABR Type 3 LSA过滤。该特性将ABR的能力扩展，使其能在OSPF区域间发布Type 3 LSAs的时候进行过滤。本特性可以限制仅允许满足某些前缀的报文从一个区域发送到另一个区域，这种类型的区域路由过滤可以应用到某个特定区域，对入区域、出区域或两者同时都可以进行控制。可惜VRP3.0是没有实现。

OSPF如何将一个区域的某条路由隐藏起来，不发布到其它区域？

A：这个问题，很多人都想当然以为用过滤列表就可以解决问题。事实上，当你知道OSPF如何过滤路由的话，你就知道不可能的。一个小把戏是，你在这个区域的ABR上配置路由聚合，聚合范围不变，比如10.1.0.0/16这个网段的路由，依然聚合成10.1.0.0/16，关键的是，聚合的语句后面有个no-advertise选项，它将帮助你不发布到骨干区，更不会发布到另外的区域去了。还有一个笨方法就是，在禁止发布的区域每个路由器用过滤列表禁止该路由加入到路由表中去。

如果你关注OSPF的新发展特性的话，你可能会知道有一个OSPF ABR Type 3 LSA 过滤特性将ABR的能力扩展，使其能在OSPF区域间发布Type 3 LSAs的时候进行过滤。本特性可以限制仅允许满足某些前缀的报文从一个区域发送到另一个区域，这种类型的区域路由过滤可以应用到某个特定区域，对入区域、出区域或两者同时都可以进行控制。

我在接口上配置了主从IP地址，OSPF是否可以以从地址建立OSPF邻居关系？

A：很遗憾，由于实现复杂度的关系，OSPF不允许以从地址建立OSPF邻居关系，而只能以接口的主地址建立邻接关系。其实其它一些协议也是如此实现的。但如果你将从地址接口的网络加入到了OSPF的配置中，它肯定是可以发布相应的路由的。需要注意的是，主从地址可以是可以配置在同一网段中。

OSPF的virtual link是否很有用处？

A：从协议的角度上来看，OSPF的虚连接Virtual Link非常有用，一是可以将不与骨干区域直接物理连接的区域连接起来，让它能正常路由，这在一些网络的合并中比较有用；二是可以提高网络的可靠性，让骨干区不至于轻易断开而不能正常路由（RFC 2328中的例子）。

但坦白的说，实际的应用中虚连接缺很少用到，我的记忆中华为VRP的OSPF协议在网络中大量应用，但几乎没有网络用到v-link。原因很简单：一般来说，网络的规划比较合理，不会让一些区域与骨干区断开物理连接，在国内网络合并而不重新全部规划的情况也非常少；二是大家也经常不会想用v-link来增加骨干区域的健壮性。

无论如何，作为OSPF的一个重要特性，对OSPF虚链路，你都应该了解。

为什么两个路由器背对背一条链路连接，我将接口shut down了，查看链路状态数据库，对端原来的LSA还在？

A：为什么不在呢？OSPF规定，只有LSA的始发者才能Flooding掉自己发布的LSA，唯一的通路都down了，LSA的Flooding自然不能更新过来了，但我自己产生的Router LSA会更新描述新的链路状态，即使你这条LSA在我的数据库中，都不会参与SPF计算，会随着时间老化。

OSPF的virtual link上MD5验证的配置为什么总是不通？

A：对于VRP3.0来说，这也是一个比较常见的问题。事实上假设OSPF的v-link是在区域2，你不能只在v-link配置的时候配置MD5认证，你首先要使能区域2的MD5认证。

如果配置Virtual link的cost，需要注意什么？

A：需要注意，这个cost是不能超过RFC规定的最大值65535的，否则不能建立建立v-link。

OSPF常用的LSA type怎么没有看到类型6呢？是否还有其它类型的？

A：Type 1 LSA：Router LSA；Type 2 LSA：network LSA；Type 3 LSA：summary LSA；Type 4 LSA：asbr-summary LSA；Type 5 LSA：AS-external-LSA；Type 7 LSA：NSSA AS-external-LSA的定义大家可能都比较熟悉了，这里没有必要介绍。

原来的OSPF报文编码不是基于TLV的，要扩展OSPF的功能只能扩展其LSA类型了。

其实type 6 LSA为group-membership-LSA，是在MOSPF中用于标识组播组成员用的，可惜MOSPF基本被淘汰。Type 8 LSA为external-attributes-LSA，用做将BGP路由引入到OSPF中互操作，保留BGP的AS路径等信息，遗憾的是完全实现的非常少，VRP，IOS都不支持。

值得一提的是RFC 2370中定义了重要的LSA类型：Opaque LSA，将OSPF的能力进一步提升，允许采用类似TLV的结构。OSPF流量工程等应用就是基于Opaque LSA扩展能力的。其中LSA Type 9为只在本链路范围扩散的Opaque LSA；LSA Type 10为只在本区域范围扩散的Opaque LSA；LSA Type 11为类似Type 5 LSA在本AS范围内扩散的Opaque LSA。

s在一个Stub区域，有两个ABR，它们产生的缺省路由，不会让它们互相指向，形成路由环路吗？

A：不会形成环路的。当一个STUB区域中的ABR接收到同一区域ABR发出的默认路由LSA的时候，它只会将其放到LSA Database里，而不会计算出路由，这样就避免了两个ABR缺省路由相互指向形成环路。这其实也是VRP OSPF一个血的教训修改的结果。

在一个NSSA区域，有两个ABR，它们都会将type 7 LSA转换成type 5 LSA吗？

A：不会的。按RFC 1587：OSPF NSSA Option的规定，在NSSA区域有多个ABR时，只有最大的Router ID的ABR将type 7 LSA负责转换为type 5 LSA。

当一条路由既有type 5 路由又有type 7的时候，我们优选哪个？

A：OSPF路由的优先级从高到低的次序是：Intra-Area>Inter-Area>type 5 AS-External>type 7 AS-External。在这种情况下，应该优选的是type 5路由。注意，对代表同一子网的路由来说，首先比较的是路由优先级才是总的cost。

OSPF DD报文中的Interface MTU是否有用？两端是否一致才能形成Full？

A：在RFC 2328中对DD报文的Interface MTU规定是要填入本链路的MTU大小的（v-link只好填为0），而且要求如果收到对方的DD报文中的Interface MTU大于本端的，这个DD报文就要被丢弃，那么邻居状态就会停留在“Exchange start”下。

事实上以前的IOS实现确实检查比较严格，VRP与其互操作会遇到问题。各主要厂商也意识到这个问题，在IOS后续的版本在接口下增加了ip ospf ignore mtu的命令，忽略对interface MTU的检查，这样并不需要两端链路配置的MTU完全一致，也可以正常达到Full状态，完成数据库同步。当然要注意，两端的MTU不能相差太悬殊。和IS-IS的hello padding要求达到链路MTU-1可以抑制样，这也是协议追求完美和现实应用的妥协。

OSPF LSA中的TOS字段做什么用的？

A：本来的意图是希望做Qos Routing用的，简单的说就是对于不同的TOS值，链路可以配置不同的cost，以实现基于TOS的路由，即目的网络相同但不同TOS值的IP报文做不同的路由。但在RFC 2328中取消了这个应用，只支持TOS 0，也就是只基于目的IP地址的路由。

但值得注意的是，从Cisco Networker 2003开始，到Networker 2004出现专门的MultiTopology Routing专题。很明显的，由于Qos这几年的兴起，Cisco IOS也再考虑重新利用TOS字段，来做Qos Routing。感兴趣的同事可以看下该文档。

是否有什么比较好的OSPF模拟程序可以用于OSPF的测试和细节学习？

A：你可能会想到测试仪器，比如Spirent的SmartBits，AX4000；Agilent的Router Tester，不错，它们确实都有OSPF的模拟程序，但我想他们不是设计用来细致的测试和学习OSPF的，做性能测试的模拟比较好用，而且设备昂贵，不是随手可得到。我认为在华为，最好还是几年前我让宋新娟写的OSPF Tester程序（集成在某个版本的测试平台VTP中）。它可以非常细致的模拟OSPF的所有的状态，协议报文中的字段可以任意填写，几乎可以用来测试OSPF的每个细节。虽然程序有些瑕疵，我还是诚意推荐。有人抱怨OSPF Tester比较难用，其实可能是对OSPF的细节并不非常精通，也并没有足够的诚意和好奇心来深入了解OSPF。在我看来，深入测试或了解OSPF，这个测试程序基本没有什么多余的。

此外，较常见的也有OSPF RFC 2328作者J. Moy写的OSPFD程序，但我没有用过。大家可以试下。Linux下也有一些OSPF的模拟程序可以使用。

在MPLS/BGP VPN环境中，PE-CE之间路由协议用OSPF，站点的区域号编码有什么限制吗？

A：如果按IOS的实现方式，一个起码的要求是，如果某个站点（site）的OSPF有区域0（骨干区）的话，必须保证PE-CE之间的链路也属于区域0。这是由于在MPLS/BGP VPN环境中，实际上OSPF将MPLS区域当作了一个super backbone来处理。VRP也是如此。

什么是OSPF sham-link，有什么用？

A：在MPLS VPN配置的时候，OSPF经常用来作为PE-CE中间的路由协议，以通过MPLS骨干网络连接VPN内的各站点。虽然OSPF PE-CE之间的连接保证了VPN站点间的连通性，但VPN站点间的backdoor后门链路也应该考虑。如果两个站点属于同一个site，那么经过后门链路的路径总是会被优先选

择，因为按OSFP协议规定，intra-area的路径总是优于inter-area路径。因此，必须考虑存在后门链路时，能通过策略来控制路由。

如果后门链路只是用来作为备份使用而不参与VPN业务提供服务，缺省的处理流程将不可接受。为了重新通过MPLS VPN骨干区建立站点间的连接，必须在相关PE路由器的入口和出口VRF间建立逻辑的intra-area链路。本特性提供一种解决方案，在两个站点间建立一个假OSPF链路：sham-link，作为intra-area通道，以使得两个站点间的之间的通讯通过MPLS骨干区域，而后门链路作为备份使用。如果两个站点间不存在后门链路，没有必要使用sham-link。

OSPF有什么非常好的书和文档可以深入学习？

A：我认为入门的话，听一次精彩的OSPF培训是个很好的方法，对于小文档，最精彩的应该是Cisco的OSPF Database Explanation Guide，再深入点的入门，也就是Jeff Doyle先生大作Routing TCP/IP volume 1 中的OSPF部分了。但如果你要测试或开发OSPF，很好的参考书有如下：

1.RFC 2328：OSPF v2

当然，你希望自己是个专业的一流的工程师，作为OSPF v2标准的RFC 2328是必须打印下来仔细读几遍，然后放在案头，随时备查的。

2.OSPF协议剖析（OSPF：Anotomy of an Internet Routing Protocol） John T. Moy

本书是OSPF协议RFC的作者John T. Moy的力作，包括了OSPF的起源、现状以及未来发展，对OSPF协议的来龙去脉和原理有一个很清晰的阐述。如果想了解OSPF的一种实现和使用OSPF的仿真程序。请看本书的姊妹篇《OSPF协议完全实现》。本书2002年8月由电力出版社出中文版。我记得我拿到本书的时候，是晚上10点看到凌晨4，5点看完的。

3.OSPF网络设计解决方案（第二版）

本书全面讲述OSPF网络的设计、配置、管理和疑难解析。全书分为三个部分：第一部分讲述OSPF基本原理，内容包括网络和路由选择基本原理、OSPF简介、OSPF通信；第二部分讲述OSPF路由选择和网络设计，内容包括设计基本原理、路由选择的概念和配置、重发布、汇总；第三部分讲述OSPF的实现、疑难解析及管理，内容包括管理和保护OSPF网络、OSPF疑难解析、OSPF网络中的BGP和MPLS。本书最后的附录部分还提供了关于OSPF的RFC文档资料。

4.Cisco OSPF 命令与配置手册

本书是一本简洁而完整的OSPF命令手册。书中提供了很多示例场景，演示了可在由最少数量的路由器组成的网络环境中实施的每条OSPF命令的正确用法。我一直认为并强调，在华为你所接触到的Internet核心协议技术，Cisco IOS对应的配置手册和命令行手册，你必须精读。

以上基本是按RFC 2328 OSPF v2的介绍，如果你想了解OSPF的诸多新的进展的话，请阅读：《OSPF新特性指南 v1.1》，文档URL在：http://tech.huawei-3com.com/article.php/838。

不要指望很短的时间内，就真的精通OSPF，在平时的工作中多磨练，多利用前人的经验和总结，你会发现：

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

i-SPF是个什么东西，有什么用？

A：所谓的i-SPF就是Incremental-SPF，增量SPF算法，是在原来的普通SPF算法改进以空间换时间，以达到加快SPF的计算速度，提高OSPF及IS-IS的收敛速度的目的。核心思想在于当路由变化时判断其对SPF树的影响，对于不涉及SPF树的变化则不需要计算，对于叶结点的增加和删除也可以实现增量计算但有时如一个down掉的链路并不在SPT上时，并不需要重新计算SPF。可以通过在SPT上保存更多的信息如：父亲列表，邻居列表来实现，基于这些增强的信息和变化，SPF再来计算反映变化。VRP3.0未实现，V5和IOS较新的版本都支持。

OSPF从1998年发布新的v2标准RFC 2328以来，有哪些新特性和发展？

A：其实OSPF不是没有发展变化。从基本的RFC 2328到现在，IOS针对网络应用发布了很多新的特性，我认为，如果你希望自己是个OSPF真正的专家，这些都是应该了解的。很幸运的是我曾经整理了《OSPF新特性指南 v1.1》，文档URL在：http://tech.huawei-3com.com/article.php/838。

文档分析了Cisco IOS最新版本IOS 12.3T和12.0S(截止2004-10-28为IOS 12.3(11)T，12.0(29)S)版本做的一些OSPF的新特性，尤其侧重于目前VRP3.0没有实现的特性，可以作为公司同事与时俱进了解OSPF新进展和模块发展规划的参考。其中的某些特性应该在VRP5上已经实现。大家可以继续关注OSPF更多的新的应用和发展，比如OSPF对MTR的支持等（可以参考Cisco Networker 2004中的相关文档）。

OSPF vs. IS-IS到底谁更好？

A：两年前IS-IS开始在中国大行其道时候很多人问的一个问题。中国电信的技术总结中也多次提到过它。事实上，专门讨论这个问题的文档不少的，最好的可能算Dave Katz，Juniper Networks 写的 IS-IS and OSPF : A Comparative Anatomy了。不过最后的结论是：They're both good protocols. Use the one that makes the most sense to you.

用句很俗的话，OSPF和IS-IS各有优缺点，而且非常近似，如何选择可能确实是个人喜好了。IS-IS的配置不容易出错，比较少耗资源，能支撑更大的网络应该是不争的事实了。

是否还有其它比较好的OSPF FAQ？

A：当然。本文确切的说应该是OSPF FAQ for Huawei，列出在华为的工程师经常询问的OSPF问题，解答也带有较强的个人观点。

Cisco的文档一向写的很好很全。我做新员工的时候就看过Cisco OSPF FAQ，现在已经上载到我们的内部技术网站，但要注意Cisco可能会更新这个文档。URL在：http://tech.huawei-3com.com/article.php/1053。文章中大致有23个FAQ。不过相信本文是不逊色Cisco对应的这个文档的。建议大家都阅读下。

不得不提的是，受人尊敬的J. Moy先生在他的大作 OSPF Anatomy of an Internet Routing Protocol（中文版本 OSPF协议剖析）中，Chapter 8 OSPF FAQ中也总结了早期IETF OSPF mail-list中常见的一些问题。

此外，对于MPLS/BGP VPN环境下，PE-CE用OSPF做路由协议，VRP团队也有一个案例集OSPF VPN 配置异常案例集可以当做FAQ。URL在：http://tech.huawei-3com.com/article.php/861

OSPF中的最短路径算法

鉴定测试中心 陈旭盛

Dijkstra算法介绍

在数学上，以某个节点为起点，计算到其他节点的最短路径的算法，称为“单源最短路径”算法。求“单源最短路径”的问题在数学上可以精确描述如下：

“单源最短路径” 问题：已知一个有n个节点(V0..n)构成的有向连通图G＝（V，E），以及图中边的权函数C（E），其中V代表节点集合，E表示所有边的集合，并假设所有权非负，求由G中指定节点V0到其他各个节点的最短路径。

Dijkstra算法是很经典的求解上述问题的算法，其基本想法是设计一种最短路径树的构造方法，按非降次序逐条构造从V0到各个节点的最短路径，第一步找到和V0相距最短的节点以及到这个节点的路径，第二步找到和V0相距次短的节点以及到这个节点的路径，如此反复，最后找到V0到所有节点的最短路径，构造出整棵最短路径树。

对上述构造方法的一个直观考虑是：和V0相距最短的节点应该在和V0直接相邻的节点中，和V0相距次短的节点要么在和V0直接相邻的节点中，要么在和这些相邻节点相邻的节点中，如此逐步扩散考虑，应该就可以找到和V0相距最短、次短、…….第n短的节点以及对应的路径，而且因为是连通图，最后肯定所有节点都能全部考虑到，也就能完成整棵最短路径树的构造。

事实上，上述直观考虑是对的，Dijkstra算法是对上述过程的一个提炼和优化：和V0相距最短的节点是和V0直接相连的节点没错；相距次短的节点范围可以缩小为，和V0直接相邻的节点，加上跟刚选中的最短节点直接相邻的节点；相距第三短的节点的范围可以类推得到，即在上一步考察的节点的基础上，加上和次短节点直接相邻的节点。如此逐步构造，可以按非降次序找到到所有节点的最短路径。

为了从数学上精确描述上述构造过程，引入了集合的概念对节点和路径进行分类。

我们把节点分成两个集合：

A：已经选入最短路径树的节点的集合。

B：剩余的其他节点的集合。

对于路径，我们分成三个集合：

I：已经选入最短路径树的路径的集合

II：候选路径集合：下一条加入最短路径树的路径将从这个集合中选入。

III：剩余的其他路径的集合（被废弃的路径或者还未考虑的路径）。

为了更好的理解，有必要对这里的路径定义进行一下强调：路径是指以V0为起点，其他节点为终点的由一条或多条边组成的一个有序集。边，可以理解为路径中的一段，只有到和V0直接相邻的节点的路径才直接对应一条边。从V0到所有节点，都可能存在一条或多条路径，非最短路径在计算过程中将会被废弃，放入集合III。

从前面的描述中可以明显看出，Dijkstra算法是一个递归构造过程，因为任何递归都必须有明确的初始状态，所以我们有必要先得到上述Dijkstra算法中定义的集合的初始值：

以V0为起点计算最短路径的话，初始状态时显然有且只有V0在集合A中，所以集合A的初始值为V0。集合B的初始值为剩余节点。

前面提到过，下一个加入集合A的节点，一定是和V0直接有边相连的节点，因此，加入最短路径树的第一条路径也必然在这些和V0直接相连的边所代表的路径中产生，所以集合II的初始值就是和V0直接相连的边构成的路径。另外，初始状态最短路径树为空，所以集合I的初始值为空。集合I、II明确了的话，集合III自然明确。

下面我们开始展开递归构造最短路径树的过程：

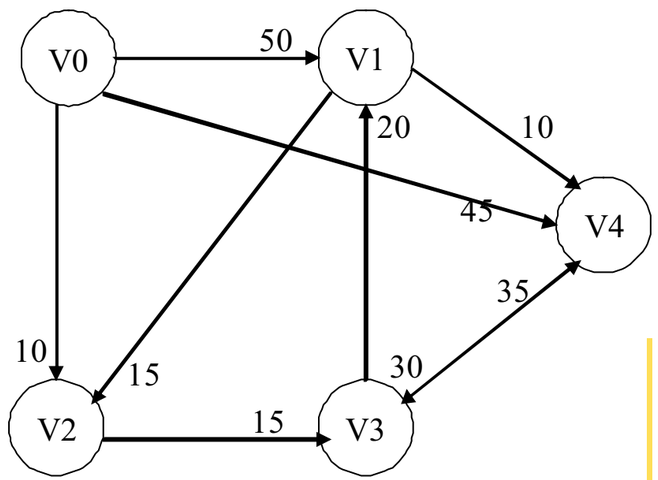
第一步：从集合II中选择一条最短的路径，放入最短路径树，相应的，这条路径的终点对应的节点（这里记为X）应该从集合B移入集合A。

第二步：考察所有从X出发的边的终点，考虑其中不属于集合A的节点，这里记为Y，计算从V0出发经X到达Y的路径值，计算方法为：最短路径树中V0到节点X的路径值加上（X，Y）这条边的值。为了描述方便，我们把从V0出发经X到达Y的路径记为（V0X）Y。接着考察集合II中的候选路径，如果其中没有到节点Y的路径，则直接把路径(V0X)Y作为候选路径加入集合II；如果集合II中已经有到节点Y的路径，则进行比较，如果这条路径值小于或等于路径（V0X)Y的路径值，那么路径(V0X)Y作为被废弃的路径放入集合III，否则原集合II中到Y的路径被废弃放入集合II，（V0X）Y作为候选路径放入集合II。对于Y节点有多个的情况，按第二步的方法一个一个的计算和比较。

重复第一步和第二步，直到集合II和集合B为空。

第二步事实上是对候选路径的一个重新计算过程，因为在集合A中引入了新的节点后，考察的范围变大了，很可能在原来节点范围内认为到某个节点的最短路径已经不再是最短路径了，这时候，需要根据第二步的方法进行调整。

为了让大家更好的理解这一构造过程，这里举一个较为典型的例子，如下是一个有向图：



计算这个有向连通图的最短路径树的过程如下：

候选路径集合	路径长度	最短路径树中的节点(集合A)	最短路径树	说明
V0V1 V0V2 V0V4	50 10 45	V0	Null	在初始状态，最短路径树中只有节点V0，候选路径就是V0直接相连的边代表的路径。
V0V1 V0V4 V0V2V3	50 45 35	V0、V2	V0V0 V0V2	V0V2在所有候选路径中最短，所以放入最短路径树，V2放入集合A。考察所有以刚放入集合A的节点V2为起点的边的终点，对其中不在集合A中的节点，这里只有节点V3，计算从V0出发经节点V2，到达V3的路径V0V2V3的值，因为候选路径中没有到V3的路径，所以V0V2V3路径直接放入候选路径集合。
~~V0V1~~ V0V4 V0V2V3V1 ~~V0V2V3V4~~	50 45 45 55	V0、V2、V3	V0V0 V0V2 V0V2V3	V0V2V3在所有候选路径中最短，所以放入最短路径树，V3放入集合A。考察所有以刚放入集合A的节点V3为起点的边的终点，对其中不在集合A中的节点，这里是节点V1，V4，计算从V0出发经节点V3，到达这两个节点的路径V0V2V3V1和V0V2V3V4的路径值，并和候选路径中已有的到达V1、V4的路径值进行比较：V0V2V3V1小于V0V1，所以V0V1从候选路径中删除，V0V2V3V1放入候选路径集合；V0V2V3V4比V0V4大，所以V0V4在候选路径中保留，V0V2V3V4路径废弃不用。
V0V2V3V1	45	V0、V2、V3、V4 V0、V2、V3、V4、V1	V0V0 V0V2 V0V2V3 V0V4 V0V0 V0V2 V0V2V3 V0V4 V0V2V3V1	V0V4和V0V2V3V1路径值相等，任意选择V0V4放入最短路径树，V4放入集合A。考察所有以刚放入集合A的节点V4为起点的边的终点，其中不在集合A中的节点没有（虽然有边V4V3，但V3已经在集合A中了），所以不进行选择和计算。 候选路径V0V2V3V1放入最短路径树，这时候选路径集合为空，并且所有节点已经放入了集合A。计算结束。

Dijkstra算法的证明

对Dijkstra算法进行证明，实际上就是要证明其构造过程构造的树一定是最短路径树。为了给出证明，我们先分析一下构造过程。

分析构造过程的第二步，可以得出结论一。

结论一：对一个还没有选入集合A的节点Y，其候选路径是所有从V0出发，中途只经过集合A中的节点到达Y的路径中路径值最小的。

这个结论根据第二步候选路径的构造过程，很容易得出：因为在第一次构造到Y的候选路径时，从V0出发经刚加入集合A的节点到Y的路径，是被直接放入候选路径集合中的，这说明第一次构造的到Y的候选路径满足条件“从V0出发，中途只经过集合A中的节点”。另外，集合A中每加入一个新节点，都会考虑从V0出发经过这个新节点到达Y的路径，并和原候选路径比较，选择两者中较小的那个，这种过程一直持续到Y被选入集合A中为止。这个过程显然保证了Y的候选路径一直保持了特性“从V0出发，中途只经过集合A中的节点”，而且因为遍历了所有A中的节点，所以是所有这种特性路径中最短的。

有了结论一，证明Dijkstra算法可以弱化为证明结论二。

结论二：假设构造过程中下一步将选择的是节点Y，那么这时到达Y的最短路径必然是从V0出发，中途只经过集合A中的节点到达Y的路径。

如果“结论二”成立的话，结合“结论一”，说明Y的最短路径和候选路径具有相同的属性“从V0出发，只经过集合A中的节点”，而“结论一”中明确说明，候选路径是这种属性的路径中的最小者，所以对于下一步即将选入集合A中的节点Y，其最短路径值就是候选路径，也即证明了算法中构造最短路径树过程的正确性。

下面我们证明“结论二”。

证明很简单，使用反证法：假设到达Y的最短路径中途不只经过集合A，那么必然经过一个不在集合A中的节点W，于是这条路径中肯定包含一条从V0到W的路径，这条路径显然比从V0出发经过W到Y的路径更短，而Y是下一步要放入集合A中的节点，根据我们按非降次序构造最短路径树的过程（第一条最短，第二条次短…..），从V0出发到W的这条路径应该已经在最短路径树中了，也即节点W应该已经在集合A中，矛盾，得证。

OSPF协议中对Dijkstra算法的使用

从理论上来说，只要描述清楚了节点之间的连接和边的属性，就描述清楚了有向图，也就可以使用Dijkstra算法计算出最短路径树。

对于使用基于Dijkstra算法的路由协议来说，如果能描述清楚整个网络拓扑（对应有向图），并让区域内的每台路由器都清楚的知道整个区域的网络拓扑，则每台路由器都可以独立的计算出最短路径树。从这个意义上说，对于一个使用Dijkstra算法的路由协议来说，需要要解决以下问题：

网络拓扑的描述问题。

网络拓扑描述信息的传播问题。

效率问题：路由协议的目的是在耗费最少资源的情况下，在最短时间内动态发现到其他网段的最优路径。

另外，还需要考虑路由协议的网络应用位置（IGP或者EGP，适合于多大网络等）以及和其他路由协议的互操作等问题。

以上几个问题有一定的关联性，互相影响。例如，网络拓扑描述的优化可以减少描述信息，从而减轻传播和计算过程的消耗，提高了效率。

本文的着力点是Dijkstra算法及其在OSPF中的应用，所以这里只说明与之相关的网络拓扑的描述和最短路径计算两个内容，网络拓扑的描述也只涉及的几类基本LSA。

OSPF对网络拓扑的描述

OSPF中使用链路状态通告（LSA）来描述网络拓扑（即有向图）。

OSPF中，Router LSA被用来描述路由器（节点）之间的连接和链路（边）的属性，具备了理论上计算最短路径的可能。但是仅仅是理论而已，从实践的角度，针对特殊的网络情况和应用场景，需要作一些优化工作。

先考虑一个有n个节点的广播网，如果使用Router LSA来描述的话，需要描述n个节点两两相连的n*(n-1)/2条链路，而且n个节点间需要建立建立n*(n-1)/2个邻接关系，描述信息需要在这些建立了邻接关系的节点间传播。如果换一种思路，我们把这个广播网虚构成一个伪节点，其他n个节点均和伪节点相连，那么就只要描述n条链路，n个节点只需要和伪节点建立总共n个邻接关系即可，能大大减少了信息描述量和信息传播量。依据这种想法，OSPF中针对广播网的描述进行了优化，使用指定路由器DR来承担这个伪节点的角色，并使用Network LSA来描述广播网内DR和各个路由器的连接情况。

对于NBMA网络的描述，处理方式和广播网基本相同。

其次，OSPF的设计目标是为一个较大的内部网络提供动态路由能力。如果内部网络较大的话，需要描述的链路会很多，存储、传播和计算这些链路信息将耗费大量内存和CPU资源。OSPF解决这个问题的办法是把较大的网络划分成多个Area，每个Area内部使用Router LSA和Network LSA把Area内部网络拓扑描述清楚，并据此使用Dijkstra算法计算出本Area内的最短路径树和路由。至于Area外的网络拓扑，区域边界路由器ABR并不把相邻Area的Router LSA和Network LSA传入本Area，而是把自己在相邻Area范围内计算得到的该Area内各个网段的路由进行汇总，把这些网段当做直接连接在自己上面，并生成Network Summary LSA来对这些网段进行描述，传入本Area。这样，对于一个有n个网段和多台路由器的相邻Area，ABR只需要生成n条网络描述信息并传入本Area即可，大大减少了进入本Area的链路描述信息，减少了存储、传播和计算消耗。需要注意的是，划分Area的做法导致了Area内部路由器中的链路状态数据库描述的Area外部分并不是真实的网络拓扑，从而计算时可能出现环路，为了避免环路出现，要求所有的Area之间的相连必须经过Backbone区域即Area 0。

另外，OSPF被设计为一个IGP，所以除了处理本自治系统内的路由外，还需要处理自治系统外的路由，这类路由在ASBR处引入，可以看做是直接连接在ASBR上，OSPF引入AS External LSA来描述这类自治系统外路由。另外，因为AS External LSA在传入其他Area时并不在ABR上重新生成，所以从计算的角度看，其他Area还必须知道自治系统外路由从哪个节点引入，所以需要引入ASBR Summary LSA来描述外部路由从哪个节点引入，ASBR Summary LSA在ABR上生成，从其他Area看，可以认为于ASBR直接连在ABR上，自治系统外部路由对应的网段又直接连在ASBR上。

OSPF中最短路径的计算

下面简单介绍一下OSPF的最短路径计算过程，具体的细节处理请参考RFC2328：

首先，计算Area内部路由。因为Router LSA和Network LSA精确的描述出了整个Area内部的网络拓扑，所以根据Dijkstra算法，可以计算出到各个节点（路由器）的最短路径。然后根据Router LSA描述的与路由器相连的网段情况，就得到了到各个网段的最短路径。注意，计算过程中，如果有多条代价相同的最短路径，都保留。

其次，计算跨Area的路由。因为从一个Area内部看，相邻Area的路由对应的网段就好像是直接连在ABR上，而到ABR的最短路径已经在上一步算出，所以直接检查Nerwork Summary LSA，就可以很容易得到这些网段的最短路径值。另外，ASBR也被看做是直接连接在ABR上，所以到ASBR的最短路径也可在这一步计算出来。注意：在这一步，如果发起计算的路由器是ABR，那么很自然，应该只考虑骨干区域的Network Summary LSA；但是对于启用了Virtual Link的拓扑来说，跨Area的路由只是逻辑上通过骨干区域，而实际是通过Transit Area到达的，因为逻辑链路可能并不是物理上最优的，所以在连接到Transit Area的ABR上，考察骨干区域的Network Summary LSA得到的跨Area路由的路径可能不最优，还需要考察Transit Area的Network Summary LSA，以得到最短路径。

最后，计算AS外部路由，因为AS外部路由可以看做是直接连接在ASBR上，而到ASBR的最短路径在上一步已经计算出来，所以逐条检查AS External LSA就可以得到到各个外部网络的最短路径。

下一跳和出接口

路由表中还有两个重要元素：下一跳和出接口。因为确定这两个元素和计算过程有一定关系，这里也提一下。

其实，一个简单事实是，对于任何一条路径来说，其下一跳和出接口，必然和他的父路径的下一跳和出接口相同。因此，求所有路径的下一跳和出接口，最后可以归结为求到“和V0直接相连的节点”的下一跳和出接口。对于OSPF来说，因为使用了DR模拟伪节点，所以“和V0直接相连的节点”是一台路由器或者是一个伪节点（即广播网、NBMA网络）。

前面Dijkstra算法已经提到，求到各个节点或网段的最短路径的过程，是一个候选路径的构造和调整过程，在这个过程中，候选路径的父路径会进行多次调整，但其根都跳不出和“V0直接相连的节点”这一范围，因此其下一跳和出接口也在“和V0直接相连的节点”的下一跳和出接口的范围内，在计算过程中继承即可。

确定“和V0直接相连的节点”的下一跳和出接口比较简单，这里不再赘述，想了解的话可以参考RFC2328的16.1.1节。

还有一类特殊的网段是发起OSPF计算的节点的直连网段，其下一跳和出接口的确定就更简单了，而且不会被其他路径继承。

OSPF Graceful Restart

鉴定测试中心 朱皓

关于Graceful Restart

简介

Graceful Restart简称GR，字面意义是平滑重起，主要实现的能力是在路由协议重起的时候保证数据转发的正常，以保证关键业务不中断。

GR技术是属于高可靠性（HA, High Availability）技术的一种。HA是一整套综合技术，主要包括冗余容错、链路保证、节点故障修复及流量工程。GR是一种冗余容错技术，目前已经被广泛的使用在主备切换和系统升级方面，以保证关键业务的不间断转发。

目前VRP3.x并不支持GR，在VRP5.x中会提供支持。

GR的实现前提

随着网络设备普遍采用了控制和转发分离的技术，GR技术就成为了可能。在传统的路由器中控制和转发是由同一个处理器（RP：Route Processor）完成的，这个处理器既通过路由协议发现并维护路由，同时也维护着路由表和转发表。为了提高设备的转发性能和可靠性，中高端设备普遍采用了多RP的结构。负责路由协议等控制模块的处理器一般位于主控板，而负责数据转发的处理器则位于线卡上。这样在主处理器重起的时候才有可能不影响线卡上的数据转发。

基于上面的原因，目前实现GR的设备都需要有以下特征：双主控结构，并且线卡具有自己独立的处理器和内存等。

传统技术和GR技术的比较

在没有使用GR的时候，因为各种原因出现的主备切换，都会造成短时间的转发中断，并且在全网造成路由振荡。通过图1可以看到问题所在。

如果R1是自治系统的ASBR，那么会通过BGP引入大量外部路由，当R1发生主备切换时，会造成下面的影响：

所有外部路由丢失，当R1恢复时，再次学到路由，造成网络路由振荡。

在R1的备份主控单元完成路由计算和发布之前，所有通过R1的数据转发中断。

R1的所有邻居需要重新进行路由计算，并改写转发表。

当R1恢复正常时，R1的所有邻居需要重新进行邻居协商和交互，并计算路由，改写转发表。

对于一个大型网络，尤其是运营商网络，对于这些路由振荡和业务中断是不可容忍的。

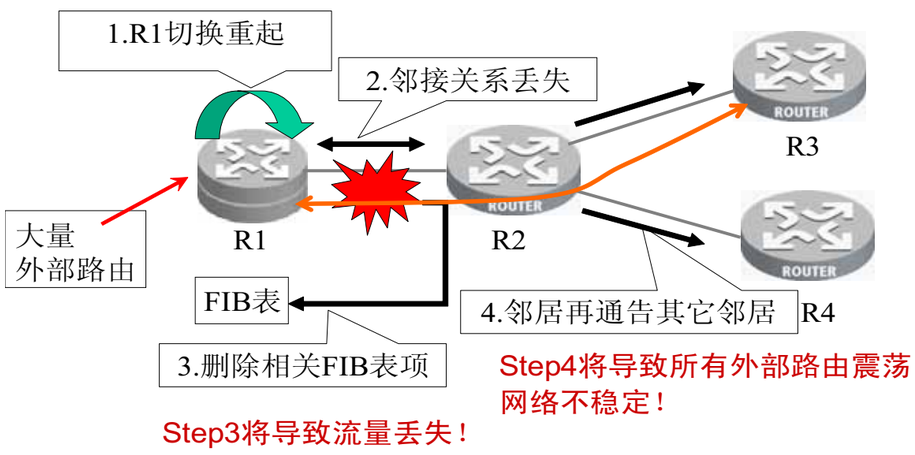


图1 传统主备切换时造成的网络问题

而使用GR技术则可以解决全部的问题。在R1与其它邻居建立邻接关系的时候，就会进行能力的协商。这样在R1进行切换的时候，它的邻居会维护邻接关系不变，并保持路由的稳定和正常的转发。除了R1的邻居，网络中的其它设备并不知道R1进行了协议的重起。

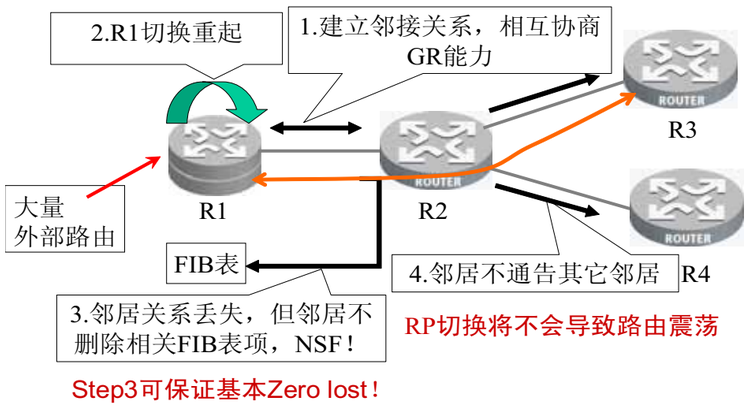


图2 采用GR技术解决了主备切换时造成的网络问题

当然，如果R1在GR的过程中，网络拓扑发生了变化，那么可能会导致短暂的路由环路或路由黑洞。对于这个问题的处理方法，目前不同厂商的实现略有不同。与业务中断的危害相比较，是否可以忍受可能出现的，短暂的路由环路和路由黑洞，对这个问题的不同观点决定了在具体实现上的不同。

小节

实现GR的前提

GR特性仅适用于带双主控、线卡具有立处理器的硬件平台。

GR用于软件或硬件错误导致Active RP重起或Active RP失效；或者管理员的主备切

换命令。

GR解决的问题

减少单点故障，提高整网可靠性。

减少对重要业务的影响。

减少RP切换时路由震荡对全网的影响。

减少丢包：基本可以达到丢包率0%，无丢包！

OSPF Graceful Restart

GR对路由协议的要求

本文主要讨论OSPF中的GR技术。众所周知，OSPF是链路状态协议，邻居间需要维护链路状态信息库LSDB的同步，并根据LSDB计算路由。因此为了完成GR，对OSPF有下面的要求：

在GR的过程中，邻接状态不能改变，必须维护在FULL的状态。

GR的周期是可选的，默认情况下使用RouterDeadInterval。

在GR过程中，必须保持转发表FIB不变，可以做相关标记。

当某些原因导致GR中断时，需要进行标准的协议重起过程，重新进行邻接的建立。

需要对OSPF做相应的扩展以支持GR特性。

GR中的角色

从前面的介绍中可以看出，对于OSPF而言，一个完整的GR过程，需要至少2个设备完成，而这两个设备可以具有相同或者不同的能力。因此，根据能力划分：

GR Capable路由器

有GR功能的路由器：通常配备双RP，能在RP切换的时候，通告周边邻居，保持自己的转发表，切换后重建路由表。

GR Aware路由器

GR感知路由器：能懂得GR路由器的信令，配合在GR路由器重起期间保持邻接关系和通过GR路由器的转发表项。GR Aware设备只是需要识别GR信令，因此可能不具备有双RP，不能进行GR。但它可以配合有GR能力的设备来完成GR过程。毫无疑问，有GR能力的设备当然是GR感知的设备。

GR Unaware路由器

不能感知GR的路由器：不懂GR信令，不能协助GR路由器在RP切换时不间断转发，更不能执行GR。一般来说是系统软件没有GR特性或者GR特性被关闭。

从GR中完成的任务来看，分为GR Restarter（就是协议重起的设备）和GR Helper（就是协助完成协议重起的设备）两个角色。

GR的流程

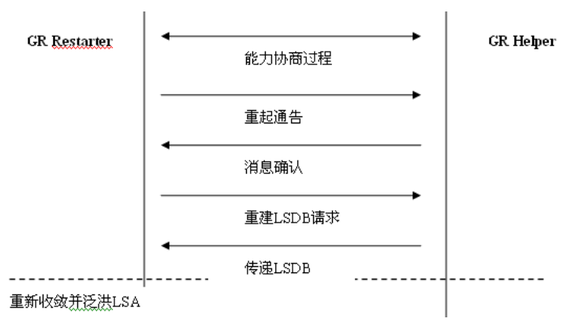


图3 GR的流程

OSPF GR的实现标准

目前有两种主流的实现标准：

RFC 3623：Graceful OSPF Restart Juniper，Redback，Nexthop，Procket等厂商采用的实现方法。

Cisco IOS，IETF草案包括

draft- nguyen-ospf-lls-05.txt

draft-nguyen-ospf-oob-resync-05.txt

draft- nguyen-ospf-restart-05.txt

下面我们对这两种实现分别进行介绍。

RFC 3623：Graceful OSPF Re-start

RFC3623中定义的GR有两个重要的原则：网络拓扑要保持稳定；重起协议的路由器可以在重起过程中保持转发表。否则需要终止GR过程，进行标准的OSPF协议重起。

在GR的过程中，定义了GR Restarter和GR Helper的行为。GR过程是通过GR Restarter发出一个Grace-LSA的通告开始的。

GR对OSPF的扩展

RFC3623定义了发起GR的信号，Grace-LSA，该LSA是一个Opaque LSA，所以需要对OSPF进行扩展，以支持Opaque LSA。Opaque LSA是通过RFC2370进行定义的，这里我们稍做一下回顾。

RFC 2370：The OSPF Opaque LSA Option 定义了新的Opaque LSA类型，以提供扩展OSPF能力的通用机制，共定义了3种类型，包括：Link State Type 9，10，11。

Type 9：link-local scope，只在本地链路（子网）传播；

Type 10：area-local scope，在本区域内传播；

Type 11：AS scope，在OSPF进程运行的路由域内传播。

原Link State ID分为Opaque Type（1个字节）和Opaque ID（3个字节）两部分。

承载信息部分：由TLVs结构组成，没有特别定义LSA内容以提供通用机制。

根据各种类型OLSA的使用范围可以看出，GR技术中的Grace-LSA应该是使用类型9，是一个本地链路的LSA。对于Grace-LSA，Opaque Type字段为3，Opaque ID字段为0。在Grace-LSA的TLVs中，主要包括GR周期、发生GR的原因和GR Restarter的接口IP地址。

对于GR Restarter行为的约定

GR Restarter设备在GR期间：

设备不会产生LSA；当收到源于自己的LSA时，不做处理，并认为有效。

在标准的OSPF过程中，当设备收到一个源于自己的LSA，会认为是上次重起前遗留的LSA。这个时候设备会生成一个序列号更大的LSA并泛洪或者提前老化该LSA。但是在GR过程中，设备通过从邻居获取LSA来得到重起前的网络拓扑，因此这时候只是接受，并认为有效。

不能下发路由项到转发表。

GR设备通过侦听接口上的HELLO知道自己是否为DR，如果是DR，还会保持。

设备收到从邻居发来的hello报文，如果其中DR为自己，那么会保持在自己的hello中宣告自己为DR。

GR Restarter设备进入GR前的准备工作

保证转发表的准确，并在重起过程中被保护。

解决保存各接口所使用的LSA序列号的问题。比如用时钟来产生序列号。这样可以保证在重起后序列号得到延续。

产生Grace-LSA，age=0，并给出GR周期。必须采用某种机制来确认邻居已经收到了Grace-LSA（采用OSPF同步LSDB的机制）。并将Grace-LSA中相关的参数保护起来，比如GR周期。需要注意的是，邻居确认收到Grace-LSA并不代表它一定具有帮助者的能力或者会帮助完成GR过程。即使一个不支持GR特性的设备，也会发出收到LSA的确认信号，但它不能帮助完成GR过程。

GR Restarter退出GR的条件

已经重建了所有邻居（可以通过从邻居那里获得的1、2类LSA中知道所有的邻接关系）

GR设备收到一个不合逻辑的LSA。比如收到的LSA中不包括自己，或者自己的LSA中不包括这个邻居。这说明网络拓扑发生了变化。

GR周期超时。

GR Restarter退出GR后的过程：

无论GR过程是否完成，都会进行标准的OSPF重新泛洪和路由计算，按照当前的链路状态产生LSA并泛洪，根据LSDB计算路由并下发到转发表。必须要完成的下面这些工作：

重新在所有关联区域泛洪1类LSA。

如果是DR，重新泛洪2类LSA。

重新计算路由并下发到转发表，产生必要的3、5、7类LSA。

删除残存在转发表中不再有效的路由。

向区域内刷新那些源于自己的LSA。

刷新掉Grace-LSA。

对于GR Helper行为的预定

GR Helper在GR期间的任务

监视网络变化，如果没有变化，就保持宣告与GR Restarter的邻接状态不变。

GR Helper进入GR前的检查

目前与GR Restarter设备在进入帮助状态的接口上是FULL的状态。

网络拓扑没有变化。检查的方法是查看与GR Restarter设备连接的重传列表（半小时一次的刷新周期除外）。如果有网络变化，那么拒绝进入帮助模式。对于LSA变化的定义如下

Options字段发生了改变。

LSA的age被设为了最大

LSA头部的长度字段发生了变化

除头部以外的内容发生了变化（除去头部目的是为了不考虑校验和及序列号）

GR周期没有过（根据Grace-LSA的age字段来判断）。

本地策略允许成为GR Helper。

本身不在GR过程中。

例外是，如果已经是某个GR Restarter设备的帮助模式，那么收到新的Grace-LSA必须接受并且因此更新GR周期。

GR Helper可能与GR Restarter设备有多个邻接关系，如果在其中的一个连接上拒绝进入帮助模式，会导致问题。因为不进入帮助模式将宣告一个邻接结束。而这样会导致LSA的更新，并被理解为网络拓扑发生了变化。

GR Helper退出帮助模式的条件

Grace-LSA被刷新了（说明GR Restarter设备已经完成GR）。

GR周期过期。

LSDB显示网络发生了变化，也就是LSA变化了。并且要满足这样的条件，如果FULL的关系还存在，那么该LSA是会被泛洪给GR设备的。也就是说这个网络拓扑的变化会影响到GR Restarter对网络设备拓扑的理解。举个例子（图3）：如果Y是一个ABR，并且是GR Helper的角色。X属于一个STUB区域，是GR Restarter的角色。如果Y收到一个新的5类LSA，那么并不会终止GR过程，因为这个LSA任何时候都不会泛洪给X。

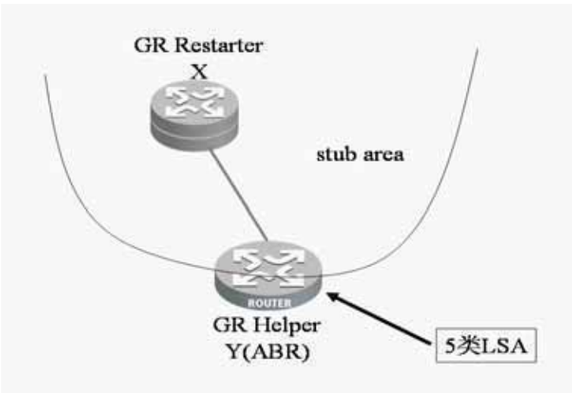


图4 GR Restarter对网络拓扑变化的理解

GR Helper退出帮助模式后的动作

重新计算DR。

重新泛洪1类LSA。

如果是DR，重新泛洪2类LSA。

如果连接了VLINK，在传输区域产生1类LSA。

关于非计划性重起的GR

通过前面的介绍，我们发现，一个GR过程似乎是有计划进行的，那么很多时候是因为主控板出现意外故障导致了主备切换，这时候的GR过程会有所不同。

GR技术的起源是用于有计划的重起，对于无计划的重起而言，要考虑到是否转发表能有效的得到保护。另一方面非计划的重起，并没有进行应做的准备，那么在控制软件可用时需要立刻发出一个Grace-LSA并要遵守下面的原则。

Grace-LSA必须在发送任何HELLO之前产生并发送出去。在广播网络中，这个LSA必须向所有路由器的组播地址发送，因为重起的设备并不知道它之前的DR状态。

Grace-LSA是封装在LSU报文中并送到所有的接口，不管是否存在邻接关系或者以前的邻接关系是如何的。

为了提高Grace-LSA被交付的概率，具体实现中可以进行多次的发送。

在Grace-LSA中重起原因中要设置为0（不可知）或者3（切换到备份控制器）。这样邻居就可以决定是否他们需要帮助路由器完成这个意外的重起。

Grace-LSA的报文格式

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            LS age=0           |    Options    | LS type= 9 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Opaque type=3 |          Oaque ID = 0                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Advertising Router                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     LS sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         LS checksum           |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                            TLVs                             -+
|                              ...                              |
```

每一个TLV都是4字节对齐的。格式如下：

```
 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              |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Value...                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Length描述了Value有效部分的长度。

TLV含义	T	L	V	在GR-LSA中
GR周期	1	4	以秒为单位	必须包含
GR原因	2	1	0 (未知原因) 1 (software restart), 2 (software reload/upgrade) 3 (主备倒换)	必须包含
接口IP地址	3	4	接口的IP地址	在广播、NBMA及点到多点网络类型中。

- GR导致的安全问题

一个虚假的Grace-LSA会导致设备进入GR Helper的角色，并孤立起来。
解决的方法是对OSPF的邻居交互做认证。

Cisco Nonstop Forwarding

- 标准及相关特性

Cisco IOS，IETF草案包括
draft- nguyen-ospf-lls-05.txt
draft-nguyen-ospf-oob-resync-05.txt
draft- nguyen-ospf-restart-05.txt
两个主要概念：
LLS(link-local signaling)：Cisco触发NSF的信令使用了LLS来进行。
Out-of-band LSDB resynchronization(OOB)：Cisco定义的无状态变迁的LSDB重新同步概
相关特性：
OSPF NSF（Nonstop Forwarding）目前在Cisco7200、7500、10000、12000系列路由器支持。因为7200系列路由器是单主控结构，因此只能做为一个NSF-aware。OSPF NSF目前不支持Virtual links。
OSPF NSF awareness（12.2提供），通过支持LLS特性，将NSF-aware的能力扩展到中低端路由器，包括26系列路由器都可以通过支持该特性成为一个NSF Helper。

LLS(link-local signaling)介绍

LLS是Cisco对标准OSPF报文做的扩展，LLS Data部分加在OSPF报文的尾部。在OSPF头部的OSPF报文长度不计算这部分数据，这样不兼容LLS能力的设备可以忽略该部分数据的存在。LLS特性在目前IOS版本中是缺省使能的，可以通过关闭该能力禁止NSF。
LLS只增加在hello和DD报文中，这两种报文就可以完成能力的协商过程。其中hello是不保证交付的，而DD则可以保证交付。
对hello和DD报文的option字段做了扩展，以表明是否携带了LLS Data。原option字段如下：

```
+-----------------------------------+
| * | * | DC | EA | N/P | MC | E | * |
+-----------------------------------+
```

图5 标准的Option字段

将其中的EA位作为LLS能力位L-bit。当该位置位时表示携带了LLS Data。

```
+-----------------------------------+
| * | * | DC | L  | N/P | MC | E | * |
+-----------------------------------+
```

图6 LLS扩展后的Option字段

LLS Data部分结构如下：

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Checksum         |     LLS Data Length           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                            TLVs                             -+
|                            ........                           |
```

其中通过不同的TLVs可以进行不同的OSPF扩展（从这里可以看出Cisco的这种实现方式可以用于多种OSPF扩展，而不仅仅是NSF）。在NSF中使用了一个叫做Extended Options TLV (EO-TLV)，它的类型为1（IANA规定1为Extended Options ），该类型是临时使用的，如果LLS的草案被通过，可能会有正式的类型。
EO-TLV的Value字段标明了设备是否支持OOB（后面介绍），或者是否是一个触发NSF的信令。Value字段的定义如下：

Extended Options Bit	Name	Reference
0x00000001	LSDB Resynchronization (LR)	[OOB]
0x00000002	Restart Signal (RS-bit)	[RESTART]

倒数第一个比特叫做LR-bit，当该位置位时，表示设备支持OOB能力。
倒数第二个比特叫做RS-bit，当该位置位时，表示设备要进行一个NSF过程。Cisco规定在同一时间，同一个网段中不能有多个设备同时将RS-bit置位。也就是在同一个网段内，同时只能进行一个NSF过程。

OOB-resync（Out-of-band LSDB resynchronization）介绍

与传统的LSDB同步过程不同的是，NSF需要在邻接状态不变化的情况下进行LSDB的重新同步。Cisco将这种无状态迁移的LSDB重新同步过程取名为OOB-resync，这里简称OOB。OOB的能力协商是通过LLS扩展实现的，在LLS的EO-TLV中LR-bit置位说明支持OOB。
当设备进行一个OOB过程时，需要发出一个信号，是通过对DD报文的扩展完成的。标准的DD报文如下：

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|版本号/Version#|       2       |     包长度/Packet length      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     路由器标识/Router ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     区域标识/Area ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       校验和/Checksum         |       验证类型/AuType         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       验证/Authentication                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       验证/Authentication                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     接口 MTU/Interface MTU    | 选项/Options |0|0|0|0|0|I|M|MS
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     DD 序号/DD sequence number                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                                                             -+
|                                                               |
+-              一个 LSA 头部/An LSA Header                    -+
|                                                               |
+-                                                             -+
|                                                               |
+-                                                             -+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            ...                                |
```

Cisco的扩展是使用了Options字段后面8个bit的第5位，将该位定义为R-bit，扩展后的报文如下：

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|版本号/Version#|       2       |         包长度/Packet length  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      路由器标识/Router ID                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       区域标识/Area ID                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         校验和/Checksum       |       验证类型/AuType         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         验证/Authentication                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         验证/Authentication                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     接口 MTU/Interface MTU    |  选项/Options |0|0|0|0|R|I|M|MS
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       DD 序号/DD sequence number              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-                                                             -+
|                                                               |
+-                 一个 LSA 头部/An LSA Header                 -+
|                                                               |
+-                                                             -+
|                                                               |
+-                                                             -+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              ...                              |
```

当设备收到R-bit置位的DD报文，说明邻居要进行一个OOB的过程，这时候设备会保持其邻接关系不变化，并且做一个标志RestartState flag，同时开始一个ResyncTimeout记时器。实际上这就是NSF周期，默认定义为RouterDeadInterval。如果在这个OOB的过程中从邻居收到R位清除的DD报文，那么就退出OOB过程，取消RestartState flag标志并且忽略ResyncTimeout记时器。

- Cisco Nonstop Forwarding过程

  首先通过在hello中将L-bit置位，协商LLS能力，并且通告对方自己的LLS数据。
  在LLS的EO-TLV中将LR-bit置位，协商OOB能力。
  在hello中的LLS数据部分，将EO-TLV的RS-bit置位，通告对方自己要进行协议重起。
  将DD报文中的R-bit置位，表明这是一个OOB的过程。
  邻居保持邻接状态不变并对邻接关系做标志RestartState flag，同时启动ResyncTimeout记时器。
  在数据同步完成后取消R-bit置位，执行标准的OSPF操作。

VRP & IOS的相关配置

- VRP 5.x的配置

  对于GR特性，在VRP5.x提供支持。如下（配置命令修改中，可能最终不同）：

```
ospf 1
  enable link-local-sigaling
  enable out-of-band-resynchronization
  graceful-restart
```

如果要关闭NSF，只需要关闭LLS能力即可。

```
router ospf 1
  no capability lls
```

对于NSF-aware设备，通过定义OOB定时器来定义可容忍的NSF周期。

```
interface fa1/0/0
  ip ospf resync-timeout 60（缺省为40s）
```

- IOS配置命令

  IOS默认使能了LLS和OOB的能力。

```
router ospf 1
  nsf
```

Graceful Restart两种主流实现方式的比较

Cisco对于NSF的实现是依靠3个方面的过程来实现的，LLS、OOB和对NSF过程的定义，并不需要支持Opaque LSA，而且LLS和OOB可以在其它的OSPF扩展中得到应用，但是技术相对复杂。对于NSF周期的定义，Cisco是由NSF Helper来决定的，虽然时间可配置，但是有可能造成NSF过程的提前终止。RFC3623的定义只是使用了9类的Opaque LSA。GR周期是由GR Restarter决定的，这样可以避免GR过程的意外中断，但同时也可能导致欺骗攻击造成孤岛。

OSPF多实例

技术支持中心 李劲松 / 鉴定测试中心 杨默寒

前言

本文是根据李劲松---李大师的PPT《OSPF Multi-instance》改编而成。

预备知识

阅读本文之前，请先熟悉一些MPLS-L3VPN的基础支持，包括：

PE和CE的概念。

OSPF协议。

MPLS-L3VPN的基本原理。

VPN site(其实就是相当于一个多实例)

MBGP（BGP进化后的结果---多协议BGP）

典型的MPLS-L3VPN拓扑：

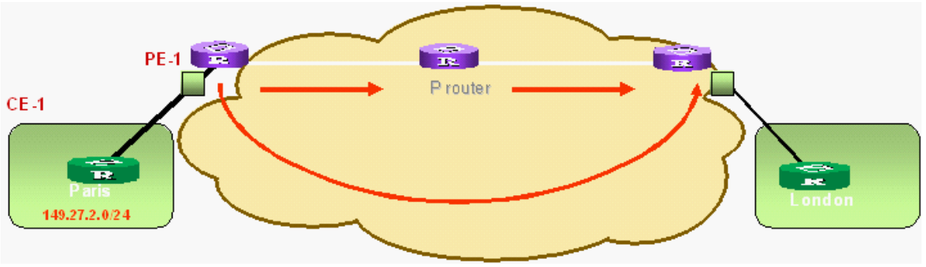


图1

路由协议的《进化论》

达尔文的《进化论》告诉我们一个原理---适者生存。当MPLS-L3VPN技术日渐成熟的时候，各个路由协议为之也要做出相应的进化或者改变（不变会被淘汰的^_^）。

我司设备目前支持的主要路由协议为：静态路由，RIP，OSPF，BGP。（IS-IS本文不涉及所以暂不介绍），这里我们先简单说一下各个路由协议的进化情况：

静态路由、RIP都是标准的协议，但是每个VRF运行不同的实例。相互之间没有干扰。与PE的MP-iBGP之间只是简单的IMPORT操作。

EBGP也是普通的EBGP，而不是MP-EBPG，只交换经过PE过滤后的本VPN路由。

OSPF则做了很多修改，可以将本site的LSA放在bgp的扩展community属性中携带，与远端VPN中的ospf之间交换LSA。每个site中的OSPF都可以存在area 0，而骨干网则可以看作是super area 0。此时的OSPF由两极拓扑（骨干区域＋非骨干区域）变为3级拓扑（超级骨干区域＋骨干区域＋非骨干区域）。

从上面我们可以看出静态路由、RIP、BGP这些实现相对简单的路由协议变化并不是很大（当然BGP的一个新功能：用MBGP传递VPN路由也算变化不小吧，这里我们就不讨论了先）。OSPF由于自身算法的太过于精妙，所以它的变化也是比较大的，不过最根本的原因还是适应网络的需求，毕竟科技以人为本啊。

OSPF多实例

OSPF多实例相关名词介绍

进程（Process）

OSPF运行是依靠运行自身的SPF算法，从而形成一个本地的链路状态数据库。不同的接口可以运行多个SPF算法从而形成多个链路状态数据库，我们把每个链路状态数据库里所涉及的接口、LSA等叫做在一个OSPF进程里面。不同的进程的链路状态数据库是完全隔离的，但是可以共同拥有一个路由表（可以通过再发布引入）。

多实例（Instance）

传统路由器只有一张路由表，但是为了适应MPLS VPN，路由器不得不在自身上再模拟一个或者多个虚拟路由器，每个虚拟路由器有独立的路由表（没有特殊处理是不可以互访问的）。每个虚拟路由器和它自己相关的路由表、接口等是存在于一个多实例的。

多实例这个概念可以说基本上是为了MPLS VPN而创造出来的，但是现在它也有别的特殊的应用，比如多实例的L2TP，或者多实例组播应用（我们不必用IGMP V3也可以把不同的源但是相同组的组播流分开）等。只要你肯想肯定还有许多未被挖掘的宝藏。

域（Domain）

域这个概念和WINDOWS2000里的域的概念比较象，或者说可以和AS的概念比较，在一个域内的OSPF可以交互LSA的详细信息，不同域的OSPF只可以交互外部LSA。

BGP和OSPF

普通服务

MPLS L3VPN网络的路由传递全是依靠BGP的扩展属性传递的，对于一般的路由协议来说（比如静态，RIP等）BGP所需要做的只是简单的再发布工作，没有什么技术含量。

VIP服务

OSPF是BGP的VIP客户，当OSPF需要通过扩展BGP在MPLS L3VPN网络中传递路由的时候，BGP除了会把OSPF的路由发布出去，同时会利用自己的扩展团体属性记录OSPF路由的一些信息。（这么做的原因也是由于OSPF本身设计的思想所导致，这个协议需要每条路由的尽量详细信息。）

BGP是如何提供VIP服务给OSPF的呢？这个还要从RT说起，RT表示路由器接受路由的喜好，当然发送路由更新的时候可以设置这个“喜好”了。现在RT除了加上了这个“喜好”，而且还对OSPF有特殊的服务，RT后面加了一些OSPF路由信息，在RT的字段有一部分是这样的：

类型	OSPF区域号	路由类型	选项

图 2

如图2是BGP扩展团体属性与OSPF相关的部分，其中类型编码为0x0306（长度16bit），接下来是OSPF的区域号（32 bit），然后是OSPF路由类型（8 bit），最后一部分是选项字段（一般用来表示外部路由的度量类型）。

我们现在具体解释一下路由类型这个字段：

第1类LSA与第2类LSA的路由类型为1或者2
第3类LSA的路由类型为3
第5类LSA的路由类型为5
第7类LSA的路由类型为7
Sham-link端点地址路由类型为 129

这里Sham-link后面会介绍到的，由于MPLS L3VPN的路由是通过BGP发布出去的那么最起码的要求是这条路由要在路由表里，当PE要发布这条路由的时候它会根据产生这条路由的LSA类型来填充路由类型这个字段，当另外一个PE的接收到某条被发布的OSPF路由后，如果它的下游也是同一个DOMAIN内的OSPF区域的话那么可以通过再发布把路由传播下去。（但是路由类型1，2的会被转化成LSA 3，其它路由类型到LSA之间的转换不会变的）

我们可以发现常见的LSA变少了，类型4的LSA没了，为什么呢？（欢迎大家讨论，以下是我的看法：由于BGP的任务是发布路由给PE，而且这条被发布的路由是要在路由表里的，但是OSPF的LSA 4只是通告一下ASBR的位置，并不会产生路由，所以发布它有点难啊.....）但是没有类型4的LSA路由器对于类型5和类型7的外部路由如何正确的算出树呢？请看下图：

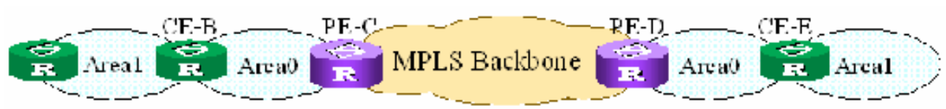


图3

如图3所示，假如与CE-B连接的区域1里某个路由器发布了一条外部路由 （LSA 5）那么PE-C发布的时候只会发布这条LSA 5产生的路由，相关的LSA 4是被忽略的，但是这条路由到了PE-D的时候，PE-D如果需要把它再发布到OSPF域的时候它会把产生者的地址写成自己，然后再发布相关的LSA 5出去，同时CE-E会产生一个相应的LSA 4（认为PE-D是ASBR），这样一切就圆满了^_^。

存在的问题

路由协议最怕的就是产生路由环路，OSPF区域里传递路由为了保证不产生环路规定：1 所有区域必须和区域0相连；2 本身用SPF算法。但是MPLS L3VPN网络中改变了这种拓扑结构，而且在跨越MPLS骨干网后一般都是类型3，5，7的LSA，这三种LSA实际上在OSPF运算产生路由的时候和DV（距离矢量）算法很接近，所以很有可能产生路由环路，比如下图：

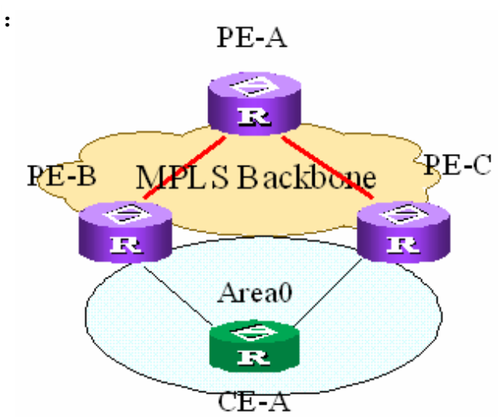


图 4

如图4（一个典型的CE双归的模型）当PE-A分别与PE-B，PE-C建立MBGP邻居后，当PE-A传递了一个类型3或者5的路由给PE-B与PE-C，PE-B与PE-C都会根据相应的LSA产生路由，但是同时就有可能PE-B产生的LSA通过CE-A传递了给PE-C然后由于路由优先级与再发布的问题，导致PE-B的路由表错误或者把这条路由又发给了PE-A，这样环路就形成了。

解决办法：首先我们先介绍一下如何避免类型3的LSA产生环路，在OSPF头部的选项字段中有一个DN位，当PE收到某条LSA 3的并且发现DN位有置位的时候，这个PE在运行SPF算法的时候会忽略这条LSA。（CISCO书上说会忽略此LSA，同时不能发布到BGP中，可是SPF算法已经忽略它了，根本就产生不了路由，怎么会发布到BGP中呢？？欲知答案，请参看附录^_^）。

接下来说一下关于类型5，7是如何避免环路的。方法和类型3用的方法类似，也是在某个地方做个标志，类型5，7的LSA都有TAG选项，当PE发布一条OSPF外部路由到VPN网络中，PE会根据这条外部路由的信息（AS号等）算出一个唯一的VPN-TAG值并且加到类型5，7的LSA里面去，当PE收到一条类型5或者7的LSA的时候如果发现有一样的VPN-TAG值的话，那么该LSA也会被忽略的。

OSPF的后门链路---Sham-link

Sham-link的作用

好的网络规划设计者都会考虑到网络的冗余、备份问题，一般情况下MPLS-L3VPN网络中PE之间存在备份链路，同时相同的VPN组之间可能也有备份链路的存在，比如下图：

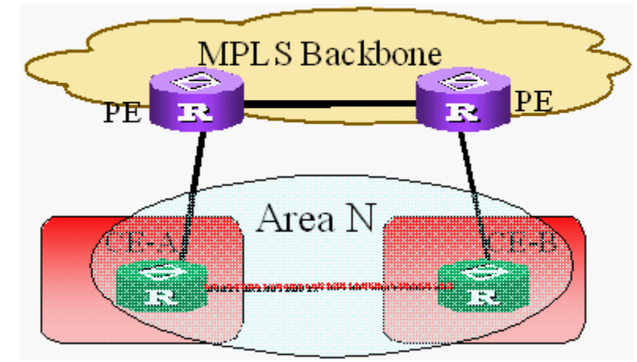


图 5

如上图5 CE-A与CE-B之间还有一个备份链路相连，问题也就来了，由于域内路由的优先级大于域间路由的优先级，所以CE-A会通过它与CE-B直接相连的链路到达另外一个VPN，而且PE也会忽略自己的BGP邻居发布过来的VPN路由。一般情况下为了网络管理与排错我们尽量会把CE-A与CE-B之间的备份链路作为“后门”链路，只有当MPLS骨干网络出现了问题数据才会通过“后门”链路转发，为了适应这种要求Sham-link产生了。通过Sham-link MPLS L3VPN骨干网的PE之间可以扩散LSA 1或者 LSA 2。

Sham-link的配置注意事项与具体实现

首先我们看一个Sham-link的配置简例：

```
{ bgp 100
 undo synchronization
 group 1 internal
 peer 1 connect-interface LoopBack0
 peer 104.0.0.3 group 1
 #
   ipv4-family vpn-instance vpna
     network 192.200.3.0
```

```
network 200.1.1.1 255.255.255.255
     import-route ospf 100
     undo synchronization

ospf 100 vpn-instance vpna
 import-route bgp
    area 0.0.0.2
    network 192.200.3.0 0.0.0.255
    sham-link 200.1.1.1 104.0.0.3   （用LOOKBACK地址）}
```

配置Sham-link一般有3条注意事项：

Sham-link的端点地址（32位）属于某个特定的多实际例

将该端点地址在BGP的多实例下通告

该端点地址不可以在相应的OSPF实例中发布

我们先来讲解一下具体实现，然后在对每个注意事项进行解释。

Sham-link与虚链路实现的手段基本一致，Sham-link在两个端点可达后，然后建立OSPF邻居，通过邻居关系把相应的LSA发送过来。现在我们来回忆一下虚链路的实现，我们在配置虚链路的时候，一般都会用到命令“vlink x.x.x.x”，这里的x.x.x.x是一个RID，也就是一个路由器的名字，也就是说在路由表里可以没有这个地址。那么Sham-link的端点地址可以不通过BGP发布吗？（答案是否定的，虚链路的虽然可以只配置对端的RID，但是这是因为本地的路由器对该区域的LSA1、LSA2都了如指掌，它完全有能力叠代的正确的出口。而Sham-link这个功能本身就是为了传递LSA1与LSA2如果不在BGP中发布端点地址，那么就好比OSPF-NBMA网络不配PEER一样）

最后我们说一下注释事项的第三点，因为一般情况下我们建立Sham-link是通过虚接口LOOKBACK建立的，而草案上说Sham-link实际上是一种地址借用的特殊情况，所以一旦把端点地址发布到OSPF里，这个时候我们的LOOKBACK接口就变成了一个启动OSPF的接口，物理接口收到的HELLO数据会送到这个LOOKBACK接口，这样就破坏了草案上说的地址借用的情况。路由器对HELLO数据包的识别也会有错误。

Sham-link 与虚链路

Sham-link与虚链路有很多表面的类似的地方，都是为了连接2个隔离的区域并且提供该区域的详细路由信息，同时也都是利用单播来建立邻居关系。但是我们在实际中应用它们的地方并不是很多，或者希望最好不要用到它们。

配置案例

以下是一个简单的配置例子（这里只介绍PE上的配置，P设备和CE的配置就不介绍了）：

首先配置一个多实例：

```
ip vpn-instance vpna
  route-distinguisher 100:1
  vpn-target 100:1 export-extcommunity
  vpn-target 100:1 import-extcommunity
```

然后配置BGP

```
bgp 65001（别忘了添加邻居）
   ipv4-family vpn-instance vpna
   import-route direct
   import-route ospf 1
```

接着配置OSPF多实例

```
ospf 1 vpn-instance vpna
    import-route bgp
    import-route direct
    area 0.0.0.1
    network 201.1.1.2 0.0.0.0
    sham-link 1.1.1.1 2.2.2.2
```

留给读者

请问下面这个拓扑哪里不合理，为什么？如何修改？

附录

答案：CISCO与我司的再发布的实现是不同的，我司的实现方法是只有IP路由表里有具体的OSPF路由，那么它才可以被发布到其它的协议当中去。而CISCO的实现是把整个可用LSA都发布到其它协议当中。

这两种实现方法各有什么好处？各有什么缺点？请大家思考。

OSPF附录E及相关问题与规避措施

鉴定测试中心 周迪

不支持附录E导致的组网缺陷

我司VRP平台V3及以前的版本都不支持OSPF协议的附录E，由此导致在实际开局与网络改造过程中个别组网存在一定的缺陷，同时也隐藏着与此相牵连的一些问题。我们以下图为例来具体阐述这类情形。

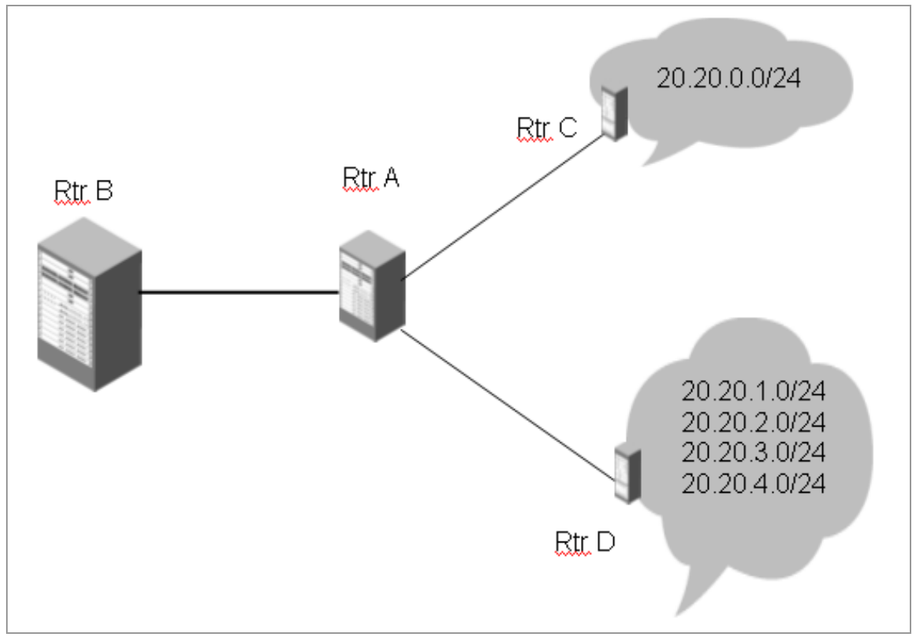


图1 OSPF协议附录E演示组网图

如图1，路由器Rtr A、Rtr B都运行OSPF协议，且所有接口都在区域0。对于上图所示的网络，一种惯常的配置步骤可能是：

1、在路由器Rtr A上配置OSPF协议引入静态路由，以通过OSPF协议将路由信息传递给路由器Rtr B。

2、路由器Rtr A上配置静态路由20.20.0.0/24，下一跳指向该网络的出口路由器Rtr C；

3、路由器Rtr A上配置静态路由20.20.0.0/16，下一跳指向该网络的出口路由器Rtr D；

这样的配置看起来十分完美，会不会隐藏着什么隐患呢？

实际的效果是：通过路由器Rtr B转发的去往目的地为20.20.1.0/24~20.20.4.0/24四个网段的流量将在Rtr B上全部被丢弃！

为什么会这样呢？

我们知道，OSPF协议是一种基于链路状态的路由协议，其路由计算是基于链路状态LSA的。当路由器引入静态路由时，通过生成相应的第5类LSA进行洪泛，以此向其他路由器传递拓扑信息。第5类LSA是以对应网段的IP地址来标识的，并协同产生该LSA的路由器ID、LSA的序列号等三个信息的组合作为其关键值进行唯一性标识，也就是说：不关心路由的掩码信息！

上述组网中，在路由器Rtr A上配置第一条静态路由20.20.0.0/24时，生成了一条第5类LSA往Rtr B洪泛；当再配置第二条静态路由20.20.0.0/16时，由于网段IP地址相同，不再生成重复的LSA。于是导致了路由器Rtr B只知道20.20.0.0/24的网段信息，而不知道Rtr D所在的网络的网段信息。

上述情形不只针对OSPF的第5类LSA，其第3、7类LSA由于对网络信息描述的关键值不包含网段的掩码信息，也存在相同的缺陷。于是RFC2328的附录E推荐了一种改进的方法。

OSPF附录E的解决方案

在OSPF协议中，关于其他区域网段和外部引入网段的计算主要依靠第3类LSA和第5、7类LSA来实现；这三类LSA分别承载OSPF区域间路由、OSPF外部路由以及OSPF的NSSA区域外部路由的信息。

由于OSPF的这三类LSA的ID是以这些网段的IP地址来标识的，每一类型的LSA是以产生该LSA的路由器ID、LSA的序列号以及被描述网段的IP地址这三个信息的组合作为关键值进行区分的。所以，同一台路由器产生的或者引入外部路由而生成的这些LSA，对于那些网段地址一样、而掩码信息不同的路由，将不能进行区分。例如，上面例子中的路由器Rtr A对于静态路由20.20.0.0/16和20.20.0.0/24的引入将只会生成一个LSA，其中必有一条路由因无法描述而被丢弃。

在讲述附录E前，我们先来深入分析下此类组网，收集下解决这类问题的一些相关要点。

首先，OSPF路由器本地必须能够区分对相同网段IP、不同掩码的多条路由的描述，这是保证解决办法有效的基础条件。

其次，对于需要的路由信息的描述必须使用网络地址，这是OSPF协议路由计算方法的根本，不应该改变，否则会涉及协议的大修改。

再次，当存在相同网段IP、不同掩码的路由时，只要保证最短掩码的那条路由传播出去，就可以解决问题。所以，合适的解决方案必须保证最短掩码的路由能够被其他路由器学习到。

最后，这类情形的发生是由于路由器本地无法对网段IP相同而掩码不同的多条路由进行描述而导致的，不涉及与其他OSPF路由器的互操作问题。所以，每台OSPF路由器可以自己选择实施合适的解决方案。

OSPF的RFC2328附录E推荐的一种改进的方法，就是本着上面四个原则进行处理的。下面以第5类外部LSA的处理为例进行介绍，相同的处理方法也同样适用于其他两类LSA。

OSPF路由器欲引入一个外部路由，网段IP地址为NA，掩码为NM_new，我们以[NA，NM_new]进行标记。该路由器的处理过程如下：

（1）查找链路状态数据库，判断是否已经存在对应IP地址为NA的路由的第5类LSA，如果没有，就产生一个第5类LSA，其ID为NA，处

理结束。

（2）如果已经存在，则从已有的第5类LSA中提取该网段的掩码NM_old，分两种情况进行处理：

如果NM_new的长度大于NM_old，也就是说，将要引入的路由的掩码比原先的那条更长，这时只需要再产生一条新的第5类LSA即可：将新的外部路由网段的主机位全部置1，子网广播地址作为新的第5类LSA的ID。例如，原先的外部路由网段是10.10.0.0/16，其LSA的ID是10.10.0.0，新的将要引入的网段是10.10.0.0/24，产生的LSA的ID就是10.10.0.255。

如果NM_new的长度小于NM_old，则需要立即替除原来的第5类LSA，并产生两条新的第5类LSA。一条LSA的ID为NA，其序列号为原来被替掉的LSA的序列号加1，花费值对应新路由的花费值；对于另一条LSA，将原路由[NA，NM_old]的主机位全部置1，用子网广播地址作为其LSA的ID。例如，原先的路由10.10.0.0/24以花费值10引入，序列号为100；新的路由10.10.0.0/16以花费值20引入。于是按上面的处理，OSPF路由器先替掉原来的LSA，并生成两条新的LSA，一个LSA的ID为10.10.0.0，花费值为20，序列号为101；另一条LSA的ID为10.10.0.255，花费值为10，序列号为1。

附录E所推荐的方法也有一定的局限性，其实施的前提是：

所有的掩码都是连续的；

不存在一条外部路由的网络IP刚好与上述处理后的子网广播地址相同。

只有基于上述两个前提，这种解决办法才能保证每次产生的LSA的ID都是唯一的。

其他相关的问题以及规避方法

我司路由交换产品中，OSPF协议除了不支持RFC2328附录E这个本身的缺陷之外，由此而引起的由于处理不完善而导致的相关问题也不少，如果在开局和网络改造过程中不加以注意，其后果也是很严重的。不过幸运的是，其规避的措施到是十分简单。

这些问题主要涉及刷新机制，所有涉及IP地址相同掩码不同的情形都可能隐藏着问题。下面以一个已经转缺陷而不作修改的问题单为例，看看这类问题是怎样由于配置的更改而引发的。

操作过程：先配置一条静态路由，目的网段为20.20.0.0/24，这时OSPF链路数据库中生成一条ID为20.20.0.0的ASE_LSA；再配置一条目的网段为20.20.0.0/16的静态路由，由于即将要生成的LSA的ID与已存在的一条一样，OSPF模块就不再生成新的ASE_LSA。此时，删除原来的目的网段为20.20.0.0/24的静态路由，OSPF模块由于不重新为后来的静态路由生成ASE_LSA，而导致路由无法传播！

与此相关的问题单还有几个，未提单的也有不少，问题类型都涉及外部路由引入过程中的增加删除操作、路由聚合过程中的删除增加操作、路由聚合过程中的指定发布与不指定发布操作等等。总之，对于所有涉及到IP地址相同而掩码不同的配置情况，都可能存在问题。

在此，建议兄弟们在开局时，在OSPF协议的配置完成以后，再作一下重起OSPF协议的操作：Clear ip ospf process，这类问题都可以规避。

后记

新的V5平台即将推出，OSPF协议的附录E也将支持，不过V3平台会在将来很长一段时间内使用，大家还是需要注意下这些问题。

此文最初写于2002年，初涉数据通信不久就舞文弄墨，沉寂两年后蒙刘宇大侠的赏识，重新修改了一下，希望大家多多批评。

参考资料

[1] www.ietf.com，RFC2328

OSPF DNA LSA

鉴定测试中心 张宇弟

背景知识

LSA Age相关知识

LSA Age定义

在OSPF中，LSA的LSA Header的第一个字段为LSA Age字段，该字段占16bit。系统产生一个LSA时，LSA Age置0。在LSA泛洪过程中，每一跳LSA Age都会递增InfTransDelay。LSA Age主要负责LSDB的LSA老化以及LSA新旧判断。

时间值定义

在LSA Age的工作需要明确几个概念：

MaxAge ：取60min。当LSA Age递增至MaxAge，该LSA将在整个Routing Domain中被清除。

MaxAgeDiff ：当具有相同的序列号和校验和的两个LSA的LSA Age之间的差值超过MaxAge-Diff时，则可以认为这两个LSA是同一个LSA的两个实例。

InfTransDelay ：从一个接口发送一个携带LSA的LSU，LSA需要估计一个在发送前的损耗延迟InfTransDelay，在LSA Age加上InfTransDelay。

LSRefreshTime ：周期发送LSA用于LSDB同步。当LSA Age递增了LSRefreshTime时会产生原本系统产生的LSA发送出去。

MinLSInterval ：该值定义了产生不同LSA实例的最小时间间隔。

MinLSArrival ： 收到同一个LSA的不同的LSA实例的最小时间间隔，若在小于该时间间隔收到一个新的LSA实例，将直接丢弃。

CheckAge ：LSA需要定期进行校验，校验的周期是CheckAge的倍数。

LSA Age工作机制

关于LSA Age我们主要从三方面进行描述：

LSA周期更新：

在LSRefreshTime的概念介绍时，我们说了，通过LSA Age递增LSRefreshTime进行周期LSA的产生和泛洪达到LSDB同步的作用。

LSA老化：

LSA Age会在LSDB中递增老化LSA，当LSA Age递增至MaxAge时，将该LSA从LSDB中清除，并将LSA Age置为MaxAge的LSA在路由域泛洪，在路由域内清除该LSA，这叫做自然老化。注意在清除LSA的操作还要做如下条件的判断：

Link state retransmission lists列表为空；

本系统的邻居没有处于Exchange或Loading状态的；

在LSA清除中还有一种所谓的“Premature aging”－提前老化，提前老化就是将LSA的LSA Age设置成MaxAge，然后同自然老化一样在路由域中清除。提前老化一般在如下几个情况下发生：

本系统产生的LSA的序列号已增至MaxSequenceNumber，此时需要在路由域中清除原LSA，生成序列号为InitialSequence-Number的新LSA。

本系统产生的AS-external-LSA携带的外部路由已不可达；

收到不期望的自己产生的LSA，参见RFC2328 13.4；

提前泛洪只对本系统产生的LSA有效，一台路由器不能去提前老化其他设备泛洪来的LSA。参见RFC2328 14.1

LSA新旧判断

对于一条LSA，判断其新旧的有三个要素：sequence number，checksum，LSA Age。

对于同一个LSA的不同LSA实例，当sequence number和checksum都相等的情况下，就通过LSA Age进行新旧判断。

一般会将LSA Age为MaxAge的LSA实例做为最新实例，这样将清除LSA；

若两个LSA实例的LSA Age都没有等于MaxAge的，则判断两个实例的LSA Age的差值是否大于MaxAgeDiff。是则取LSA Age小的做为最新的LSA实例接收。当差值小于MaxAgeDiff时，认为两个实例为同一实例。参见RFC2328 13.1

按需线路

在实际组网中，有一些链路是按时或按流量收费的，例如X.25 SVC，ISDN和拨号电路等即用即连的电路。按需线路的特点决定了哪怕是最小的协议通信量，按需线路也不希望有。

OSPF对于按需线路的支持称为OSPF对按需线路的扩展，这种扩展特性虽然是为了即用即连的链路设计的，但是在任何带宽有限的链路上也是有用的。

DNA LSA详解

DNA LSA的由来

按需线路就是要求在没有数据传输任务时，链路层是DOWN的。当在按需线路上跑OSPF协议，我们知道OSPF邻居之间会周期发送hello报文以保持邻接关系，周期发送LSA更新同步LSDB。根据按需线路的特点，可想而知需要对周期发送hello报文和LSA更新报文进行抑制。换句话说，OSPF协议报文只在OSPF邻接关系初始建立阶段进行交互。

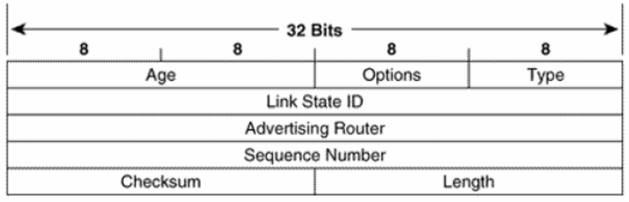
对于按需线路来说，需要抑制的是周期的hello报文和LSA更新。抑制hello报文，OSPF协议会有一个可达性的假定，即OSPF协议必须假定在按需线路连接的时候这条线路可用并有效。

LSA的抑制将导致LSDB中的LSA无法得到更新，在前面我们学习过，LSA会通过LSA Age进行老化。LSA的老化会最终导致LSDB的不同步。那么如何解决这个问题呢？在邻接关系建立过程中，LSDB会完成同步过程。此后，可以不对LSA进行老化工作。这就是DNA LSA(DoNotAge LSA)提出的原因。

DNA LSA的工作机制

DNA LSA关键字段

首先我们还是再回顾一下LSA Header的格式：



在LSA Header的第一个16bit字段是LSA Age字段，该字段的最高位为DoNotAge位，该位置位的LSA为DNA LSA。

紧接着是选项字段Options字段。Options字段包含在hello报文，DD报文以及所有的LSA报文中，用于协商可选能力参数。对于DNA LSA需要关注的是Options字段中的DC位：

*	*	DC	EA	N/P	MC	E	T

DC位_DemandCircuit位，路由器在发送的LSA中将该位置1表示其能够支持OSPF对按需线路的扩展特性。置位DC位的路由器并不意味着该设备一定就是按需线路的端设备，只是表明本设备能够正常处理DNA LSA。

DC扩展能力的协商

在讲述DNA LSA处理机制之前，我们先讲述一下DC扩展能力的协商以及hello报文的抑制情况。在hello报文和DD报文的Option字段中通过设置DC位标志支持DC扩展。对于DC的扩展需要通过配置命令启动，具体命令在后文详述。在此需要明确的是，链路的两端只需要一端配置该命令启动DC扩展即可。在邻接协商的初期，未配置的一端会收到启动DC扩展一端的hello报文，该报文中携带了DC位信息。若本端支持DC扩展，则在后续的hello报文和DD报文中置位DC位。若不支持DC扩展，对于收到的hello报文中的DC位忽略，通过发送DC位为0的DD报文指示对端本端不支持DC扩展。收到该DD报文后，配置启动DC扩展命令的一端不再抑制hello报文的周期发送。

对于hello报文，我们需要注意一点，OSPF在Broadcast和NBMA网络上需要通过hello报文选举DR/BDR，因此在network type为Broadcast和NBMA的链路上不可以抑制hello报文周期发送。所以对于周期hello报文的抑制只在network type为P2P和P2MP有效，因此在配置按需线路的时候要注意将network type设置为P2P或P2MP。同时对于P2MP网络，是按照多个P2P链路处理的，也就是说DC扩展能力也可以在每个虚拟的P2P之间独自协商。

DNA LSA处理机制

DNA LSA在LSDB中与普通LSA的区别在于DNA LSA不会周期产生LSA更新，并且不会老化。对于DNA LSA的Age采用DoNotAge＋X的形式。在比较两个LSA实例的时候，DoNotAge+X和X被认为等值的。即比较不同LSA实例的LSA Age的时候，DoNotAge将被忽略。

因此，LSA Age的取值有两个范围：0－MaxAge和0－DoNotAge＋MaxAge。对于不在该范围的LSA Age取值处理为MaxAge。

注意LSA Age是不在checksum校验范围内的，路由器的出错有时可能会导致DoNotAge的误置位。因此需要强调的是，路由器对于本系统产生的LSA不会将本系统LSDB的LSA的DoNotAge置位。这样就能通过泛洪将路由域中误置位DoNotAge的LSA更新。

LSDB的资源是有限的，为了保证这种资源的合理有效使用，必须将过期的，废弃的LSA进行清除。那么难到DNA LSA就特殊，将永远保留在LSDB中吗？这样当然是不合理的，那就牵涉到对于废弃的DNA LSA进行清除的处理，注意DNA LSA的清除是OSPF LSA清除的特例，在OSPF中常规LSA清除是由该LSA的始发者完成，但是DNA LSA的清除无法按照正常LSA泛洪来完成，所以对于DNA LSA的清除可以由任一满足下面清除条件的路由器完成。我们知道DNA LSA的LSA Age在递增至DoNotAge＋MaxAge是不会清除的，但是满足如下条件就可以清除该DNA LSA：

1）DNA LSA Age等于MaxAge的整数倍；
2）产生DNA LSA的源设备变得不可达已超过MaxAge时长；

DNA LSA的后向兼容

并不是所有的路由器都支持OSPF对按需线路的扩展特性。如图1，路由器C不支持DC扩展，那么会出现DNA LSA的反复重传。路由器D学习到DNA LSA，发往路由器C。路由器C不支持DC扩展，不识别DNS LSA，将该LSA认为是MaxAge LSA。那么根据LSA的处理原则，会发送该MaxAge LSA的响应给路由器D。路由器D会认为MaxAge LSA不是先前发送的DNA LSA，因此会重传DNA LSA。

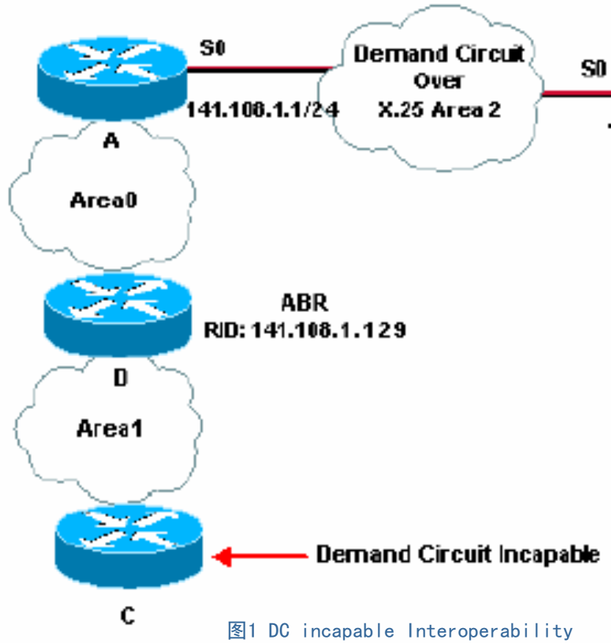


图1 DC incapable Interoperability

对于这种情况的处理方法是，当支持DC扩展的路由器的LSDB中发现LSA的DC清零。则将本LSDB中所有的DNA LSA清除，重新学习正常的LSA，即DoNotAge清零的LSA。

同时做为ABR的路由器D还要产生一个“Indication-LSA”告知路由域，有不支持DC扩展的路由器存在，请不要继续DNA LSA的产生。如图1，路由器D向Area0发送Indication-LSA，告知不要向自己发送DNA LSA。路由器A收到Indication-LSA之后，向Area2发送Indication-LSA，通知该区域不产生DNS LSA。并且周期更新的LSA会经过路由器A,B之间的按需线路，导致按需线路UP。这样就违反了按需线路的初衷，如何解决呢？按需线路理想的状况是，路由域中的所有设备都支持OSPF对DC的扩展特性，但是当存在不支持扩展特性的路由器时，难道按需线路就无法正常工作吗？

我们先对Indication-LSA进行一下介绍。Indication-LSA是由ABR产生的，该LSA是4类LSA—ASBR-summary-LSAs。其特殊性在于，link-state ID取ABR的ROUTER ID，LSA的metric设置为LSInfinity，并且DC为清零。图1中，将Area2设置为STUB或NSSA区域，这样路由器A对收到的Indication-LSA，因为是ASBR-summary-LSAs，所以不会发布到Area2中去。这样问题就迎刃而解了。

因此，建议对于按需线路的区域中所有的路由器均为支持DC扩展的路由器，并将该区域设置为STUB或NSSA。

ABR负责产生Indication-LSA，对于Indication-LSA的产生也需要做一定的限制：

1） 当本区域已经包含DC为清零的LSA时，不会产生Indication-LSA；

2） 当一个区域有多个ABR，并且这些都有产生Indication-LSA的可能性，那么本区域的Indication-LSA由ROUTER ID大的路由器优先产生。即，当路由器X，Y都连接在Area0时，并各自连接Area1，Area2。Area1，Area2都包含不支持DC扩展的路由器，那么X,Y都会向Area0产生Indication-LSA。若X的ROUTER ID大于Y的ROUTER ID，则由X产生Indication-LSA就可以了。

Flooding Reduction应用

Flooding Reduction特性是对DC扩展特性的改动，区别在于配置Flooding Reduction的链路，只抑制周期LSA更新，而不抑制周期hello报文的发送。其工作机制同DC扩展是一样的，同样使用DNA LSA进行LSA抑制。

虚链路的DC处理

按照RFC规定，虚链路是按照DC扩展处理的。也就是说，虚链路的处理基本是和按需线路的处理机制一致的。

不同之处在于，按需线路的接口在邻接关系形成后是保持在down状态的。而虚链路不同。虚链路上是通过是否有intra-area path来判断邻居是否可达的。

配置和示例

配置需求

在CISCO上，DC扩展和Flooding Reduction的配置是非常简单的。

DC扩展的配置只需要在链路的一端配置即可，因为一端一旦配置了DC扩展，则在建立连接关系时发送hello报文双方就是否支持此特性进行协商。只需在接口模式下配置命令就完成了DC扩展或Flooding Reduction的配置：

ip ospf demand?circuit / ip ospf flood?reduction

同时根据前文的描述，需要遵循如下规则：

1 将配置DC扩展的接口网络类型强制为P2P或P2MP；

2 将DC链路的两个接口所在的区域设置为STUB或NSSA区域，并保证该区域中其他的设备也支持DC扩展；

示例

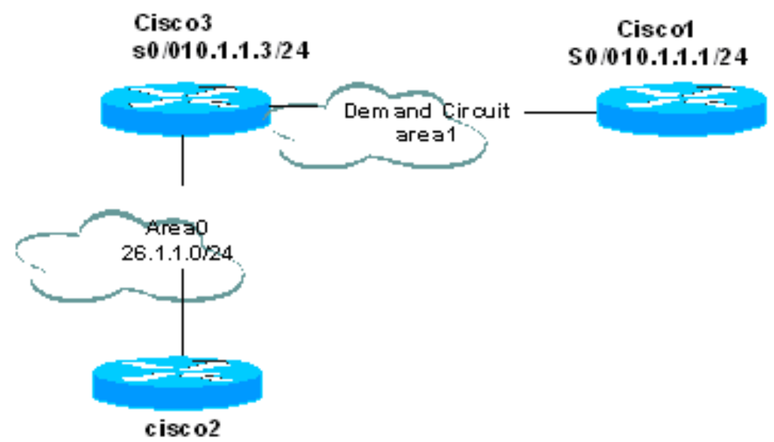


图2 OSPF DNA LSA示例

Cisco1与Cisco3通过demand circuit相连，接口启动OSPF。下面我们看看Cisco1的链路状态数据库：

```
cisco1#show ip ospf database

            OSPF Router with ID (10.1.1.1) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.1.1        10.1.1.1        1038        0x80000008 0x007CA5 0

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
10.1.1.1        10.1.1.1        300         0x8000002A 0x00D365 2
10.1.1.3        10.1.1.3        1     (DNA) 0x8000002E 0x00A7E8 2

                Summary Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
26.1.1.0        10.1.1.3        1     (DNA) 0x80000004 0x00A766
```

总结

本文主要是对LSA Age的工作机制以及在按需线路上OSPF的DNA LSA的工作原理进行了介绍。对于按需线路的深入学习可参见RFC1793。

附录

参考资料：

J. Moy, Extending OSPF to Support Demand Circuits RFC1793
Syed Faraz Shamim, OSPF Demand Circuit Feature
Syed Faraz Shamim, Why OSPF Demand Circuit Keeps Bringing Up the Link

OSPF缺省路由

鉴定测试中心 姜杏春

OSPF区域类型

OSPF根据网络的需求可以定义为下列几种类型：

普通区域
STUB区域
完全STUB区域
NSSA区域
完全NSSA区域

普通区域

当区域被缺省定义时，它被认为是普通区域。普通区域可以是标准区域或骨干区域。标准区域是最通用的区域，它携带区域内路由，区域间路由和外部路由。骨干区域是连接所有其它OSPF区域的中央区域。

STUB区域

STUB区域是一个不允许AS外部LSA在其内部泛洪的区域。STUB区域只可以携带区域内路由和区域间路由。在这些区域中路由器的OSPF数据库和路由表规模以及路由信息传递的数量都会大大减少，为了保证到自治系统外的路由依旧可达，由该区域的ABR生成一条缺省路由0.0.0.0 传播到区域内，所有到自治系统外部的路由都必须通过ABR才能到达。

完全STUB区域

完全STUB区域是区域中最受限的形式，它不仅不允许携带外部路由，甚至连区域间路由也不允许携带，只可以携带区域内路由。在这些区域中路由器的OSPF数据库和路由表规模以及路由信息传递的数量都会大大减少，为了保证到区域外的路由依旧可达，由该区域的ABR生成一条缺省路由0.0.0.0 传播到区域内，所有到该区域外部的路由都必须通过ABR才能到达。

NSSA区域

NSSA区域允许一些外部路由通告到OSPF自主系统内部，而同时保留自主系统的区域部分的STUB区域的特征。假设一个STUB区域中的路由器连了一个运行其他路由进程的自治系统并且引入到该OSPF进程，现在这台路由器就变成了ASBR，所以这个区域就不能再称为STUB区域了。然而如有把这个区域配置成一个NSSA区域，ASBR会产生NSSA外部LSA（类型7），可以泛洪到整个NSSA区域。这些7类LSA在NSSA ABR上会转换成5类LSA并且泛洪到整个OSPF域中。

完全NSSA区域

和NSSA区域相似，完全NSSA区域允许一些外部路由通告到OSPF自主系统内部，而同时保留自主系统区域部分的完全STUB区域的特征。该区域的ASBR会产生NSSA外部LSA（类型7）在其区域内部泛洪并通过该区域的ABR转换成5类LSA在整个OSPF域泛洪。同时，该区域的ABR也会产生一条缺省路由0.0.0.0传播到区域内，所有域间路由都必须通过ABR才能到达。

缺省路由的产生

普通区域

缺省情况下，在普通OSPF区域内的OSPF路由器是不会产生缺省路由的，即使它有缺省路由。当网络中缺省路由通过其他路由进程产生时，必须能够将缺省路由通告到整个OSPF域中。这个时候要想产生缺省路由必须在ASBR上OSPF协议视图下手动配置：

VRP: default-route-advertise [always]
IOS: default-information originate [always]

使用了该命令将在整个OSPF域中通告缺省路由0.0.0.0，但前提是该ASBR自己已经有缺省路由，否则不会通告缺省路由。

如果在该命令上加上关键字always的话，则无论ASBR是否有缺省路由都将在整个OSPF域中通告缺省路由0.0.0.0，这将强制缺省路由总是出现在路由表中，所以慎用关键字always。

使用了该命令后将会产生一个Link State ID为0.0.0.0，网络掩码为0.0.0.0的ASE LSA（5类），并且通告到整个OSPF域中。

STUB区域

由于STUB区域不允许外部LSA在其内部泛洪，所以该区域内的路由器除了ABR外没有自治系统外部路由，如果它们想到自治系统外部时应该怎么办？在STUB区域里的路由器将本区域内ABR作为出口，ABR会产生缺省路由0.0.0.0通告给整个STUB区域内的路由器，这样的话到达自治系统外部的路由可以通过ABR到达。

配置了STUB区域之后，ABR自动会产生一条Link State ID为0.0.0.0，网络掩码为0.0.0.0的SUMMARY LSA（3类），并且通告到整个STUB区域内。

完全STUB区域

完全STUB区域不仅不允许外部LSA在其内部泛洪，连区域间的路由也不允许携带，所以在完全STUB区域里的路由器要想到别的区域或自治系统外部时应该怎么办呢？同样的，在完全STUB区域里的路由器也将本区域内ABR作为出口，ABR会产生缺省路由0.0.0.0通告给整个完全STUB区域内的路由器，这样的话到达本区域外部的路由都通过ABR到达就可以了。

配置了完全STUB区域之后，ABR自动会产生一条Link State ID为0.0.0.0，网络掩码为0.0.0.0的SUMMARY LSA（3类），并且通告到整个完全STUB区域内。

NSSA区域

NSSA区域允许少量外部路由通过本区域的ASBR通告进来，它不允许携带其他区域的外部路由，这样的话到达自治系统外部路由只能通过本区域的ASBR到达，如果该ASBR没有通告该外路由的，则不能到达。

在只配置了NSSA区域的时候，是不会自动产生缺省路由的。

如果只希望到达自治系统外部的某些路由通过该区域的ASBR到达，其它外部路由通过其它区域出去的话，有没有办法呢？在NSSA ABR上产生缺省路由0.0.0.0通告给整个NSSA区域内的路由器，这样的话除了某少部分路由通过NSSA的ASBR到达，其它都可以通过NSSA ABR到达其它区域的ASBR出去。但是这只是一个需求，并不是必须的，所以这条缺省路由不是配置了NSSA区域就会自动产生

缺省路由0.0.0.0，可以在NSSA ABR上手动配置：

VRP：nssa default-route-advertise （NSSA区域视图）

IOS：area area-id nssa default-information-originate （OSPF协议视图）

使用了该命令后，在NSSA ABR上就会产生一条Link State ID为0.0.0.0，网络掩码为0.0.0.0的NSSA LSA（7类），将在整个NSSA区域内通告缺省路由0.0.0.0。

但是如果希望所有的外部路由只通过本区域的NSSA ASBR到达，希望ASBR通告一条缺省路由0.0.0.0，可以在NSSA ASBR上手动配置：

VRP：nssa default-route-advertise （NSSA区域视图）

IOS：area area-id nssa default-information-originate （OSPF协议视图）

NSSA ASBR和NSSA ABR使用相同的命令，与NSSA ABR不同的是NSSA ASBR必须是在自身已经有一条缺省路由的情况下才会产生一条Link State ID为0.0.0.0，网络掩码为0.0.0.0的NSSA LSA（7类），在NSSA区域内通告缺省路由0.0.0.0。

因为缺省路由只是在本NSSA区域内泛洪，并没有泛洪到整个OSPF域中，只能实现本NSSA区域内的路由器在找不到路由之后从该NSSA ASBR出去，不能实现整个OSPF域的路由器从这个出口出去。

完全NSSA区域

完全NSSA区域和NSSA区域不同的是，它不允许携带区域间路由，如果要到其他区域的时候应该怎么办呢？同样的，缺省路由又出场了，在该区域ABR上会产生一条缺省路由0.0.0.0，通告给整个完全NSSA区域，所有的域间路由都将NSSA ABR作为出口。

配置了完全NSSA区域后，就会自动产生一条Link State ID为0.0.0.0，网络掩码为0.0.0.0的SUMMARY LSA（3类），在NSSA区域内通告缺省路由0.0.0.0。

与NSSA区域ABR上缺省路由产生的方式不同的是，在完全NSSA区域ABR上的缺省路由是配置好区域之后自动产生类型3的缺省LSA，在NSSA区域上ABR的缺省路由是自己可配置的，因为在完全NSSA区域产生的缺省路由是必须的，它起着指导本区域内路由器区域间路由的作用。

配置实例

以下配置实例中配置均以VRP为例。

普通区域

缺省情况下，普通区域不产生缺省路由。图1中为Area1和Area2都是普通区域。路由器2是普通区域的ASBR。

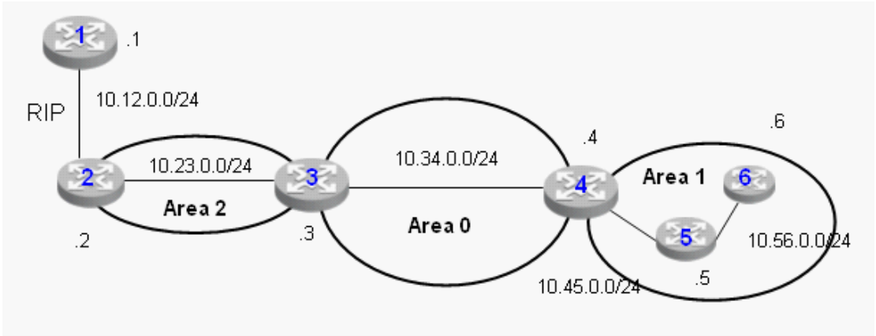


图1 普通区域实验组网图

路由器2已经有一条其他进程的缺省路由0.0.0.0，该条缺省路由需要在OSPF域通告，在路由器2上配置：

default-route-advertise

```
#
ospf 1
 import-route rip
 import-route static
 default-route-advertise
 area 0.0.0.2
  network 2.2.2.2 0.0.0.0
  network 10.23.0.0 0.0.0.255
#
 ip route-static 0.0.0.0 0.0.0.0 10.12.0.1 preference 60
#
return
```

观察路由器2的链路状态数据库可以发现，路由器2产生一条类型5的缺省LSA，Link State ID为0.0.0.0。

```
[2]dis ospf lsdb
                    OSPF Process 1 with Router ID 2.2.2.2
                              Link State Database
                              Area: 0.0.0.2
                              ………………
                          AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  0.0.0.0          2.2.2.2            50 36     80000002          1 Ase List
ASE  6.6.6.6          6.6.6.6            163 36    80000004          1 Uninitialized
                              ………………
ASE  1.1.1.1          2.2.2.2            1174 36   80000091          1 Ase List
```

路由器2生成的缺省LSA：

```
Type        : ASE
Ls id       : 0.0.0.0
Adv rtr     : 2.2.2.2
Ls age      : 606
Len         : 36
Seq#        : 80000004
Chksum      : 0xdaec
Options     : (Non DC)
Net mask    : 0.0.0.0
    Tos 0 metric: 1
    E type      : 2
    Forwarding Address :0.0.0.0
    Tag: 1
```

观察路由器3、4、5的链路状态数据库和路由表，看看路由器2的缺省路由的泛洪。

路由器3的信息：

```
<3>dis ospf lsdb
                          AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  0.0.0.0          2.2.2.2            491 36    80000003          1 Uninitialized
ASE  6.6.6.6          6.6.6.6            332 36    80000005          1 Uninitialized
                              ………………
ASE  1.1.1.1          2.2.2.2            1352 36   80000092          1 Uninitialized

<3>dis ip ro 0.0.0.0
Destination/Mask   Protocol Pre  Cost      Nexthop        Interface
0.0.0.0/0          O_ASE    150  1         10.23.0.2      Ethernet1/0
```

路由器4的信息：

```
<4>dis ospf lsdb
                    OSPF Process 1 with Router ID 4.4.4.4
                              Link State Database
                              Area: 0.0.0.0
………………
                              Area: 0.0.0.1
………………
                          AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  0.0.0.0          2.2.2.2            1403 36   80000003          1 Uninitialized
ASE  6.6.6.6          6.6.6.6            1244 36   80000005          1 Uninitialized
………………
ASE  1.1.1.1          2.2.2.2            516 36    80000093          1 Uninitialized

<4>dis ip ro 0.0.0.0
Destination/Mask   Protocol Pre  Cost      Nexthop        Interface
0.0.0.0/0          O_ASE    150  1         10.34.0.3      Ethernet0/1
```

路由器5的信息：

```
<5>dis ospf lsdb
                    OSPF Process 1 with Router ID 5.5.5.5
                            Link State Database
                            Area: 0.0.0.1
                                ………………
                         AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE   0.0.0.0         2.2.2.2            1727 36   80000003         1 Uninitialized
                                ………………
ASE   1.1.1.1         2.2.2.2            831 36    80000093         1 Uninitialized

<5>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost        Nexthop         Interface

0.0.0.0/0           O_ASE    150  1           10.45.0.4       Ethernet2/0
```

路由器3、4、5都有类型5的缺省LSA，通告者是路由器2，并产生相应的缺省路由，可以看出，普通区域的ASBR产生的缺省路由0.0.0.0不仅仅在本区域内泛洪，还泛洪到整个OSPF域中去。

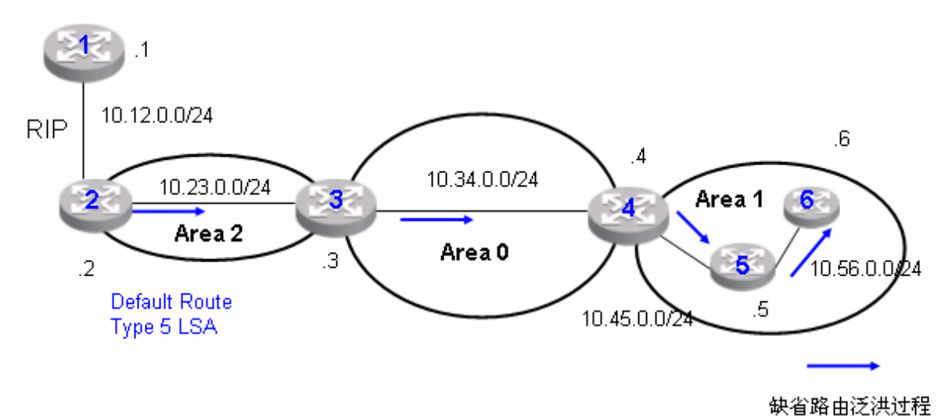


图2 普通区域缺省路由泛洪

如果路由器2自身没有缺省路由，但是仍希望能够在OSPF域中通告缺省路由0.0.0.0，可以在路由器上2配置：

```
default-route-advertise always
```

STUB区域和完全STUB区域

配置一个区域成为STUB区域，必须在本区域的所有路由器上区域视图配置：

```
stub
```

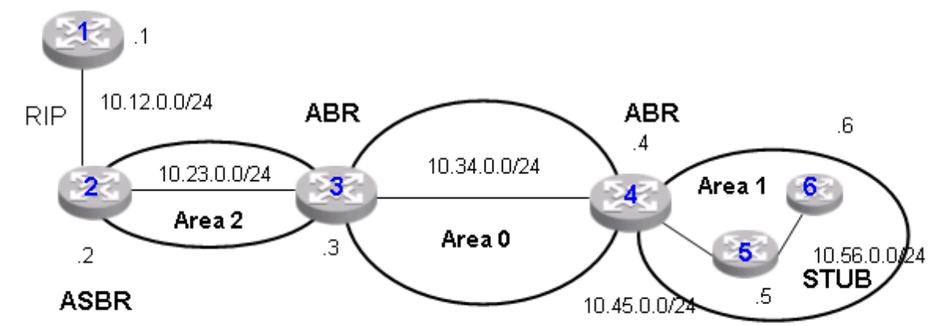


图3 STUB区域实验组网图

如图，配置区域1为STUB区域，路由器4为区域的ABR。

```
#
ospf 1
 default-route-advertise
 area 0.0.0.1
  network 10.45.0.0 0.0.0.255
  stub
 #
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 10.34.0.0 0.0.255.255
#
```

观察路由器4的链路状态数据库可以发现，路由器4在区域1生成一条类型3缺省LSA，Link State ID为0.0.0.0。

```
<4>dis ospf lsdb
                    OSPF Process 1 with Router ID 4.4.4.4
                              Link State Database
                             Area: 0.0.0.0
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 4.4.4.4          4.4.4.4            1383 24   0                 0 SpfTree
                                  ……………………
ASB  2.2.2.2          3.3.3.3            773 28    80000088          1 Uninitialized
                             Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 5.5.5.5          5.5.5.5            1525 24   0                 0 SpfTree
                                  ……………………
Net  10.56.0.5        5.5.5.5            377 32    80000003          0 SpfTree
SNet 0.0.0.0          4.4.4.4            238 28    80000002          1 SumNet List
SNet 2.2.2.2          4.4.4.4            238 28    80000002          12 Inter List
                                  ……………………
SNet 10.34.0.0        4.4.4.4            238 28    80000089          10 Inter List
                         AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  1.1.1.1          2.2.2.2            1483 36   800000e7          1 Uninitialized
```

路由器4成的缺省LSA：

```
  Area: 0.0.0.1

  Type        : SumNet
  Ls id       : 0.0.0.0
  Adv rtr     : 4.4.4.4
  Ls age      : 697
  Len         : 28
  Seq#        : 80000002
  Chksum      : 0x37f5
  Options     : (DC)     (Non DN)
  Net mask    : 0.0.0.0
     Tos 0 metric: 1
```

观察路由器3、4、5的链路状态数据库和路由表，看看路由器4缺省路由的泛洪。

路由器3的信息：

```
   OSPF Process 1 with Router ID 3.3.3.3
                    Link State Database
                   Area: 0.0.0.0
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 4.4.4.4          4.4.4.4            2283 24   0                 0 SpfTree
Stub 3.3.3.3          3.3.3.3            2291 24   0                 0 SpfTree
Rtr  4.4.4.4          4.4.4.4            1231 48   800000b9          0 SpfTree
Rtr  3.3.3.3          3.3.3.3            7 48      8000008e          0 SpfTree
Net  10.34.0.4        4.4.4.4            1231 32   80000003          0 SpfTree
SNet 2.2.2.2          3.3.3.3            1675 28   80000085          2 Inter List
SNet 10.45.0.0        4.4.4.4            1136 28   80000059          10 Uninitialized
SNet 10.56.0.0        4.4.4.4            1136 28   80000002          20 Uninitialized
SNet 5.5.5.5          4.4.4.4            1136 28   80000002          11 Uninitialized
SNet 10.23.0.0        3.3.3.3            1675 28   80000085          1 Inter List
ASB  2.2.2.2          3.3.3.3            1675 28   80000088          1 SumAsb List
                             Area: 0.0.0.2
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 2.2.2.2          2.2.2.2            -1 24     0                 0 SpfTree
                                  ……………………
ASB  4.4.4.4          3.3.3.3            1675 28   80000002          10 SumAsb List
                         AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  1.1.1.1          2.2.2.2            641 36    800000e8          1 Uninitialized
<3>dis ip ro
 Routing Table: public net
```

```
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
1.1.1.1/32          O_ASE    150  1         10.23.0.2       Ethernet1/0
2.2.2.2/32          OSPF     10   2         10.23.0.2       Ethernet1/0
3.3.3.3/32          DIRECT   0    0         127.0.0.1       InLoopBack0
4.4.4.4/32          OSPF     10   11        10.34.0.4       Ethernet2/0
5.5.5.5/32          OSPF     10   21        10.34.0.4       Ethernet2/0
10.23.0.0/24        DIRECT   0    0         10.23.0.3       Ethernet1/0
10.23.0.3/32        DIRECT   0    0         127.0.0.1       InLoopBack0
10.34.0.0/24        DIRECT   0    0         10.34.0.3       Ethernet2/0
10.34.0.3/32        DIRECT   0    0         127.0.0.1       InLoopBack0
10.45.0.0/24        OSPF     10   20        10.34.0.4       Ethernet2/0
10.56.0.0/24        OSPF     10   30        10.34.0.4       Ethernet2/0
127.0.0.0/8         DIRECT   0    0         127.0.0.1       InLoopBack0
127.0.0.1/32        DIRECT   0    0         127.0.0.1       InLoopBack0
```

路由器5信息：

```
<5>dis ospf lsdb
                 OSPF Process 1 with Router ID 5.5.5.5
                         Link State Database
                          Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 5.5.5.5          5.5.5.5            2688 24   0                0 SpfTree
                                  ……………………
Net  10.56.0.5        5.5.5.5            1527 32   80000003         0 SpfTree
SNet 0.0.0.0          4.4.4.4            1389 28   80000002         1 Uninitialized
SNet 2.2.2.2          4.4.4.4            1389 28   80000002        12 Uninitialized
                                  ……………………
SNet 10.34.0.0        4.4.4.4            1389 28   80000089        10 Uninitialized

                       AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  6.6.6.6          6.6.6.6            3379 36   80000059         1 Uninitialized
ASE  10.56.0.0        6.6.6.6            -1 36     80000058         1 Uninitialized

<5>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
0.0.0.0/0           OSPF     10   11        10.45.0.4       Ethernet2/0
```

路由器6信息：

```
<6>dis ospf lsdb
                 OSPF Process 1 with Router ID 6.6.6.6
                         Link State Database
                          Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            158 24    0                0 SpfTree
                                  ……………………
Net  10.56.0.5        5.5.5.5            346 32    80000004         0 SpfTree
SNet 0.0.0.0          4.4.4.4            190 28    80000003         1 Uninitialized
                                  ……………………
SNet 10.34.0.0        4.4.4.4            190 28    8000008a        10 Uninitialized
                       AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE  6.6.6.6          6.6.6.6            163 36    8000005d         1 Ase List
ASE  10.56.0.0        6.6.6.6            163 36    8000005c         1 Ase List

<6>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
0.0.0.0/0           OSPF     10   21        10.56.0.5       Ethernet1/0
```

从上述信息发现，在STUB区域内的路由器5和6都有类型3的缺省LSA，并产生相应的缺省路由，区域0的路由器3没有缺省LSA也没有缺省路由，可以看出，STUB区域产生的缺省路由只泛洪到整个STUB区域。

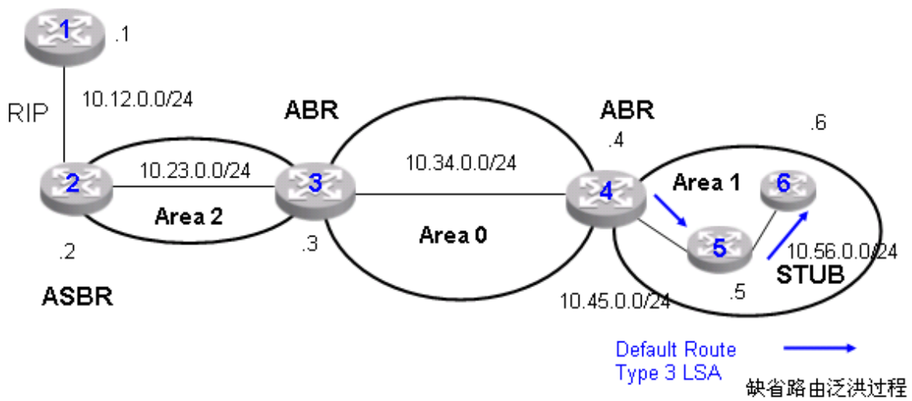


图4 STUB区域缺省路由泛洪

完全STUB区域产生的缺省路由LSA类型、方式和STUB区域一样，并且在STUB区域内泛洪。

配置完全STUB区域，在完全STUB区域的ABR上配置：

```
stub no-summary
```

- NSSA区域

配置一个区域成为NSSA区域，必须在本区域的所有路由器上区域视图配置：

```
nssa
```

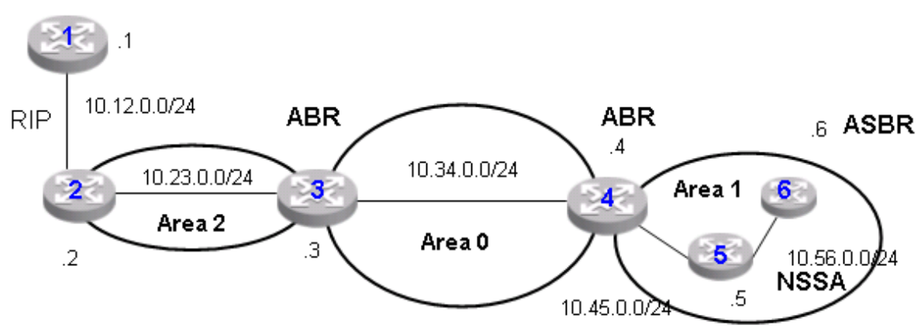


图5 NSSA区域实验组网图

观察路由器4、5、6的链路状态数据库和路由表，并没有产生任何缺省路由。

要在NSSA区域ABR上产生缺省路由，在区域视图下使用该命令：

```
nssa default-route-advertise
```

如图，区域1是NSSA区域，路由器4是其区域中的ABR，现在要在路由器4上产生一条缺省路由。

```
ospf 1
 area 0.0.0.1
  network 10.45.0.0 0.0.0.255
  nssa default-route-advertise
 #
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 10.34.0.0 0.0.255.255
```

观察路由器4的链路状态数据库可以发现，路由器4在区域1生成一条类型7缺省路由的LSA，Link State ID为0.0.0.0。

```
<4>dis ospf lsdb
                 OSPF Process 1 with Router ID 4.4.4.4
                        Link State Database
                            Area: 0.0.0.0
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 4.4.4.4          4.4.4.4            3398 24   0                 0 SpfTree
                                  ...................
ASB  2.2.2.2          3.3.3.3            570 28    80000097          1 Uninitialized
                            Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            1051 24   0                 0 SpfTree
                                  ...................
SNet 10.34.0.0        4.4.4.4            1504 28   80000099         10 Inter List
NSSA 0.0.0.0          4.4.4.4            1051 36   80000005          1 Nssa List
                                  ...................
NSSA 10.56.0.0        6.6.6.6            282 36    8000000c          1 Uninitialized
```

```
                    AS External Database:
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
ASE   6.0.0.0         4.4.4.4            44 36     80000002         1 Ase List
                                  ……………………
ASE   1.1.1.1         2.2.2.2             7 36     800000f8         1 Uninitialized
```

路由器4成的缺省LSA：

```
Type        : NSSA
Ls id       : 0.0.0.0
Adv rtr     : 4.4.4.4
Ls age      : 928
Len         : 36
Seq#        : 80000002
Chksum      : 0xa4fa
Options     : (No Type 7/5 translation, DC)
Net mask    : 0.0.0.0
    Tos 0 metric: 1
    E type      : 2
    Forwarding Address :0.0.0.0
    Tag: 1
```

观察路由器3、5、6的链路状态数据库和路由表，看看路由器4缺省路由的泛洪。

路由器3的信息：

```
<3>dis ip ro
 Routing Table: public net
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
1.1.1.1/32          O_ASE    150  1         10.23.0.2       Ethernet1/0
2.2.2.2/32          OSPF     10   2         10.23.0.2       Ethernet1/0
3.3.3.3/32          DIRECT   0    0         127.0.0.1       InLoopBack0
4.4.4.4/32          OSPF     10   11        10.34.0.4       Ethernet2/0
5.5.5.5/32          OSPF     10   21        10.34.0.4       Ethernet2/0
6.0.0.0/24          O_ASE    150  1         10.34.0.4       Ethernet2/0
6.6.6.6/32          OSPF     10   31        10.34.0.4       Ethernet2/0
10.6.0.1/32         O_ASE    150  1         10.34.0.4       Ethernet2/0
10.6.0.2/32         O_ASE    150  1         10.34.0.4       Ethernet2/0
10.6.0.3/32         O_ASE    150  1         10.34.0.4       Ethernet2/0
10.23.0.0/24        DIRECT   0    0         10.23.0.3       Ethernet1/0
10.23.0.3/32        DIRECT   0    0         127.0.0.1       InLoopBack0
10.34.0.0/24        DIRECT   0    0         10.34.0.3       Ethernet2/0
10.34.0.3/32        DIRECT   0    0         127.0.0.1       InLoopBack0
10.45.0.0/24        OSPF     10   20        10.34.0.4       Ethernet2/0
10.56.0.0/24        OSPF     10   30        10.34.0.4       Ethernet2/0
127.0.0.0/8         DIRECT   0    0         127.0.0.1       InLoopBack0
127.0.0.1/32        DIRECT   0    0         127.0.0.1       InLoopBack0
```

路由器5信息：

```
<5>dis ospf lsdb
                    OSPF Process 1 with Router ID 5.5.5.5
                          Link State Database
                         Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6         1775 24      0                0 SpfTree
                                  ……………………
NSSA 0.0.0.0          4.4.4.4           791 36     80000003         1 Uninitialized
                                  ……………………
NSSA 10.56.0.0        6.6.6.6          1199 36     8000000d         1 Uninitialized

          All areas is NSSA area, AS external database is disabled.

<5>dis ip ro
 Routing Table: public net
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
0.0.0.0/0           O_NSSA   150  1         10.45.0.4       Ethernet2/0
```

路由器6信息：

```
<6>dis ospf lsdb
                    OSPF Process 1 with Router ID 6.6.6.6
                              Link State Database
                             Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            1976 24   0                 0 SpfTree
                                   ……………………
NSSA 0.0.0.0          4.4.4.4            994 36    80000003          1 Uninitialized
                                   ……………………
NSSA 10.56.0.0        6.6.6.6            1400 36   8000000d          1 Nssa List

          All areas is NSSA area, AS external database is disabled.

<6> dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost        Nexthop          Interface
0.0.0.0/0           O_NSSA   150  1           10.56.0.5        Ethernet1/0
```

从上述信息发现，在NSSA区域内的路由器5和6都有类型7的缺省LSA，并产生相应的缺省路由，区域0的路由器3没有该条LSA，可以看出，NSSA区域ABR产生的缺省路由只泛洪到整个NSSA区域，类型7的缺省路由LSA并没有在ABR上转换成类型5的LSA泛洪到整个OSPF域中。

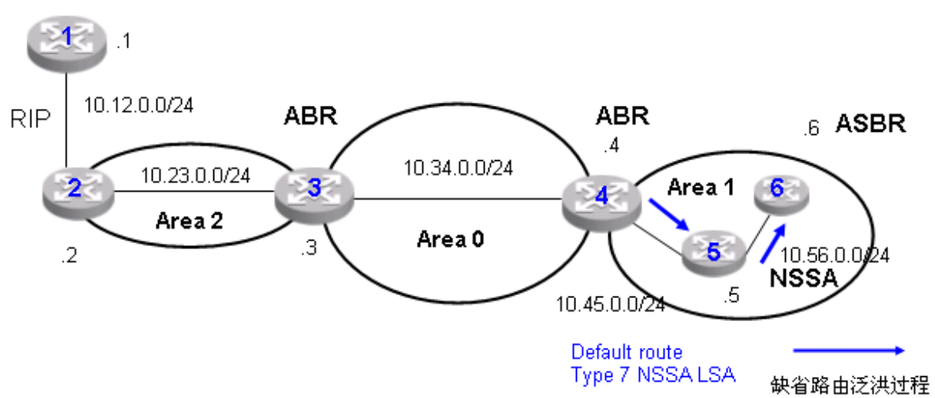


图6 NSSA区域ABR缺省路由泛洪

要在NSSA区域ASBR上产生缺省路由，在区域视图下使用该命令：

nssa default-route-advertise

该命令生效的前提条件是NSSA ASBR已经有一条缺省路由。

如图，区域1是NSSA区域，路由器6是其区域中的ASBR，现在要在路由器6上产生一条缺省路由。

```
ospf 1
  import-route direct
  area 0.0.0.1
    network 6.6.6.6 0.0.0.0
    network 10.56.0.0 0.0.0.255
    nssa default-route-advertise
#
  ip route-static 0.0.0.0 0.0.0.0 6.0.0.2 preference 60
#
```

观察路由器6的链路状态数据库可以发现，路由器6在区域1生成一条类型7缺省路由的LSA，Link State ID为0.0.0.0。

```
<6>dis ospf lsdb
                    OSPF Process 1 with Router ID 6.6.6.6
                              Link State Database
                             Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            1424 24   0                 0 SpfTree
                                   ……………………
NSSA 0.0.0.0          6.6.6.6            423 36    80000003          1 Nssa List
                                   ……………………
NSSA 10.56.0.0        6.6.6.6            1382 36   8000000e          1 Nssa List

          All areas is NSSA area, AS external database is disabled.
```

路由器6生成的缺省LSA：

```
Type        : NSSA
Ls id       : 0.0.0.0
Adv rtr     : 6.6.6.6
Ls age      : 753
Len         : 36
Seq#        : 80000003
Chksum      : 0xeda0
Options     : (Type 7/5 translation, DC)
Net mask    : 0.0.0.0
    Tos 0 metric: 1
    E type      : 2
    Forwarding Address :0.0.0.0
    Tag: 1
```

观察路由器3、4、5的链路状态数据库和路由表，看看路由器6缺省路由的泛洪。
路由器3的信息：

```
<3>dis ip ro
 Routing Table: public net
Destination/Mask   Protocol Pre  Cost      Nexthop        Interface
1.1.1.1/32         O_ASE    150  1         10.23.0.2      Ethernet1/0
2.2.2.2/32         OSPF     10   2         10.23.0.2      Ethernet1/0
3.3.3.3/32         DIRECT   0    0         127.0.0.1      InLoopBack0
4.4.4.4/32         OSPF     10   11        10.34.0.4      Ethernet2/0
5.5.5.5/32         OSPF     10   21        10.34.0.4      Ethernet2/0
6.0.0.0/24         O_ASE    150  1         10.34.0.4      Ethernet2/0
6.6.6.6/32         OSPF     10   31        10.34.0.4      Ethernet2/0
10.6.0.1/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.6.0.2/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.6.0.3/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.23.0.0/24       DIRECT   0    0         10.23.0.3      Ethernet1/0
10.23.0.3/32       DIRECT   0    0         127.0.0.1      InLoopBack0
10.34.0.0/24       DIRECT   0    0         10.34.0.3      Ethernet2/0
10.34.0.3/32       DIRECT   0    0         127.0.0.1      InLoopBack0
10.45.0.0/24       OSPF     10   20        10.34.0.4      Ethernet2/0
10.56.0.0/24       OSPF     10   30        10.34.0.4      Ethernet2/0
127.0.0.0/8        DIRECT   0    0         127.0.0.1      InLoopBack0
127.0.0.1/32       DIRECT   0    0         127.0.0.1      InLoopBack0
```

路由器4信息：

```
<4>dis ospf lsdb
                    OSPF Process 1 with Router ID 4.4.4.4
                            Link State Database
                          Area: 0.0.0.0
Type LinkState ID    AdvRouter          Age Len   Sequence    Metric Where
Stub 4.4.4.4         4.4.4.4            311 24    0                0 SpfTree
                                  ……………………
ASB  2.2.2.2         3.3.3.3            430 28    800000bb         1 Uninitialized
                          Area: 0.0.0.1
Type LinkState ID    AdvRouter          Age Len   Sequence    Metric Where
Stub 6.6.6.6         6.6.6.6            1434 24   0                0 SpfTree
                                  ……………………
SNet 10.34.0.0       4.4.4.4            301 28    80000001         10 Inter List
NSSA 0.0.0.0         6.6.6.6            956 36    80000023         1 Uninitialized
                                  ……………………
NSSA 10.56.0.0       6.6.6.6            174 36    8000002f         1 Uninitialized
                       AS External Database:
Type LinkState ID    AdvRouter          Age Len   Sequence    Metric Where
ASE  6.0.0.0         4.4.4.4            65 36     80000002         1 Ase List
                                  ……………………
ASE  1.1.1.1         2.2.2.2            539 36    80000001         1 Uninitialized

<4>dis ip ro 0.0.0.0
Destination/Mask   Protocol Pre  Cost      Nexthop        Interface
0.0.0.0/0          O_NSSA   150  1         10.45.0.5      Ethernet0/0
```

路由器5信息：

```
<5>dis ospf lsdb
                OSPF Process 1 with Router ID 5.5.5.5
                        Link State Database
                        Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            1702 24   0                0 SpfTree
SNet 10.34.0.0        4.4.4.4             557 28   80000001        10 Uninitialized
NSSA 0.0.0.0          6.6.6.6            1219 36   80000023         1 Uninitialized
NSSA 6.6.6.6          6.6.6.6            1114 36   80000030         1 Uninitialized
NSSA 6.0.0.0          6.6.6.6             834 36   80000031         1 Uninitialized
NSSA 10.6.0.1         6.6.6.6             771 36   80000031         1 Uninitialized
NSSA 10.6.0.2         6.6.6.6             771 36   80000031         1 Uninitialized
NSSA 10.6.0.3         6.6.6.6             771 36   80000031         1 Uninitialized
NSSA 10.56.0.0        6.6.6.6             428 36   8000002f         1 Uninitialized

<5>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost        Nexthop         Interface
0.0.0.0/0           O_NSSA   150  1           10.56.0.6       Ethernet1/0
```

从上述信息发现，在NSSA区域内的路由器4和5都有类型7的缺省LSA，并产生相应的缺省路由，区域0的路由器3没有该条LSA，可以看出，NSSA区域ASBR产生的缺省路由只泛洪到整个NSSA区域，类型7的缺省路由LSA并没有在ABR上转换成类型5的LSA泛洪到整个OSPF域中。

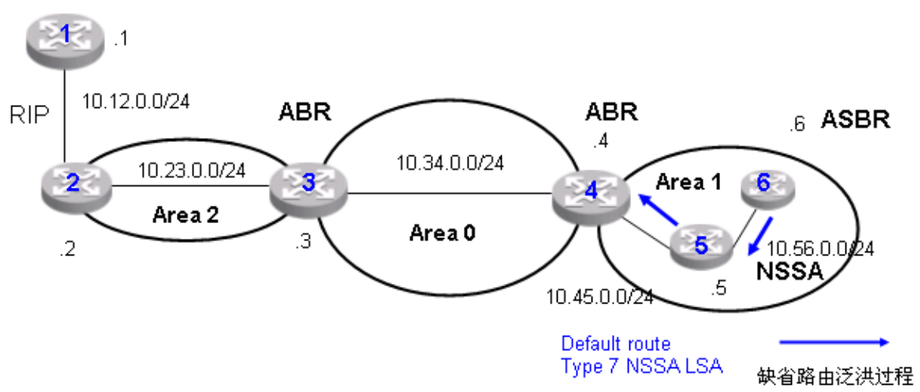


图7 NSSA区域ASBR缺省路由泛洪

完全NSSA区域

配置一个NSSA区域成为完全NSSA区域，在本区域的ABR上区域视图配置：

```
nssa no-summary
```

与NSSA区域不同的是，完全NSSA区域配置后区域内的ABR上会产生一条类型3的缺省路由的LSA，并在NSSA区域内泛洪。

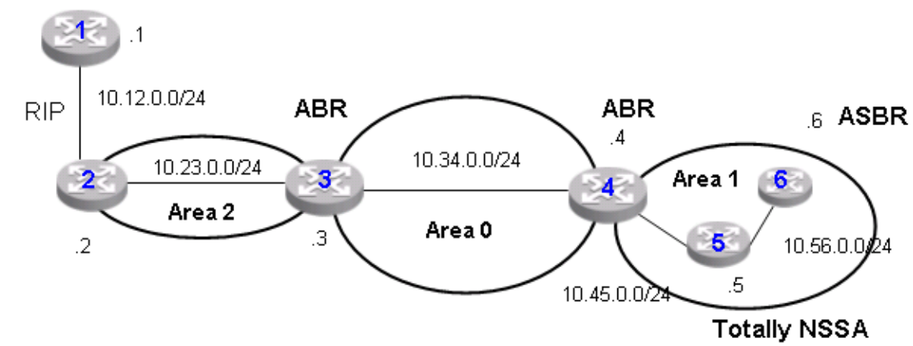


图8 完全NSSA区域实验组网图

如图，区域1是NSSA区域，路由器4是其区域中的ABR，现在要在路由器4上产生一条缺省路由。

```
ospf 1
 area 0.0.0.1
  network 10.45.0.0 0.0.0.255
  nssa no-summary
 #
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 10.34.0.0 0.0.255.255
```

观察路由器4的链路状态数据库可以发现，路由器4在区域1生成一条类型3缺省路由的LSA，Link State ID为0.0.0.0。

```
                OSPF Process 1 with Router ID 4.4.4.4
                        Link State Database
                        Area: 0.0.0.0
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
```

```
Stub 4.4.4.4        4.4.4.4        1826 24    0          0 SpfTree
.......................
ASB   2.2.2.2        3.3.3.3        267 28    800000c0     1 Uninitialized
                    Area: 0.0.0.1
Type LinkState ID   AdvRouter      Age Len   Sequence   Metric Where
Stub 6.6.6.6        6.6.6.6        615 24    0          0 SpfTree
                    .......................
SNet 0.0.0.0        4.4.4.4        453 28    80000001     1 SumNet List
                    .......................
NSSA 10.56.0.0      6.6.6.6        1335 36   80000033     1 Uninitialized
                    AS External Database:
Type LinkState ID   AdvRouter      Age Len   Sequence   Metric Where
ASE   6.0.0.0       4.4.4.4        436 36    80000002     1 Ase List
.......................
ASE   1.1.1.1       2.2.2.2        49 36     80000006     1 Uninitialized
```

路由器4成的缺省LSA：

```
Type        : SumNet
Ls id       : 0.0.0.0
Adv rtr     : 4.4.4.4
Ls age      : 480
Len         : 28
Seq#        : 80000001
Chksum      : 0x39f4
Options     : (DC)      (Non DN)
Net mask    : 0.0.0.0
    Tos 0 metric: 1
    Tag: 1
```

观察路由器3、5、6的链路状态数据库和路由表，看看路由器4缺省路由的泛洪。
路由器3的信息：

```
<3>dis ip ro
 Routing Table: public net
Destination/Mask   Protocol Pre  Cost      Nexthop        Interface
1.1.1.1/32         O_ASE    150  1         10.23.0.2      Ethernet1/0
2.2.2.2/32         OSPF     10   2         10.23.0.2      Ethernet1/0
3.3.3.3/32         DIRECT   0    0         127.0.0.1      InLoopBack0
4.4.4.4/32         OSPF     10   11        10.34.0.4      Ethernet2/0
5.5.5.5/32         OSPF     10   21        10.34.0.4      Ethernet2/0
6.0.0.0/24         O_ASE    150  1         10.34.0.4      Ethernet2/0
6.6.6.6/32         OSPF     10   31        10.34.0.4      Ethernet2/0
10.6.0.1/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.6.0.2/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.6.0.3/32        O_ASE    150  1         10.34.0.4      Ethernet2/0
10.23.0.0/24       DIRECT   0    0         10.23.0.3      Ethernet1/0
10.23.0.3/32       DIRECT   0    0         127.0.0.1      InLoopBack0
10.34.0.0/24       DIRECT   0    0         10.34.0.3      Ethernet2/0
10.34.0.3/32       DIRECT   0    0         127.0.0.1      InLoopBack0
10.45.0.0/24       OSPF     10   20        10.34.0.4      Ethernet2/0
10.56.0.0/24       OSPF     10   30        10.34.0.4      Ethernet2/0
127.0.0.0/8        DIRECT   0    0         127.0.0.1      InLoopBack0
127.0.0.1/32       DIRECT   0    0         127.0.0.1      InLoopBack0
```

路由器5信息：

```
<5>dis ospf lsdb
                    OSPF Process 1 with Router ID 5.5.5.5
                            Link State Database
                           Area: 0.0.0.1
Type LinkState ID   AdvRouter      Age Len   Sequence   Metric Where
Stub 6.6.6.6        6.6.6.6        897 24    0          0 SpfTree
                              .......................
SNet 0.0.0.0        4.4.4.4        208 28    80000002     1 Uninitialized
                              .......................
NSSA 10.56.0.0      6.6.6.6        1623 36   80000033     1 Uninitialized
```

```
<5>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
0.0.0.0/0           OSPF     10   11        10.45.0.4       Ethernet2/0
```

路由器6信息：

```
<6>dis ospf lsdb
                    OSPF Process 1 with Router ID 6.6.6.6
                            Link State Database
                            Area: 0.0.0.1
Type LinkState ID     AdvRouter          Age Len   Sequence     Metric Where
Stub 6.6.6.6          6.6.6.6            1004 24   0                 0 SpfTree
                                   ......................
SNet 0.0.0.0          4.4.4.4            316 28    80000002          1 Uninitialized
                                   ......................
NSSA 10.56.0.0        6.6.6.6            1729 36   80000033          1 Nssa List
          All areas is NSSA area, AS external database is disabled.
<6>dis ip ro 0.0.0.0
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
0.0.0.0/0           OSPF     10   21        10.56.0.5       Ethernet1/0
```

从上述信息发现，在NSSA区域内的路由器5和6都有类型3的缺省LSA，并产生相应的缺省路由，区域0的路由器3没有该条LSA，可以看出，完全NSSA区域ABR产生的缺省路由只泛洪到整个NSSA区域。

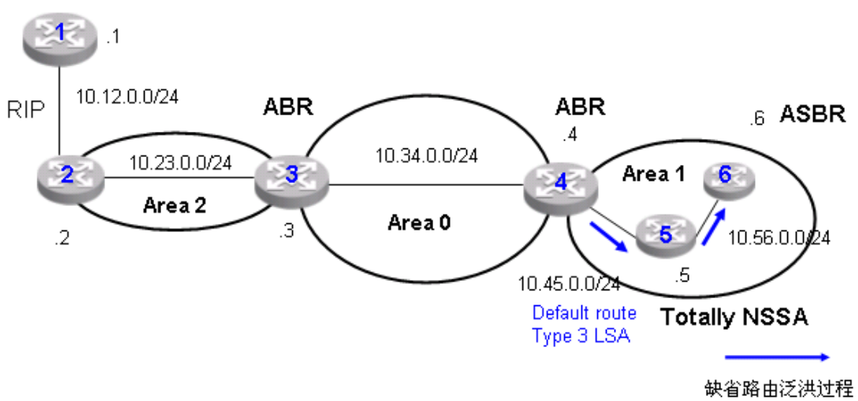


图9 完全NSSA区域缺省路由泛洪

FAQ

- **为什么有的STUB区的ABR没有正确产生缺省路由？**

A：在某些特定组网下（主要是Area 0只采用loopback接口，没有形成邻居），STUB区的ABR没有产生缺省路由。看来这样的组网方式还是有些市场，这其实是出了网上问题几方讨论后的一个妥协结果。因此对于区域边界路由器（ABR），一旦失去跟骨干区域的连接（骨干区域没有FULL邻居），那么就要停止发布缺省路由。这主要用于解决当区域存在多个出口的ABR时，此时可以通过别的ABR出口继续转发报文。

- **在一个STUB区域，有两个ABR，它们产生的缺省路由，不会让它们互相指向，形成路由环路吗？**

A：不会形成环路的。当一个STUB区域中的ABR接收到同一区域ABR发出的默认路由LSA的时候，它只会将其放到LSA Database里，而不会计算出路由，这样就避免了两个ABR缺省路由相互指向形成环路。这其实也是VRP OSPF一个血的教训修改的结果。

如果OSPF路由器已经发布了缺省路由LSA，那么不再学习其它路由器发布的相同类型缺省路由（即路由计算时不再计算其它路由器发布的相同类型缺省路由LSA）。

原因主要有以下两点：

本路由器自身已经具有对外的出口，所以不需要学习其它路由器发布的缺省路由。

如果学习其它路由器发布的缺省路由，就会形成缺省路由的下一条相互指向，造成路由环路。

- **在一个NSSA区域，有两个ABR，它们都会将type 7 LSA转换成type 5 LSA吗？**

A：不会的。按RFC 1587：OSPF NSSA Option的规定，在NSSA区域有多个ABR时，只有最大的Router ID的ABR将type 7 LSA负责转换为type 5 LSA。

如果OSPF路由器已经发布了缺省路由LSA，那么不再学习其它路由器发布的相同类型缺省路由（即路由计算时不再计算其它路由器发布的相同类型缺省路由LSA）。

小结

OSPF缺省路由发布的方式取决于引入缺省路由的区域类型，主要分普通区域、STUB区域和完全STUB区域、NSSA区域和完全NSSA区域。

OSPF缺省路由发布的方法：

配置特定的区域自动生成，STUB/Totally STUB/Totally NSSA

配置生成缺省路由的命令：普通区域 default-route-advertise [always]

NSSA区域 nssa default-route-advertise

OSPF缺省路由通常应用于下面两种情况：区域边界路由器（ABR）发布和自治系统边界路由器（ASBR）发布。

产生条件 / 区域类型	自动产生	配置	ABR	ASBR	有出口	已有缺省路由	产生LSA类型	泛洪范围
普通区域		default-route-advertise		√		√	类型5	整个OSPF域
普通区域		default-route-advertise always		√			类型5	整个OSPF域
Stub区域	√		√		√		类型3	STUB
完全STUB区域	√		√		√		类型3	STUB
NSSA区域		nssa default-route-advertise	√				类型7	NSSA
NSSA区域		nssa default-route-advertise		√		√	类型7	NSSA
完全NSSA区域	√		√				类型3	NSSA

表1 OSPF缺省路由总结表

参考资料

[1] 魏秀刚，《OSPF缺省路由总结》
[2] 刘宇，OSPF FAQ
[3] Cisco，How Does OSPF Generate Default Routes
[4] Cisco，How OSPF Injects a Default Route into a Normal Area
[5] Cisco，How OSPF Injects a Default Route into a Stub or Totally Stub Area
[6] Cisco，How OSPF Injects a Default Route into a Not So Stubby Area

OSPF协议新特性指南

鉴定测试中心 刘宇

OSPF协议/模块简介

OSPF（Open Shortest Path First）协议是IETF为IP协议族开发的高效率、开放的IGP（Internal Gateway Protocol）路由协议，截止2004-10-28，最新的标准为RFC 2328。

OSPF是基于链路状态技术的路由协议，作为曾经广泛使用的基于D-V算法的路由协议如RIP的替代者，OSPF引入了路由验证、支持VLSM、路由聚合、支持层次性网络等特征，经过多年的发展，OSPF被几乎所有主流的设备提供商支持并广泛部署，已经成为被IEFT推荐Internet最广泛使用的IGP。

OSPF新特性指南

以下的新特性是从OSPF基本功能（按RFC 2328为基准实现）以来，IOS所做的OSPF协议扩展及功能增强各IOS特性详细文档包括最新的IOS Feature guide和IOS Command References Books，文档可以在如下站点获取：

无Cisco CCO账号：
http://www.cisco.com/univercd/cc/td/doc/product/software/ios122s/index.htm

有Cisco CCO账号
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1839/products_feature_guides_list.html获取。

接口开销自动计算参考基数：auto-cost reference-bandwidth

特性描述

本命令用来控制OSPF计算接口缺省cost，在OSPF配置模式下使用，赋予一个基值，接口没有明确配置OSPF cost的时候，cost按配置的基值除以接口带宽来计算。缺省为100M。

IOS 12.2中本特性命令为： auto-cost reference-bandwidth

特性优点

当运行OSPF的路由器存在多个1000M以上的高速接口时候，如果接口没有明确赋予OSPF cost，按缺省公式自动计算的cost将都为1，不能反映链路速率。可以用此命令来调整基数，正确反映链路速度，优选路径。

需求分析

VRPv5提供。

RFC 2328兼容RFC 1583：compatible rfc1583

特性描述

RFC 2328修改了部分RFC 1583的路由计算方法，对选择不同途径得到的AS-external-LSAs 的选路优先级做出了新的规定（RFC 2328 Section 16.4）。为了避免在网络中与按RFC 1583的路由器组网运行OSPF由于算法的细微差别造成可能的路由计算错误（可能形成路由环路），RFC 2328规定必须实现可配置的RFC1583compatibility的全局参数。

在整个OSPF路由域中，该参数必须一致，以尽量减少可能出现的路由环路。

特性优点

两点：一是如果OSPF的协议实现是按照RFC 2328实现的，按协议要求必须实现；二是在遇到和按RFC 1583实现的路由器OSPF一起组网的时候，可以打开兼容开关，避免可能出现的路由环路。

需求分析

VRP3虽然号称按RFC 2328实现，但一直没有此命令，市场和产品也一直没有反馈。由于要对协议细节非常熟悉，了解这个的人不多。VRPv5已经实现该特性。

OSPF ABR Type 3 LSA 过滤

特性描述

本特性在Cisco网站上的URL如下：
http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080087be9.html。IOS 12.2(4)T及以上支持。

OSPF ABR Type 3 LSA 过滤特性将ABR的能力扩展，使其能在OSPF区域间发布Type 3 LSAs的时候进行过滤。本特性可以限制仅允许满足某些前缀的报文从一个区域发送到另一个区域，这种类型的区域路由过滤可以应用到某个特定区域，对入区域、出区域或两者同时都可以进行控制。在IOS 12.2中本特性命令为： area {area-id} filter-list prefix {prefix-list-name in | out}

特性优点

OSPF ABR Type 3 LSA 过滤特性能给管理员在OSPF区域间提供增强的路由控制能力。在某些网络中，需要区域间路由隔离可以起到很好的应用作用。

需求分析

事实上这个特性是比较有用的，VRP应该提供。

域名查找：ip ospf name-lookup

特性描述

IOS本命令用来启用DNS查询OSPF的show信息中的路由器。在OSPF协议配置模式下使用。类似于IS-IS的动态主机名转换。由于IS-IS协议报文的扩展性比较好，IS-IS为该机制已经有了RFC 2763: Dynamic Hostname Exchange Mechanism for IS-IS。OSPF只是一个扩展特性而已。

特性优点

本特性使得在OSPF的show信息中更容易识别一个路由器，因为是用路由器名字代替了路由器的Router ID，进一步方便了网络管理人员。

需求分析

很少人知道该特性，更少人使用，Cisco的资料中，show ip ospf database中的显示也基本是按Router id来显示的，说明Cisco自己也很少使用。由于VRP的DNS系统做的不够完善，实现该特性有一定难度。

OSPF Support for Fast Hellos

特性描述

本特性在Cisco网站上的URL如下：
http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00801039b1.html；IOS 12.2(15)T引入。

OSPF快速hello报文特性提供了在1秒内发送多个hello报文的途径。

命令参数为在接口模式下配置：ip ospf dead-interval minimal hello-multiplier x

特性优点

本特性可以更快侦测到OSPF邻居的变化，加快OSPF的收敛速度，尤其是LAN上的收敛速度。

需求分析

IOS是在IS-IS有了相应的特性以后，为OSPF新做的该特性，作为路由收敛的一个综合的解决方案提出。加快hello报文的发送频率，虽然可以加快路由收敛速度，但对于启动OSPF接口较多的路由器，频繁的hello报文将对性能造成一定的影响，因此除非收敛速度非常重要，基本上不建议使用该特性。VRPv3的Timer实现机制造成实现Fast hello是有相当困难的。

OSPF Forwarding Address Suppression in Translated Type-5 LSAs

特性描述

本特性在Cisco网站上的URL如下：http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00801541c9.html ；IOS 12.2(15)T引入。

按协议规定，对于NSSA的ABR，在将type 7的LSA转换成Type 5的LSA时，要填写一个非0的forwarding address，本特性可以强制使得转换的LSA中forwarding address填为0.0.0.0。

特性优点

本特性使得NSSA ABR向OSPF骨干区发出的Type 5 LSA配置强制产生0.0.0.0的forwarding address，以便使流量直接流到ABR。

需求分析

IOS对协议的修改优化，事实上OSPF的RFC 2328，包括OSPF NSSA的RFC，对forwarding address的填写一直没有很好的规定，导致VRP最早版本的实现与Cisco有相当大的差异。尤其是NSSA转换的时候的填写方式，曾经导致VRP和IOS路由器组网出现问题，路由计算错误。增加此特性能在很大程度上避免互操作可能出现的问题。暂无人提出该需求，了解的人也非常少。但能解决一些组网上遇到的问题。

OSPF Nonstop Forwarding (NSF) Awareness

特性描述

本特性在Cisco网站上的URL如下：http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080153edd.html；IOS 12.2(15)T引入。

OSPF Nonstop Forwarding (NSF) Awareness特性能使得具有CPE路由器 (customer premises equipment)协助有NSF能力(NSF-aware)的路由器执行不停顿转发(nonstop forwarding)。

特性优点

OSPF NSR Awareness特性允许能感知NSF(NSF-aware)的customer premise equipment (CPE) 路由器协助有NSF能力的路由器(NSF-capable)执行报文的不停顿转发。NSF感知的特性是IOS软件代码的一部分，并不需要配置。本地路由器并不需要执行NSF，但它对NSF的感知能力使得在倒换过程中RIB和有NSF能力的邻居产生的LSA保持一致性和准确性。

需求分析

OSPF NSR Awareness实际上是去年Cisco主推的GRIP (Globally Resilient IP)的一个组成部分，GRIP包含了很多组件，它的部署能很大程度地提高网络的可靠性和稳定性，对于该特性的实现，需要实现对多个协议进行扩展并统一协作。VRPv5已经实现。

OSPF log-adj-changes

特性描述

OSPF log-adj-changes特性使得路由器能再OSPF邻居变化时发送syslog信息到控制台或log sever。命令在OSPF协议配置模式下配置： log-adj-changes [detail] 。

特性优点

OSPF邻居关系的变化将引起一系列的重要变化，了解邻接状态的变化将非常有助于OSPF网络故障的定位和查找。本特性能提供方便的手段来得知邻接关系的变化。

需求分析

log-adj-changes不只是在OSPF中才有，事实上IOS对于维护邻居关系的路由协议都有相应的命令，而VRP做的不够统一，目前VRPv3的实现没有，而IS-IS等有此命令。

由于能增强维护手段，不需要修改协议，实现简单，建议实现。已经有人提出对路由协议增强维护手段，邻接关系的变化也应该是其中之一。

OSPF database-filter

需求分析

本特性能过滤发给某个邻居的LSA或从某个接口发出的LSAs。 有两种应用方式，一是在OSPF协议配置模式下：neighbor database-filter ；二是在接口配置模式下配置：ip ospf database-filter all out。

特性优点

本特性能在Full mesh的情况下，过滤掉产生的重复多余的LSAs，减少协议报文流量对网络的影响和路由器对协议报文的处理。

需求分析

对于同样用途的特性，IS-IS有RFC 2973: IS-IS Mesh Groups，VRPv3和IOS都实现了。但对于OSPF，没有相应的协议机制来实现，本特性是Cisco IOS对协议的应用优化。和IS-IS的应用一样，此特性的应用不多，尤其在应用的时候，如果出现冗余链路故障的话，将造成路由计算错误，有一定的危险。目前没有人对此提出需求。没有什么必要做这个特性。

OSPF flood-reduction

特性描述

在OSPF中，如果LSA没有变化，缺省每30分钟会更新一次OSPF LSA。对一个很稳定的网络，LSA的周期性flooding是不必要的。本特性应用在稳定的网络中，通过LSA加DoNotAge bit置位发送，来使得LSA的周期性flooding和老化失效，用来抑制周期性的flooding。但如果网络拓扑发生变化，LSA都将扩散，不管是否有该命令。在接口模式下使用： ip ospf flood-reduction 。IOS 12.1(2)T开始引入。

特性优点

本特性能在能抑制周期性的LSAs的Flooding。

需求分析

该特性与database-filter作用同样适用于稳定的，冗余度比较高的网络。个人认为勿需要支持。

OSPF DC扩展

特性描述

本特性是OSPF的协议扩展，在接口下配置ip ospf demand-circuit将使得该接口成为DC接口。早在IOS 11.2开始引入。

特性优点

本特性能在线路上形成OSPF邻接关系LSAs交换完成后，抑制以后周期性的hello报文发送和LSA刷新。在低速链路和拨号链路上能节约网络带宽，节省费用。

需求分析

该特性有特定的RFC 1793：Extending OSPF to Support Demand Circuits。VRP1.x最早是支持该特性的，但由于不稳定，基本没有使用。VRPv3和VRPv5都没有实现。在低速链路和拨号链路上还是有一定的需求的，作为OSPF协议的一个基本扩展，尤其是海外应用和部分测试需要。

OSPF：ignore lsa mospf

特性描述

当路由器不支持MOSPF时，为了抑制收到Type 6 Multicast OSPF（MOSPF）报文而发出的syslog信息，可用此命令关闭发送。在OSPF协议模式下配置：ignore lsa mospf

特性优点

能抑制发送不需要的syslog信息。

需求分析

由于MOSPF实现的厂商非常少，Juniper和Cisco都没有实现，而且IP组播技术经过多年发展，MOSPF基本被淘汰，组网时采用MOSPF的可能性非常少。目前没有人提过相关需求，也没有必要实现。

OSPF Nonstop Forwarding（NSF）Aware-ness

特性描述

RFC 2328：OSPF version 2，Appendix E：An algorithm for assigning Link State IDs如下描述：OSPF type3，type5 LSAs是用来描述特定的网络地址的。但可能需要在LSID中设置多位的主机位。这就允许路由器能产生相同的网络地址但不同的掩码的LSAs，这样的情况在部署supernet和子网零的情况下经常发生。附录E给出了一种在LSID中设置主机位的算法。这样的算法是只本地选择，只影响产生LSAs的路由器，不同的路由器可以采用不同的算法。本特性不需要任何额外命令来支持。

特性优点

毫无疑问，在部署超网和子网零或引入路由某些情况下，支持本附录，将能够生成正确的可区分的LSAs，如果OSPF的实现不支持，可能根本计算不出正确路由。

需求分析

实际上，Appendix E是RFC 2328不可分割的一部分，在网络部署的时候很可能会遇到。由于各种原因，VRP一直没有实现该附录。测试部、用服已经多次提出要实现，多年依然。VRPv5已经支持。

OSPF Update Packet-Pacing可配置定时器

特性描述

本特性在Cisco网站上的URL如下：http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00800b5d35.html ；IOS 12.2(4)T引入。

在某些情况下，如需要发送超大量LSAs的时候，可能需要修改OSPF的缺省定时器以改变OSPF的报文发送速率，减轻CPU和buffer的利用率。本特性允许配置OSPF LSA泛洪(flooding)步调速率，重传步调速率和分组更新步调速率。主要包含如下几个方面：1.配置OSPF 泛洪步调定时器能控制OSPF发送队列中相继的LSA更新报文间隔时间；2.配置OSPF重传步调定时器能控制OSPF重传队列中相继的LSA更新报文间隔时间；3.配置成组定时器能控制重新LSAs重新定时刷新时，每组间的刷新间隔时间，但此定时器并不改变单个LSAs的刷新频率（缺省30分钟）。本处分组应用在大型网络中，当定时刷新大量LSAs时候通过分组来调节LSAs刷新的步调。

IOS在OSPF协议模式下使用，对应如下三条命令：timers pacing flood milliseconds；timers pacing retransmission milliseconds；timers pacing lsa-group seconds。

特性优点

本特性为管理员提供了一种在某个区域内有超大量的LSAs需要泛洪时来控制LSA updates发送速率的手段，以减轻CUP和buffer的超高利用率，避免对其它协议和业务造成的影响。

需求分析

在最近的一次C公司对VRP所实现的路由协议的攻击提到过这个特性，他们认为“不能调整LSA的产生控制时间和传播控制时间，使得网络的可控性和灵活性降低，并影响网络的优化”。事实上VRPv3已经实现了：设置接口重传LSA的时间间隔 ip ospf retransmit-interval；设置接口LSA的传输延迟时间：ip ospf transmit-delay；但都是基于接口实现的。IOS保留了这两条接口调节命令，新增加了本特性来做整体的调整。VRPv5由于是参考以前IOS版本实现，也没有实现该特性。另外，请注意，对于LSAs定时刷新，对Type 5 LSA数目巨大的情况下，这种刷新将占用很大的网络带宽资源，将对网络流量和其他路由器的处理造成冲击；而VRP采用分组的方式处理这种刷新，这样将减少流量冲击，有利于网络的稳定运行。具体见专利《OSPF协议第五类LSA分组刷新技术》。

OSPF Retransmissions Limit

特性描述

本特性在Cisco网站上的URL如下：http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00800e95e7.html ；IOS 12.2(11)T引入。

Cisco IOS Release 12.2(4)加入了在链路上(DC或普通链路)数据库交换或update报文发送时候对重传次数的限制，一旦到达该重传次数，就停止重传，以防止在某些特殊情况下邻居在形成邻接关系的过程中不再相应而过多的重传将不必要的浪费带宽。重传的限制是24次。可以通过limit-retransmissions命令设置为1-25次。

特性优点

本特性能在某些特殊情况下，节约带宽，并和其它不支持此特性的IOS版本兼容。

需求分析

IOS对协议的修改优化。事实上IOS与VRP的OSPF实现对重传处理是不一致的，IOS在邻接形成过程中，重传达到一定次数的时候将会把邻居的状态置为初始化，而VRPv3不改变状态(RFC 2328没有明确规定如何处理)。该特性确实有一定的优点，优化的很小的特性，没有必要实现。

OSPF Stub Router Advertisement

特性描述

本特性在Cisco网站上的URL如下：http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00800b5d37.html#wp1030973；IOS 12.1(8)E引入。

OSPF Stub Router Advertisement特性能允许你将一个新的路由器放入OSPF网络中后，并不立即将流量引过来；或者在你因为某种原因需要重起路由器前，实施优雅重起（gracefully shut down），以保证到其它网络的报文不丢失。本特性提供三种配置选项以允许你让运行OSPF的路由器传播一个无限大的metric到其所有的邻居。任选一种配置，路由器将产生并发布一个带有最大metric（LSInfinity：0xFFFF）到所有非stub的链路，这样高metric的LSA将导致其它路由器在选路的时候选择其它的路由器，新加入的路由器将不会成为到其它网络的中转路由器（transit router），直到新路由器的路由表完全收敛后才可能成为中转路由器。

特性优点

两个优点：1.增强网络的稳定性和可靠性：在路由器启动阶段和重起的阶段发布一个最大的metric到所有的邻居将避免邻居路由器用发布的

路由器做中转路由器，因此可以减少报文被丢弃的可能性增强网络的可靠性和稳定性；2. 平滑地移去一个路由器：在shutdown一个路由前发布最大的metric到所有的邻居，将让其它路由器在经过重起路由器的中转路径不可达改选其它路径。

需求分析

实际上IS-IS早就有类似的特性，也就是IS-IS对Overload bit的支持。这是一个比较好的想法，OSPF协议的发展，包括IOS的增强借鉴了IS-IS的很多优点，本特性已经形成了标准RFC 3137：OSPF Stub Router Advertisement。VRPv5实现。

OSPF Route Map-Based Filtering

特性描述

本特性在Cisco网站上的URL如下：
http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a00800b5d37.html#wp1030973；IOS 12.0(24)引入。
OSPF基于route-map的入口路由过滤功能允许使用route-map来对要加入路由表中的OSPF路由进行过滤。用户可以在route-map中使用OSPF路由的任何属性来过滤。在OSPF协议配置模式下使用：distribute-list route-map map-tag in。

特性优点

用户可以使用route-map来限制OSPF路由加入到路由表中，这个动作发生在通过OSPF LSDB计算出OSPF路由后，要加入到路由表中前。需要注意的是，由链路状态路由协议的本质决定，本特性对LSA的泛洪机制没有任何影响。在route-map中，用户可以使用任何与OSPF路由相关的属性来匹配，如下：match interface；match ip address，match ip next-hop，match ip route-source，match metric，match route-type，match-tag；本特性在ASBRs上引入路由对不同前缀加入tags时，在其他路由器需要将这些路由加入路由表前过滤非常有用。

需求分析

VRPv3已经实现类似过滤(直接加acl和IP前缀列表)，但由于不是基于route-map的，没有这样强大的功能，尤其是不能通过tag进行过滤。可以考虑相关路由协议一起增强。

OSPF Shortest Path First Throttling

特性描述

本特性在Cisco网站上的URL如下：
http://www.cisco.com/en/US/products/sw/iosswrel/ps5012/products_feature_guide09186a0080134ad8.html ；IOS 12.2(14)S引入。
OSPF Shortest Path First Throttling特性可以用来以毫秒级配置SPF调度，能潜在地在网络不稳定的时候延迟SPF计算。SPF计算用来在网络发生变化时调用，一次SPF计算可能包括多个网络拓扑的变化事件。SPF计算中间的间隔是动态的，它基于网络拓扑的变化频率，被选定的间隔是界于用户配置的范围内的。如果网络拓扑不稳定，SPF throttling 计算将延长间隔直到网络重新稳定。
IOS在OSPF协议配置模式下配置：timers throttle spf spf-start spf-hold spf-max-wait。

特性优点

本特性能根据网络拓扑变化来动态调整SPF算法的计算间隔,提高网络的可控性和灵活性,优化网络.

需求分析

VRPv3已经实现类似过滤spf-schedule-interval，但是不如IOS对链路状态路由协议都采用统一的“指数后退”算法（可参考Cisco Networkers 2002中的胶片RST-440：Advance in Routing Technology），根据网络变化来动态调整。对网络变化的适应性和灵活性差了很多。由于特性比较新，用户较少用到，没有人提过需求。但C公司对VRP的路由协议攻击中提到过“不支持OSPF的快速收敛（不能调整SPF控制时间）”。建议和IS-IS统一考虑，实现指数后退算法。

OSPF SPF算法增强优化

特性描述

由于SPF算法是OSPF的内部实现，Cisco IOS没有提供专门的文档说明所做的增强特性。但在Cisco Netwoker2002年会的胶片RST-440：Advances in Routing Technology中提到过。
在SPF算法中，当收到一个新的LSP（LSA）时，每个路由器将检查LSP发生的变化，SPF将基于这些信息重新计算来反映变化。但有时如一个down掉的链路并不在SPT上时，并不需要重新计算SPF。
IOS对普通的SPF算法主要在两个方面做了增强：
一是PRC：Partial Route Calculation ：
主要针对IS-IS的SPF算法来做的，在IS-IS中，IP前缀是SPT（shortest path tree）的叶子节点，是利用System ID来标识路由器的。当收到新的LSP时，如果只有叶子路由变化了，没有必要重新构建SPT（在这个LSP发布的到邻居的metric没有改变的前提下），此时我们只需要重新将IP前缀加入路由表即可。
二是Incremental-SPF(i-SPF)：
在路由变化时判断对SPF树的影响，对于不涉及SPF树的变化则不需要计算，对于叶结点的增加和删除也可以实现增量计算但有时如一个down掉的链路并不在SPT上时，并不需要重新计算SPF。可以通过在SPT上保存更多的信息如：父亲列表，邻居列表来实现，基于这些增强的信息和变化，SPF再来计算反映变化。
内部实现，无配置命令。

特性优点

PRC：不需要重新构建SPT，提高收敛速度。
i-SPF：当执行SPF计算的路由器远端频繁变化的时候，该路由器就能越节约SPF计算时间。虽然i-SPF的计算时间依然不好预测，但不会比原来的SPF计算时间差，据测试能提高80%的收敛速度。

需求分析

算法内部实现，在一定条件能相当程度提高路由协议的收敛速度，一般用户是难以了解，因此没有人提出需求，VRPv3实现有一定困难：
一是算法中我们每次在重新计算前都重新清空SPF树，要做到不清空而比较还有一定困难。二是我们现在缺少算法理论基础，要做到优化必须先有一定理论研究（当然可能会有公开资料、论文来说明优化细节）。
V5实现。

OSPF路由聚合控制cost增强

特性描述

按RFC 2328规定，ABR聚合路由时，重新生成的LSA中cost是被聚合的路由中最大值，本特性可以加强控制，对生成的聚合路由可以配置其cost。
IOS命令为在OSPF协议配置模式下配置：
area area-id range ip-address mask [advertise | not-advertise] [cost cost]；
IOS 12.2开始引入此增强可选参数。

特性优点

本特性让用户根据需要配置聚合路由的cost，在一个区域多ABR时，能够灵活地根据需要通过发布的cost来控制网络流量。

需求分析

事实上很多公司，如AVICI的IPriori的OSPF实现早就做了这个增强，Cisco IOS接纳是其优化协议，适应客户需求和网络网络不断改进的一个很好的例子。

OSPF缺省路由产生控制

特性描述

很简单，实际上只不过是缺省路由发布能根据route-map来控制，发布前先查找自己是否有某条或某些路由，满足route-map的要求后在发布。
IOS在OSPF协议配置模式下配置：default-information originate [always] [metric metric-value] [metric-type type-value] [route-map map-name]

特性优点

本特性能根据网络拓扑变化来动态决定本地是否产生缺省路由，提高网络的可控性和灵活性，优化网络设计。

需求分析

如果对IOS比较熟悉的话，应该知道Cisco在缺省路由发布时候的策略是统一的，各个路由协议都能实现这样的控制。在某些组网应用中，确实有这样的需求。实际上曾经有市场的同事提出过。
VRP的各路由协议是否统一考虑修改，待讨论。

OSPF Sham-Link Support for MPLS VPN

特性描述

本特性在Cisco网站上的URL如下：
http://www.cisco.com/en/US/products/sw/iosswrel/ps5012/products_feature_guide09186a00800a8570.html；IOS 12.2(8)T引入。
在MPLS VPN配置的时候，OSPF经常用来作为PE-CE中间的路由协议，以通过MPLS骨干网络连接VPN内的各站点。虽然OSPF PE-CE之间的连接保证了VPN站点间的连通性，但VPN站点间的backdoor后门链路也应该考虑。如果两个站点属于同一个site，那么经过后门链路的路径总是会被优先选择，因为按OSFP协议规定，intra-area的路径总是优于inter-area路径。因此，必须考虑存在后门链路时，能通过策略来控制路由。
如果后门链路只是用来作为备份使用而不参与VPN业务提供服务，缺省的处理流程将不可接受。为了重新通过MPLS VPN骨干区建立站点间的连接，必须在相关PE路由器的入口和出口VRF间建立逻辑的intra-area链路。本特性提供一种解决方案，在两个站点间建立一个假OSPF链路：sham-link，作为intra-area通道，以使得两个站点间的之间的通讯通过MPLS骨干区域，而后门链路作为备份使用。如果两个站点间不存在后门链路，没有必要使用sham-link。

特性优点

在同一VPN的不同站点间存在后门（backdoor）链路的时候，提供优化控制，可以让站点间的流量优先走MPLS骨干网络，后门较低速链路作为备份使用。

需求分析

VRPv3已经接受需求，启动了OSPF VPN项目，由于是按draft-rosen-vpns-ospf-bgp-mpls-05实现，必然支持OSPF sham-link。有版本已经支持。

OSPFv3支持IPv6

特性描述

最新发布的Cisco IOS 12.2(15)T和Juniper JUNOS 5.7已经正式支持OSPFv3（RFC 2740）。链路状态的路由协议中，由于IS-IS相对简单，并且天生从设计上就支持多协议，Cisco等业界领先的公司最先实现的支持IPv6的链路状态路由协议是IS-IS。但从IPv4到IPv6，OSPF一直是IETF所推荐的IGP，提供完善的支持是势在必行。
OSPFv3关于OSPFv2的一些保留和改变，不再此详述，具体细节可以参考RFC 2328和RFC 2740及Cisco的文档：OSPFv3 Overview。

特性优点

OSPFv3依然是IETF推荐的IPv6使用的IGP，以后的广泛应用是不可避免的，是提供完善的IPv6动态路由协议支持的必然要求。

需求分析

VRPv5正在实现努力中。

MPLS流量工程OSPF扩展

特性描述

流量工程是对于给定资源和拓扑的网络采用的各种优化利用率的技术的通称，MPLS TE是MPLS技术的一个重要应用。OSPF作为最常用的链路状态路由协议，可以通过扩展，利用type 10 LSA（Opaque LSA）来泛洪网络拓扑和资源使用信息，再依据网络的限制条件利用CSPF来计算并建立单向的tunnels， 流量将根据不同的目的地和其它条件被映射到不同的tunnel上，以达到对网络的充分利用的目的。

特性优点

MPLS TE的实现需要对链路状态路由协议进行一定的扩展，OSPF作为最常用的链路状态路由协议，其TE扩展是必然的需求。

需求分析

TE实现的必然途径，V5支持。

以下部分参考了JUNOS OSPF比较好的实现特性，作为VRP OSPF增强的参考.....

OSPF引入路由数量限制

特性描述

把外部路由作为type 5 LSA引入OSPF路由域时，按协议规定type 5 LSA将在整个OSPF路由域内泛洪。JUNOS 5.7实现了一个新的特性，能限制OSPF引入路由数目，
[edit protocols ospf level level-number] prefix-export-limit number ;

特性优点

在各种测试和应用中，OSPF经常引入外部路由，以学习到更多的路由做极限测试，但由于基本实现并不限制路由数目，路由器可能耗尽内存而重起，本特性能限制引入路由数目，在一定程度保护路由器，提高可靠性。

需求分析

在测试中已经反馈需要提高性能和做好保护，VRP可以考虑实现，这在所有的路由协议中都可以统一考虑做下保护。

OSPF Area Transit Capability

特性描述

在某些场合下，transit区域可能有多个ABR，而穿过transit区域配置的Virtual－link只连接某个ABR，此特性允许v-link另外一端的ABR选择更优于沿V-link的路径传送流量。事实上这个特性是与RFC 2328兼容的。
此特性在IOS中是缺省打开的，如果需要取消，可以在ABR的OSPF协议配置模式下配置：
no capability transit
这样将强制流量在v-link上传输。

特性优点

本特性能在多个transit区域有多个ABR的情况下，选择不通过v-link的最优路由传输流量。

需求分析

一般情况下，没有太大必要对此进行控制，选择total cost最低的最佳路由是最好的方案，但需要了解下IOS的此特性。
VRP没有必要实现此特性。

OSPF Link-State Database Overload Protection

特性描述

本特性能限制每个OSPF进程所能接收的非自己产生的OSPF LSA数量。过多的OSPF LSA将耗尽CPU和内存资源。

当此特性使能的时候，路由器保持一个收到的非自己产生的LSA的计数器，当接收到的LSAs数目到达配置的阀值时，产生错误信息，并记录。当接收到LSAs数目超过配置的的最大的LSAs数超过时，路由器将发出通告。如果接收的LSAs的数量在一分钟内还在增加，OSPF进程将关闭所有的邻居并清除OSPF数据库。在此时的“ignore”状态中，OSPF进程忽略本进程收到的OSPF报文并且不产生如何OSPF报文。OSPF将在ignore状态中保持在max-lsa命令中配置的“ignore-time ”时间。每次当OSPF进程进入ignore状态时，有一个计数器加一。当这个计数器超过配置的 ignore-count 参数时，OSPF将永远停留在同样的ignore状态，只有手动干预才能将OSPF进程从ignore状态下解除。当OSPF进程在正常状态下运行了reset-time 参数配置的时间内，ignore状态计数器将清零。

本特性及上述参数在OSPF进程配置模式下配置：

max-lsa maximum-number [threshold-percentage] [warning-only] [ignore-time minutes] [ignore-count count-number] [reset-time minutes]

特性优点

本特性在协议级别提供可靠性保护，限制OSPF进程所能接收非自己产生的OSPF LSA。在网络上，如果路由器配置错误，有可能产生大量的LSA，比如引入大量的路由，此特性能保护路由器因此产生的对CPU利用率和内存资源的消耗。

需求分析

出于对路由器高可靠性的考虑，综合各协议都做了对接收路由数目限制的特性，建议VRP平台实现该特性。

OSPF per-Interface Link-Local Signaling

特性描述

本特性允许OSPF以接口为单位选择性的使能/去使能Link-local Signaling(LLS)特性，而不管OSPF的全局配置如何。

当LLS在路由器级别上使能时，它将自动作用到所有的接口。此特性允许你以接口为单位使能/去使能Link-local Signaling(LLS)特性。在网络中可能存在不遵循RFC 2328的非Cisco路由器，在某个特定接口上去使能LLS，能协助正常形成OSPF邻接关系。

在接口模式下配置：[no | default] ip ospf lls [disable]

特性优点

当LLS在路由器级别上使能时，它将自动作用到所有的接口。此特性允许你以接口为单位使能/去使能Link-local Signaling(LLS)特性。在网络中可能存在不遵循RFC 2328的非Cisco路由器，在某个特定接口上去使能LLS，能协助正常形成OSPF邻接关系。

需求分析

V5提供。也是实现OSPF GR的必要特性。

在接口上用ip ospf area命令启用OSPF

特性描述

本特性允许你用ip ospf area命令在接口上显示的使能OSPF并指定所属的区域，而不用在OSPF配置模式下用network area命令了，可以简化在unnumbered的接口下的配置。在接口模式下配置：ip ospf area x

特性优点

似乎看不到有什么优点，很有意思的是以前VRP1.x就是如此配置OSPF区域的，后来VRP3改到OSPF配置模式下。但要注意的是对于unumbered接口来说，原来的命令只能让这个接口和被借用的接口在同一区域，而本命令可以取消这个限制。

对于IOS来说，这个命令配置接口区域和通过netwrok area配置的命令来说，具有更高的优先级别。

本命令在IOS 12.0(29)S以上版本支持，T版本目前还不支持。

需求分析

暂无需求实现。对于一些熟悉IOS的海外客户来说，可能会提出该需求。

BFD for OSPF

特性描述

Bi-directional Forwarding Detection (BFD) 是一种新的HA机制。它在保持很小的开销的基础上，提供了转发平面的快速故障检测功能。BFD是一种通用的技术，它也可以为任何协议在任何介质上提供单一的标准化的链路/设备/协议的故障检测。BFD和OSPF的绑定，能让OSPF从原来的依靠Hello协议的秒级的邻居故障发现降低到毫秒级。

在特权模式下配置：[no] bfd interval <50-999> min_rx <1-999> multiplier <3-50> 启动BFD，再在OSPF配置模式下配置：bfd all-interface 绑定BFD，使得OSPF的失效检测与BFD相关联。

详细文档可以参考：BFD for OSPF

特性优点

BFD和OSPF的绑定，能让OSPF从原来的依靠Hello协议的秒级的邻居故障发现降低到毫秒级。本命令在IOS 12.0(31)S以上版本支持部分硬件平台，T版本预计在12.4以上版本支持。

需求分析

由于Juniper早已实现，IETF成立了专门的BFD的工作组，在CN2项目中，我司受引导已经被要求实现BFD。在最近的IOS中，已经初步实现了BFD的特性及与路由协议的绑定。在高端产品中HA的要求从目前业界的发展趋势来看，在我司软件平台中实现BFD是必然的需求。

VRP3 vs. IOS OSPF实现差异

本部分内容可以参考全面的专利分析，实现标准分析和所缺少特性分析，以下介绍些其它的差异。由于工作原因长时间没有具体和开发测试团队实质接触了，仅以几年前的记忆做参考。

命令行部分

由于最近的IPR问题，VRP已经将所有的命令行进行了整改，80-90%已经和IOS不一样了，OSPF也不在本文详述。具体请参考VRP最新的配置手册OSPF模块和IOS的配置手册。

相关特性

上一章节中列出的一些内容可以作为VRP和IOS的实现差异，这些特性VRP都实现了较简单方式，而Cisco做了进一步的加强，如：

4.14 OSPF Update Packet-Pacing Configurable Timers;
4.15 OSPF Retransmissions Limit;
4.16 OSPF Stub Router Advertisement;
4.17 OSPF Route Map-Based Filtering;
4.18 OSPF Shortest Path First Throttling;
4.20 OSPF路由聚合控制cost增强
4.21 OSPF缺省路由产生控制

其它

这里提供一些其它比较细节的差异，这些作为华为公司相关测试和研发人员都应该知道：

1. OSPF router id的配置：VRP在全局模式下配置；IOS新加入没有多久，在OSPF配置模式下使用。
2. 邻居建立过程：对重传处理是不一致的，IOS在邻接形成过程中，重传达到一定次数的时候将会把邻居的状态置为初始化，而VRPv3不改变状态(RFC 2328没有明确规定如何处理)。请参考4.15 OSPF Retransmissions Limit。
3. 对等价路由数量的支持：VRP3所有的路由协议共用一个宏，用户不可调整，3条；而IOS对各个路由协议都提供max-paths命令，提供单独的调节，缺省4条，最大6条。VRPv5和IOS实现一样。
4. NBMA网络上cost的处理：VRP不能对单独的邻居配置不同的cost；IOS可以对单独的邻居配置不同的cost，加强了网络控制能力。
5. 对于路由优先级的控制：IOS能依据ACL和IP前缀列表对不同的路由前缀设置；VRP不能；这个差异，实际上存在各个路由协议。
6. loopback接口缺省的OSPF cost：VRP缺省是1562，IOS缺省是1；IOS显然比较合理些。但由于是loopback是虚拟接口，不形成邻居，不影响全局的计算。
7. 路由引入：IOS的OSPF路由引入做的比较差，引入带子网的路由必须加入subnet关键字，对于初学者经常会发生疑惑，为什么路由没有引入到OSPF？VRP没有这个参数。
8. 配置区域：最新的IOS S版本可以在接口下配置区域，VRP3取消了本功能。

总结

由于R&S在国内的流行，很多人都认为自己精通OSPF，实际上大部分人对OSPF的一些新的特性，OSPF实现的一些细节并不清楚。本文总结了IOS目前的最新的一些关于OSPF的进展和新特性，可以作为与时俱进的参考资料。我认为，如果你希望自己成为一个真正精通OSPF的高手，那么这些细节和进展你都是需要精通的。

参考资料

[1] Cisco System，Networker2002，“RST 440-Advances in Routing Technology”
[2] Cisco System，IOS 12.3 “IOS Feature guide”；
[3] Cisco System，IOS 12.3 “Command References Books”
[4] 华为技术，《VRP3.x OSPF配置手册》，《命令行手册》
[5] 华为3COM内部技术网站（测试中心维护）：tech.huawei-3com.com

OSPF中5类LSA的Forwarding Address

鉴定测试中心 朱皓

RFC中对于FA的描述

原文

RFC2328中2.3小节

"The "forwarding address" has one other application. It enables routers in the Autonomous System's interior to function as "route servers". For example, in Figure 2 the router RT6 could become a route server, gaining external routing information through a combination of static configuration and external routing protocols. RT6 would then start advertising itself as an AS boundary router, and would originate a collection of OSPF AS-external-LSAs. In each AS-external-LSA, Router RT6 would specify the correct Autonomous System exit point to use for the destination through appropriate setting of the LSA's "forwarding address" field." [1]

FA还有另一个应用，FA可以使一个AS中的某一台路由器成为一个"路由服务器"。这样该路由器宣告自己为一个ASBR，并且把将通过静态配置或者其它路由协议得到的路由信息以5类LSA的形式引入到OSPF区域之中，并且通过设定FA来标明正确的AS出口位置。

RFC2328中16.4小节

"Call the destination described by the LSA N. N's address is obtained by masking the LSA's Link State ID with the network/subnet mask contained in the body of the LSA. Look up the routing table entries (potentially one per attached area) for the AS boundary router (ASBR) that originated the LSA. If no entries exist for router ASBR (i.e., ASBR is unreachable), do nothing with this LSA and consider the next in the list.

Else, this LSA describes an AS external path to destination N. Examine the forwarding address specified in the AS-external-LSA. This indicates the IP address to which packets for the destination should be forwarded.

If the forwarding address is set to 0.0.0.0, packets should be sent to the ASBR itself. ……

If the forwarding address is non-zero, look up the forwarding address in the routing table.[24] The matching routing table entry must specify an intra-area or inter-area path; if no such path exists, do nothing with the LSA and consider the next in the list." [2]

如果LSA描述的目标网络为N。N的地址通过LSA中LS ID和LSA体中的网络掩码计算而来。在路由表中查找生成该LSA的ASBR。如果不存在该ASBR的表项（比如ASBR不可达），那么对该LSA不做操作。

否则，该LSA就描述了到达AS外部网络N的路径。检查AS-external-LSA中的'转发地址'，这个地址就是发往该目标网络N的数据包应该被送达的地址。

如果转发地址被设为0.0.0.0，包应当被转发到ASBR。……..

如果转发地址不是零，那么在路由表中查找FA。匹配FA的路由表项必须是区域内或区域间路径，如果没有符合条件的表项存在，那么对该LSA不做操作。

例子说明

从RFC中对于FA的定义可以看出，FA为外部路由提供了额外的转发信息。FA的重点有两个部分，一个是ASBR与外部路由器的连接必定是通过一个多点可达的网络（NBMA或者以太网）；另一个重点是，FA非0时，SPF计算会根据FA来计算目标网络的下一跳，因此需要对覆盖FA的路由信息做检查。从下面这个例子中可以看到FA的作用。

见下图。RouterC为OSPF区域外部的路由器，它通过一个多点可达的网络与ASBR-1和RouterB连接。ASBR-1将RouterC的网络引入OSPF。当OSPF区域内部的路由器RouterA访问外部网络时，流量会达到RouterB并转发给ASBR-1，然后到达RouterC，也就是图中红线路径。很显然，这样的转发路径在RouterB这里走了一个弯路，因为RouterB完全可以将数据直接通过多点可达网络转发给RouterC，也就是蓝色的路径。当出现这样的情况时，ASBR-1在引入RouterC的外部网络时，将FA项填写为RouterC的接口地址2.2.2.2，这样RouterB会通过SPF计算，将2.2.2.2作为这些外部网络的下一跳，就避免了数据多传一跳情况的发生。当然在实际情况中可能ASBR-1和RouterB之间有多台设备，虽然只有一个ASBR，但是FA使OSPF区域中不同设备访问外部网络时，可以根据FA计算出更合理的下一跳。

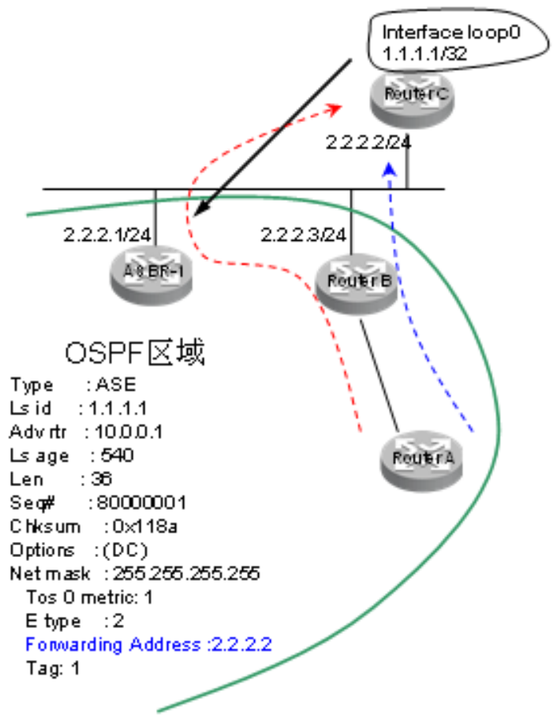


图1 FA的作用

配置

```
RouterC
interface Ethernet0/0
 ip address 2.2.2.2 255.255.255.0
 rip version 2 multicast
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
#
rip
 undo summary
 network 2.0.0.0
 network 1.0.0.0
ASBR-1
router id 10.0.0.1
#
interface Ethernet2/0
 ip address 2.2.2.4 255.255.255.0
 rip version 2 multicast
#
interface LoopBack0
 ip address 10.0.0.1 255.255.255.255
#
ospf 1
 import-route rip
 area 0.0.0.1
network 2.2.2.0 0.0.0.255
network 10.0.0.1 0.0.0.0
#
rip
 undo summary
 network 2.0.0.0
```

```
RouterB
router id 10.0.0.2
#
interface Ethernet0/0/0
 ip address 2.2.2.3 255.255.255.0
#
interface LoopBack1
 ip address 10.0.0.2 255.255.255.255
#
ospf 1
 area 0.0.0.1
  network 2.2.2.0 0.0.0.255
  network 10.0.0.2 0.0.0.0
```

结果

```
<RouterB>display ospf lsdb ase 1.1.1.1

         OSPF Process 1 with Router ID 10.0.0.2
                 Link State Database

 Type      : ASE
 Ls id     : 1.1.1.1
 Adv rtr   : 10.0.0.1
 Ls age    : 304
 Len       : 36
 Seq#      : 80000001
 Chksum    : 0x118a
 Options   : (DC)
 Net mask  : 255.255.255.255
     Tos 0 metric: 1
     E type      : 2
     Forwarding Address :2.2.2.2
     Tag: 1
```

```
<RouterB>display ip routing-table
 Routing Table: public net
Destination/Mask    Protocol Pre  Cost      Nexthop         Interface
1.1.1.1/32          O_ASE    150  1           2.2.2.2         Ethernet0/0/0
2.2.2.0/24          DIRECT   0    0           2.2.2.3         Ethernet0/0/0
2.2.2.3/32          DIRECT   0    0           127.0.0.1       InLoopBack0
10.0.0.1/32         OSPF     10   2           2.2.2.4         Ethernet0/0/0
10.0.0.2/32         DIRECT   0    0           127.0.0.1       InLoopBack0
127.0.0.0/8         DIRECT   0    0           127.0.0.1       InLoopBack0
127.0.0.1/32        DIRECT   0    0           127.0.0.1       InLoopBack0
```

FA对路由计算的影响

- 目前VRP对FA填写的规定

  OSPF在ASBR的下一跳接口启用；

  ASBR的下一跳接口没有被设置为被动接口；

  ASBR的下一跳接口不是OSPF P2P或P2MP类型的；

  ASBR的下一跳接口地址是落在OSPF协议中发布的网络范围之内；

  除此之外，其它情况FA都填为0.0.0.0。

这个规定与CISCO的IOS相同。

- 覆盖FA的路由问题

在RFC中规定，覆盖FA的路由必须是区域内或区域间路由。为什么要做这样一个检查呢。我们知道OSPF是一个无环的链路状态协议，区域内的无环是依靠最短路径树来保证，而区域间的无环是依靠非骨干区域和骨干区域连接来实现，这些方法保证了AS内部路由计算的可靠性。但是OSPF无法控制AS外部的路由信息是否可靠，FA本身就是用于转发到达外部网络的地址，如果用一个外部路由来迭代查找FA并进行路由计算，那么很可能会导致路由问题。在VRP早期的版本中，对FA的查找可以迭代到一个外部路由上，这样会导致问题。我们来看下面的例子。

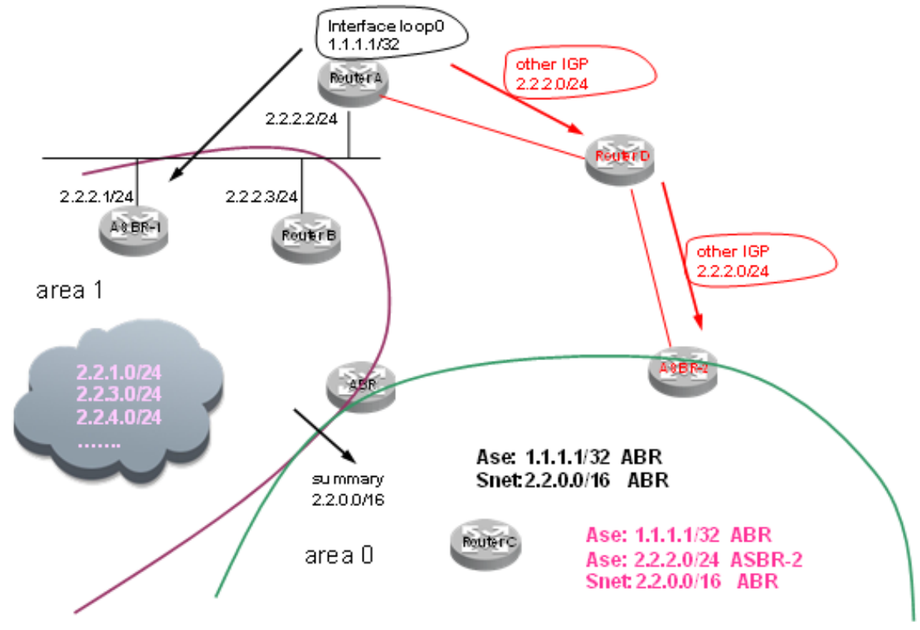


图2 外部路由迭代FA

先不考虑所有红色的拓扑部分。正常情况下ASBR-1从RouterA获得了外部路由1.1.1.1/32并且将FA填写为2.2.2.2。在ABR处，实施了路由聚合策略，区域1的所有2网段路由被汇总了。RouterC是一个骨干区域内的路由器，它会获得3类汇总的LSA2.2.0.0/16和5类LSA1.1.1.1/32并且计算出正确的路由，下一跳都会指向ABR。这个时候跟前面1.2中的例子基本上是完全相同的。

当考虑到红色的拓扑部分时，就有了一些变化。在OSPF区域外部RouterA和RouterD之间运行IGP并且将2.2.2.0/24传递给RouterD，然后ASBR-2做路由引入，将2.2.2.0/24这样一个5类的LSA泛洪进骨干区域。这个时候骨干区域内的路由器RouterC的LSA会包括这样3条：

Ase: 1.1.1.1/32 FA=2.2.2.2

Ase: 2.2.2.0/24 Adv router=ASBR-2

Snet:2.2.0.0/16 Adv router=ABR

如果按照更精确的外部路由来迭代FA，那么路由计算的结果就是1.1.1.1/32的下一跳指向ASBR-2，接着ASBR-2会将数据报文送到RouterD，而RouterD可能没有到达1.1.1.1/32的路由，于是出现问题。

可见对于迭代FA的路由进行限制是必要的。但是这样的限制又产生了新的问题。

FA导致的问题

在CISCO的IOS早期版本的实现中，对迭代FA的路由限制导致了新的问题。在Common Routing Problem with OSPF Forwarding Address文档中，OSPF区域内部的路由器无法计算出正确的外部路由。是两个原因导致了这样问题的产生。

一个是，在早期的IOS版本，当OSPF引入RIP路由的时候，同时在RIP和OSPF宣告了的网络都会被作为5类LSA引入到OSPF的LSDB中。第二个问题是，IOS会查找精确匹配FA的路由信息，如果该路由不是区域内或者区域间路由，那么对该5类LSA就不做处理。我们看一下原文中的这个例子。

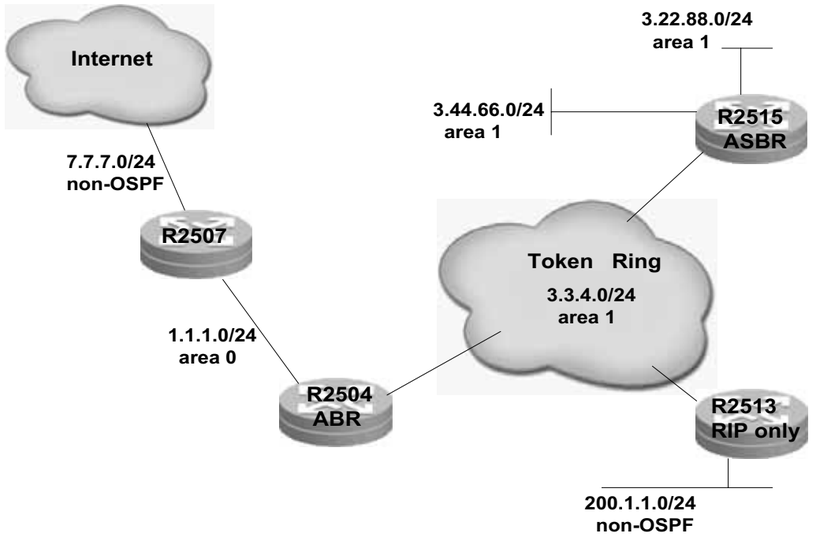


图3 FA产生的问题

R2513是一个只运行RIP的路由器，Token Ring网络是多点可达的。R2515同时运行了RIP和OSPF，并将R2513的外部网络200.1.1.0/24引入OSPF。这时候因为前面提到的第一个原因，在R2515的LSDB中包括这样几个5类LSA

Type-5 AS External Link States

Link ID	ADV Router	Age
3.22.88.0	3.44.66.3	718
3.3.4.0	3.44.66.3	718
3.44.66.0	3.44.66.3	718
200.1.1.0	3.44.66.3	718

R2515将这几个5类LSA传递给R2504，并且对于200.1.1.0/24这个5类LSA，填写FA为R2513的接口地址3.3.4.4。R2504作为ABR在区域边界对3类LSA做了汇总为3.0.0.0/8。这时候R2507的LSDB就变成这样一个情况：4个5类LSA不变，同时有一个覆盖FA的3类LSA（被汇总过的3.0.0.0/8）。进行200.1.1.0/24的路由计算时，对覆盖FA的路由进行精确匹配，发现3.3.4.0/24是一个外部路由，于是对该5类LSA不做处理，结果是R2507的路由表中没有到达200.1.1.0/24的路由，从而导致错误。

规避的方法很多，其实主要就是解决一个问题：当对FA进行路由查找时，防止外部路由成为最精确匹配的路由。所以下面的方法都可以解决问题。

一是在R2504这个ABR处不要对3类LSA进行汇总，那么在R2507上路由计算的结果3.3.4.0/24就是一个区域间路由，符合RFC的规定，因此5类LSA200.1.1.0/24也会得到正确的处理。

二是在ASBR处不将3网段作为5类LSA引入，这个方法在早期的IOS版本上就是在ASBR处进行引入路由的过滤。

当然我们也可以在ASBR处对5类LSA也做汇总，只要汇总的结果不比ABR汇总的结果更小，就可以了。

解决2.3问题的方法

上一节谈到了FA可能导致的问题，以及解决的方法。其实除了前面讨论到的，还有其它的方法，新的IOS做了改进。当一个接口所属的网络同时在RIP和OSPF中宣告的时候，不会再额外产生一个5类的LSA。这样就避免了2.3中出现的问题。

但是IOS目前对FA进行迭代查找时还是会选择最精确的路由，如果最精确匹配到了一条外部路由，那么该ASE就不进行处理。也就是说对于2.2中的情况，IOS还是存在路由问题。不过在IOS的12.3中增加了一个新的特性，可以使用area area-id nssa translate type7 suppress-fa命令。在一个NSSA的ABR处，7类LSA转换为5类LSA时，将FA项改为全0并泛洪进骨干区域，这样就可以进行正确的路由计算了。

VRP的改进方法则不同，首先，同样是当一个接口所属的网络同时在RIP和OSPF中宣告的时候，不会再额外产生一个5类的LSA（但VRP与IOS在引入路由时的实现还是小有区别，有兴趣的同学可以做实验比较一下）。更彻底的是，VRP不管是否存在更精确的外部路由来匹配FA。SPF计算只会用区域内或区域间的路由来迭代查找FA，如果找不到，那么就不处理该5类LSA。这也就是在2.2中VRP可以正确计算出路由的原因。在2.2中，做一点小小的变化，在ABR上改变聚合策略，聚合为2.2.0.0/16后做not-advertise，这时候骨干区域内的路由器RouterC因为找不到区域内或区域间路由可以覆盖FA，会放弃对5类LSA1.1.1.1/32的计算。

相比较而言，似乎VRP的实现方法更安全。

到底在哪个路由表里查找ASBR和FA？

在RFC中规定，当收到5类LSA并做路由计算时，要求首先看引入该5类LSA的ASBR是否路由可达，否则就不对该LSA做处理。当FA非0时，存在类似的问题，需要查找覆盖FA的区域内或者区域间路由，如果存在才进行处理。

ASBR的可达性

在VRP中做如下实验验证是如何对ASBR进行可达性查找的。比如2.2中的例子，如果RouterA这里进行路由过滤，只允许1.1.1.1/32放入路由表。这时候虽然全局路由表中没有ASBR的路由，但是也会计算出正确的外部路由进入路由表。在全局路由表中并不存在ASBR的路由（多数情况下ASBR只是作为一个LSA用于SPF计算，确实并不会产生全局路由的条目），那么是不是说只要LSA存在，能构造出正确的SPF计算树，就可以计算出正确的外部路由呢。

```
RouterA
  router id 10.0.0.3
  #
  interface Ethernet0/0/0
   ip address 2.2.1.1 255.255.255.0
  #
  interface LoopBack0
   ip address 10.0.0.3 255.255.255.255
  #
  ospf 1
   filter-policy ip-prefix ase import
   area 0.0.0.0
    network 2.2.1.0 0.0.0.255
    network 10.0.0.3 0.0.0.0
 #
  ip ip-prefix ase index 10 permit 1.1.1.1 32
[RouterA] display ip routing-table
  Routing Table: public net
Destination/Mask   Protocol Pre   Cost      Nexthop        Interface
1.1.1.1/32         O_ASE     150   1         2.2.1.2        Ethernet0/0/0
2.2.1.0/24         DIRECT    0     0         2.2.1.1        Ethernet0/0/0
2.2.1.1/32         DIRECT    0     0         127.0.0.1      InLoopBack0
10.0.0.3/32        DIRECT    0     0         127.0.0.1      InLoopBack0
127.0.0.0/8        DIRECT    0     0         127.0.0.1      InLoopBack0
127.0.0.1/32       DIRECT    0     0         127.0.0.1      InLoopBack0
```

在RFC的定义中使用了routing table这个词来定义对ASBR可达性，或者FA迭代查找的位置。 其实这里的routing table是指OSPF的路由表。无论是IOS还是VRP都有一个专门的ABR&ASBR路由表。在VRP中的命令为display ospf abr-asbr

```
                OSPF Process 1 with Router ID 104.0.0.6

 Routing Table to ABR and ASBR

  I = Intra i = Inter A = ASBR B = ABR S = SumASBR
 Destination        Area          Cost    Nexthop        Interface
 IB 104.0.0.3       0.0.0.0       1       104.1.36.1     Ethernet1/0
 iS 1.1.1.1         0.0.0.0       1563    104.1.36.1     Ethernet1/0
```

在IOS中的命令为

```
Router# show ip ospf border-routers

      SPF Process 109 internal Routing Table
   Codes: i - Intra-area route, I - Inter-area route
i 192.168.97.53 [10] via 172.16.1.53, Serial0, ABR, Area 0.0.0.3, SPF 3
i 192.168.103.51 [10] via 192.168.96.51, Serial0, ABR, Area 0.0.0.3, SPF 3
I 192.168.103.52 [22] via 192.168.96.51, Serial0, ASBR, Area 0.0.0.3, SPF 3
I 192.168.103.52 [22] via 172.16.1.53, Serial0, ASBR, Area 0.0.0.3, SPF 3
```

相信对于ASBR可达性的查询是以这个表为依据的。

迭代FA路由的问题

当ASBR可达性查找完成了后，就会看5类LSA的FA是否为0，当非0时，要在routing table中查找是否有合法的表项来迭代FA（合法的为intra&inter）。这里的routing table很显然也并非全局路由表。细心的人可能已经注意到这个问题，在1.2小节中的例子里，路由器RouterB使用了FA项作为转发ASE的下一跳。但是在RouterB的路由表中覆盖FA的只有一条直连路由而已！那么迭代FA的查找肯定不是在全局路由表中进行的。

在VRP中，使用display ospf routing可以看到这样一张OSPF的路由表。

```
[RouterB-ospf-1] display ospf routing

                OSPF Process 1 with Router ID 10.0.0.2
                         Routing Tables

Routing for Network
Destination        Cost Type NextHop        AdvRouter        Area
2.2.2.0/24          1 Stub 2.2.2.3          10.0.0.2         0.0.0.1
……
```

与全局路由表比较一下，会发现即使是直连网段，只要OSPF被使能，就会存在于OSPF路由表中。

```
[RouterB-ospf-1] display ip routing-table
 Routing Table: public net
Destination/Mask    Protocol Pre   Cost        Nexthop          Interface
1.1.1.1/32          O_ASE    150   1           2.2.2.2          Ethernet0/0/0
2.2.2.0/24          DIRECT   0     0           2.2.2.3          Ethernet0/0/0
2.2.2.3/32          DIRECT   0     0           127.0.0.1        InLoopBack0
```

这解释了前面对FA迭代合法性的问题。

那么如果在直连网段上OSPF没有被使能，会出现什么情况呢？这时候在OSPF路由表中存在可迭代FA的intra类型OSPF路由

```
                OSPF Process 1 with Router ID 10.0.0.2
                         Routing Tables

Routing for Network
Destination        Cost Type NextHop        AdvRouter        Area
2.2.2.0/24          3 Stub 104.1.23.3       10.0.0.1         0.0.0.1
……(104.1.23.3为ASBR-1与RouterB建立OSPF邻接关系的地址)
```

这时候路由计算的结果将变为：

```
[RouterB-ospf-1] display ip routing-table
 Routing Table: public net
Destination/Mask    Protocol Pre   Cost        Nexthop          Interface
1.1.1.1/32          O_ASE    150   3           104.1.23.3       Ethernet0/0/1
2.2.2.0/24          DIRECT   0     0           2.2.2.3          Ethernet0/0/0
2.2.2.3/32          DIRECT   0     0           127.0.0.1        InLoopBack0
```

前一小节中提到VRP对于FA的迭代是直接在合法路由中进行，所以虽然2.2.2.2地址直连可达，但是因为OSPF没有在2.2.2.3接口上被使能，所以RouterB的OSPF路由表中只计算出通过ASBR-1到达2.2.2.0/24的路由，按照OSPF路由表计算的结果自然是将下一跳指向了ASBR-1。

这个实验说明，如果计算ASE的路由器并不存在与ASBR的直接邻接，并且FA直连可达。为了使FA实现转发的优化，必须在直连FA的接口也使能OSPF，可以同时将该接口配置为被动接口以减少对网络的影响。比如如下的拓扑。

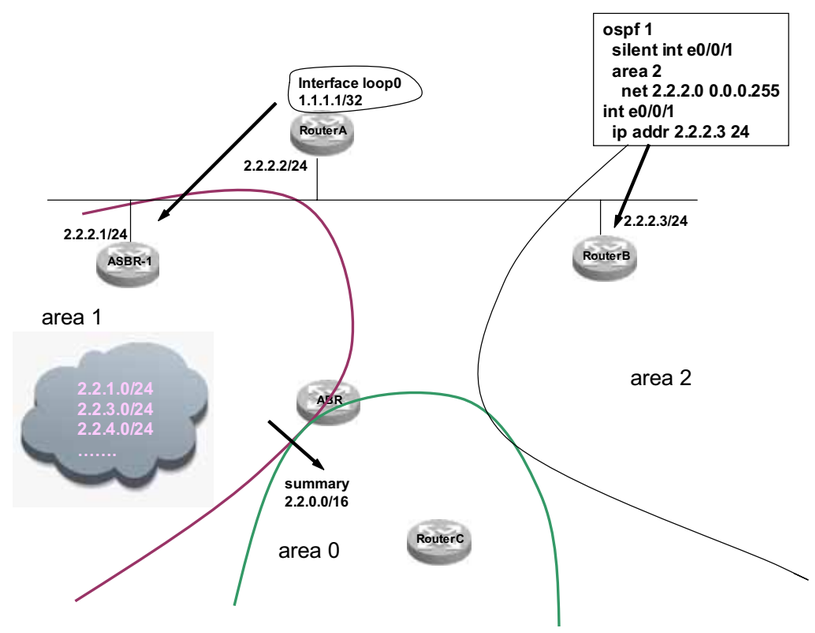


图4 FA对转发的优化

IOS是看不到OSPF路由表的，是不是对于FA的迭代直接在全局路由表中进行了呢？将上面的例子中，将RouterB换成是一个CISCO的设备，如果在2.2.2.3这个接口不使能OSPF，那么外部ASE是不会被计算。使能2.2.2.3接口后，虽然在全局路由表中的到的结果仍然是直连路由的形式，但这个时候外部ASE得到了正确的计算。可见在IOS中对于FA的迭代合法性检查是在全局路由表中进行的，只不过又做了额外检查：直连FA的接口是否使能了OSPF。

比较一下VRP和IOS的实现：

当2.2.2.3接口使能OSPF的时候，两个系统都会根据OSPF路由表迭代FA，正确的计算出ASE的下一跳。

当2.2.2.3接口未使能OSPF的时候，IOS会丢弃ASE，而VRP相当于按照FA为0进行了路由计算。

从这两种实现的比较来看，类似2.4的情况，VRP目前的实现更合理一些。

总结

FA为0时

当一个ASE中的FA为0时，按照Adv Rtr（也就是ASBR）来计算该ASE的下一跳。

FA为非0时

同时满足如下条件时，ASBR会在ASE的FA域内填写转发地址。

OSPF在ASBR与外部网络连接的下一跳接口启动；

ASBR与外部网络连接的下一跳接口没有被设置为被动接口；

ASBR与外部网络连接的下一跳接口不是OSPF P2P或P2MP类型的；

ASBR与外部网络连接的下一跳接口地址是落在OSPF协议中发布的网络范围之内；

当一个ASE中的FA为非0时，就不考虑Adv Rtr了，而是用FA来计算该ASE的下一跳。

对于IOS，会在全局路由表中寻找精确匹配FA的路由条目，如果这个路由条目不合法（不是ospf inter/intra路由），此LSA就丢弃，不计算。有一种特例就是当匹配FA的路由为直连路由并且该接口OSPF使能的情况下，该LSA被正确计算。

对于VRP，会在OSPF路由表中寻找区域内或区域间路由迭代查找FA，如果找不到，该LSA就丢弃，不计算。

组网中需要注意的问题

在组网中可能遇到多种版本的IOS或者是VRP，因为这些版本的差异（2.2和2.3介绍），需要在配置规划中注意一些问题。

尽量在ASBR处对外部路由进行聚合。

如果不在ASBR处进行路由聚合，那么进行路由过滤，只引入希望引入的外部路由。

如果ASBR连接外部网络的接口上使能了OSPF，并且该接口连接了一个多点可达的网络，那么在所有OSPF内的路由器，要可以通过区域内或者区域间路由访问ASBR连接外部网络的接口地址。

如果路由器并不存在与ASBR的直接邻接，并且FA直连可达。为了使FA实现转发的优化，必须在直连FA的接口也使能OSPF，可以同时将该接口配置为被动接口以减少对网络的影响。

参考资料

J. Moy，RFC2328—OSPFv2
http://tech.huawei-3com.com/article.php/27
刘宇，《OSPF FAQ》
http://tech.huawei-3com.com/article.php/1079
Cisco，The Effects of the Forwarding Address on Type 5 LSA Path Selection
http://tech.huawei-3com.com/article.php/85
Cisco，Common Routing Problem with OSPF Forwarding Address
http://tech.huawei-3com.com/article.php/71

路由协议选择 OSPF vs. EIGRP

鉴定测试中心　刘宇

路由协议规划选择原则

在互联网飞速发展的今天，TCP/IP协议已经成为数据网络互联的主流协议。各种网络上运行的大大小小各种型号路由器，承担着控制本世纪或许最重要信息的流量，而这成百上千台路由器间的协同工作，离不开路由协议。因此在大型网络的规划构建中，选择适当的路由协议是非常重要的。目前常用的单播路由协议有多种，如RIP、OSPF、IS-IS、BGP，以及Cisco私有的IGRP/EIGRP协议等。不同的路由协议有各自的特点，分别适用于不同的条件之下。

互连是网络构建最基础和最本质的要求，选择适当的路由协议需要以此为目标，并综合考虑以下因素：

- **路由选择协议的开放性**

开放性的路由协议保证了不同厂商都能对本路由协议进行支持，这不仅保证了目前网络的互通性，而且保证了将来网络发展的扩充能力和用户构建网络时的设备选择空间，这点在很多情况下是需要重点考虑的

- **网络的拓扑结构**

网络拓扑结构直接影响协议的选择。例如RIP这样比较简单的路由协议不支持分层次的路由信息计算，对复杂网络的适应能力较弱。对于比较复杂的网络，需要使用处理能力更强的协议，如OSPF、EIGRP等。

- **网络结点数量**

不同的协议对于网络规模的支持能力有所不同，需要按需求适当选择，有时还需要采用一些特殊技术解决适应网络规模方面的扩展性问题。农发展银行全国网络节点较多，路由信息也非常多，而且网络状况会千变万化，将导致路由刷新相对频繁，所以对路由协议的性能提出很高的要求。如能支持的节点数、路由选径是否最佳、路由算法必须具有鲁棒性、快速收敛性、灵活性等。

- **网络间的互通及关联要求**

通过划分成相对独立管理的网络区域，可以减少网络间的相关性，有利于网络的管理和扩展。可通过划分区域等形式，路由协议要能支持减少网络间的相关性。必要时还要考虑路由信息安全因素和对路由交换的限制策略管理。

- **管理和安全上的要求**

通常要求在可以满足功能需求的情况下尽可能简化管理。但有时为了实现比较完善的管理功能或为了满足安全的需要，例如对路由的传播和选用提出一些人为的要求，就需要路由协议对策略的支持。

根据以上原则，现在各种大型网络构建中，为节省投资、保证网络的持续扩展性，都在使用开放、标准而又健壮的协议。如整个金融系统网络都在由IBM专有的SNA体系向开放标准的TCP/IP体系过渡，由于竞争的作用，这将会大大降低银行网络的构建及维护费用；又如建行网络在改造构建中，全网使用开放标准的OSPF+BGP路由协议，将使得网络在以后的扩展中具有更多的选择空间，不会受到使用某一封闭标准而带来的扩展限制。

OSPF vs. EIGRP路由协议特性比较

OSPF和EIGRP都是近年来出现的比较好的动态路由协议，OSPF以协议标准化强，支持厂家多，受到广泛应用，是IETF推荐使用的内部网关路由协议（IGP）；而EIGRP协议由厂商Cisco公司发明，并靠其在业界的影响力和市场份额，也受到部分用户的认同。然而这两种协议究竟哪种更好，谁更适合网络未来发展的需要？这部分就用户普遍关心的问题，我们从技术角度客观分析这两种协议各自的优缺点，以便网络集成商和企业用户在网络设计规划时能作为参考。

- **OSPF协议**

OSPF协议简介

OSPF是Open Shortest Path First（即“开放最短路由优先协议”）的缩写。它是IETF组织开发的一个基于链路状态算法应用在自治系统内部的路由协议。在IP网络上，它通过收集和传递自治系统内部设备的链路状态信息来动态地发现并传播路由。

每一台运行OSPF协议的路由器总是将本地网络的连接状态，（如可用接口信息、可达邻居信息等）用LSA（链路状态广播）描述，并扩散到整个自治系统中去。这样，每台路由器都收到了自治系统中所有路由器生成的LSA，这些LSA的集合组成了LSDB（链路状态数据库）。由于每一条LSA是对一台路由器周边网络拓扑的描述，则整个LSDB就是对该自治系统网络拓扑的真实反映。

根据LSDB，各路由器运行SPF(最短路径优先)算法。构建一棵以自己为根的最短路径树，这棵树给出了到自治系统中各节点的路由。在图论中，“树”是一种无环路的连接图。所以OSPF计算出的路由也是一种无环路的路由。

OSPF协议为了减少自身的开销，提出了以下概念：

DR（Designated Router）

在各类可以多址访问的网络中，如果存在两台或两台以上的路由器，该网络上要选举出一个“指定路由器”(DR)。“指定路由器”负责与本网段内所有路由器进行LSDB的同步。这样，两台非DR路由器之间就不再进行LSDB的同步。大大节省了同一网段内由于协议报文造成的带宽开销。

区域（Area）

OSPF可以根据自治系统的拓扑结构和层次管理需求划分成不同的区域（Area），这样区域边界路由器（ABR）向其它区域发送路由信息时，以网段为单位生成摘要LSA。这样可以减少自治系统中的LSA的数量，以及路由计算的复杂度。

OSPF使用4类不同的路由，按优先顺序来说分别是：

区域内路由
区域间路由
第一类外部路由
第二类外部路由

区域内和区域间路由描述的是自治系统内部的网络结构，而外部路由则描述了应该如何选择到自治系统以外目的地的路由。一般来说，第一类外部路由对应于OSPF从其它内部路由协议所引入的信息，这些路由的花费和OSPF自身路由的花费具有可比性；第二类外部路由对应于OSPF从外部路由协议所引入的信息，它们的花费远大于OSPF自身的路由花费，因而在计算时，将只考虑外部的花费。

OSPF协议特点

总的来说，由于OSPF发展成熟，厂商支持广泛，已经成为世界上使用最广泛的IGP，尤其在企业级网络，也是IETF推荐的唯一的IGP。其他路由协议所能适应的网络和具备的主要

优点，OSPF都能适应。

1. OSPF是真正的loop-free（无路由自环）路由协议：源自其采用算法本身（链路状态及最短路径树算法）的优点；
2. OSPF收敛速度快：能够在最短的时间内将路由变化传递到整个自治系统并完成路由重新计算；
3. 支持等价路由负载分担，能更有效地利用链路资源；
4. 提出区域（area）划分的概念，将自治系统划分为不同区域后，通过区域之间的对路由信息的摘要，大大减少了整个自治系统所需传递的路由信息数量，减轻了对路由器的性能需求和管理难度，也使得路由信息不会随网络规模的扩大而急剧膨胀；
5. 协议设计精巧，将协议自身的报文开销控制到最小。主要采用的技术如下：

用于发现和维护邻居关系的是定期发送的是不含路由信息的hello报文，非常短小。包含路由信息的报文时是触发更新的机制（有路由变化时才会发送）。但为了增强协议的健壮性，每1800秒全部更新一次。

在广播网络中，使用组播地址（而非广播）发送报文，减少对其它不运行OSPF的网络 设备的干扰。

在各类可以多址访问的网络中（广播，NBMA），通过选举DR，使同网段的路由器之间的路由交换（同步）次数由 O（N*N）次减少为 O（N）次。

提出STUB区域的概念，使得STUB区域内不再传播引入的AS外部路由，并可以控制其它区域LSA的传入。

在ABR（区域边界路由器）上支持路由聚合，进一步减少区域间的路由信息传递。

在点到点接口类型中，通过配置按需拨号属性（OSPF over On Demand Circuits），使得OSPF不再定时发送hello报文及定期更新路由信息，保证低速链路上能节约网络带宽的消耗。只在网络拓扑真正变化时才发送更新信息。

6. 通过严格划分路由的级别（共分四级），提供更可信的路由选择。
7. 良好的安全性，OSPF支持基于接口的明文及MD5协议报文验证，可以很好地防止恶意攻击和错误的配置；
8. OSPF适应各种规模的网络，经过适当的规划可以支持多达数千台。
9. 具备链路状态路由协议能感知全局网络拓扑相关信息的特点，可以扩展支持流量工程，最大程度地提高骨干网络资源的使用效率。

EIGRP协议

EIGRP协议简介

EIGRP和早期的IGRP协议都是由Cisco发明，是基于距离向量算法的动态路由协议。

EIGRP(Enhanced Interior Gateway Routing Protocol)是增强版的IGRP协议。它属于动态内部网关路由协议，仍然使用矢量－距离算法，但它也具有一些链路状态路由协议的特点，因此有些文献也称其为“混合型算法路由协议”。但它的实现比IGRP已经有本质上的改进，其收敛特性和操作效率比IGRP有显著的提高。

EIGRP的收敛特性是基于DUAL（Distributed Update Algorithm）算法的。DUAL 算法的特性使得路径在路由计算中根本不可能形成环路。它的收敛速度可以与已存在的其它任何路由协议相匹敌。

EIGRP协议特点

EIGRP协议主要具有如下特点：

精确的路由开销计算和多网络协议的支持

EIGRP协议继承了IGRP协议的最大的优点：混合矢量路由权。EIGRP协议在路由计算中要对网络带宽，网络时延，信道占用率，信道可信度等因素作全面的综合考虑，所以EIGRP的路由计算更为准确，更能反映网络的实际情况。同时EIGRP协议支持多种网络协议，如IPX，CLNP。

较少的带宽占用

使用EIGRP协议的路由器之间周期性的发送很小的hello报文，以此来保证从前发送报文的有效性及邻居的可达性。路由更新的发送使用增量发送方法，即每次只发送发生变化的路由。发送的路由更新报文采用可靠传输，如果没有收到确认信息则重新发送，直至确认。EIGRP还可以对发送的EIGRP报文进行控制，减少EIGRP报文对接口带宽的占用率，从而避免连续大量发送路由报文而影响正常数据业务的事情发生。

无环路由和较快的收敛速度

路由计算的无环路和路由的收敛速度是路由计算的重要指标。EIGRP协议由于使用了DUAL算法，使得EIGRP协议在路由计算中不可能有环路路由产生，同时路由计算的收敛时间也有很好的保证。因为，DUAL算法使得EIGRP在路由计算时，只会对发生变化的路由进行重新计算；对一条路由，也只有此路由影响的路由器才会介入路由的重新计算。

协议安全支持协议报文MD5认证

为确保路由获得的正确性，运行EIGRP协议进程的路由器之间可以配置MD5认证，对不符合认证的报文丢弃不理，从而确保路由获得的安全。

任意掩码长度的路由聚合

EIGRP协议可以通过配置，对相关的EIGRP路由进行任意掩码长度的路由聚合，从而减少路由信息传输，节省带宽。

支持等价和非等价路由负载分担

EIGRP一大特点就是不但可以支持等价路由负载分担，而且可以支持非等价路由负载分担，在几条不同开销的链路上按比例发送流量，能更有效地利用链路资源，这点是OSPF所做不到的。

但由于支持非等价路由负载分担对路由器造成一定的压力，并且可能会有路由环路，因此即使是Cisco公司，也不推荐使用此特性。

协议配置简单

使用EIGRP协议组建网络，路由器配置非常简单，它没有复杂的区域设置，也无需针对不同网络接口类型实施不同的配置方法。使用EIGRP协议只需在路由器上启动EIGRP 路由进程，然后再使用network 命令使能网络范围内的接口即可

OSPF和EIGRP的比较

OSPF和EIGRP都是收敛速度较快并且不会形成环路的算法，网络带宽占用较小，使用灵活，安全性较好的路由协议。但是从以上分析可以看出，各自还是有优缺点。

OSPF的缺点

配置相对复杂：由于网络区域划分和网络属性的复杂性，需要网络分析员有较高的网络知识水平才能配置和管理OSPF网络，但由于OSPF使用的广泛性，绝大部分管理员都熟悉OSPF,问题并不大。

路由负载均衡能力较弱：OSPF虽然能根据接口的带宽等信息，自动生成接口路由开销，但通往同一目的的不同开销路由，OSPF只选择最优路径转发（当然，可以支持等价路由负载分担），不同开销的路径，不能实现负载分担，不象EIGRP那样可以配置一个范围，根据链路开销不同，按比例发送流量。

EIGRP的缺点

1. EIGRP没有区域（Area）的概念，不适合

层次性较强的网络部署。而OSPF在大规模网络的情况下，可以通过划分区域来规划和限制网络规模。所以EIGRP适用于网络规模相对较小的网络，这也是矢量-距离路由算法（RIP协议就是使用这种算法）的局限所在。EIGRP虽然也可以划分自治系统并互相引用，但是这样的组网方案在实际中很少使用，效果不及OSPF的划分区域理想，也不如OSPF+BGP的实现方案。

2. 没有在拨号链路上的DC扩展：运行EIGRP的路由器之间必须通过定时发送HELLO报文来维持邻居关系，这种邻居关系即使在拨号网络上，也需要定时发送HELLO报文，这样在按需拨号的网络上，无法定位这是有用的业务报文还是EIGRP发送的定时探询报文，从而可能误触发按需拨号网络发起连接，尤其在备份网络上，引起不必要的麻烦。所以一般运行EIGRP的路由器，在拨号备份端口还需配置Dialer list和Dialer group，以便过滤不必要的报文，或者运行TRIP协议，这样做增加路由器运行的开销。而OSPF可以提供对拨号网络按需拨号的支持，只用一种路由协议就可以满足各种专线或拨号网络应用的需求。

3. EIGRP的无环路计算和收敛速度是基于分布式的DUAL算法的，这种算法实际上是将不确定的路由信息（active route）散播（向邻居发query报文），得到所有邻居的确认后（reply报文）再收敛的过程，邻居在不确定该路由信息可靠性的情况下又会重复这种散播，因此某些情况下可能会出现该路由信息一直处于active状态（这种路由被称为stuck in active route），并且，如果在active route的这次DUAL计算过程中，出现到该路由的后继（successor）的metric发生变化的情况，就会进入多重计算，这些都会影响DUAL算法的收敛速度。而OSPF算法则没有这种问题，所以从收敛速度上看，虽然整体相近，但在某种特殊情况下，EIGRP还有不理想的情况。比如说对于狭长形的网络拓扑结构，如果一端路由器的网络路由丢失，丢失路由的路由器将向所有邻居发起查询，每个邻居又会向各自的所有邻居发起查询。在狭长形的网络拓扑结构中，一端的路由信息变化需要较长时间才能传递到另外一端，然后才能从远端向回收敛，效率不理想。

4. 在共享网段上，EIGRP路由器将两两交换路由信息，但是OSPF因为选举DR、BDR，所有路由器将只和DR、BDR交换路由信息。这样EIGRP将会使用更多的带宽来传送协议报文，当一个共享网段上有大量EIGRP路由器时，协议报文对带宽的占用更加严重。

5. EIGRP是Cisco公司的私有协议。Cisco公司是该协议的发明者和唯一具备该协议解释和修改权的厂商，没有受到外界监督与完善。Cisco公司对该协议的修改没有义务通知任何其他厂家和使用该协议的用户，因此会在网络升级扩展中导致运行的安全隐患；从另外一个角度，如果用户使用了这种协议组网，由于担心网络扩展而出现的隐患，从而选择了单一设备，则又将陷入封闭网络的怪圈，步入使用SNA这种私有网络体系的后尘，最终使得网络的构建及维护费用都大大增加。而OSPF是开放的协议，是IETF组织公布的标准。世界上主要的网络设备厂商都支持该协议，所以它的互操作性和可靠性由于公开而得到保障，并因为竞争的存在也使得网络的构建在合理的价格范围内，而在众多的厂商支持下，该协议也会不断走向更加完善。

OSPF与EIGRP的比较总结

下面是OSPF和EIGRP路由协议简单的比较：

比较点	OSPF	EIGRP
协议标准	IETF 标准协议，协议设计完美成熟，绝大部分厂商支持，组网不受厂商选择限制	Cisco 私有协议，不能与其它厂商联合组网；协议实现在根据使用经验不断优化，比如增加对 stub 的处理，不如 OSPF 成熟。
部署范围	IETF 推荐的 IGP，世界上使用最广泛的路由协议	仅有少数网络使用，而且部署越来越少
核心算法	SPF 算法收敛快，无环路	采用分布式的 DUAL 算法，中间状态不可预测，可能陷入 SIA 状态，查询可能扩散到全网
网络拓扑支持	支持层次性网络拓扑，具有很好的可管理性和扩展性	不支持层次性网络
对新技术的支持	可以扩展支持 OSPF-TE	无 TE 扩展支持，大型网络中采用 EIGRP 作 IGP，将限制 TE 的部署

表1 OSPF和EIGRP比较总结

从EIGRP网络到OSPF网络的迁移

对于一个已经部署EIGRP协议的网络，如何在尽量减少对现有业务和流量的冲击的前提下，整网切换到OSPF协议，是很多已经选择部署EIGRP协议用户在网络路由协议迁移和网络扩容时的疑虑。很容易想到的是删除原来路由器上与EIGRP相关的配置，再在各个路由器上配置OSPF，但这样粗暴的迁移肯定会对正在运行的业务造成中断，无疑是不可取的。实际上路由协议的平滑迁移远没有想象的困难。按照一定正确的切换操作步骤，可以保证在切换过程中基本上始终保持网络的连通性，对现有业务基本没有什么影响，所有的工作只是简单的配置命令的修改而已。

路由协议的平滑迁移主要是利用路由优先级（Huawei VRP称为preference，Cisco IOS称为distance）的差异，当多个路由协议运行的时候，路由是基于各自协议的优先级来决定谁加入路由表中（通常数值大的优先级反而低）。主要的操作步骤如下：

1. 做好OSPF的网络规划，比如区域划分，路由策略等；

2. 在已经运行EIGRP的路由器上迭加配置OSPF，首先将OSPF的路由优先级设置比EIGRP的要低（数值更大），再启动相应接口，加入相应的区域；

3. 按第二步完成所有相关路由器的配置，在此过程中，由于EIGRP的路由更优先，即使OSPF学习到相关路由，也不会加入到路由表中，所有路由器的路由表都是完整，而且路径都是连通的；

4. 等待一定时间（视网络规模而定），OSPF路由收敛后再将所有路由器上的EIGRP的优先级改为比第二步中设置的OSPF优先级更低（或在OSPF配置中将OSPF的优先级改为比迁移前EIGRP的更高），此时由于OSPF已经收敛，OSPF路由马上会替换EIGRP路由到路由表中去来指导报文转发。

5. 删除所有路由器上的EIGRP配置，并按需要将OSPF路由优先级改为缺省值（由于只有OSPF运行，可以不修改），由于在第四步路由已经切换到OSPF，一般来说现有流量并不会中断，至此，整网的路由协议就由EIGRP平滑切换到OSPF。

虽然采用上述方法切换短暂的路由循环和黑洞路由依然有可能存在，但可能性很小，而

且迁移风险小，此外，由于在迁移过程中最后删除EIGRP进程之前保留了原来的路由协议，如果由于某种原因希望改回原来的配置是非常容易的。需要考虑的一点是，由于在迁移过程中要启动两套路由协议，如果网络过大，要注意所涉及的路由器的性能需求是否满足（CPU性能和内存大小）及所需要配置的工作量，可以先做好配置的脚本，通过网管系统来配合采用自动更新配置的手段来操作。

IPv6 OSPFv3协议详解_RFC2740

测试研究部 许亮

概述

本文基于RFC2740介绍了OSPFv3与OSPFv2的区别和实现，对继承OSPFV2的内容可以参考RFC2328，这里不再赘述。

OSPFV3是在原有OSPF V2基础上开发的支持IPV6的协议，它使用的仍然是链路状态算法，OSPFV2的基本机制，如洪泛（flooding）、DR选举、区域划分、SPF算法保持不变。相对于OSPFV2，OSPFV3最大的变化就是对IPV6地址的支持，以及对IPv6体系架构的兼容。另外，OSPFv3在OSPFv2的基础上，对功能做了增强。

语义

IPv6中使用地址/前缀长度描述地址，而没有子网掩码的概念。IPv6中的链路（Link）就类似于IPv4中子网或网络。因此，本文中也使用链路（Link）表达类似的意思。

与Version2的差别

OSPFv3保留了OPSFv2的大部分机制。为了支持IPv6地址、IPv6报文结构和体系，OSPFv3主要做了以下修改：

从per-sub到per-link

IPv6使用链路（link）表示节点赖以在链路层通信的媒介或工具。Interface连接到链路上。多个IPv6地址前缀可以分配到一个单独的链路上；对连接到链路上的两个节点，即使他们的IPv6地址前缀不同，也可以直接通信。

相应的，OSPFv3也是运行在链路上，而不象IPv4中是基于网段的。链路的概念取代了OSPFv2中的网络和子网。因此，我们说OSPF接口是连接到链路上而不是子网上。这一变化影响了Hello报文的接收以及Hello报文和Network-LSA的内容。

地址

在OSPFv3中，OSPF协议报文头和大部分LSA中去掉了地址部分，从而与网络层协议无关。具体说来有以下方面：

除了LSU报文中的LSA载荷包含IPv6地址信息外，其他的协议报文中都不含有地址字段；

Router-LSA和Network-LSA中去掉了网段地址，用拓扑信息取而代之；

OSPFv3中的Router ID、Area ID和Link State ID保留IPv4中的格式——32bit；

在所有类型的网络中，包括广播和NBMA，都使用Router ID来识别邻居；

洪泛范围

LSA的洪泛范围归纳为3类，放在LSA的LS类型域（field）。这三类是：

区域范围：LSA只在一个OSPF区域内部传播，这类LSA包括Router-LSA、Network-LSA、Inter-Area-Prefix-LSA、Inter-Area-Router-LSA和Intra-Area-Prefix-LSA；

自治系统范围：LSA在整个自治系统内传播，这类LSA只有AS-external-LSA。

多实例

OSPFv3支持在单链路上运行多实例。这使得多个供应商在共享一台甚至多台网络设备的情况下，仍然可以保持各自网络的独立运行。在OSPFv2中，是通过设置不同的OSPF验证来实现的。

在单链路上运行多实例是通过在OSPF报文头和OSPF接口数据结构中包含实例号（Instance ID）做到的。实例号只影响OSPF报文接收。

使用link-local地址

IPv6中的本地链路地址用于单链路上的邻居发现，无状态自动配置等等。对于以本地链路地址作为源地址的报文，IPv6路由器不作转发。本地链路单播地址范围是FF80/10。

OSPFv3假定每个路由器的物理接口都分配了本地链路单播地址。除了虚连接以外，所有的OSPFv3接口都使用本地链路单播地址作为报文的源地址。路由器从链路上学到其他路由器的本地链路单播地址，再使用这些地址作为转发报文的下一跳。

虚连接使用全球范围地址或本地站点地址作为OSPF报文源地址。

本地链路地址只出现在Link-LSA中，其他OSPF LSA不使用本地链路地址。Inter-area-prefix-LSA、AS-external-LSA和intra-area-prefix-LSA中决不能携带本地链路地址。

认证

OSPFv3本身没有认证（Authentication）功能。因此，OSPFv3报文头中去掉了AuType和Authentication字段。相应的，所有的OSPF区域和接口数据结构都去掉了认证相关域（field）。

OSPFv3的认证依赖于IPv6报文的认证头和IP封装安全有效载荷报头。OSPFv3通过这些IP报文头来确保路由交换的完整性和认证/保密。

OSPFv3报文利用IPv6标准的16位完整校验和防止报文数据的随机错误。该校验和覆盖了整个OSPF报文和伪IPv6头。

报文格式

OSPFv3报文直接封装在IPv6报文中。OSPFv3报文头中去除了素有的地址相关字段，做到与网络协议无关。所有的地址信息都放在各种LSA中。OSPFv3报文的具体变化有（分别见图一——图三）：

OSPF版本变为3；

Hello报文和DD报文的Option字段扩展到24位；

OSPF报文头除去了Authentication和AuType字段；

Hello报文不含有任何地址信息，包含Interface ID。发送路由器为链路上的接口分配Interface ID，并用Interface ID唯一的标识这些接口。在Network-LSA中，使用DR的Interface ID作为Link State ID。

在报文Option字段增加了’R-bit’和’V6-bit’。在SPF运算处理Router-LSA时会用到这两个参数。若’R-bit’为空则路由器可以在OSPF拓扑分发时不转发transit流量；this can be used in multi-homed hosts that want to participate in the routing protocol。’V6-bit’对’R-bit’作出了更详细的规定：若’V6-bit’为空，则路由器可以在OSPF拓扑分发时不转发IPv6报文。若设置了’R-bit’但’V6-bit’为空，路由器不转发IPv6报文，但可能转发其他协议体系的报文。

OSPFv3支持在单个链路上运行多个OSPF协议实例（OSPF protocol instance）。在OSPFv3报文头中增加了’Instance ID’字段。

OSPFv3报文头格式

OSPFv3的报文头如下图所示：

```
Version(8bit) |  Type(8bit)   |     Packet length(16bit)
-------------------------------------------------------------
                        Router ID  (32bit)
-------------------------------------------------------------
                         Area ID   (32bit)
-------------------------------------------------------------
          Checksum (16bit)     | Instance ID(8bit) |     0
```

Version：3
Type:：报文类型，1---Hello，2---DD，3---LSR，4---LSU，5---LSAck
Packet length：包括标准OSPF报文头的OSPF报文长度，长度单位Byte。
Router ID：每个OSPFv3进程的唯一标识
Area ID：区域标识符
Checksum：包括IPv6 pseudo-header的OSPF报文校验和
Instance ID：VPN实例号。
OSPFv3中，LSR和LSAck报文的发送保持不变。Hello，DD和LSU报文的发送有所不同。

Hello报文格式

```
                        OSPFv3 Header
                          (128bit)

-------------------------------------------------------------
                      Interface ID (32bit)
-------------------------------------------------------------
  Rtr Priority |            Options (24bit)
-------------------------------------------------------------
        HelloInterval        |        RouterDeadInterval
-------------------------------------------------------------
                  Designated Router ID
-------------------------------------------------------------
               Backup Designated Router ID
-------------------------------------------------------------
                       Neighbor ID #1
-------------------------------------------------------------
                          ......
-------------------------------------------------------------
                       Neighbor ID #n
```

图2 Hello报文格式

OSPFv3 Header：参见OSPFv3报文头格式
Interface ID：接口标识，在路由器上唯一标识接口
Rtr Priority：路由器优先级
HelloInterval：发送Hello 报文的间隔
RouterDeadInterval：此计时器超时后，邻居Down掉
Neighbor ID：邻居的Router ID
Options：V6：为0时，不参加路由计算
E ---- 在Hello报文中，此位为0时，表示此区域不传播AS-External-LSA
MC -- 是否多播
N --- Type-7 LSA 处理相关
R --- 为0时，表明产生此条LSA的路由器状态down掉了
DC-- 环回相关

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8  9  0  1  2  3
| | | | | | | | | | | | | | | | |DC| R| N|MC| E|V6
```

图3 Option选项

DD报文格式

```
                    OSPFv3 Header  (128bit)

-------------------------------------------------------------
      0       |               Options
-------------------------------------------------------------
       Interface MTU         |       0       |0|0|0|0|0|I|M|MS
-------------------------------------------------------------
                    DD sequence number
-------------------------------------------------------------

                    List of LSA Header
```

图4 DD报文格式

Options：与Hello报文中的Options相同
Interface MTU：本地接口的MTU值
I：Initial，初始化位，为1时表明是第一个DD报文
M：More，表明接下来还有DD报文
MS：主/从位（Master/Slave），为1时，表明为主(Master)
DD sequence number：DD报文的序列号
List of LSA Header：LSA头部的集合，用于接受方检查LS数据库

LSR报文
LSR（LS Request）请求所需的LSA

```
                    OSPFv3 Header  (128bit)
-------------------------------------------------------------
              0                  |        LS type
-------------------------------------------------------------
                        Link State ID
-------------------------------------------------------------
                     Advertising Router
-------------------------------------------------------------
                              ...
```

图5 LSR报文格式

Link State ID：LSA的标识
LS Type：参考LSA格式部分

LSU报文
LSU（LS Update）回复所请求的LSA

```
                    OSPFv3 Header  (128bit)
-------------------------------------------------------------
                            # LSAs
-------------------------------------------------------------
                             LSAs
                              ...
```

图6 LSU报文格式

LSAs：报文里包含的LSA数目

LSAck报文
确认收到的LSA

```
                    OSPFv3 Header  (128bit)
-------------------------------------------------------------
                             LSA Headers
                                ...
```

图7 LSAck报文格式

LSA格式

LSA头和Router-LSA、Network-LSA中去掉了所有的地址内容。Router-LSA和Network-LSA描述了路由域的拓扑结构，而该结构与具体的网络层协议无关。增加了新的LSA用来分发IPv6地址信息和下一跳信息。一些IPv4中的LSA名称改变了，以保持所有LSA意思的一致性。LSA的具体变化有：

Option字段从LSA头中拿走放到了Router-LSA、Network-LSA、Inter-Area-Router-LSA和Link-LSA中。同时，Option字段扩展到24bit；

LSA类型字段扩展到16bit（占据了原来Option与LSA Type的位置），最高的3位表示洪泛范围和对未知类型LSA的处理；

LSA中的地址格式为：[前缀，前缀长度]取代过去的[地址，掩码]。缺省路由的前缀长度为0；

Router-LSA和Network-LSA不包含地址信息，即完全独立于具体的网络层协议；

完整的路由器接口信息可能分布在多个Router LSA中，接收路由器在进行SPF运算时需要综合同一路由器发出的多个Router-LSA；

出现一类新的LSA：Link-LSA。这类LSA只在本地链路上传播，不能被扩散到链路以外的范围。增加Link-LSA有三个目的：

为连接到同一链路上的其他路由器提供本设备的本地链路地址；

向连接到同一链路上的其他路由器通告本链路的IPv6前缀列表；

声明本链路上生成的Network-LSA的Options字段集；

在IPv4中，router-LSA携带了路由器的IPv4接口地址，该地址与IPv6中的本地链路地址相当。只在OSPF计算路由的下一条时会用到这些地址信息，因此，Router-LSA中的这些地址信息不需要传播到本地链路以外的区域。IPv6中使用Link-LSA分发这些地址更为有效。而仅仅通过接收Hello报文，路由器是无法学到本地链路地址的：在NBMA网络上，路由器和下一条路由器不一定会交互Hello报文，他们是通过DR来感知对方的存在。

the Option field in the Network LSA is set to the logical OR of the Options that each router on the link advertises in its Link-LSA.路由器将它的Link-LSA中的本链路路由器的

Option字段值相与，得到自己的Options字段值。

3类Summary-LSA更名为’Inter-Area-Prefix-LSA’；

4类Summary-LSA更名为’Inter-Area-Router-LSA’；

Inter-Area-Prefix-LSA、Inter-Area-Router-LSA和AS-external-LSA中的Link State ID失去了原来的地址意义，只用来标识链路状态数据库中的一项。过去用Link State ID来标识的地址信息和Router ID被放在LSA体中。

唯有Network-LSA和Link-LSA的Link State ID带有附加意义：Link State ID也是本链路上发送路由器的Interface ID。因此，这两类LSA也是目前唯一对长度不作限定的LSA：Network-LSA必须包含本链路上的所有路由器；Link-LSA必须包含本链路上的所有路由器的本地链路地址。

添加了一类新的LSA：Intra-Area-Prefix-LSA。这类LSA携带了所有的IPv6地址前缀信息。

AS-external-LSA中有两个字段是可选的：forwarding address和external route tag。此外，为了包含额外的OSPF自治系统外路由属性AS-external-LSA可以引用其他的LSA；

处理未知LSA类型

OSPFv3能够处理未知类型的LSA。对于未知类型的LSA，可以只在本地链路范围发送，也可以把它当作一种已知的类型洪泛。具体的处理方式取决于link state header中的LS Type字段的定义。

在IPv6中，需要路由器在单个链路上提供对报文的多种处理。IPv4中对未知类型报文直接丢弃的做法已经不适用了。当DR比链路上其他路由器支持的Option类型少时，直接丢弃报文会产生问题。

Stub区域

在IPv4中设置Stub区域减小了区域内部路由器的链路数据库和路由表规模。这使得在大规模OSPF路由域内的路由器只占用了很少的资源。

OSPFv3保留了Stub区域的概念，类似的，Stub区域内只有router-LSA、network-LSA、Inter-Area-Prefix-LSA、Link-LSA和Intra-Area-Prefix-LSA。

与上一版本不同的是，OSPFv3允许不能识别类型的LSA被当作已知类型洪泛。无节制的加入未知类型LSA可能导致Stub区域的链路状态数据库超过路由器的处理能力。

为防止这种现象的发生，必须同时满足下列两个条件的LSA才可以在Stub区域内洪泛：

LSA的洪泛范围为区域（area）或本地链路范围；

LSA的’U-bit’置为0。

识别邻居

在OSPFv2中，对p2p，Virtual link上的邻居用他们的Router ID来识别；对broadcast，NBMA，p2mp上的邻居用他们的IPv4接口地址来识别。在OSPFv3中，所有链路上的邻居路由器是由他们的Router ID来识别的。这样的变化使得OSPF报文的接收、邻居发现有了相应的变化。

0.0.0.0的Router ID是保留未用的。Router ID决不能设为0.0.0.0。

实现细节

从IPv4到IPv6，基本的OSPF机制保持不变。详细内容请参加RFC2328 Section4。归纳起来，两者的共性主要体现在以下几个方面：

都维护链路状态数据库；

构成数据库的基本单位LSA需要在邻居间同步；

首次同步是通过交换DD报文，LSR报文和LSAck报文来完成的；此后链路状态数据库的维护是通过洪泛，发送LSU报文和LSAck报文来实现的；

使用Hello报文来发现和维护邻居关系；

只在broadcast和NBMA链路上选举DR和BDR；

邻接关系的建立；

区域内路由和用ASE标识引入的区域外部路由的思想；

各种路由的计算。

特别的，OSPFv3完全继承了RFC2328中描述的OSPFv2的下列机制：

2328 Section4.3描述的OSPF报文，包括Hello报文，DD报文，LSR报文，LSU报文，LSAck报文。尽管有些报文的格式有一些变动，但这些报文的用法是完全不变的；

对运行OSPF的系统要求不变，只是OSPFv3要求系统必须运行在IPv6的协议栈下；

邻居关系的发现和维持机制，邻接关系的建立，包括RFC2328的Section 7，7.1-7.5；

OSPF支持的链路类型（或者叫接口类型）不变，有：p2p，broadcast，NBMA，p2mp和虚连接；

接口状态机，包括接口状态和事件列表，DR、BDR选举机制保持不变，可参见RFC2328 Section 9.1-9.4；

邻居状态机，包括OSPF邻居状态和事件，可参见RFC2328 Section 10.1-10.4；

链路状态数据库老化机制、LSA的域内传播和提前老化处理，可参见RFC2328 Section 14和14.1；

在保留大部分OSPFv2机制的同时，部分机制有了变化。下面描述OPSFv3的具体实现方法。

协议数据结构

主要的数据结构是不变的，分为：区域数据库，接口数据库，邻居数据库，链路状态数据库和路由表。Top-level数据结构与RFC2328 Section 5中列出的一致。对数据库主要做了下列修改：

所有未知的LS类型和AS洪泛范围出现在顶层数据库中，这些内容过去只限于特定的区域或链路。AS-external-LSA是唯一具有AS洪泛范围的LSA。未知LS类型’U-bit’设为1。

区域数据结构

OSPFv3区域数据结构包括了RFC2328 Section 6中定义的所有元素。此外，区域数据结构中包含了所有已知类型的LSA洪泛范围，包括：router-LSA、network-LSA、inter-area-prefix-LSA、inter-area-router-LSA和intra-area-prefix-LSA。区域数据结构中也包含未知LS类型的洪泛范围：area。实现MOSPF的路由器会在区域数据结构中包含group-menbership-LSA。7类LSA放在NSSA区域的数据结构中。

接口数据结构

在OSPFv3中，路由器通过接口连接到链路。

OSPFv3的接口数据结构相对于OSPFv2做了以下修改：

Interface ID

路由器的每个接口都分配了接口ID，用来唯一的标识这些接口。例如有些应有会用MIB-II IfIndex作为接口ID。接口ID出现在：该接口发送的Hello报文中；路由器向连接该接口的链路发送的link-local-LSA中；所属区域的Router-LSA中。链路上的DR发送的network-LSA中把本设备该链路上的接口ID作为link State ID。

Instance ID

每个接口都分配了Instance ID，缺省为0。只有当存在多个彼此独立的OSPF团体（community of OSPF Router）时，才有必要分配各自不同Instance ID。

例如，有两个完全独立的OSPF团体。给团体1分配Instance ID 0，即给所有的OSPF团体1接口分配Instance ID 0；对团体2分配Instance ID 1，即给其余的接口分配Instance ID 1。这样OSPF的发送和接收过程就能保证两个团体的独立。

本地链路范围的LSA列表

该列表包括了所有洪泛范围为本地链路范围的LSA，也包括这样一些LSA，他们是由连接到该链路的接口的接口数据结构中包含的链路产生或传播的。该列表包含了本链路的Link-LSA。

未知LS类型的LSA列表

该数据结构包括所有未知LS类型和’U-bit’设为0（对未知LS类型，认为该LSA传播范围为本地链路范围）。

接口IP地址

OSPFv3中几乎所有报文的源地址都是发送接口的本地链路地址，只有虚连接除外：在虚连接上发送的报文必须使用路由器的本地站点地址或IPv6全球地址作为接口IP地址。

链路前缀列表

路由器在Link-LSA中会通告本链路上的IPv6前缀列表。这样链路上的DR就可以在intra-area-prefix-LSA中通告这些地址。

OSPFv3中，每个路由器接口有唯一的metric，该metric代表了该接口发送报文的花费。OSPFv3的认证依赖于IPv6报文的认证头和IP封装安全有效载荷报头。OSPFv3通过这些IP报文头来确保路由交换的完整性和认证/保密。因此，关键字AuType和Authentication与OSPF接口无关。

接口状态，事件和接口状态机保持不变，，可参见RFC2328 Section 9.1-9.3。DR和BDR选举机制也保持不变，可参见RFC2328 Section 9.4。

邻居数据结构

OSPFv3中邻居数据结构的作用不变：收集所有建立邻接关系需要的信息。每个数据结构绑定到单个的OSPF接口上。IPv6中定义的邻居数据结构与RFC2328中的定义有以下差异：

邻居接口ID
邻居数据结构必须记录邻居发出的Hello报文中的接口ID。在两种情况下，路由器发布的Router-LSA会包含邻居的接口ID：
发送到p2p邻居；
向DR通过与之相连的链路；

邻居IP地址
在OSPFv3中，邻居IP地址是对方的本地链路地址。虚链路除外。

DR
邻居选举出的DR用Router ID来标识，而不象过去用IP地址标识。

BDR
邻居选举出的BDR用Router ID来标识，而不象过去用IP地址标识。

邻居状态、事件和邻居状态机保持不变，可参加RFC2328 10.1、10.2、10.3。判断是否建立邻接关系不变，可参加RFC2328 10.4。

报文处理

OSPFv3直接运行在IPv6网络层之上，因此，OSPFv3报文被封装在一个或多个IPv6报文头之后，而其紧邻的前一个报文头Next Header域值为89。
与OSPFv2一样，除了保持邻居关系的Hello报文外，其他的OSPF协议报文只发给建立邻接关系的邻居。OSPFv3报文类型和功能与OSPFv2报文一致，在标准OSPF报文头Type域中标明。

发送协议报文

IPv6路由器发送OSPF协议报文时，如下填充标准IPv6 OSPF报文头的各个域：

版本
设为3。

类型
OSPF报文类型，如Hello，LSU等。

包长度
包括标准OSPF报文头在内的整个OSPF报文的字节数。

Router ID
标识发送报文的路由器本身。
Area ID
接收OSPF报文的区域号。

Instance ID
绑定发送OSPF报文的接口的OSPF实例号。

校验和
标准IPv6 16位报文完整校验和，覆盖了整个OSPF报文和IPv6伪头。

对OPSFv3协议报文的IPv6源地址和目的地址选择与OSPFv2一致，可参见RFC2328 Section 8.1。目的地址从建立邻接路由器的AllSPFRouters，AllDRouters和邻居IP地址中选取。在IPv6中，除了虚连接外，都是选择对方的本地链路地址。
发送LSR报文和LSAck报文不变，可参见RFC2328 Section 10.9，13.5。发送Hello报文详见本文3.2.1.1，DD报文发送详见本文3.2.1.2，发送LSU报文详见3.5.2。
3.2.1.1 发送Hello报文
OSPFv3中，发送Hello报文有如下改变：
Hello报文必须包含发送接口的接口ID；
由于OSPFv3运行基于链路而不是网段，因此Hello报文中不再携带子网掩码；
被选举出的DR，BDR在Hello报文中用他们的Router ID标识，而不是接口地址。报文中将DR/BDR置为0.0.0.0表示DR/BDR尚未选出。
扩展了Hello报文中的Option域，如下的一些位必须设置正确：
当且仅当发送接口连接到non-stub区，才设置’E-bit’；
当且仅当发送接口连接到NSSA区，才设置’N-bit’；
当且仅当接口抑制Hello报文发送，才设置’DC-bit’；
Hello报文中Option域不能识别的位应清空。

NBMA链路上的Hello报文发送与OSPFv2一致，可参见RFC2328 Section9.5.1。
3.2.1.2 发送DD报文
OSPFv3中，DD报文的发送与OSPFv2有如下差异：
扩展了Hello报文中的Option域，如下的一些位必须设置正确：
当且仅当路由器转发按照MOSPF转发多播报文时，才设置’MC-bit’；
当且仅当接口抑制Hello报文发送，才设置’DC-bit’；
DD报文中Option域不能识别的位应清空。

接收协议报文

当路由器收到OSPF协议报文时，他用接收接口对报文进行标记。对于配置虚连接的接口，可能无法立即找到标识接口。在下面的描述中，我们都假设报文不是来自虚连接链路。
处理OSPFv3报文前，必须对封装OSPFv3报文的IPv6报文头做如下检验：
报文的目的IP地址必须是接收接口的单播地址，包括本地链路地址，或者是AllSPFRouters、AllDRouters的多播IP地址；
紧邻的IPv6头必须在Next Header域指明OSPF协议（89）；
必须验证所有的IP验证头和IP封装安全载荷以保证OSPF路由交换的安全认证以及机密性；
本地产生的OSPF报文不送往OSPF：必须检查IPv6源地址，以确保没有路由器自己产生的多播报文。

在处理完IPv6报文头后，紧接着处理OSPF报文头。OSPFv3报文头的字段必须服从如下规则，否则报文将被丢弃：
类型域必须为3；
标准的IPv6 16位完整校验和必须通过校验，该校验和覆盖了整个OSPF报文和IPv6伪头；
区域ID必须通过校验：区域ID可以是如下的任一种情况，如果以下情况都不满足，应丢弃该OSPF报文：
是接收接口的区域ID。与IPv4不同，IPv6中源地址不必与接收接口地址处于统一网段；
为骨干区。这时，报文在虚连接上传送，接收路由器是ABR，报文中的源路由器Router ID是虚连接另一端的路由器Router ID。接收接口也配置在该虚连接中。通过以上检查后，路由器接收该报文并把该报文绑定到虚连接上。
实例ID必须与接收接口实例ID相同；
仅有DR或BDR接收目的IP地址为AllDRouters。
报文头处理完后，报文按照不同的OSPF报文类型来处理。
对Hello报文，送给Hello协议处理。其他报文则只在建立邻接的双方收发。这也意味着发送报文的路由器是active的邻居。
除了Hello报文外，其他报文的接收机制不变。

接收Hello报文
Hello报文的接收有如下变化：
在所有类型的链路上，邻居由他们的OSPF Router ID来唯一的标识。除了虚连接外，路由器用接收报文的源地址作为邻居的IP地址；
不含地址掩码；
选出的DR，BDR用他们的Router ID来标识。

路由表结构

OSPFv3的路由表项与OSPFv2相似：包含IPv6地址前缀表项，ASBR表项。后者只是建立路由表过程中的一个中间结果。
为了保持SPF运算中每个区域的中间结果，每个区域有各自的路由表，路由表包含的表项如下：
区域中的每个路由器：路由器由他们的Router ID来标识；这些路由表项包含到路由器的最短路径集，路由器用Intra-area-prefix-LSA发布自己产生的IP地址前缀时，会用到这些信息；对于ABR，这些信息也用于计算到区域内部地址前缀的路径。
区域中每个transit连接：Network-LSA用到了Transit连接。Transit连接和Network-LSA都是用链路上DR的接口ID和DR的Router ID的组合来标识的。这些路由表项使得intra-area-prefix-LSA可以得到Transit Link的IP地址前缀。

一些OSPF路由表的域保持不变：capabilities（Router only），path type，cost，type 2 cost，link state origin，到同一目的地的等价路由，下一跳，发布路由器。

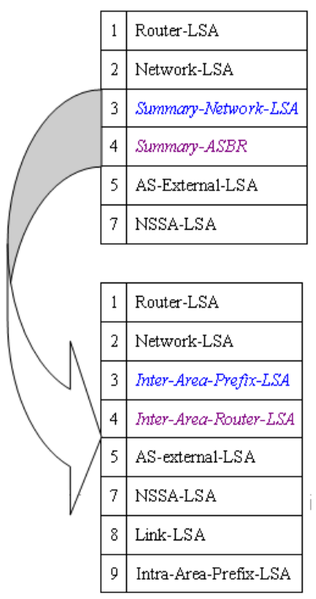
在OSPFv3中，路由表每项中的link-state origin域是直接或间接产生该路由项的router-LSA或network-LSA。

路由表查找

路由表查找（即决定指导IP转发的最佳路由项）保持不变。

链路状态广播

OSPFv3中，LSA头有少许的改变：LS类型域扩展了；Option字段放到了LSA中；一些LSA的格式变化了（router-LSA，network-LSA，ASE）；一些LSA更名了（3类和4类路由分别更名为inter-area-prefix-LSA和inter-area-router-LSA）；增加了两类LSA（Link-LSA和Intra-Area-prefix-LSA），如下图所示；LSA中去掉了Type of Service（TOS）字段。



下面将详细介绍这些变化。

LSA报文头

在IPv4和IPv6中，OSPF LSA头都是20字节，但是20字节中的内容是不同的。其中，Advertising Router，LS Sequence Number，LS Checksum和length域不变，可参见RFC2328 Section12.1.1，12.1.5，12.1.6，12.1.7和A4.1。变化的域有：

Options：

LSA头中去掉了Option字段，在router-LSA，network-LSA，Inter-area-router-LSA和Link-LSA中加入了Option。Option字段从原来的8位扩展到24位，一些位的定义有了变化。另外，LSA中的每个前缀跟有一个单独的8位的PrefixOption域。

LS type：

该域从8位扩展到16位。前两位表示洪泛的范围，接下来的位表示对未知的LSA类型的处理。

Link State ID：

该字段长度保持32位不变，但是除了network-LSA和link-LSA外，其他LSA中的该值已经没有了地址意义。比如，一个支持IPv6的路由器在生成多个ASE时，可以给他们依次分配Link State ID：第一个分配0.0.0.1；第二个分配0.0.0.2等等。这时，Link State ID已不具备地址意义，只是为了区分单个路由器产生的多个LSA的一个标识。

对于network-LSA，Link State ID是DR在链路上的接口地址。当路由器产生link-LSA时，该LSA的Link State ID是该路由器在这个链路上的接口地址。

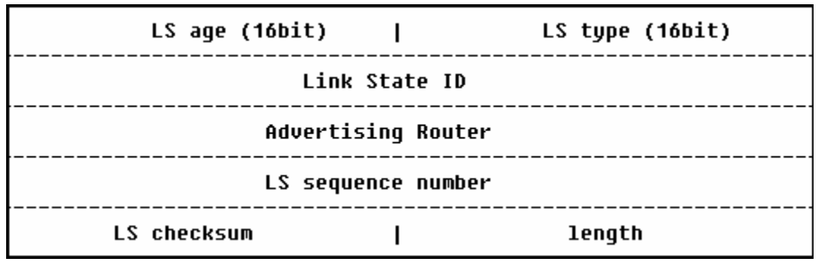


图9 OSPFv3 LSA Header

LS Type：有9种类型，其中6和7两种不属于基本类型

0x2001	Router-LSA
0x2002	Network-LSA
0x2003	Inter-Area-Prefix-LSA
0x2004	Inter-Area-Router-LSA
0x4005	AS-External-LSA
0x2006	Group-membership-LSA
0x2007	Type-7-LSA
0x0008	Link-LSA
0x2009	Intra-Area-Prefix-LSA

链路状态数据库

与OSPFv2一样，OSPFv3中，一条LSA是由它的LS类型，Link State ID和发布路由器三个字段的组合唯一确定的。对同一条LSA的两个实例而言，通过LSA的LS Sequence Number来判定哪一条是最近发布的。当出现一样的情况，使用LS checksum和LS age来做判定。可参见RFC2328 Section 13.8。

OSPFv3中，Link-State database被分割为3个独立的数据结构：

AS洪泛范围的LSA放在顶层数据（top-level date structure）结构中，包括已知的ASE和’U-bit’置1的LSA；

区域洪泛范围的LSA放在appropriate area structure中，包括router-LSA，network-LSA，inter-area-prefix-LSA，inter-area-router-LSA和intra-area-prefix-LSA以及’U-bit’置1的LSA；

本地链路洪泛范围的LSA，包括link-LSA和’U-bit’置0的未知类型LSA。

生成LSA

重新生成LSA的过程不变：LSA的LS sequence number增加，LS age置0，重新计算checksum后加到链路状态数据库中，从特定的接口发布。

除了RFC2328 Section12.4中描述的触发重新生成LSA的事件外，OSPFv3增加了如下的触发事件/动作：

? 路由器的某个接口状态改变：这是路由器可能需要（重新）生成或发布它的Link-LSA和一条或多条router-LSA和/或intra-area-prefix-LSA；

链路的DR identity改变：路由器可能需要（重新）生成或发布该链路的network-LSA和一个或多个router-LSA和/或intra-area-prefix-LSA；

邻居状态迁移到’full’或从’full’迁移。路由器可能需要（重新）生成或发布该链路的network-LSA和一个或多个router-LSA和/或intra-area-prefix-LSA；

邻居的接口ID改变：路由器可能在相关区域内重新发布router-LSA，一条或条intra-area-prefix-LSA；

链路上增加了一个新的前缀或删除了一个前缀：这使得路由器重新发布该链路上的link-LSA，如果这台路由器是链路上唯一的一台设备，它会重新生成intra-area-prefix-LSA；

由于收到新的link-LSA而使得链路上的前缀集合改变：对于链路上的DR会生成一条新的intra-area-prefix-LSA。

下面的章节详细叙述了7类LSA的结构。

Router-LSA

----- 每个路由器生成，描述本路由器的链路状态和花费，只在路由器所处区域内传播

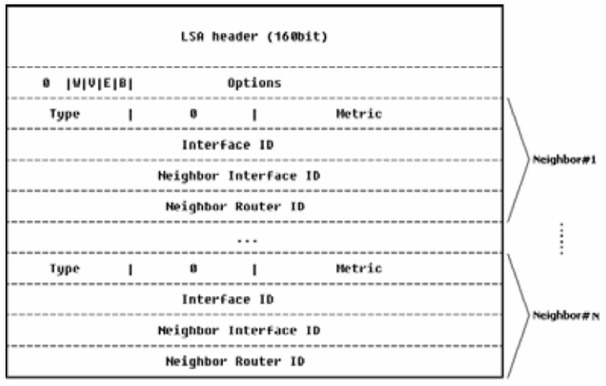


图10 Router-LSA格式

Type：1--- PPP连接

2---Transit网络

3---保留

4---VLink

Options：与Hello报文中的Options相同

Network-LSA

----- 广播网络和NBMA网络的DR生成，描述本网段链路状态，只在DR所处区域传播，列出了DR所连接的所有Router ID。 Link-LSA和Network-LSA的 LS ID 与 Interface ID 相同。

```
                    LSA header
-------------------------------------------------------------
     0              |              Options (21bit)
-------------------------------------------------------------
                  Attached Router #1
-------------------------------------------------------------
                          ...
-------------------------------------------------------------
                  Attached Router #n
```

图11 Network-LSA格式

Attached Router：DR所连接的Router ID
Options：与Hello报文中的Options相同

Inter-Area-Prefix-LSA

----- 和OSPFv2中的Type-3 LSA(net sum_LSA)对应，该LSA由区域边界路由器ABR生成，在与该LSA相关的区域内传播。每一条Inter-Area-Prefix-LSA描述了一条到达本自治系统的、其他区域的IPv6地址前缀（IPv6 Address Prefix）路由。

```
                    LSA Header
-------------------------------------------------------------
     0              |              Metric
-------------------------------------------------------------
PrefixLength  | PrefixOptions |             (0)
-------------------------------------------------------------
                  Address Prefix
                          ...
```

图12 Inter-Area-Prefix-LSA格式

Metric：这条路径的花费
PrefixLength：前缀长度
PrefixOptions：

```
0  1  2  3  4  5  6  7
|  |  |  | P|MC|LA|NU
```

NU：No Unicast，如果为1，则不参与单播路由计算
LA：为1，表示这个前缀为产生LSA的路由器的接口地址
MC：是否多播
P：NSSA区域前缀被ABR重新公布时设为1

Inter-Area-Router-LSA

----- 和OSPFv2中的Type-4 LSA(asbr sum_LSA)对应，该LSA由区域边界路由器ABR生成，在与该LSA相关的区域内传播。每一条Inter-Area-Router-LSA描述了一条到达本自治系统的、其他区域的去往边界路由器ASBR的路由。

```
                    LSA Header
-------------------------------------------------------------
     0              |          Options  (21bit)
-------------------------------------------------------------
     0              |          Metric
-------------------------------------------------------------
                Destination Router ID
```

图13 Inter-Area-Router-LSA格式

Options：与Hello报文中的Options相同
Metric：到其他区域的路由器的花费
Destination Router ID:目标路由器的Router ID

AS-external-LSA

----- 由自治系统边界路由器ASBR生成，描述到达其它AS的路由，传播到整个AS（Stub区域除外）。AS的缺省路由也可以用AS-external-LSAs来描述。

```
                    LSA Header
-------------------------------------------------------------
     |E|F|T|              Metric
-------------------------------------------------------------
PrefixLength  | PrefixOptions |     Referenced LS Type
-------------------------------------------------------------
                  Address Prefix
                          ...
-------------------------------------------------------------
          Forwarding Address (Optional) (128bit)
-------------------------------------------------------------
          External Route Tag (Optional) (32bit)
-------------------------------------------------------------
     Referenced Link State ID (Optional) (32bit)
```

图14 AS-External-LSA格式

E：为1时，表明为Metric为第2类外部路由的花费
F：为1时，LSA里带转发地址(Forwarding Address)
Forwarding Address：转发地址，数据流将经过这个地址
T：为1时，LSA里带外部路由标记(External Route Tag)
External Route Tag：外部路由标记，用于ASBR间交流
Referenced LS type：非0时，LSA带Referenced Link State ID
Referenced Link State ID：所引用的LS ID，用于ASBR间交流

Link-LSA

----- 路由器为每一条链路生成一个Link-LSA，在本地链路范围内传播。每一个Link-LSA描述了该链路上所联系的IPv6地址前缀，包含Link-local地址。

1）向链路上其他路由器发布自己的Link-local地址
2）发布这条链路上的IPv6前缀
（注：Link-LSA的LS ID与发布此LSA的路由器的Interface ID相同， Network-LSA也是如此）

```
                    LSA Header     (128bit)
-------------------------------------------------------------
  Rtr Pri    |              Options
-------------------------------------------------------------
        Link-local Interface Address  (128bit)
-------------------------------------------------------------
                    # prefixes
-------------------------------------------------------------
PrefixLength | PrefixOptions |             (0)
-------------------------------------------------------------
                  Address Prefix
                          ...
-------------------------------------------------------------
                          ...
-------------------------------------------------------------
PrefixLength | PrefixOptions |             (0)
-------------------------------------------------------------
                  Address Prefix
                          ...
```

图15 Link-LSA格式

各个域的意义同上

Intra-Area-Prefix-LSA

----- 每个Intra-Area-Prefix-LSA包含路由器上的IPv6前缀信息，stub网段信息或传输网段（Transit network）的网段信息，该LSA在区域内传播，发布区域内的前缀。由于Router-LSA和Network-LSA不再包含地址信息，导致了Intra-Area-Prefix-LSAs的引入。

```
                    LSA Header     (128bit)
-------------------------------------------------------------
     # prefixes          |     Referenced LS type
-------------------------------------------------------------
              Referenced Link State ID
-------------------------------------------------------------
           Referenced Advertising Router
-------------------------------------------------------------
PrefixLength | PrefixOptions |          Metric
-------------------------------------------------------------
                  Address Prefix
                          ...
-------------------------------------------------------------
                          ...
-------------------------------------------------------------
PrefixLength | PrefixOptions |          Metric
-------------------------------------------------------------
                  Address Prefix
                          ...
```

图16 Intra-Area-LSA格式

prefixes：本条LSA包含的前缀数目
Referenced LS type，Referenced Link State ID，Referenced Advertising Router：这3个参数表明引用的是哪条router-LSA 或network-LSA，以及产生这条LSA的路由器。报文中的地址前缀信息即为相应路由器或网段/链路的地址前缀。
其余各个域的意义同上。

洪泛

大多数的OSPFv2中的洪泛机制保持不变，可参见RFC2328 Section13。完全继承的部分有：

判定更新的LSA实例，参见RFC2328 Section13.1；
对自己产生的LSA的处理，RFC2328 Section13.4；
发送LSAck报文，RFC2328 Section13.5；
LSA的重传，RFC2328 Section13.6；
LSAck的接收，RFC2328 Section13.7。

洪泛范围的修改和对未知LS类型的报文处理使得洪泛机制有了变化：LSU的收发必须考虑LSA的洪泛范围和’U-bit’设置；另外，由于IPv6中对LSA的类型和内容进行重组，在数据库中加入一条LSA也会触发不同的事件，下面对这些内容进行详细描述。

接收LSU报文

LSU的接收与以前不同，这些变化是因为在LS type中包含了洪泛范围，以及处理未知LS类型报文的需要。在IPv4中，要检验LSU中的每一条LSA，检验的8个步骤详见RFC2328 Section 13。在OSPFv3中，步骤2和3有如下修改：

（2）检验LS type。丢弃下列LSA，转而处理LSU中的下一条：

未知LS类型；

在Stub区中收到洪泛范围为AS flooding scope或’U-bit’置1的LSA；

（3）LSA的洪泛范围置为’reserved’。丢弃该LSA，转而处理LSU中的下一条。

步骤5b和5d也做了修改，见3.5.2和3.5.3。

发送LSU报文

发送LSU的步骤不变，可参见RFC2328 Section 13.3。在OSPFv3中，洪泛LSA的合法接口不同了，接口的选择基于以下因素：

LSA的洪泛范围；

对于洪泛范围为区域或本地链路的LSA，LSA与之关联的区域或接口；

是否能够识别该LSA的LS类型；

‘U-bit’设置：

为0：未识别LS类型的LSA洪泛范围为本地链路范围；

为1：未识别LS类型的LSA被当作已知LS类型的LSA接收和洪泛；

因此，合法接口的选择有以下三种情况：

情况1：

LSA的LS类型已知

此时，合法接口集取决于LS type中的洪泛范围。对洪泛范围为’AS-flooding-scope’的LSA，合法接口集合包括除虚连接外的所有路由器接口，此外，ASE不从连接到Stub区域的接口发送。

对洪泛范围为’area flooding scope’的LSA，合法接口集合包括连接到该区域的所有接口。对洪泛范围为’link-local flooding scope’的LSA，合法接口只包含一个接口，就是路由器连接到这条链路上的接口，该接口也是收到这条LSA的接口。

情况2：

未知LS类型且’U-bit’置0

认为该LSA洪泛范围为本地链路范围。合法接口只包含一个接口，就是路由器连接到这条链路上的接口。

情况3：

未知LS类型且’U-bit’置1

这时，合法接口的选择与情况1类似，要排除连接到Stub区域的接口。

有时候，在将一条LSA加入指定邻居的链路状态重传列表时，还需要其他条件。如果路由器能识别LS Type而他的邻居不能（可以从邻居发出的DD报文的Option字段判定），并且此时LSA的’U-bit’置为0，当且仅当邻居是DR或BDR时，将该LSA加入邻居链路数据库重传队列。除了group-membership-LSA外，我们假定前面提到的9类LSA都能被所有的路由器识别。对于group-membership-LSA，只有MOSPF路由器才能识别，它的’U-bit’是置0的。仅当一个non-MOSPF邻居路由器是DR或BDR时，该邻居才转发这类LSA。

在数据库中添加LSA

由于数据库被分为独立的三个部分，添加LSA也相应的分为三种情况：

已知LS type或’U-bit’置1的自治系统洪泛范围的LSA被加入global OSPF数据结构中；

已知LS type或’U-bit’置1的区域洪泛范围的LSA被加入appropriate area数据结构中；

未知LS type且’U-bit’置0的LSA和本地链路洪泛范围的LSA被加入appropriate interface数据结构中。

在向链路状态数据库中加入LSA时，必须检查LSA的内容是否改变。当LSA的内容改变时，根据LSA类型的不同，路由表的如下部分必须重新进行计算：

Router-LSA，Network-LSA，Intra-Area-Prefix-LSA和Link-LSA：整个路由表需要重新计算；

Inter-Area-Prefix-LSA和Inter-Area-Router-LSA：通过这些LSA得到的到达目的的最优路由必须重新计算。当目的地是ASBR时，所有的ASE可能也需要重新计算；

AS-external-LSA：通过这些LSA得到的到达目的地的最优路由必须重新计算。

当得到一条LSA的新的实例时，该LSA的旧的实例必须从数据库中清除，同时所有邻居的重传请求队列中也应该去掉该LSA。

自己产生的LSA

OSPFv3中，对自己产生的LSA的描述简化了：自己产生的LSA就是那些Advertising Router与自己的Router ID一致的LSA。

虚连接

除了以下情况外，OSPFv3对虚连接的定义不变：

具有自治系统传播范围的LSA决不向虚连接邻居传播，也不在数据库交换的过程中将这样的LSA聚合起来传给虚连接邻居。

接口范围必须是站点本地或全球范围地址，而不是被其他接口类型使用的本地链路地址。站点本地或全球范围地址才能用于虚连接的协议报文源地址；

为了在路由计算时得到虚连接邻居的IPv6地址，邻居在Intra-Area-Prefix-LSA中向虚连接传播区域发布他的虚连接IPv6接口地址；

虚连接被分配唯一的接口ID。这个ID用于虚连接上Hello报文的发布和router-LSA的发布。

路由表计算

路由表计算过程中，OSPFv3和OSPFv2的区别有：

Router-LSA不发布前缀信息，改由intra-area-prefix-LSA发布。因此，在Stub区将对intra-area-prefix-LSA发布的前缀做检查；

3类和4类LSA更名为inter-area-prefix-LSA和inter-area-router-LSA；

地址信息不在link state ID中，在LSA体中；

路由器在当区域中可以生成多条router-LSA，多条router-LSA之间用link state ID区分。在SPF运算中，这些router-LSA必须被当作一个整体；

路由表变化触发的事件以及等价路由保持不变。

计算区域内路由

OSPFv3中对inter-area路由的计算有如下变化：

3类和4类LSA分别更名为：inter-area-prefix-LSA和inter-area-router-LSA；

地址前缀信息放在inter-area-prefix-LSA体中，不再放在link sate ID中；路由器的OSPFRouter ID 放在inter-area-router-LSA体中；

Prefix Options中设置了’NU-bit’的前缀应被inter-area路由计算忽略；

当单个inter-area-prefix-LSA或inter-area-router-LSA改变时，运用RFC2328 Section16.5的机制，而不是重新计算整个路由表。

检查Tansit区Summary-LSA

OSPFv3对这部分的修改参加Section3.8.3中对inter-area路由的部分。

计算ASE

ASE的计算有如下修改：

ASE的link state ID不再是网段信息。地址前缀信息被放在ASE体中；

默认路由由ASE发布，其中的前缀长度为0；

不通过forwarding address是否置0.0.0.0判断是否使用了forwarding address，判断external-LSA的F位。当且仅当设置了F位时，使用了forwarding address；

区域内的路由计算应该忽略设置了Prefix Options域’NU-bit’的前缀；

当单个ASE改变时，可以不对整个路由表进行重新计算。

多接口共享链路

在OSPFv3中，路由器可能有多个接口连接到单个链路上，所有的接口都收发数据报文，但只有一个接口收发OSPF控制信息，具体实现如下：

每个接口都有各自不同的接口ID。这样一来，路由器可以自动检测到何时多个接口连接到同一个链路上：当收到的Hello报文来自它本身的Router ID，但是接口ID不是接收接口的ID时；

对单个链路OSPF关掉其他接口的OSPF控制报文收发功能，只留下该链路上的一个接口。对这个接口的选择取决于实现。例如，选择接口ID最大的接口。如果路由器是DR，那么这个接口的ID也是network-LSA的link state ID；

单个链路上的多个接口都将出现在

router-LSA中，每个接口也都将产生link-LSA。因此，单个链路上的多个接口都将参与到SPF运算中；

如果选出的收发控制报文的接口失败，另一个接口将接替他的工作，从新建立邻接关系。通过其他接口停止收到该接口发出的hello报文，路由器可以侦测到接口失败。

参考文献

【1】 Moy, J., “OSPF Version 2”, RFC 2328, April 1998
【2】 Moy, J., “OSPF Version 3”, RFC 2740, December 1999

OSPF MTR

鉴定测试中心　朱皓

关于MTR

简介

MTR是Multi Topology Routing的简称，是指多拓扑路由。多拓扑路由的意思就是指在一个物理拓扑下，划分成多个逻辑的拓扑，这些逻辑的拓扑可能是交叉或者重叠的。

MTR是属于智能路由的范畴，不同的数据流可以被划归不同的逻辑拓扑，这样可以为实施转发策略提供了一个有效的手段。MTR虽然类似QoS或者TE技术，但却与它们不同。MTR属于一种路由技术。无论是QoS或者TE或者路由技术，其实都是帮助完成数据的转发，不过实施的方法和关注的重点不同而已。

目前VRP不支持MTR。

MTR的实现

实现MTR技术，肯定需要对路由协议进行扩展。另一个方面需要对流进行分类，以达到寻找拓扑归属的目的。所以MTR必然是与流分类技术共同使用的。

目前OSPF的MTR技术标准主要是IETF的草案draft-ietf-ospf-mt-04.txt。

我们用图1可以很好的理解MTR技术。对于传统的路由技术，一个物理拓扑是依靠路由来建立逻辑结构的，同一目的的不同业务报文必然是通过相同的链路来进行转发。当然我们可以通过PBR来改变下一跳，或者是通过TE来进行流量的规划。但是MTR提供了另一个选择，与PBR相比，MTR的优势在于是基于拓扑而不是下一跳。与TE相比，MTR的部署要方便的多。

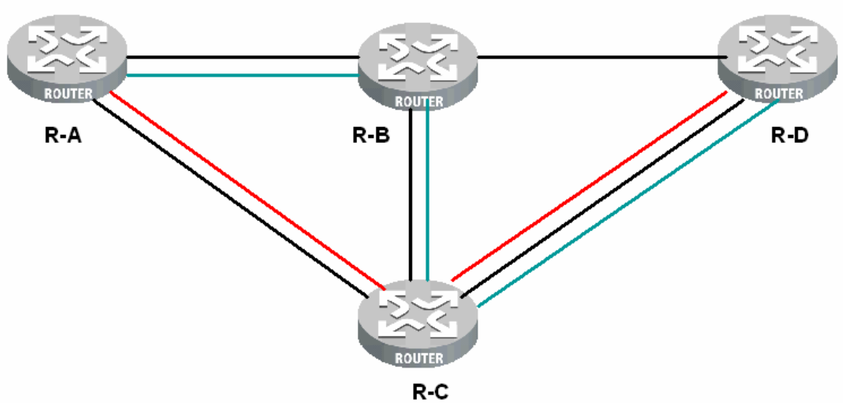


图1 多拓扑示意图

图1中黑色线条代表传统的拓扑结构，从R-A侧去往R-D侧的流量所经过的路径是通过路由协议选择的结果。而对于MTR而言，可以在这个物理拓扑上建立多个逻辑拓扑。黑色的完整拓扑叫做全局拓扑，除此以外的其它拓扑我们可以叫做有色拓扑。由红色线条代表的红色拓扑和由绿色线条代表的绿色拓扑。对于红色拓扑而言，R-B并不存在。而对于绿色拓扑而言，它认为R-A和R-C间，以及R-B和R-D间并没有可用的链路相连。每一个单独的拓扑都根据路由协议计算出自己的RIB和FIB。属于本拓扑的流量则根据本拓扑的RIB进行路由。这就同时要求有色拓扑是连续的，毕竟MTR实现的是一种端到端的路由技术。

为了增加MTR的灵活度和可靠性，允许回退现象，也就是说如果在有色拓扑中不存在目的路由，那么可以使用全局拓扑中的路由来进行转发。当然，也可以在某一台路由器上配置策略，使某些有色拓扑或者全部有色拓扑的流量直接使用全局拓扑的路由来进行转发。

MTR为以网络前缀为目的的路由结构增加了新的维度，流量首先是选择拓扑，然后才根据目的前缀来进行路由匹配。

MTR的流程

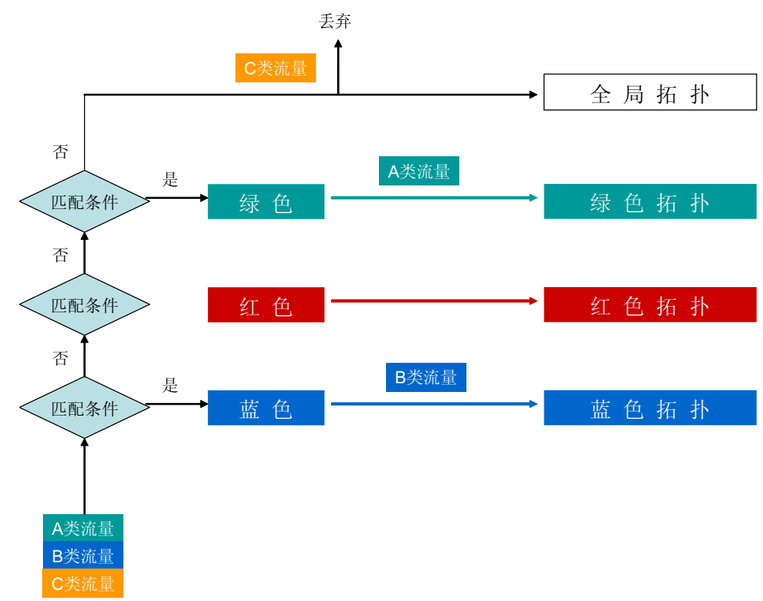


图2 MTR流程

到达的报文首先匹配流分类策略，如果匹配到某一个有色拓扑，就根据该拓扑转发。否则就走全局拓扑或者被丢弃。路由器本地的出接口不一定相同，它一定是属于全局拓扑以及某一个或者多个有色拓扑。

OSPF MTR

协议的扩展

RFC1583定义了TOS字段，但是一直没有被使用，在RFC2328中删除了关于TOS的部分。于是在IETF的MTR草案中使用了这个字段。TOS数量被更改为MT的数量，也就是MT-ID。相应的，TOS的TOS度量值，也成为了对应MT-ID的度量值。一条LSA可能属于不同的有色拓扑，每一个有色拓扑的归属都是通过对应的MT字段来标明的，在MT字段中包括MT-ID和MT-ID的度量值。

这样通过对LSA的扩展，在OSPF协议报文交互的过程中，不需要额外更多的LSA就可以交互链路的拓扑归属，在两个路由器之间的一个邻接关系交互了所有的拓扑归属信息。

MTR是通过有色拓扑来划分层次，并且基于有色拓扑来给链路赋予不同的开销，进而达到路由变化的目的，因此MTR并不会也没有必要改变DR、ABR和ASBR的位置。而且MTR的参数是通过LSA来传递的，也就是说MTR的信息实质上仍然是通过全局拓扑来进行传递的。

在LSA方面2类LSA是一个特例，它并不携带任何关于MTR的信息。因为2类LSA描述的是多点网络中DR的位置和该网络中的路由器，它本身并不携带任何度量信息。

对于OSPF而言，路由计算得到的度量值是根据1、3、4、5、7类LSA产生的，因此MT字段主要用在这些LSA上。下面是几种LSA中MT字段的情况。

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            LS age             |    Options    |      1        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link State ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Advertising Router                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     LS sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         LS checksum           |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|*|*|*|N|W|V|E|B|        0      |            # links            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Link ID                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link Data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      | # MT-ID       |            metric             |
            该链路属于多少个有色拓扑
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     MT-ID     |       0       |          MT-ID  metric        |
  第一个有色拓扑的 ID              在该有色拓扑中的度量值
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   ...  可能存在的其他有色拓扑的 MT 字段          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     MT-ID     |       0       |          MT-ID  metric        |
  最后一个有色拓扑的 ID            在该有色拓扑中的度量值
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Link ID                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link Data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            ...                                |
```

图3 1类LSA的MTR扩展

3、4类LSA有类似的结构

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            LS age             |    Options    |    3 or 4     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link State ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Advertising Router                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     LS sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         LS checksum           |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Network Mask                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       0       |                  metric                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     MT-ID     |          MT-ID  metric                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     MT-ID     |          MT-ID  metric                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

图4 3、4类LSA的MTR扩展

5、7类LSA的结构，7类与5类相似。

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            LS age             |    Options    |      5        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Link State ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Advertising Router                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     LS sequence number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         LS checksum           |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Network Mask                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|E|     0       |                  metric                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Forwarding address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     External Route Tag                        |
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|E|    MT-ID    |             MT-ID  metric                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Forwarding address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   External Route Tag                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ...                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|E|    MT-ID    |             MT-ID  metric                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Forwarding address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   External Route Tag                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

图5 5类LSA的MTR扩展

从几个图中，大家可以注意到，因为5、7类LSA使用了E-bit来标明外部路由的类型，因此MT-ID只能使用7个比特，有效范围为0-127。1被保留作为缺省多播拓扑使用，0被保留作为MTRoutingExclusionCapability，也就是说如果MT-ID为0，那么该LSA是被排除在缺省拓扑之外的。

在IETF的MTR草案中定义了更多灵活的选择，在全局拓扑上形成基本拓扑的概念，根据策略的需要在全局拓扑中保留某些链路不被MTR所使用，这是通过MTRoutingExclusionCapability来完成的，支持该能力的MTR路由器互相之间进行协商并建立基本拓扑。基本拓扑相当于是MT-ID为0的有色拓扑，它的度量值是从缺省度量值继承而来的。而全局拓扑中则包括所有支持与不支持MTR特性的路由器，它的度量值使用缺省度量值。

为了对MTRoutingExclusionCapability能力进行协商，对hello报文进行重新定义。将option字段中用于TOS的位作为MT-bit来使用。当MT-bit被置位时，标志MTRoutingExclusionCapability能力被使能。为了与非MTR路由器兼容，该能力默认应该是非使能的状态。只有当一个区域中所有的路由器都支持MTR特性时，才能使能该位。当一个路由器置位了MT-bit，如果收到MT-bit清除的hello报文，将会丢弃。

```
+---+---+---+---+---+---+---+---+
|DN |O  |DC |EA |NP |MC |E  |MT |
+---+---+---+---+---+---+---+---+
```

图6 MTRoutingExclusionCapability能力标志位

Cisco OSPF MTR

Cisco在NETWORKS2004中介绍了将要推出的MTR技术包括配置，但在目前的销售版本中还没有MTR特性。Cisco的计划是在2005年下半年在包括7系列使用的12.2以及17、26、37系列的T版本上支持MTR特性。

定义拓扑

在全局下定义拓扑，考虑到每个有色拓扑都需要有独立的RIB和FIB会占用资源，规定每个地址族最多支持32个有色拓扑。

```
address-family ipv4
topology RED
[no] shutdown
[no] forward-base
topology YELLOW
topology GREEN
```

从命令行可以看出，有色拓扑的可用性是通过shutdown & no shutdown来管理的。Forward-base应该是指当目的地址无路由可匹配时，是否允许使用基本拓扑来进行转发。

流分类

流分类使用了类映射的方式，可以根据应用业务类型、DSCP值等，对流进行分类。最后将类与对应的逻辑拓扑进行绑定。不能匹配条件约束的数据成为默认类，默认类使用全局拓扑转发。

```
class-map VIDEO_CLASS
  match {App type DSCP ...}

policy-map class-routing CLASS_ROUTE_POLICY
class VIDEO_CLASS
select-topology RED
class VOICE_CLASS
select-topology YELLOW
class DATA_CLASS
select-topology GREEN
```

应用MTR

最后是将MTR在本地或路由协议内或接口下进行应用。

```
全局下：
service-policy input CLASS_ROUTE_POLICY

路由视图：
router ospf 33
topology router {topology name } tid {mtid value}

接口视图下：
interface fa1/0
ip ospf cost 20
ip topology {topology name}
ip ospf cost 10
```

MTR与相关技术的比较

MTR的一些特点

MTR可以根据流特性采用不同的转发路径，不同路径关注的重点不同，比如高吞吐量或者小延时等等，这样就可以满足不同的应用流需求。

MTR技术成为另一种链路备份的手段。

在链路负载方面，可以实现基于流的多路不等价负载分担。

对于安全方面，可以将可疑的流量通过一个安全拓扑来处理，在这个安全拓扑中进行安全鉴别甚至直接丢入黑洞路由。

MTR技术可以分离组播流量，限制组播流量在某一个固定的有色拓扑中。

MTR不能用于VPN解决方案，因为MTR并不支持不同VPN的相同网络前缀。

MTR并不是一种隧道技术，它完全是一种3层的路由技术，并且根据动态路由协议进行动态的适应。

MTR与QoS

很显然，因为MTR是基于拓扑的，一旦某个流被映射到一个有色拓扑中，相当于获得一个到达目的的完整的路径。而QoS则只是对下一跳进行了区分。

MTR与PBR

MTR基于动态路由协议，可以说是动态的有色拓扑。而策略路由技术相对而言属于静态配置。MTR的流量转发最终还是依靠拓扑计算得出的路由，而PBR并不关心路由表的指向。

MTR与VPN多实例

VPN多实例也可以看做是多个拓扑结构，每个VPN都有自己独立的拓扑和独立的路由表。但是在VPN中，一个接口只能属于某一个VPN，而MTR中，一个接口可以被多个有色拓扑使用。VPN多实例实质上需要多个路由协议的进程，而MTR只要一个进程就够了。因此在协议方面的设备开销要小的多。当然，MTR并不是一种VPN技术，所以它不能解决VPN地址重叠的问题。

MTR与MPLS TE

首先，MTR是一种IP路由技术，与MPLS完全不同。MPLS TE是一种隧道技术，并且是基于MPLS技术的。MTR的配置非常简单并且可以更有效的适应拓扑的变化。但是受到每地址族最多32个有色拓扑的限制，在对流量分类要求更细致的网络中，还是需要使用MPLS TE技术。

任何一种技术都有它主要的关注点和使用范围，并且在复杂的综合网络中，多种技术总是综合使用的。MTR是一种新的路由技术，这个概念的提出是否会被广泛接受，最终还是要经过市场的检验。

我司OSPF链路状态数据库分析

国际市场部 石松

前言

一直以来，关于路由协议及OSPF的培训胶片在公司已经相当丰富了。一代一代的数通华为人前赴后继，留下了大量的关于路由协议的培训胶片。对于OSPF这样一个复杂的路由协议，我零零碎碎的接触了一年了，还是有很多概念不是很清楚，记得刚到北研的时候一领导说如果用半年的时间把OSPF搞的很精通已经很不容易了，当时还真是不屑。现在已经一年多过去了，还是有很多问题啊。（呵呵，不是偶笨啦，多方面原因啦）可OSPF绝对是个复杂的协议，其RFC文档就200多页，据说在所有的RFC中名列前茅。一个路由协议的配置应该没什么可说的，关键是有没有投入时间和精力对协议的内部机制、算法、思想进行深入研究。可目前的现状是，很多同事没有时间对协议的细节进行深入分析。我一直觉得需要一篇介绍OSPF路由计算过程的案例。于是乎，在对OSPF的LSDB有了一定的了解后就迫不及待的写下这篇自己一直需要的文档，希望读者也真的是需要。本文不在协议的细节上纠缠，会跳过很多繁琐的概念。

OSPF概述

IETF(Internet Engineering Task Force)于1988年提出的OSPF是一个基于链路状态的动态路由协议，协议的基本思路如下：在自治系统中每一台运行OSPF的路由器收集各自的接口/邻接信息称为链路状态，通过Flooding算法在整个系统广播自己的链路状态，使得在整个系统内部维护一个同步的链路状态数据库，根据这一数据库，路由器计算出以自己为根，其它网络节点为叶的一根最短的路径树，从而计算出自己到达系统内部各可定的最佳路由。OSPF是一类Interior Gateway Protocol(内部网关协议IGP)，它处理在一个自治系统中，路由器的网络的路由表信息。

OSPF目前在网络上的应用广泛，是很有生命力的一个路由协议。OSPF协议中引入了很多概念，正确理解这些概念是理解OSPF的前提。下面我们对OSPF路由协议术语做一下简单的介绍。

OSPF路由协议：

Open Shortest Path First 最短路径优先协议，实际上指OSPF的最短生成树算法。

路由器（router）：

第三层的IP包交换机，以前在IP文献中也叫做网关

自治系统(autonomy system)：

一群路由器通过相同的路由协议来交换路由信息，缩写为AS

区域(area)：

自治系统的划分单元,一个自治系统可以划分为多个区域.

区域 ID(area ID)：

自治系统内区域的32-bit标识.

内部网关协议(internal gateway protocol)：

属于一个自治系统的路由器上运行的路由协议，缩写为IGP，每一个自治系统有一个单独的IGP，不同的自治系统可能运行不同的IGP。OSPF是内部网关协议的一种.

路由器的ID(router ID)：

一个32位的标号对每个运行OSPF的路由器，在自治系统内是唯一的。具有相同Router ID的OSPF路由器往往使网络路由计算混乱，这在组建OSPF网络时是需要考虑和规避的。

网络(network)：

在这种意义下，是IP网络/子网/超网，可能是标记了多重复合IP地址的子网，我们把它们看成是分离的网络，点到点的物理网络是个特例，它们只是被当作简单的网，不管对它们怎样指定IP号。

网络掩码(network mask)：

一个32位的数指示IP网络的IP地址范围，以十六进制数显示，例如一个C类网的掩码是0xffff00，在文字上写成255.255.255.0。

点到点网络(point to point network)：

由一对路由器简单组成的网络，例如一个56Kb的串口线的连接

广播网：

网络支持许多(两个以上)的路由器。都有能力将地址信息发送到所有连接的路由器上(广播)邻居路由器被OSPF的hello协议动态地发现，OSPF使这种广播能力能得到更大的应用，如果它存在，它上的每一对路由器都假定能和对方直接相连，以太网是一个广播网的例子。

非广播网：

网络支持许多(两个以上)路由器，但没有广播能力，邻结点也是通过OSPF hello报文来维持，但是由于设有广播能力一些邻居需要靠配置来发现，在邻居间，OSPF协议报文也是互相传送的，X.25网是一个例子。

OSPF可以在两种非广播网上运行，一种是非广播多重访问，(NBMA)它类似于OSPF在广播网上的操作，第二种类型叫点到多点，可看为多个点到点的连接的集合。非广播网类型的判别依靠于对网络的操作模式。

接口(interface)：

一个路由器与它连接的一个网络的连接称为接口，一个接口有它的状态信息，可以通过底层协议或是路由协议本身来得到。每个接口有一个唯一一个IP地址和掩码(除非是不标号的点到点连接)，一个接口有时也指一个连接。

邻居路由器(neighbor)：

两台路由器有接口连向共同的网络，邻居关系通过OSPF hello协议被维持(通常是动态的)。

- **邻接(adjacency)：**

为交换路由而在邻居间建立的关系，不是每对邻居都建立邻接关系。Adjacency关系是为了交换LSA而建立的一种逻辑关系。

- **连路状态通告(Link State Advertise)：**

描述本地路由器或网络的数据单元对路由器来说，它描述了路由器的接口状态和邻接状态，第一个连接状态传送会发送到整个路由领域，所有的连接状态传送组成了协议的连接状态数据库，这在全局范围内使用,缩写为LSA，LSA是OSPF的一种报文类型。深入理解LSA是理解OSPF的关键。

- **连接状态数据库(Link State DataBase)：**

所有的连接状态传送（LSA）组成了连接状态数据库.LSDB是进行最短生成树算法的基石。

- **stub网络：**

只有一个接口与外部相连的网络,如一个PPP可视为一个stub网络

- **骨干区域(Backbone Area)：**

所有区域边界路由器和它们之间的路由组成骨干区域.OSPF保留区域0为骨干区。

- **自治系统外部路由(AS external route)：**

指由非OSPF协议得到的路由，如BGP（边界网关协议），RIP（Routing Information Protocol),系统的静态配置路由等,系统的静态路由是由配置得到的,其他协议的路由是通过引入操作得到的,外部路由的指定是由用户决定的.

- **路由(route)：**

指两节点之间的连通路径。

- **路由表（routing table）：**

到每个目的地有路由，这样的表叫路由表。

OSPF路由计算过程

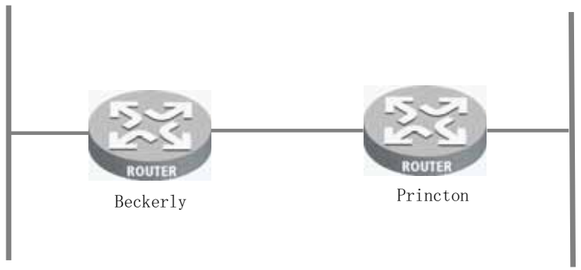


图1

本小节简略介绍OSPF从启动到路由计算完成的过程，更详细的描述见第四节。由于OSPF是链路状态路由协议，所以相比与RIP及EIGRP这种D-V算法的路由协议，其实现的复杂程度高出几倍。能够把握OSPF路由建立的大致过程，对于大多数用户已经足够了。

我们通过上图中的简单OSPF网络描述OSPF的路由计算过程，两个路由器一个叫Beckerly，一个叫Princton，简单起见，我们将Beckely和Princton设置在同一区域中。

1、在两路由器的的接口上运行ospf协议后，两路由器开始从ospf接口发送hello报文。

2、两路由器进行DR/BDR选举，选定DR的目的是为了给链路选一个代表，由DR负责这条链路信息的发布。

3、建立邻接关系，选出Master 和Slave。以便决定DD报文交换的主从关系。

4、根据DD报文信息，请求和更新lsa.同步LSDB

5、每个路由器自己根据自己的LSDB，生成SPF树。

6、根据最小生成树，计算OSPF路由。

7、由路由管理根据规则把OSPF路由下发到路由表。

其实既便是这样一个简单的不能再简单的OSPF网络，其报文交换角色协商的过程也是复杂的。我们不再此细化描述目的在于使其看起来简单些。

下面一节的描述将集中在LSDB上，所以LSDB形成前的DR选举、adjacency的建立、flood过程的阐述过程都不是很详细。（详细内容参见《ROUTING TCP/IP（volume 1）》）

路由计算例析

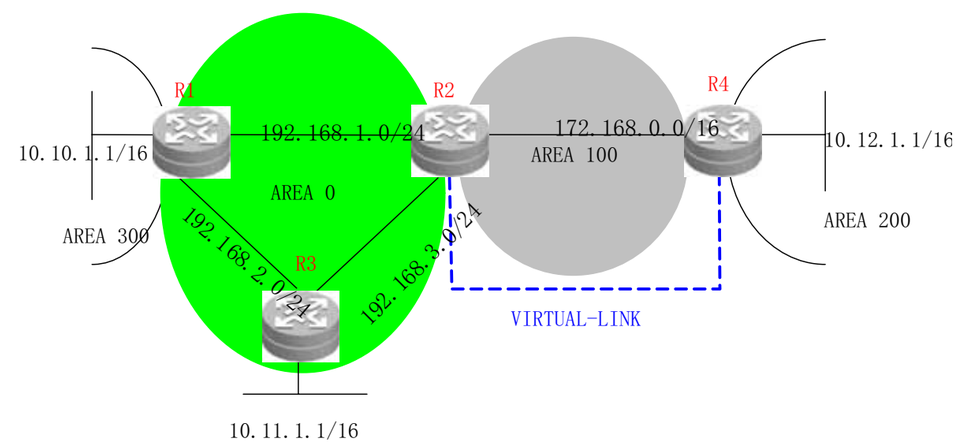


图2

我们通过这样一个实际的组网，通过实际的数据说明ospf的路由计算过程，在本文的前言部分已经阐述了本文的目的，就是让读者了解ospf路由协议产生背后的一些故事。

组网说明

R1:

Router id：1.1.1.1

Area0：接口192.168.1.1　　接R2接口：192.168.1.2

　　接口192.168.2.1　　接R3接口：192.168.2.3

Area300：10.10.1.1

R2:

Router id：2.2.2.2

Area0：接口192.168.1.2　　接R1接口：192.168.1.1

　　接口192.168.3.2　　接R3接口：192.168.3.3

Area100：接口172.168.1.1　　接R4接口：172.168.2.1

与R4（4.4.4.4）建立虚连接

R3:

Router id: 3.3.3.3

Area0: 接口192.168.2.3 接R1 192.168.2.1

接口192.168.3.3 接R2 192.168.3.2

Area400: 10.11.1.1

R4:

Router id: 4.4.4.4

Area100: 接口172.168.2.1 接R2 172.168.1.1

与R3(3.3.3.3)建立虚连接

Area200: 10.12.1.1

各路由器在网络中的属性:

列出各路由器上接口的信息（这里只列出了R1和R2）:

R1:

```
Area: 0.0.0.0
IP Address      Type    State       Cost  Pri    DR          BDR
192.168.1.1     Bcast   BackupDR    10    1      192.168.1.2 192.168.1.1
192.168.2.1     Bcast   BackupDR    10    1      192.168.2.3 192.168.2.1
Area: 0.0.1.44
IP Address      Type    State       Cost  Pri    DR          BDR
10.10.1.1       Bcast   DR          10    1      10.10.1.1   None
```

R2:

```
Area: 0.0.0.0
IP Address      Type    State       Cost  Pri    DR          BDR
192.168.1.2     Bcast   DR          10    1      192.168.1.2
192.168.1.1
192.168.3.2     Bcast   BackupDR    10    1      192.168.3.3 192.168.3.2
Area: 0.0.0.100
IP Address      Type    State       Cost  Pri    DR          BDR
172.168.1.1     Bcast   BackupDR    10    1      172.168.2.1 172.168.1.1
Virtual Links:
Transit Area    Router ID    Remote IP    Local IP    Type State  Cost
0.0.0.100       4.4.4.4      172.168.2.1  172.168.1.1 Virt PtoP   10
```

由以上信息可知各网段的DR:

IP	DR	BDR
192.168.1.0	192.168.1.2	192.168.1.1
192.168.2.0	192.168.2.3	192.168.2.1
192.168.3.0	192.168.3.3	192.168.3.2
172.168.0.0	172.168.2.1	172.168.1.1

此外，还可以知道:

1. 网段中的ABR是: R1,R2,R3,R4;
2. 没有ASBR;
3. 存在3个Stub Area: Area200，Area300，Area400。

LSDB的形成过程:

那么，明确了各自的职责后，各路由器就开始相互之间通过建立邻接关系来传播和交换各自的LSA了。这里，邻接关系的建立不作为重点讨论，我们重点关注一下LSA的生成和传播。

不同职责的路由器所要负责的LSA也不同，在本组网中用到的LSA分为以下几种:

Router LSA:

每一台路由器都要产生自己的Router LSA，它记录了该路由器在本Area的每一条连接。Rtr LSA只在本Area内传播:

在Area0，先来看看R1的Rtr LSA。R1在该区域有两条连接:

```
Type      : Router
Ls id     : 1.1.1.1
Adv rtr   : 1.1.1.1
Ls age    : 244
Len       : 48
Seq#      : 80000007
Chksum    : 0x2130
Options   : (DC) ABR      <<<<<<<<<因为R1与Area0和Area300相连，所以是ABR
Link count : 2            <<<<<<<<<可以看出R1在Area0中有两条连接
```

Link ID : 192.168.1.2 <<<<<<<<<<第一条，从192.168.1.1.到192.168.1.2
Data : 192.168.1.1
Type : TransNet
Metric : 10
Link ID : 192.168.2.3 <<<<<<<<<<第二条，从192.168.2.1.到192.168.2.3
Data : 192.168.2.1
Type : TransNet
Metric : 10

R2和R3的Router LSA与R1相类似，这里就不再列举了。比较特别的是R4的Rtr LSA：

Type : Router
Ls id : 4.4.4.4
Adv rtr : 4.4.4.4
Ls age : 259
Len : 36
Seq# : 80000007
Chksum : 0x5052
Options : (DC) ABR
Link count : 1
Link ID : 2.2.2.2
Data : 172.168.2.1
Type : Virtual <<<<<<<<R4并没有直接连在Area0上，而是通过虚连接与R2相连
Metric : 10

可以看出，Rtr LSA很简单但也是最基础的。在整个LSDB以及由此得到的网络拓扑树中，起到了一个“节点”的作用。每个路由器作为网络中的一个节点，他们的基本信息和基本连接状态都是由Router LSA通告天下的。

Network LSA

这种LSA是由DR生成的，他列出了所有与自己相连的路由器（包括他自己）。其实就是把那些选举它作为DR的所有接口所在的路由器都列举了出来。

该LSA与Router LSA相同，也是只在本Area中传播。

还是以Area0中的R1为例：列出R1在Area0上的接口信息：

OSPF Process 1 with Router ID 1.1.1.1
Area: 0.0.0.0

IP Address	Type	State	Cost	Pri	DR	BDR
192.168.1.1	Bcast	BackupDR	10	1	192.168.1.2	192.168.1.1
192.168.2.1	Bcast	BackupDR	10	1	192.168.2.3	192.168.2.1

可以看出在网段192.168.1.0 /24 中，DR为192.168.1.2，所以在该网段中只有192.168.1.2所在的路由器才会发布Network LSA。该LSA如下：

Type : Net
Ls id : 192.168.1.2
Adv rtr : 2.2.2.2
Ls age : 1678
Len : 32
Seq# : 80000002
Chksum : 0x2595
Options : (DC)
Net mask : 255.255.255.0
Attached Router: 2.2.2.2 <<<<<<<<<<192.168.2.1自己所在的路由器
Attached Router: 1.1.1.1 <<<<<<<<<<以他为DR的192.168.1.1所在的路由器

可以看出，与DR192.168.1.2相连的接口只有192.168.1.1。而这两个接口所在的路由器就是2.2.2.2 和1.1.1.1，于是，他将该信息列在这个LSA中，告诉本Area中所有的路由器。

此外，在Area0中，还有两个网段192.168.2.0和192.168.3.0，他们的DR也会在本Area中发布自己的Network LSA。

Network Summary LSA:

这种LSA是由ABR生成的，他告诉与它相连的某个区域能通过该路由器访问到哪些其他Area内的路由器。

在Area0如下中的路由器会接收到的Network Summary LSA：

Area : 0.0.0.0
Type : SumNet

Ls id : 10.10.0.0
Adv rtr : 1.1.1.1
Ls age : 813
Len : 28
Seq# : 80000003
Chksum : 0xee2c
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<<路由器1.1.1.1

说："通过我能到达网络10.10.0.0 /16，距离是10"

Type : SumNet
Ls id : 10.11.0.0
Adv rtr : 3.3.3.3
Ls age : 778
Len : 28
Seq# : 80000003
Chksum : 0xa66b
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<<路由器3.3.3.3

说："通过我能到达网络10.11.0.0 /16，距离是10"

Type : SumNet
Ls id : 10.12.0.0
Adv rtr : 4.4.4.4
Ls age : 828
Len : 28
Seq# : 80000004
Chksum : 0x7a91
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<<路由器4.4.4.4

说："通过我能到达网络10.12.0.0 /16，距离是10"

Type : SumNet
Ls id : 172.168.0.0
Adv rtr : 4.4.4.4
Ls age : 828
Len : 28
Seq# : 80000005
Chksum : 0xdeec
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<路由器4.4.4.4

说："通过我能到达网络172.168.0.0 /16，距离是10"

Type : SumNet
Ls id : 172.168.0.0
Adv rtr : 2.2.2.2
Ls age : 867
Len : 28
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<<路由器3.3.3.3

说："通过我能到达网络10.11.0.0 /16，距离是10"

Type : SumNet
Ls id : 10.12.0.0
Adv rtr : 4.4.4.4
Ls age : 828
Len : 28
Seq# : 80000004
Chksum : 0x7a91
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<<路由器4.4.4.4

说："通过我能到达网络10.12.0.0 /16，距离是10"

Type : SumNet
Ls id : 172.168.0.0
Adv rtr : 4.4.4.4
Ls age : 828
Len : 28
Seq# : 80000005
Chksum : 0xdeec
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric: 10 <<<路由器4.4.4.4

说："通过我能到达网络172.168.0.0 /16，距离是10"

Type : SumNet
Ls id : 172.168.0.0
Adv rtr : 2.2.2.2
Ls age : 867
Len : 28
Seq# : 80000003
Chksum : 0x1fb6
Options : (DC) (Non DN)
Net mask : 255.255.0.0
Tos 0 metric : 10 <<<<<<路由器2.2.2.2说："通过我能到达网络172.168.0.0.0.0 /16，距离是10"

当然，并不是只有Area0才接收到Network Summary LSA的。各个ABR在向Area0发Network Summary LSA的同时也会向其他与他们相连的Area发送。比如在Area300，R1就会发以下的Network Summary LSAs：

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Snet	10.11.0.0	1.1.1.1	179	28	80000002	20	Inter List
Snet	10.12.0.0	1.1.1.1	209	28	80000001	30	Inter List
Snet	172.168.0.0	1.1.1.1	219	28	80000001	20	Inter List
Snet	192.168.1.0	1.1.1.1	264	28	80000001	10	Inter List
Snet	192.168.2.0	1.1.1.1	229	28	80000001	10	Inter List
Snet	192.168.3.0	1.1.1.1	179	28	80000002	20	Inter List

该表中，Type表示LSA的类型，SNET表示Network Summary LSA。

由上面的例子可以看出，SNET LSA的使用有点像Rip协议中的距离向量算法，ABR把自己作为一个Next hoop告诉区域内的路由器，并告知他们通过自己能到达哪些接口。所以对于不同Area的路由器

他们并不知道对方的拓扑结构。

那么，这样不就会形成令Rip协议无比头疼的环路了吗？

这是我们就得感谢Area0的作用了。作为Back bone，所有的ABR都必须与之相连，也就是说，各Area之间形成了一个以Area0为根的二层树，自然不会有环路产生了。

网络中的各个路由器的各接口之间通过选举DR，建立Adjacency关系来传播上面提到的这些LSA，最终，每个路由器都生成了自己的LSDB。

下面列出了R4的LSDB：

Link State Database

Area: 0.0.0.0

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Rtr	2.2.2.2	2.2.2.2	367	60	8000000f	0	SpfTree
Rtr	4.4.4.4	4.4.4.4	365	36	80000006	0	SpfTree
Rtr	1.1.1.1	1.1.1.1	359	48	80000006	0	SpfTree
Rtr	3.3.3.3	3.3.3.3	358	48	80000006	0	SpfTree
Net	192.168.1.2	2.2.2.2	405	32	80000001	0	SpfTree
Net	192.168.2.3	3.3.3.3	358	32	80000002	0	SpfTree
Net	192.168.3.3	3.3.3.3	370	32	80000001	0	SpfTree
Snet	10.10.0.0	1.1.1.1	438	28	80000001	10	Uninitialized
Snet	172.168.0.0	4.4.4.4	329	28	80000003	10	Inter List
Snet	172.168.0.0	2.2.2.2	481	28	80000001	10	Uninitialized
Snet	10.11.0.0	3.3.3.3	282	28	80000002	10	Uninitialized
Snet	10.12.0.0	4.4.4.4	329	28	80000002	10	Inter List

Area: 0.0.0.100

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Rtr	2.2.2.2	2.2.2.2	381	36	80000007	0	SpfTree
Rtr	4.4.4.4	4.4.4.4	380	36	8000000a	0	SpfTree
Net	172.168.2.1	4.4.4.4	405	32	80000002	0	SpfTree
Snet	10.10.0.0	2.2.2.2	397	28	80000001	20	Uninitialized
Snet	10.11.0.0	2.2.2.2	362	28	80000001	20	Uninitialized
Snet	192.168.1.0	2.2.2.2	397	28	80000002	10	Uninitialized
Snet	192.168.2.0	2.2.2.2	362	28	80000002	20	Uninitialized
Snet	10.12.0.0	4.4.4.4	329	28	80000002	10	Inter List
Snet	192.168.3.0	2.2.2.2	397	28	80000001	10	Uninitialized

Area: 0.0.0.200

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Stub	10.12.0.0	4.4.4.4	415	24	0	0	SpfTree
Rtr	4.4.4.4	4.4.4.4	415	36	80000005	0	SpfTree
Snet	10.10.0.0	4.4.4.4	329	28	80000002	30	Inter List
Snet	172.168.0.0	4.4.4.4	329	28	80000003	10	Inter List
Snet	10.11.0.0	4.4.4.4	329	28	80000002	30	Inter List
Snet	192.168.1.0	4.4.4.4	329	28	80000002	20	Inter List
Snet	192.168.2.0	4.4.4.4	329	28	80000002	30	Inter List
Snet	192.168.3.0	4.4.4.4	329	28	80000002	20	Inter List

从上表可以看出，R4虽然是通过虚连接与Area0相连，而不是与Area0直接相连，但他与Area0内的路由器拥有一样的关于Area0的LSDB。

同时，需要注意，在Area200，与R4相连的是一个Stub类型的Area，所以R4生成了一个Stub类型的LSA。

路由表的产生：

有了上面的LSDB，在每个路由器上，OSPF就可以通过最小生成树算法得到路由表：

我们来研究一下R2的LSDB：

OSPF Process 1 with Router ID 2.2.2.2

Link State Database

Area: 0.0.0.0

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Rtr	2.2.2.2	2.2.2.2	283	60	8000000f	0	SpfTree
Rtr	4.4.4.4	4.4.4.4	284	36	80000006	0	SpfTree
Rtr	1.1.1.1	1.1.1.1	276	48	80000006	0	SpfTree

Rtr	3.3.3.3	3.3.3.3	275	48	80000006	0	SpfTree
Net	192.168.2.3	3.3.3.3	275	32	80000002	0	SpfTree
Net	192.168.3.3	3.3.3.3	287	32	80000001	0	SpfTree
Net	192.168.1.2	2.2.2.2	322	32	80000001	0	SpfTree
Snet	10.10.0.0	1.1.1.1	355	28	80000001	10	Uninitialized
Snet	10.11.0.0	3.3.3.3	199	28	80000002	10	Uninitialized
Snet	10.12.0.0	4.4.4.4	248	28	80000002	10	Uninitialized
Snet	172.168.0.0	4.4.4.4	248	28	80000003	10	Uninitialized
Snet	172.168.0.0	2.2.2.2	398	28	80000001	10	Inter List

Area: 0.0.0.100

Type	LinkState ID	AdvRouter	Age	Len	Sequence	Metric	Where
Rtr	2.2.2.2	2.2.2.2	298	36	80000007	0	SpfTree
Rtr	4.4.4.4	4.4.4.4	299	36	8000000a	0	SpfTree
Net	172.168.2.1	4.4.4.4	324	32	80000002	0	SpfTree
Snet	10.10.0.0	2.2.2.2	313	28	80000001	20	Inter List
Snet	10.11.0.0	2.2.2.2	278	28	80000001	20	Inter List
Snet	10.12.0.0	4.4.4.4	248	28	80000002	10	Uninitialized
Snet	192.168.1.0	2.2.2.2	313	28	80000002	10	Inter List
Snet	192.168.2.0	2.2.2.2	278	28	80000002	20	Inter List
Snet	192.168.3.0	2.2.2.2	313	28	80000001	10	Inter List

在Area0，

首先：R1--R4发出了自己的Rtr LSA，他们形成了本区域最小生成树的四个节点；

然后：Area0中的3个DR192.168.2.3，192.168.3.3，192.168.1.2公布了自己的NET LSA；

有了这两种LSA，我们很容易就可以得到Area0内的一张连结图：

```
R1————R2
|      |
+————R3
```

那么，R4哪去了呢？可别忘了他是以虚连接与Area0相连的，我们不能直接在区域内找到他。

由于组网很简单，所以由上面那张图，路由器R2得最小生成树马上就出来了（当然，就算是很复杂的情况用算法也很容易得到）：

```
R1————R2
      |
      R3
```

在Area100，情况更简单了，R3，R4发布了Rtr LSA；R4的172.167.2.1作为DR发布了自己的NET LSA，最小生成树就得到了：

```
R2————R4
```

有了这两棵树，路由表就出来一部分了：

Destination	Cost	Type	NextHop	AdvRouter	Area
172.168.0.0/16	10	Net	172.168.1.1	4.4.4.4	0.0.0.100
192.168.1.0/24	10	Net	192.168.1.1	2.2.2.2	0.0.0.0
192.168.3.0/24	10	Net	192.168.3.2	3.3.3.3	0.0.0.0
192.168.2.0/24	20	Net	192.168.1.1	3.3.3.3	0.0.0.0

怎么样，很简单吧。

当然，光靠这些可不够，其他Area的路由器怎么办？

我们再回到LSDB，看看那些类型为SNET的LSA：前面说过，这种类型的LSA是通过向量距离算法处理的。路由器X收到某个ABR Y的SNET LSA，该LSA的目标为Z。X把自己到Y的距离加上LSA中记录的Y到Z距离得到距离M。X在从路由表中找到自己到Y的下一跳N。这样，X到Z的路由就产生了：距离为M，next hoop 为N。

回到我们的组网：R2的LSDB中Snet类型的有：

Area0：

Snet	10.10.0.0	1.1.1.1	355	28	80000001	10	Uninitialized
Snet	10.11.0.0	3.3.3.3	199	28	80000002	10	Uninitialized
Snet	10.12.0.0	4.4.4.4	248	28	80000002	10	Uninitialized
Snet	172.168.0.0	4.4.4.4	248	28	80000003	10	Uninitialized
Snet	172.168.0.0	2.2.2.2	398	28	80000001	10	Inter List

因为到172.168.0.0的路由已经有Area内的路由了，所以不管最后两条，按上面提到的流程，得到以下路由：

Destination	Cost	Type	NextHop	AdvRouter
10.10.0.0/16	10+10	Snet	192.168.1.1	1.1.1.1
10.11.0.0/16	10+10	Snet	192.168.3.3	3.3.3.3
10.12.0.0/16	10+10	Snet	192.168.3.3	3.3.3.3

Area100：

Snet	10.10.0.0	2.2.2.2	313	28	80000001	20	Inter List
Snet	10.11.0.0	2.2.2.2	278	28	80000001	20	Inter List
Snet	10.12.0.0	4.4.4.4	248	28	80000002	10	Uninitialized
Snet	192.168.1.0	2.2.2.2	313	28	80000002	10	Inter List
Snet	192.168.2.0	2.2.2.2	278	28	80000002	20	Inter List
Snet	192.168.3.0	2.2.2.2	313	28	80000001	10	Inter List

处理与Area0相同，到192.168.1.0，192.168.2.0，192.168.3.0的因为已有Area内的路由不用管，其他的得到以下路由：

Destination	Cost	Type	NextHop	AdvRouter
10.10.0.0/16	20+0	Snet	192.168.1.1	1.1.1.1
10.11.0.0/16	20+0	Snet	192.168.3.3	3.3.3.3
10.12.0.0/16	10+10	Snet	192.168.3.3	3.3.3.3

最后，将路由表合在一起就得到了最终的路由表：

OSPF Process 1 with Router ID 2.2.2.2

Routing Tables

Routing for Network

Destination	Cost	Type	NextHop	AdvRouter	Area
172.168.0.0/16	10	Net	172.168.1.1	4.4.4.4	0.0.0.100
192.168.3.0/24	10	Net	192.168.3.2	3.3.3.3	0.0.0.0
10.10.0.0/16	20	Snet	192.168.1.1	1.1.1.1	0.0.0.0
192.168.1.0/24	10	Net	192.168.1.1	2.2.2.2	0.0.0.0
10.12.0.0/16	20	Snet	172.168.2.1	4.4.4.4	0.0.0.0
10.11.0.0/16	20	Snet	192.168.3.3	3.3.3.3	0.0.0.0
192.168.2.0/24	20	Net	192.168.1.1	3.3.3.3	0.0.0.0

Total Nets: 7

Intra Area: 4 Inter Area: 3 ASE: 0 NSSA: 0

注意：以上所描述的LSDB以及路由表的生成以及部分具体细节，在路由器的实际实现过程中是同步的，并没有严格的先后顺序。本文只是为了理解容易而将他们分解开来介绍。具体的实现细节有的并未提及。

关于OSFP不同区域负载分担的问题

技术支持中心 陈志刚

问题的提出

办事处在用户处实施工程时遇到下面的组网，见图1。用户想实现上行方向（从2830到NE05）的负载分担，但不愿意改变当前的OSPF区域划分。在该组网下，能否实现负载分担？

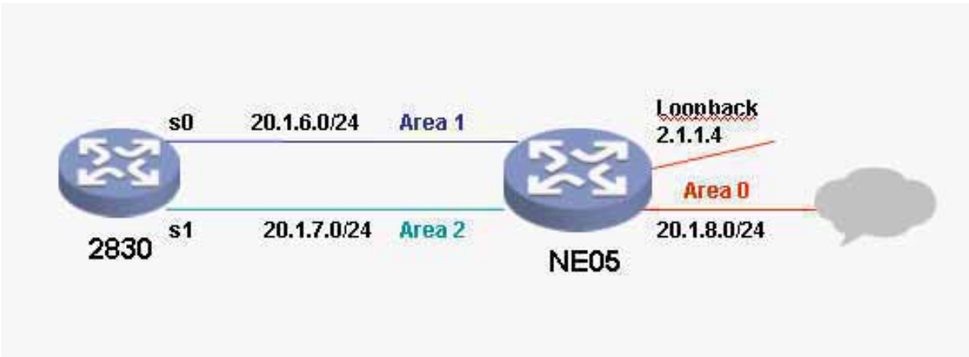


图1 用户组网

问题分析

问题初识

回顾通常的做法，如果市级路由器NE05以下下挂的县级路由器2830数量较多，为减小链路振荡变化导致的OSPF计算负担，可以通过划分区域的方法，将NE05下每个县划分到一个不同区域中，如果有2条广域网链路，可以做MP捆绑当做同一链路或者都划到相同区域中，例如区域1。如果县级路由器数量少，不考虑链路振荡，也可不分区域，同市级路由划分到同一区域，都属于骨干区域0或者相同的非骨干区域。两种通常的做法，都可以实现上行流量在s0 s1上的负载分担，不再赘述。

如图所示，该组网特殊之处在于OSPF非骨干区域的划分：两条广域网链路（以下分别简称为s0和s1，二者链路带宽相同）分属不同的非骨干区域：区域1和区域2（以下简称A1 A2）。在当前的区域划分下，能否实现负载分担？经过实验，显示2830上的路由表display ip route，到达网段20.1.8.0/24和2.1.1.4/32只有1条路由，出口是s0或s1，即无法形成负载分担。并且发现：最后OSPF配置的如果是s0，则下一跳是s0，最后配置的如果是s1，则下一跳是s1。

为何出现这样的情况呢？下面展开分析。

2830上的路由计算

2830上如何能计算出到达属于NE05区域0的路由20.1.8.0/24？

需要2种LSA：

1、网段20.1.8.0/24 通过第3类summary-network LSA（以下简称T3 LSA），由ABR NE05发布到A1 A2，2830相应从A1 A2学习到。

2、NE05发布第1类router LSA（以下简称T1 LSA），2830计算如何到达ABR NE05。

有了这2种LSA，就可以计算出到达目的网段的路由。

2830计算出到达网段20.1.8.0/24后，怎么加入路由表？

根据OSPF对于不含AS外部路由的路由选择规则，优先级由高到低为：

1、区域内路由

2、骨干区域内路由

3、区域间路由（指从经过了两个区域的路由但又不是虚连接构成的骨干区域内路由）。

对于网段20.1.8.0/24，2830从A1 A2学习到都是区域间路由，广域网链路cost也相同，不同的是从不同的区域学到。仔细复习RFC 2328的描述，在章节16.8 “Equal-cost multipath”中描述：“Each one of the multiple routes will be of the same type (intra-area, inter-area, type 1 external or type 2 external), cost, and will have the same associated area. However, each route may specify a separate next hop and Advertising router.”[1]

原来OSPF规定路由需要从相同的区域学习到，才能形成等价路由。这里网段是从不同区域A1 A2学习到的，因此不能形成等价路由。最终将哪条路由加入路由表，协议没有详细规定。考虑2830产品可能是这样实现的：回顾前述的最后配置OSPF的接口，成为下一跳，通过实验发现，最后做shutdown 和undo shutdown操作的接口，也成为下一跳。猜测可能产品实现在计算时，考虑所学习的LSA的新旧程度，最新收到的做为最真实的状态加以应用。仅仅是猜测，也许需要纠正。

问题的解决

原配置不能形成负载分担的原因有了，最新配置的OSPF接口成为下一跳也得到一定程度的解释。用户要求解决负载分担问题，如果不改变原区域配置，有什么办法？

考虑以下如图2的组网。

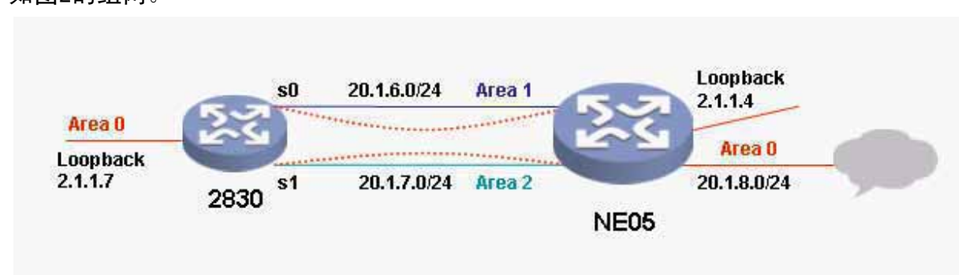


图1 用户组网

在2830上配置loopback接口（网段.1.1.7/32），划分骨干区域0，分别通过转换区域A1 A2，在2830、NE05之间配置虚链路，实现骨干区域的连通。经过实验配置，验证实现了上行流量的负载分担。

解决办法的分析

建立虚链路后，2830在区域0中，从两条虚链路上都学习到NE05发布的T1 LSA，描述拥有到达网段20.1.8.0/24的能力。同章节2中网段20.1.8.0/24是区域间路由相比，域内路由的优先级更高，因此会将域内路由加入路由表。另外由于域内的2条路由都是从相同区域学习到的，因此最终都加入路由表，实现了负载分担。

当然，最佳的办法是前文描述的通常的做法。本文的配置只是在学习掌握协议的过程中做的一些实验配置，用于说明理解协议，不建议这样配置。

思考过程的补充

从虚链路上学习到的路由表，是否能替换从实链路上学习到的相同路由？

原先认为虚链路这个架构是建立在区域A1 A2内路由计算结果这个基础上的，这个架构得出的结论能否修改原先的基础？仔细考虑发现，虚链路的存在，是区域内路由计算的结果（如何到达ABR）。而2830上网段20.1.8.0/24的计算是依靠这个基础再加上T3 LSA（ABR NE05发布的）得出的，因此关于网段20.1.8.0/24，从虚链路的得到的路由可以同实链路上学习到的进行比较，删除从实链路学习的该网段，不会影响到虚链路存在的基础。

关于网段2.1.1.4/32和ROUTER ID 2.1.1.4

考虑中曾经混淆的地方，写到这里供参考。关于网段2.1.1.4/32，是NE05的loopback接口，分配在骨干区。这同NE05的router id是不同的。在区域A1 A2 A0里， NE05发布 LSA 使用 router id 2.1.1.4，并不要求2830要知道如何到达网段2.1.1.4/32，router id仅仅是标识，即使NE05上没有网段2.1.1.4/32存在也是可以的。

在NE05上引入外部路由，是否能实现负载分担？

当不采用在NE05上配置network发布接口路由的方式，而是通过import 引入外部路由的方式，同样对于网段20.1.8.0/24，是否能实现负载分担？

对于这个问题，考虑LSA：

1、NE05引入外部路由，将在NE05上(ASBR，也是ABR)生成T5 LSA（缺省是第2类外部路由）在A1 A2 A0中发布；

2、因为NE05是ASBR，又是ABR，因此将不产生T4 LSA（描述如何到达ASBR的主机路由）发送到非骨干区域。这种情况下，非骨干区域的其他路由器通过T1 LSA一样能计算出到达 T5 LSA 发布者的路径。同样，也不会发布T4 LSA到骨干区域，骨干区域的其他路由器，采用类似非骨干区的方式，得到到达ASBR的路径。特别地，骨干区域的其他ABR将会主动生成T4 LSA向其相连非骨干区发送。

因此，2830到ASBR（NE05），从A1 A2的实链路各有1条路径到达，从A0有2条链路（虚链路）可以到达。因此有4条路径可到达ASBR。而且到达ASBR的路由都是intra-AS路由。关于外部路由的花费值，只计算外部路由引入时的花费值，当只有在外部路由花费值相同的情况下，才比较到ASBR的花费值，因此最终计算出到达网段20.1.80/24的花费亦相同。

根据RFC2328对于引入外部路由选择的描述，花费相同的情况下选择区域ID大的：

“when there are multiple least cost routing table entries the entry whose associated area has the largest OSPF Area ID (when considered as an unsigned 32-bit integer) is chosen.”[1]

配置示例

2830的路由

```
[2830]dis ip rou
Routing Tables:
  Destination/Mask   Proto   Pref      Metric      Nexthop     Interface
        2.1.1.4/32     OSPF   10        3124        20.1.6.2    Serial0
                                                    20.1.7.2    Serial1
        2.1.1.7/32   Direct    0           0       127.0.0.1    LoopBack0
       20.1.6.0/24   Direct    0           0        20.1.6.2    Serial0
       20.1.6.1/32   Direct    0           0       127.0.0.1    LoopBack0
       20.1.6.2/32   Direct    0           0        20.1.6.2    Serial0
       20.1.7.0/24   Direct    0           0        20.1.7.2    Serial1
       20.1.7.1/32   Direct    0           0       127.0.0.1    LoopBack0
       20.1.7.2/32   Direct    0           0        20.1.7.2    Serial1
       20.1.8.0/24     OSPF   10        3124        20.1.6.2    Serial0
                                                    20.1.7.2    Serial1
      127.0.0.0/8    Direct    0           0       127.0.0.1    LoopBack0
      127.0.0.1/32   Direct    0           0       127.0.0.1    LoopBack0
```

因此选择从A2到ASBR，最终只将1条路由加入路由表。因无法将A0（存在2条链路到达ASBR）学到的加入路由表，因此不能实现负载分担。

CISCO路由器的实现

将2830更换成CISCO3700（Version 12.3(10)），观察其实现方式。

1、在图1组网下，对于不同区域学习的域间路由，形成了ECMP。即未按照RFC2328 章节16.8的不同区域不能形成ECMP规定实现。

2、在图1组网下，对于引入外部路由的情况，亦形成ECMP。即未按照RFC2830的对于引入外部路由选择区域ID大的规定实现。

这样实现利弊如何，还不清楚。

■ NE05的路由

```
[NE05]dis ip rou
 Routing Table: public net
Destination/Mask    Protocol Pre  Cost        Nexthop         Interface
2.1.1.4/32          DIRECT   0    0           127.0.0.1       InLoopBack0
2.1.1.7/32          OSPF     10   3124        20.1.6.1        Serial4/0/1
                                              20.1.7.1        Serial4/0/2
20.1.6.0/24         DIRECT   0    0           20.1.6.2        Serial4/0/1
20.1.6.1/32         DIRECT   0    0           20.1.6.1        Serial4/0/1
20.1.6.2/32         DIRECT   0    0           127.0.0.1       InLoopBack0
20.1.7.0/24         DIRECT   0    0           20.1.7.2        Serial4/0/2
20.1.7.1/32         DIRECT   0    0           20.1.7.1        Serial4/0/2
20.1.7.2/32         DIRECT   0    0           127.0.0.1       InLoopBack0
20.1.8.0/24         DIRECT   0    0           20.1.8.2        Serial4/0/3
20.1.8.1/32         DIRECT   0    0           20.1.8.1        Serial4/0/3
20.1.8.2/32         DIRECT   0    0           127.0.0.1       InLoopBack0
127.0.0.0/8         DIRECT   0    0           127.0.0.1       InLoopBack0
127.0.0.1/32        DIRECT   0    0           127.0.0.1       InLoopBack0
```

参考资料

1、本文参考了测试中心论坛关于该问题的讨论和技术支持部关于该问题的邮件，进行总结补充。
2、RFC 2328　《OSPF Version 2》

如何测试OSPF

测试研究部 许亮 程新娥 李敏 李玉 陶豆 金炜

OSPF协议概述

OSPF是路由协议是一种典型的链路状态路由协议，在实际网络中得到广泛应用，是主要的IGP（内部网关协议），在RFC2328中对该协议有详细描述。运行OSPF协议的路由器之间通过交换链路状态信息（连接状态，连接花费），得到整个网路的路由。运行过程包括建立进程和使能、建立邻居、交换LSA、计算路由。

整个OSPF网络划分为多个区域，区域类型为2大类：骨干区（0区）和普通区域。其目的是将网络层次化，各类链路状态信息只在一定区域内交换。从而减少网络流量和存储开销，也方便网络的管理。并且规定，所有非骨干区必须同骨干区连接，对于无法直连的区域可以使用虚连接（virtual－link）方式连接到骨干区。在某些特殊网络中，我们不需要区域内部路由器学习到所有网络路由，特别是外部路由，为此，OSPF可以将区域设置为STUB和NSSA类型，用于完全或部分过滤调外部路由。

在同一区域内，路由器会和同一链路上的其他路由器通过发送Hello、DD、LSR、LSU、LSAck报文建立邻居关系，Hello报文用于发起这个建立过程，并在建立邻居后定期发送以维护邻居关系，DD（数据库描述）报文描述路由器所拥有的所有链路状态信息的概要信息，邻居检查DD报文，发现自己缺少和需要更新的链路状态信息，就会发送LSR（链路状态信息请求）请求对方发送包含该条链路状态信息的完整内容的LSU（链路状态信息更新）报文，收到后回复LSAck（链路状态信息确认），最终邻居关系达到Full状态。

路由器收到链路状态信息后，会作相应的区域内路由计算、区域间路由计算、外部路由计算。计算区域内路由时，先以路由器自己为根，用Dijkstra算法由链路状态信息生成区域的最小生成树，即选出到达区域内其他路由器具有最小花费的路径，然后加入地址信息，生成路由。

区域间路由计算则是路由器到区域边界路由器（ABR）的路径加上ABR到其他区域的路径之和。如果存在多个ABR，则出现多条路径，选取花费最小的。

计算外部路由的规则：路由器到自治系统边界路由器（ASBR）的路径加上ASBR到达此外部地址的路径之和。任何导入其他路由协议（包括静态和直连路由）的IPv6路由的路由器，都称为ASBR，不一定需要位于自治系统的拓扑边缘。

测试方法描述

要对OSPF进行一次全面的测试需要包括下面几个方面的测试：

配置功能测试
功能及协议一致性测试
性能测试
组合测试
互通测试

❶ 配置测试方法

这是对OSPF进行最基本的配置功能测试，主要验证设备是否有完整、正确的配置命令接口和调试命令接口。命令行测试中，不但需要关注各种OSPF基本配置能够正确执行、生效，还应该对命令行的解释、命令行执行后的提醒信息以及输出调试信息进行测试。

基本配置测试时应该重点注意：设备提供的配置命令行和调试命令行是否完整；对每条配置命令和调试命令的解释是否准确，有无拼写、语法错误；正确的配置命令执行后是否有效；配置命令的中需要输入的参数是否合理，对边界值和异常值的处理是否正确；对于错误的配置是否有准确的错误提示信息；配置能否正确保存、Buildrun和配置恢复；反复配置和取消配置后设备有无异常等。

特别应该注意的是我司设备都提供了英语和中文两种配置环境，所以在测试中还需要对照这两种环境中对同一个命令的描述是否相同、准确

❷ 功能及协议一致性测试方法

基本功能测试

基本功能测试，是按照OSPF协议规定的功能模块对设备的功能完备性、正确性进行测试，是对OSPF最基本的测试方法。由于OSPF协议的复杂性，我们按照其功能划分为邻居状态机、LSA传播、路由计算和虚连接几个部分，分别对其测试方进行描述。

接口状态机测试

这部分的测试主要集中在测试DR、BDR的选举过程上。测试要点主要包括：

路由器处于2-WAY状态时，构造HELLO报文的优先级、DR和BDR等字段，使路由器进入不同的DR和BDR的选举状态；

DR的三条原则：民主选举制，终身制以及继承制；

在不同的链路上DR的选举，包括：广播、NBMA。

测试的方法：

第一个测试要点由于处于动态的中间状态，主要应该依靠测试仪完成，比如：AX4000等，对这部分有详细的测试用例（依据RFC2328）；

第二、三测试要点可以分别设计基本功能、多个原则以不同顺序组合，以及基于不同链路类型上进行测试（依据RFC2328）。

邻居状态机测试分析

这部分的测试要点主要集中在测试INIT、2-WAY、EXSTART、LOADING、FULL等几个状态机的迁移过程上。测试要点包括：

各个状态机之间的迁移过程及迁移条件；

DD报文的交互过程；

报文的正确性检查，包括：HELLO报文、DD报文、REQUEST报文、UPDATE报文以及ACK报文。

在不同链路上，对以上三个测试点分别进行验证，包括：广播、NBMA、P2P、P2MP。

主要测试方法包括：

前两个测试要点由于处于动态的中间状态，主要应该依靠测试仪完成，比如：AX4000，对这部分有详细的测试用例（依据RFC2328）；

第三个测试要点也主要依靠测试仪完成，比如：ROUTERTESTER，对这部分有详细的测试用例（依据RFC2328）；

第四个测试要点需要在不同的链路上通过测试仪完成，比如：AX4000、ROUTERTESTER等（依据RFC2328）。

LSA的传播测试

（一）各种网络类型上LSA的发送测试

OSPF利用LSU发送LSA，从而保证LSDB的同步更新。在广播网络上，DR/BDR发送LSU的目的地址是AllSPFRouters(224.0.0.5)，DROther对收到的LSU回应ACK报文；DROther发送LSU的目的地址是AllDRouters(224.0.0.6)，DR收到DROther发来的LSA后再转发给AllSPFRouters。在P2P网络上，发送LSU的目的地址都是AllSPFRouters。而在其它网络（包括virtual links）上，则是以单播形式发送，即直接发送给对端邻居。

进行这一部分的测试，主要是通过debug信息来判断。在设备上打开debug ospf packet update开关，然后reset ospf process，根据路由器的各自身份（是DR、BDR还是DROTHER）查看其发送的LSU报文的目的地址是否与上面给出的相符。

（二）LSA的flooding范围

OSPF中各类LSA的flooding范围是不一样的，type 1,2,3,4类的flooding范围是区域内，type5类的flooding范围是整个OSPF域，type7类只能在NSSA区域中，stub区域中只能存在type1,2,3类LSA，total-stub区域中只能存在type1,2,3类LSA，且type3类LSA只有一个缺省路由。NSSA区域中不能有type5类的LSA，由ABR负责type7-type5之间的转换。

这一部分的测试主要是通过LSDB中的信息来判断。通过查看不同类型的LSDB信息，对照各类LSA的flooding范围，看结果是否相符。

（三）LSA的周期性更新

为了避免LSDB中的LSA被老化，从而引起不

必要的路由计算，OSPF定期（LSRefresh）向其它OSPF更新自己发出的LSA，LSRefresh一般为30分钟。

该测试项也是通过debug信息来验证的。在设备上引入一条外部路由，然后在邻居上打开debug ospf packet update开关，发现收到对端发来的LSU，其中包含一个type 5类的LSA；等待LSRefresh时间后，观察邻居的屏幕输出，会发现再次收到对端发来的LSU，里面包含这个LSA。

OSPF路由计算测试方法

通过泛洪收到LSA后，路由器将根据LSA的信息，使用Dijkstra算法来计算出自己所能到达的路由以及相应的cost值和下一跳。计算是按照先计算路由器所处区域内的路由，再计算区域间路由，最后计算AS外部路由的顺序进行的。下面就介绍一下目前经常使用的对路由计算的测试方法：

路由计算正确性测试：即在一些特定组网环境下，测试OSPF区域中的路由器能否由学到的正确的LSA完整，正确的计算出OSPF路由。这就要求，在测试时必须尽可能的将各种情况考虑全面，包括：同一区域中路由的计算，区域间路由的计算，外部引入路由（由ASE和NSSA两种LSA生成）的计算，Stub区路由的计算,NSSA区路由的计算，有虚连接时路由的计算等。另外，在测试时要考虑变化的情况，如区域改变、路由器接口变化、虚连接状态变化时，能正确及时的计算出变化后的路由以及相应的cost值、下一跳。

路由优选比较规则测试：由于OSPF在计算路由时是逐级计算的，因此要验证当有多条LSA包含同一条路由信息时，路由器在这些LSA中进行选择的顺序是否正确。也就是要考虑：同一区域不同LSA之间的比较（比较cost等）；不同区域学到路由之间比较（规则有区域内路由优于区间路由优于外部路由等）；ASE路由和NSSA路由之间的比较；以及同为ASE或同为NSSA路由时type-2与type-1路由的比较等情况。而在测试时，要针对每种情况构造出正确的网络环境，测试计算出的路由是否满足各项比较规则。

对forwarding address的测试：在某些情况下引入外部路由时，LSA中会包含forwarding address来指导计算下一跳。在测试时，要针对ASE类引入路由和NSSA类引入路由两种情况，测试在forwarding address满足条件时，计算出的路由将以forwarding address作为下一跳，而在forwarding address不满足条件时，能够正常计算出下一跳而无异常情况出现。

虚连接测试方法

OSPF要求骨干区连续，非骨干区间的通信必须经过骨干区。为了保证在任何组网下都能满足上述两个条件，我们需要用到Virtual Link。OSPF把Virtual Link连接当作unnumbered point-to-point处理，metric是穿越的区域的路由器出端口metric之和。

对Virtual Link的测试可以分为两个方面：

未配置Virtual Link，骨干区分离；

配置Virtual Link后，骨干区连续。

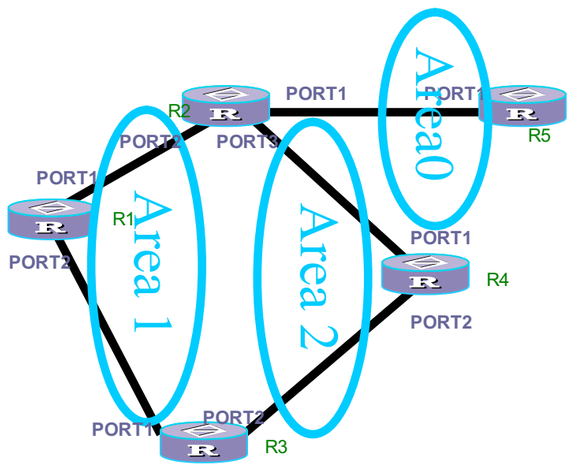


图1 虚连接图示

组网如上图所示：R3跨越Area1，Area2，但没有与骨干区路由器建立虚连接。

未配置Virtual Link，骨干区分离：由于非骨干区必须通过骨干区进行通讯，因此R1上的Area2网段信息都是R2发布的。R3虽然跨两个区域，但不能产生3类LSA，也不能担任ABR的责任。

配置Virtual Link后，骨干区连续：在RT2、RT3上配置Virtual Link后，骨干区连续，路由学习正常。这时，R3向Virtual Link穿越的Transit区域发布的LSA应该遵循水平分割原则。同时，Virtual Link的metric是穿越的区域的路由器出端口metric之和。

协议一致性测试

RouterTester和Ixia等测试设备都提供了进行协议一致性测试测试套，仪器会严格根据RFC上的描述对OSPF的实现进行测试。协议一致性测试对于验证我司设备对协议标准实现的准确性，保证能够顺利同其他设备商的产品进行互联。测试方法相对简单，只需要按照测试仪器提供的组网图将设备与仪器互联，并运行自动化脚步即可。对于测试结果，不同厂家（包括测试仪器厂商及设备厂商）对RFC的理解及遵照程度不一样，会有一些测试不通过的项目，对于测试过程中使DUT重启或发生异常，肯定是bug，需要解决。对于和RFC不一致或和测试仪器厂家不一致的测试项，需要具体分析。

❸ 性能测试方法

性能测试在整个测试当中能够起到举足轻重的作用，所以能够选择好性能测试的方法并以适当的测试内容进行测试是非常重要的。OSPF的性能测试大致可以分为规格验证测试、压力异常测试和拷机持久性测试。下面简单的介绍一下OSPF性能测试方面的内容。

OSPF的规格验证测试

规格验证测试作为路由测试的基本测试项，能够很好的评测被测设备的基本能力。规格验证测试中应包含OSPF内、外部路由容量的测试、最大邻居数的测试、路由学习时间的测试、路由振荡收敛时间的测试（包括Failover和Recovery形式）、路由振荡的丢保率和振荡中的平均延时的测试。这些测试项都可以使用测试仪器来进行，其中比较好的测试仪器为SmartBits的TRT软件。

在进行规格测试中的容量测试时，被测设备与测试仪器的两个端口相连，从仪器的一个端口发布路由，另一个端口转发到发布路由的很小的流量（正常转发情况下不至于丢包的流量），察看是否能够转发所有的报文，如果都能转发，说明发布的路由都能够正常的学到，之后增加发布的路由再次进行测试，直到测试出设备的路由容量。

在进行最大邻居数的测试时，使用测试仪器可以很方便的模拟，与被测设备的单个端口相连的多个邻居的环境，并在每个邻居上发布很小的路由，察看被测设备是否可以完全的建立所有的OSPF邻居，并且转发的流量也非常正常的情况下可以继续增加模拟的邻居数目，使用同样的方式重复进行测试。最大邻居数的测试与内存CPU资源的大小相关性比较大。

路由学习时间的测试和路由振荡收敛时间的测试都可以通过测试仪器来实现，像TRT软件就具有Flapping的功能，在发布路由的同时可以进行路由振荡事件的添加。在进行振荡收敛时间时，可以从仪器的一端发布路由，等被测设备全部学到所有路由时再通过Flapping中的收回事件把所有的路由收回，并重新发布路由事件来实现路由振荡收敛事件的测试。这个部分的测试操作起来可能有些烦琐，结果也必须通过计算的方式获得，所以推荐仪器厂家提供的脚本化的程序进行。

OSPF的压力异常测试

压力异常测试是考量被测设备稳定性和安全性的很好的方法。在OSPF的压力异常测试当中主要包含如下的相关测试项：在较大流量的情况下，反复路由振荡、引入路由/取消引入路由、邻居反复出现取消、设备重启或者反复Up/Down OSPF的接口或者重启OSPF进程等操作，察看设备是否可以正常稳定地恢复；OSPF的接口板热插拔时OSPF是否还正常恢复；大路由、大流量下进行路由振荡等操作时的系统稳定性。

如上的压力异常测试的前提条件是在被测设备上必须一直存在大路由或者是大流量，在这种压力环境中再进行其他平常易出现异常现象的操作时，设备的稳定性、设备的正常恢复能力。对于异常压力测试的环境搭建也可以通过测试仪器来实现的，所有仪器厂家的测试仪器都可以实现如上的环境，从仪器的一个端口上发布大量的OSPF路由并从另一个端口转发到构造路由的流量即可。

另一种异常测试就是通过构造一些非法的协议报文攻击路由器，验证设备能够对这些非法报文的处理。测试方法包括在测试状态机时，利用仪器和工具软件构造某些字段错误的协议报文发给设备，验证设备是否能正确处理。另一种就是在邻居关系正确建立以后，向设备发送大量错误的协议报文攻击设备，验证设备是否异常。

❹ OSPF的考机持久性测试

持久性测试是考量被测设备的长期稳定性的能力察看设备在长时间的，不同的环境下是否可以正

常的运行。在OSPF的持久性测试中主要包含：在大路由、大流量下长时间运行时的系统的稳定性；在大路由、多邻居、大流量下长时间运行时系统的稳定性；在大路由、大流量、频繁路由振荡下长时间运行时的系统稳定性等。持久性测试中首先要满足长时间的要求，一般持续时间为72小时左右。在OSPF的持久性测试中一般结合压力测试会很容易发现设备的问题的。

❺ 组合测试方法

由于OSPF用于很多其他IGP路由协议不具备的优点，在实际网络应用中大部分都采用OSPF作为AS内部网关协议，保证网络连通性。然而，实际网络并不会使用单一一种路由协议。同时，ISP为了提供有效的服务，合理利用网络资源，在网络上不光会配置路由协议，还会有如Qos、组播甚至MPLS TE等上层应用协议。此时，光OSPF本身基本功能正确还不能保证设备在实际网络中正常工作，组合测试的目的就是为了验证OSPF协议在这样复杂的实现网络环境下能够同其他网络协议配合工作，共同维护网络的通畅和提供各种服务。

与其他路由协议的组合测试：ISP一方面需要利用BGP发布本AS的可达路由，OSPF与BGP协议之间能够相互有策略地引入对方路由。同时，根据实际网络拓扑和规模需要，ISP可能会在同一个网络中部署不同的IGP协议，OSPF协议是否能够同其他路由协议共同组合工作也是组合测试验证的重点。路由协议组合测试的主要测试方法包括：利用路由策略等手段有策略地引入其他网络协议路由；多种IGP协议共同存在时能否有效防止路由环路，不同协议类型路由的优选等。

与组播协议的组合测试：随着IPTV等组播业务的不断推广，OSPF与组播协议相结合的测试显得迫切而重要。其主要测试方法包括：OSPF网络与PIM SM、PIM DM结合，通过振荡路由、切换链路、RESET OSPF 邻居和PIM邻居等方法测试路由协议不稳定性对组播的影响；在存在组播流的网络中OSPF功能的正确性。OSPF路由可能影响到组播RP等角色主机的选举、共享树和最短路径树的计算、组播流量转发等。而大量组播报文也可能影响OSPF报文的传递、处理，甚至导致邻居状态变迁、路由振荡等。

与链路层协议的组合测试：在实际网络应用中，除了大量使用到以太网接口外，ATM、POS、E1等接口被广泛应用在广域网络中，对应的链路层协议包括：ATM、PPP、FR和HDLC等。对于不同的链路层协议和网络拓扑，OSPF采用不同的网络方式建立邻居关系、交换路由信息。和链路层协议得组合测试验证我司设备能否在各种网络环境中正常运行。在测试过程中主要关注：通过ATM、PPP等协议连接的设备之间能否正常建立OSPF邻居、传播LSA；在改变链路协议后，OSPF能否正常工作；在相同链路上使用不同OSPF网络类型建立邻居（包括：分别使用P2P、P2MP、NBMA和Broadcast）。

以上几种组合测试可以利用一个较为复杂、含盖多种链路类型和网络结构的组网环境同时进行测试，并利用测试仪器模拟多种实际流量和适当路由进行测试。测试手段包括：振荡路由、改变网络拓扑、端口物理端口等方法。

❻ 互通测试方法

在实际网络中，我司设备不可避免的需要同其他厂商的设备进行互联。尽管所有厂商都使用同样的协议标准进行开发，不同的实现方法和数据结构往往会带来相象不到的问题。互通测试就是为了验证我司产品能否正确与其他厂商产品进行互联互通、交换信息。目前，主流的数据通信设备供应商包括：Cisco、Juniper、港湾等。考虑到OSPF协议应用位置，这里我们主要考虑与Cisco和Juniper设备的互通。OSPF互通测试主要考虑邻居关系建立、路由交换、验证互通以及STUB区域、NSSA区域和虚连接的互通等内容。对于邻居关系、路由交换和验证的测试方法可以使用两台设备简单直连组网方式，链路协议可以使用以太网、PPP、FR等验证在不同OSPF网络类型上的邻居互通、DR选举是否正确、一致。对于STUB、NSSA区域和虚连接的验证，需要采用较为复杂的组网，分别让我司设备和其他厂商设备作为ABR和ASBR，验证是否能正确接收对端设备发布的缺省路由、7类路由转换为5类，以及在NSSA区域存在多个ABR时对于7类—>5类路由转换的路由器的选举是否正确等功能。

RouterTester之测试OSPF初步

鉴定测试中心 姜杏春

OSPF概述

OSPF是Open Shortest Path First（开放最短路由优先协议）的缩写。它是IETF组织开发的一个基于链路状态的内部网关协议。目前使用的是版本2（RFC2328），其特性如下：

适应范围——支持各种规模的网络，最多可支持几百台路由器

快速收敛——在网络的拓扑结构发生变化后立即发送更新报文，使这一变化在自治系统中同步。

无自环——由于OSPF根据收集到的链路状态用最短路径树算法计算路由，算法本身保证了不会生成自环路由。

区域划分——允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，一方面减少了占用的网络带宽，另一方面减少了计算量。

等值路由——支持到同一目的地址的多条等值路由。

路由分级——使用4类不同的路由，按优先顺序来说分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。

支持验证——支持基于接口的报文验证以保证路由计算的安全性。

组播发送——支持组播地址。

区域（Area）

随着网络规模日益扩大，当一个网络中的OSPF路由器数量非常多时，会导致LSDB变得很庞大，占用大量存储空间，并消耗很多CPU资源来进行SPF计算。并且，网络规模增大后，拓扑结构发生变化的概率也会增大，导致大量的OSPF协议报文在网络中传递，降低网络的带宽利用率。

OSPF协议将自治系统划分成多个区域（Area）来解决上述问题。区域在逻辑上将路由器划分为不同的组。不同的区域以区域号（Area ID）标识，其中一个最重要的区域是区域0，也称为骨干区域（backbone area）。

骨干区域完成非骨干区域之间的路由信息交换，它必须是连续的，对于物理上不连续的区域，需要配置虚连接（virtual links）来保持骨干区域在逻辑上的连续性。

路由器类型

OSPF路由器被生成和区域相关的4个类型，如下：

所有接口都属于同一个区域的路由器称作内部路由器。

连接骨干区域和非骨干区域的路由器称作区域边界路由器（Area Border Router，简称为ABR）。

OSPF中还有一类自治系统边界路由器（Autonomous System Boundary Router，简称为ASBR），实际上，这里的AS并不是严格意义的自治系统，连接OSPF路由域routing domain）和其它路由协议域的路由器都是ASBR，可以认为ASBR是引入OSPF外部路由信息的路由器。

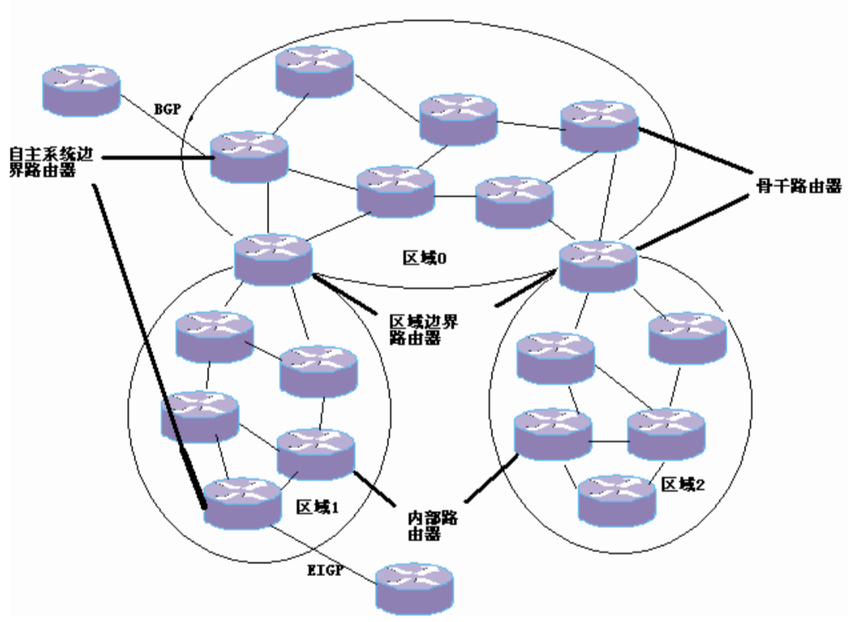


图1 路由器类型

LSA的类型

Router-LSAs：第一类LSA（Type-1），由每个路由器生成，描述本路由器的链路状态和花费，只在路由器所处区域内传播。

Network-LSAs：第二类LSA（Type-2），由广播网络和NBMA网络的DR生成，描述本网段的链路状态，只在DR所处区域内传播。

Summary-LSAs：包含第三类LSA和第四类LSA（Type-3，Type-4），由区域边界路由器ABR生成，在与该LSA相关的区域内传播。每一条Summary-LSA描述一条到达本自治系统的、其它区域的某一目的地的路由（即区域间路由：inter-area route）。Type-3 Summary-LSAs描述去往网络的路由（目的地为网段），Type-4 Summary-LSAs描述去往自治系统边界路由器ASBR的路由。

AS-external-LSAs：第五类LSA（Type-5），由自治系统边界路由器ASBR生成，描述到达其它AS的路由，传播到整个AS（Stub区域除外）。AS的缺省路由也可以用AS-external-LSAs来描述。

NSSA-external-LSAs：第七类LSA（Type-7），在NSSA区域内部产生和发布；但NSSA区域内不会产生或发布Type-5类LSAs。

Opaque LSAs：为了使OSPF能够支持更多新的业务应用，在RFC2370（The OSPF Opaque LSA）中定义了用于对OSPF进行扩展的Opaque LSAs。Opaque LSAs包含三种类型的LSA，不同类型的L A扩散范围不同：Type-9：扩散范围为link-local，可以认为只在某一个接口所在的网段扩散，不会发布到本地网段或本地子网以外。Type-10：扩散范围为area-local，即，只在本区域以内扩散。Type-11：与Type-5 LSAs具有相同的扩散范围，可以在除STUB区域和NSSA区域之外的整个自治系统内部扩散。

RouterTester测试OSPF

RouterTester简介

RouterTester是高端和核心路由器专用测试仪器，提供FE，GE，POS，ATM网络接口，速度从FE到10GPOS，10GE基本完全覆盖。

RouterTester测试OSPF功能简介

支持OSPF协议仿真/统计/解码功能

支持OSPFv2/OSPFv3协议验证和压力测试

可以模拟复杂真实的OSPF环境测试

支持运行模拟环境的同时动态添加路由器/网络/LSAs

支持动态通告或删除仿真拓扑的连接

支持OSPF流量的测试统计

OSPF路由协议测试原理

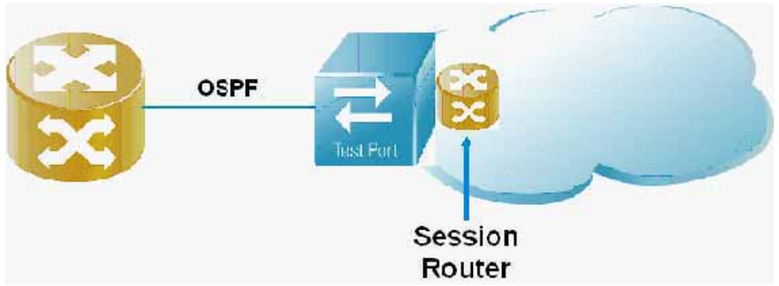


图2 Session Router

当使用Router Tester创建一个OSPF Session时，一个模拟的路由器也被创建了，称作Session Router，是和被测设备（SUT）接口直接相连的。
Router Tester 路由测试的基本思想是Router Tester首先扮演一个Session Router角色，然后在Session 后面可以根据使用者设置的数据模拟一个可构造数量的Router组成的拓扑网络。在进行测试的时候根据形成的路由器拓扑网络生成巨大的LSAs，通过Session Router引入到我们的SUT中来对SUT进行路由测试。

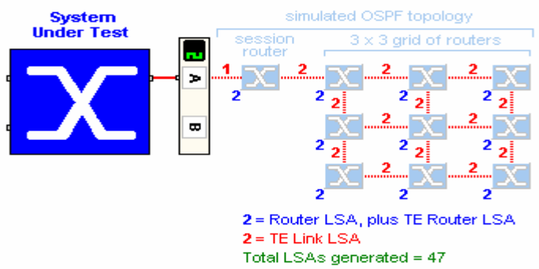


图3 路由测试拓扑

配置被测设备（SUT）

按照需要，配置被测设备：

配置接口

启动OSPF协议

配置OSPF协议的各个参数

配置RouterTester

选择端口，配置物理层、链路层

首先选择使用的端口，图4选择了102的两个端口102/1, 102/2。

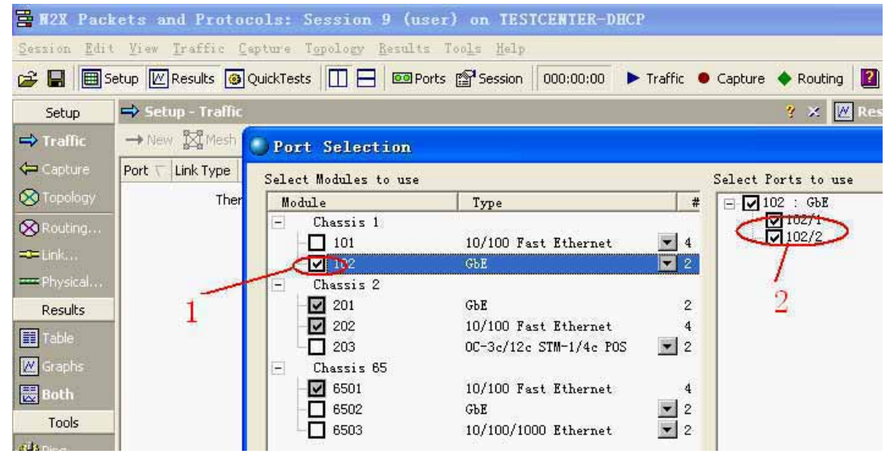


图4 选择端口

选择端口后，建立配置与被测设备的连接，配置物理层参数，配置链路层参数。
点击左侧Setup菜单的 Link...，配置链路层参数。图5链路层协议是Ethernet的一个例子。

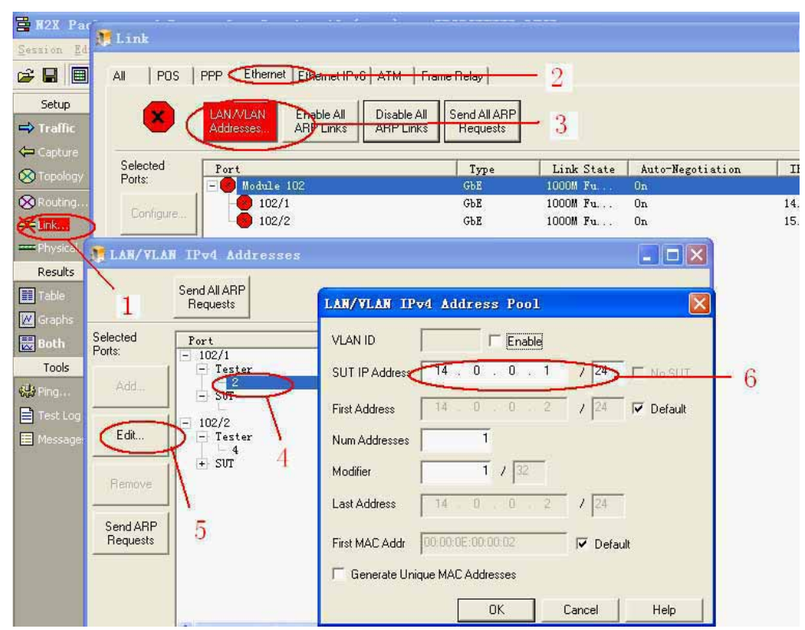


图5 配置链路层参数

配置完链路层参数后，发送ARP请求，获取被测设备的MAC地址，如图6示。

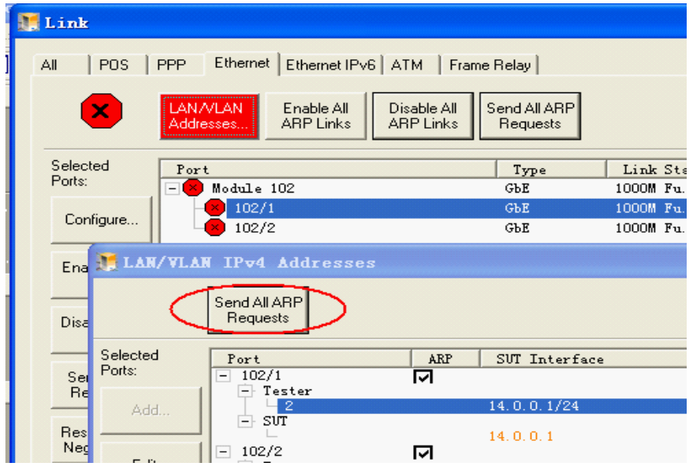


图6 发送ARP请求

发送ARP请求后，端口状态图标由 转变成 ，见图7说明已经与被测设备建立连接。

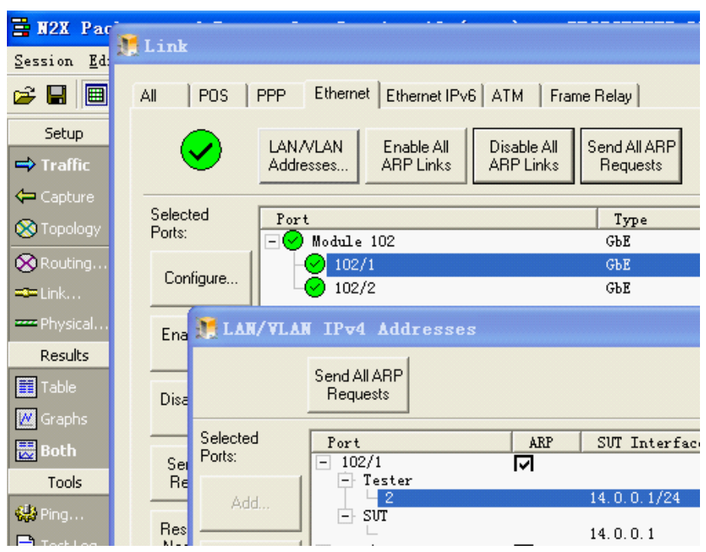


图7 与被测设备建立连接

创建一个OSPF session

接下来就是配置OSPF路由参数，模拟拓扑网络，点击左侧Setup菜单的 Routing...，显示Routing对话框，如图8。

选择Routing对话框上侧的OSPF栏，如图8，表示要模拟一个OSPF网络拓扑。

要模拟OSPF网络拓扑，先得构造一个OSPF session。

选择一个我们需要添加session的测试端口，图8中选择102/1端口，然后再点击Routing对话框左侧的 Add Session... ，添加一个session，这时会显示一个session对话框。

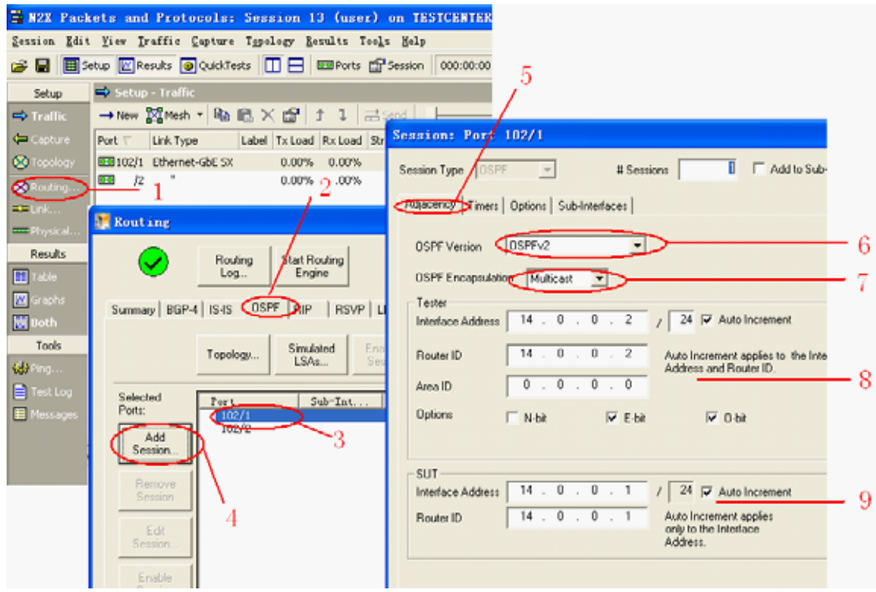


图8 构造一个OSPF session

然后就可以配置OSPF的邻接关系参数了。

你可以在OSPF Version栏里选择OSPF的版本（即OSPFv2或OSPFv3），只有在端口选择的类型是IPv6的情况下才会选择OSPFv3。

OSPF Encapsulation是用来选择以单播(Unicast)或多播(Multicast)方式发送报文。被测设备(SUT)匹配的网络类型配置必须能够确保测试端口可以识别。OSPF的网络类型通常取决于接口类型：

	Unicast	Multicast
POS	PTOP_UNICAST	PTOP
Ethernet	NBMA	BROADCAST

配置session Router接口IP地址、Router ID和Area ID等。在Tester栏的Options，N-bit表示session Router在一个NSSA，如果选了N-bit则与之相连的被测设备（SUT）接口也必须是NSSA的。E-bit表示测试session可以处理自治系统外部LSAs，如果不是Stub区域，就要选上E-bit。O-bit表示可以接收和发送Opaque LSAs。

配置SUT的接口IP地址和Router ID，注意SUT的Router ID是否填正确，默认的值是使用SUT的接口IP地址，通常这和SUT的Router ID不符的。

构建网络拓扑结构

点击Routing对话框上侧的 Topology... 显示一个OSPF拓扑仿真窗口，如图9示。

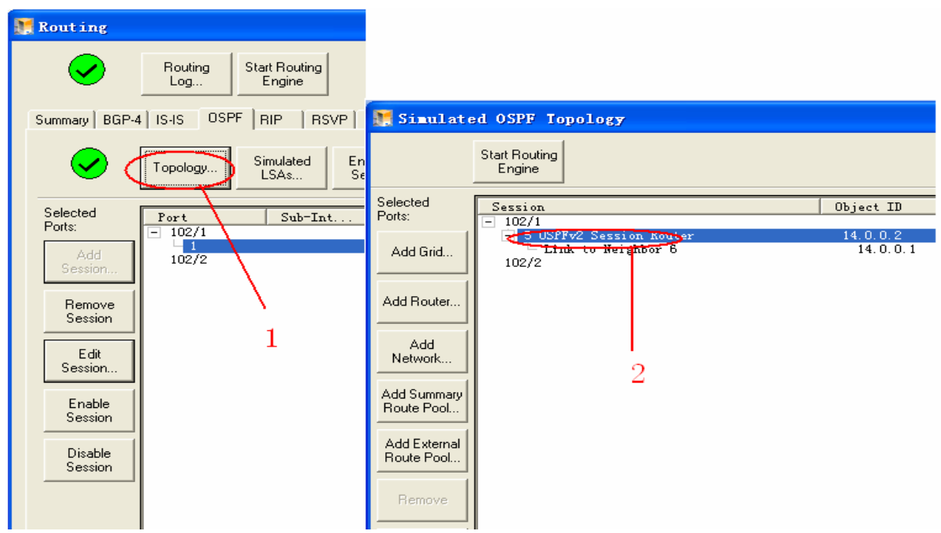


图9 构建网络拓扑

仿真窗口里你会看见一个Session Router，是模拟一个与被测设备SUT直接相连的路由器。在OSPF拓扑仿真窗口下，我们就可以模拟各种复杂大型的OSPF网络拓扑了。

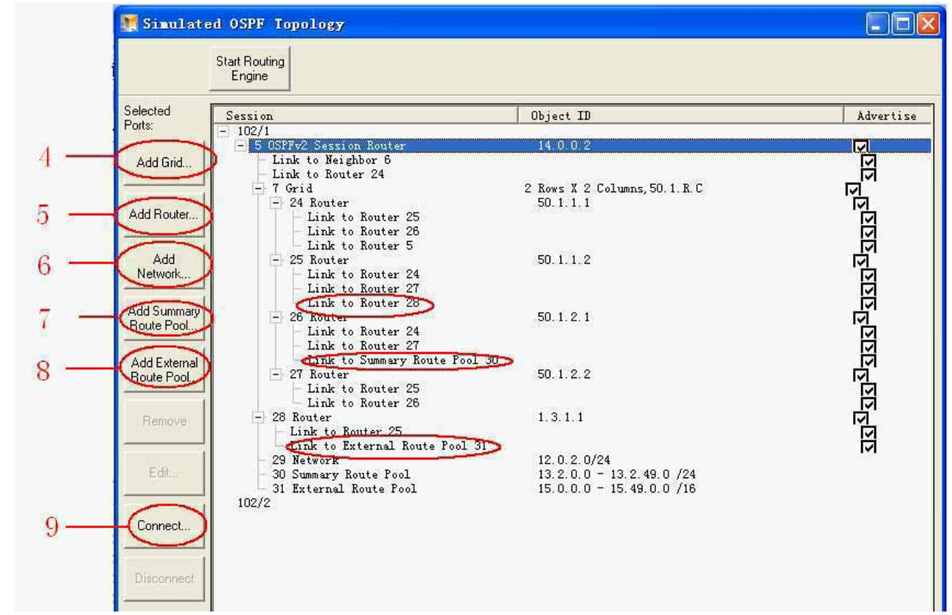


图10 OSPF拓扑仿真窗口

可以使用 ，，模拟一个内部互连的矩阵拓扑。如图11中，我们模拟了一个2×2的矩阵，将它第1行第1列的路由器与Session Router相连。

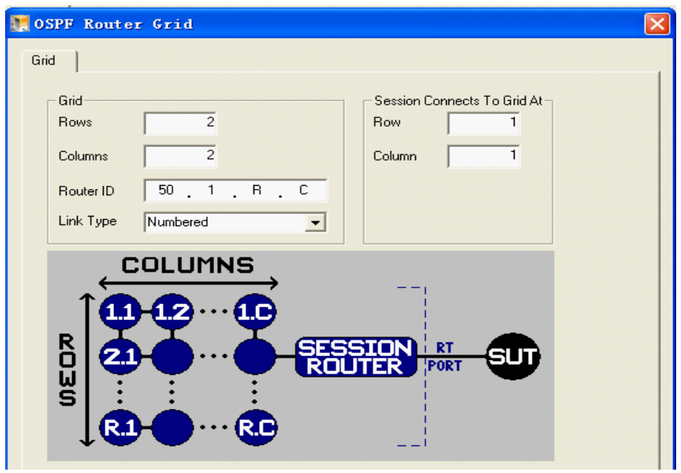


图11 创建Grid拓扑

图11中模拟的拓扑如图12示。

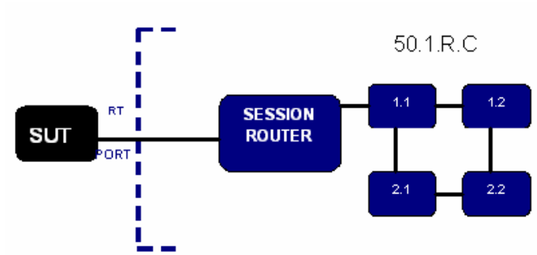


图12 Grid 拓扑

可以使用 Add Router... ，来模拟一个路由器，可以配置路由器的Router ID，路由器的类型等参数。

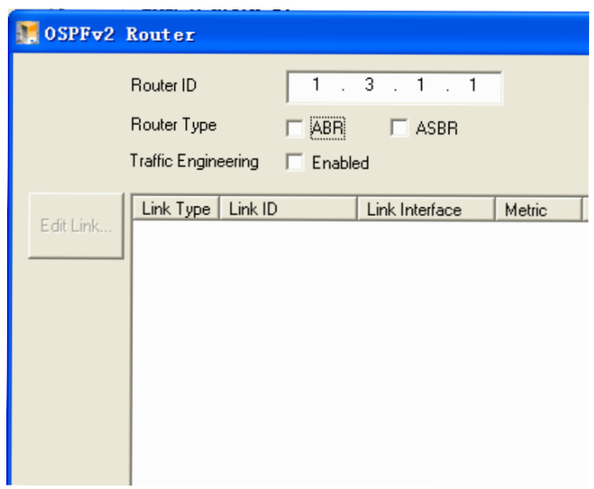


图13 创建一个Router

可以使用 Add Network... ，来加一个网段，注意这里不是添加Network LSA，因为RouterTester不能做DR。

可以使用 Add Summary Route Pool... ，来添加Summary LSAs，模拟其他区域的路由。

可以使用 Add External Route Pool... ，引入外部路由，需要注意如果是往NSSA区域添加外部路由，需要选上配置窗口下侧的NSSA，如图14。

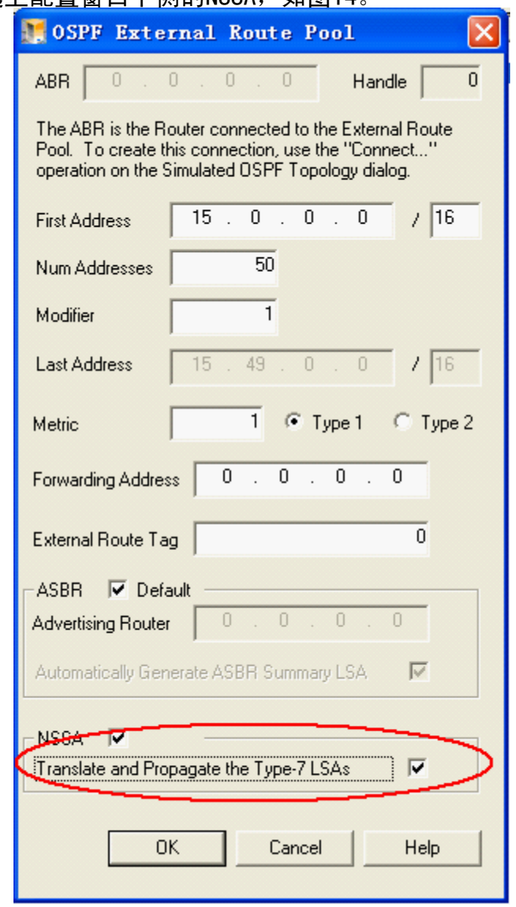


图14 引入外部路由

除了 Add Router... 模拟矩阵拓扑的是和ession Router相连的，其他几种拓扑都需要建立连接 Connect... ，将各种拓扑结构连接在一起。

如图15所示，从对话框左侧选择要连接的两个拓扑添加到右侧，图例中将Router ID为1.3.1.1的路由器连接到矩阵拓扑1行2列的路由器上。

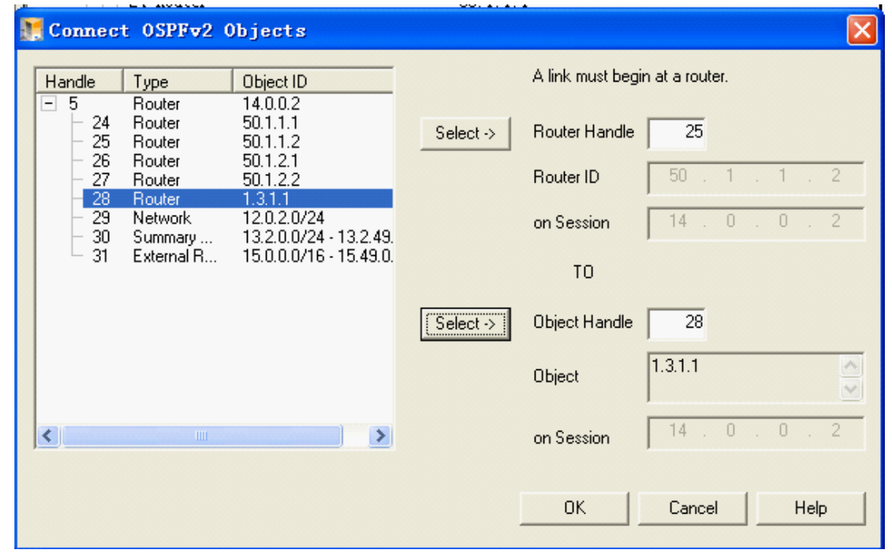


图15 建立连接

仿真LSAs

构建完网络拓扑，建立好连接，关闭仿真窗口，可以点击Routing对话框上侧的 来查看链路状态数据库，在LSAs仿真窗口可以根据需要添加、编辑和删除LSA，也可以选择是否通告某些LSA，也可以使用，造成LSAs的振荡。

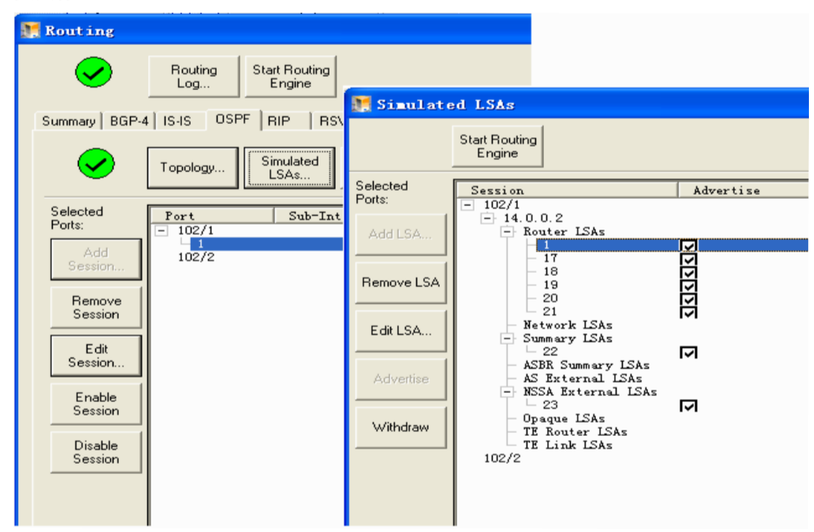


图16 LSAs仿真窗口

通告网络拓扑

点击仿真窗口上侧的 Start Routing Engine ，启动路由引擎，通告构建的OSPF网络拓扑，等待OSPF的状态变成Full。在路由器上可以看到打入的路由和建立的ospf邻居等，完成Router Tester的拓扑模拟功能。可以很方便的用于进行路由性能等方面的测试。

我们可以使用 Routing Log... 来实时监控跟踪路由引擎的活动，包括Tester和SUT之间的路由协议交换信息等。

我们还可以使用 OSPF Stats... 来实时监控测试端口和SUT端口之间的OSPF报文交换，记录交换报文数量。记录值只是针对当前的OSPF Session，如果当前OSPF Session关闭，则显示的是上次连接的信息。

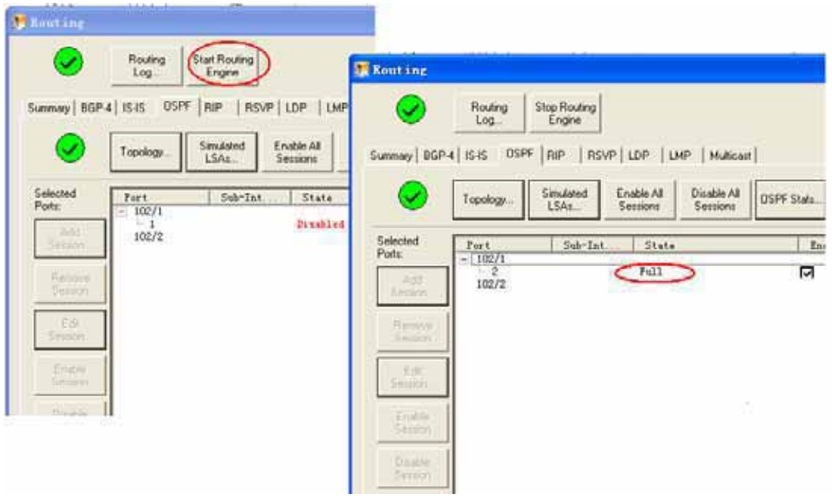


图17 启动路由引擎

注：启动路由引擎之后，你可以动态添加或删除OSPF session和各种拓扑结构。
现在我们就可以测试了。

构造验证数据流

完成OSPF网络拓扑结构的模拟，接下来就该构造数据流验证SUT是否学到路由了。
在Router Tester的OSPF测试里有一项功能，可以自动生成验证路由的数据流。那么怎样构造自动生成的数据流呢？
在OSPF拓扑仿真窗口里有Traffic Destination这个选项，见图18(a)，图中的Summary, External Route Pool在这一栏都可选，我们在这里选中了两个External Route Pool。

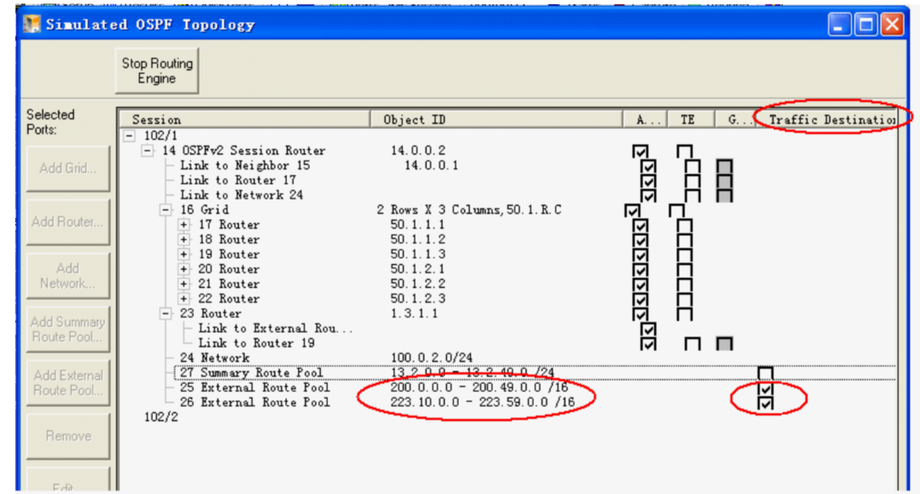


图18 (a) Traffic Destination 选项

现在我们来构造一个Traffic Mesh，当其他测试端口创建目的端口是该Session测试端口的Traffic Mesh时，见图18(b)。

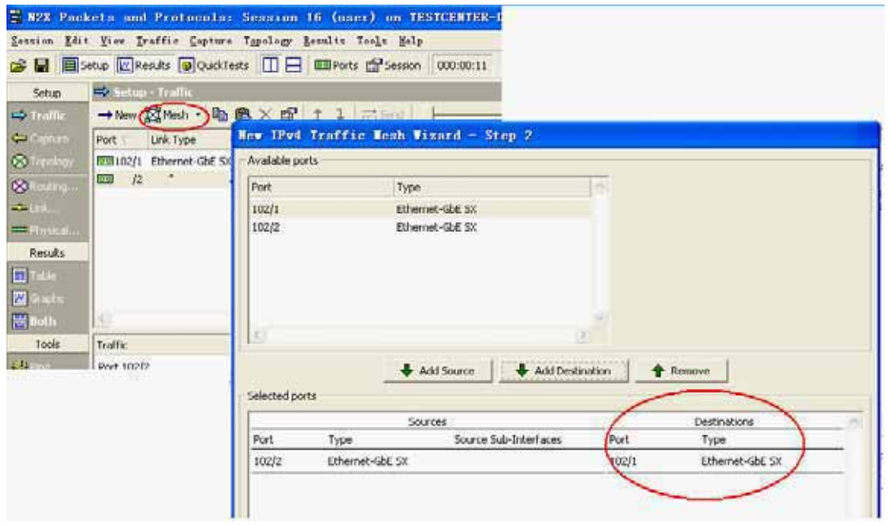


图18 (b) 构造一个Traffic Mesh

- Traffic Mesh会自动生成选中该选项的地址为目的地址的数据流，见图18(c)，比较图18(a)、(c)，(c)中的数据流的目的地址就是(a)选中的路由池的地址，这样验证路由的数据流就构造出来了。

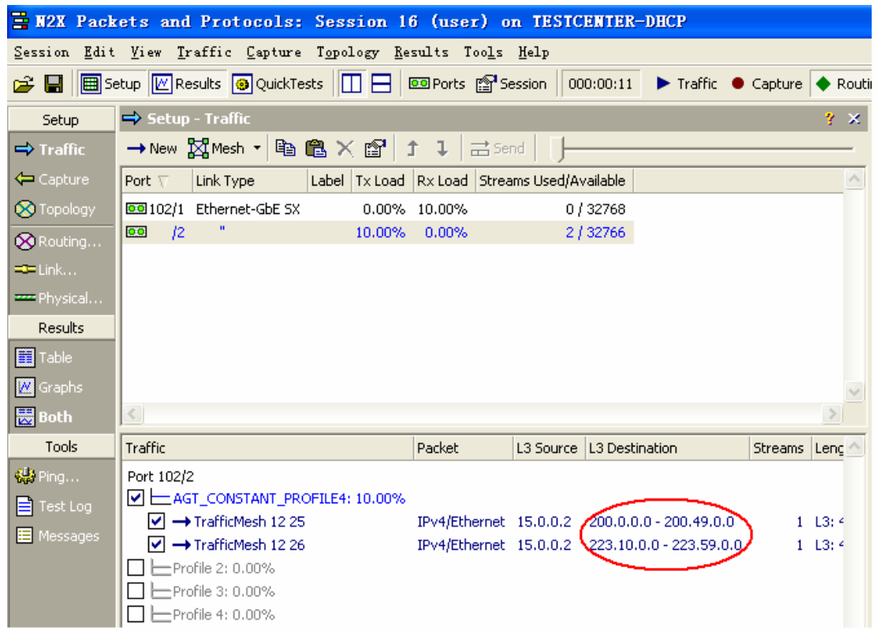


图18（c）生成的数据流

不过这项功能在OSPFv2里只适用于summary，external路由池，在OSPFv3里还适用于 inter-area prefix 路由池，其他的路由就得我们自己来构造合适的数据流来验证了。
构造完验证路由的数据流，我们就可以发送数据流了，点击主界面上侧的 ▶ Traffic ，查看结果，如果发送的数据流全部都可以正确收到，说明SUT的路由学习正确。

使用TRT测试OSPF

鉴定测试中心 张敬军

软件概述

TERAROUTING TESTER（TRT）是思博伦（SPIRENT）公司开发的一种基于SMARTBITS硬件系统的针对路由协议的功能测试及性能分析软件。TRT可以构造各种路由协议下的环境，包括BGP、RIP、OSPF等，并且可以同时模拟二到四层的简单业务流量，并且支持路由震荡，能够按照用户制定的规则制造不稳定的路由环境。

路由协议测试基本思想

对于测试方法学的理解有助于我们正确地设计测试及分析结果，能否将TRT的作用发挥到极致，在一定程度上取决于对于测试思想的理解，当然还有对于测试对象的理解。

路由稳定性测试：对于DUT执行路由查找动作、添加路由表项、根据路由表项转发报文能力的测试。

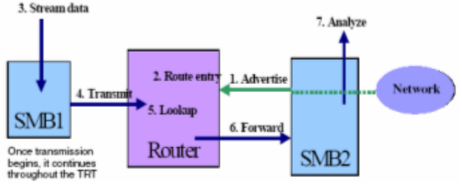


图1 路由稳定性测试

路由震荡测试：我们可以通过利用物理链路的UP/DOWN、连接（邻居关系等）的建立/断开、特定路由信息或者LSA的通告/撤销来制造震荡，测试设备的稳定性。

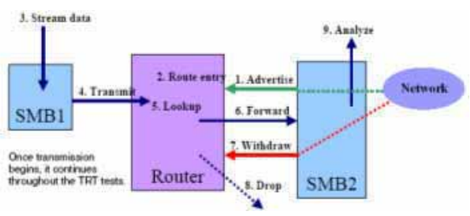


图2 路由震荡测试

路由恢复测试：对于DUT路由性能分析测试的关键在于对于DUT从路由波动中恢复的能力的考察。TRT可以帮助我们检验DUT如何在大规模的路由波动后恢复和处理路由信息及相关的转发流量。

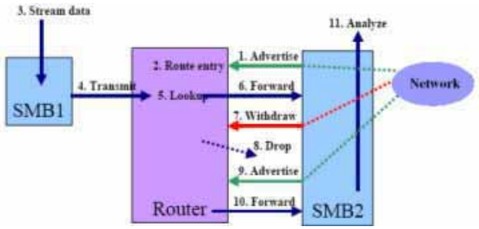


图3 路由恢复测试

路由切换测试：我们需要利用TRT来测试DUT的路由切换能力，在最优路由失效的时候一台性能良好的设备应该可以迅速有效的将流量从最优路径切换到次优路径上以尽可能减少丢包。

这种情况在实际网络中比较容易出现，因此基于这种场景的路由切换测试还是显得非常有意义的，也是我们测试的一个重点。

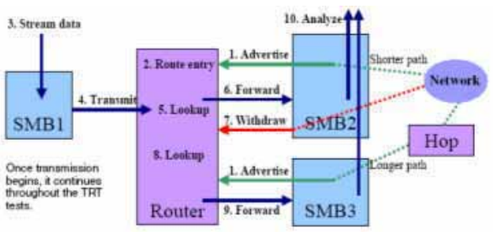


图4 路由切换测试

TRT基本操作

连接

确认好物理连接后，启动TRT图形界面，选择Tools > Setup Chassis Connections，点击Add IP按钮，键入测试仪IP地址及端口号（默认16385，不需修改）。选择Actions > Connect即可连接测试仪。

TRT用户界面

TRT用户界面如下：

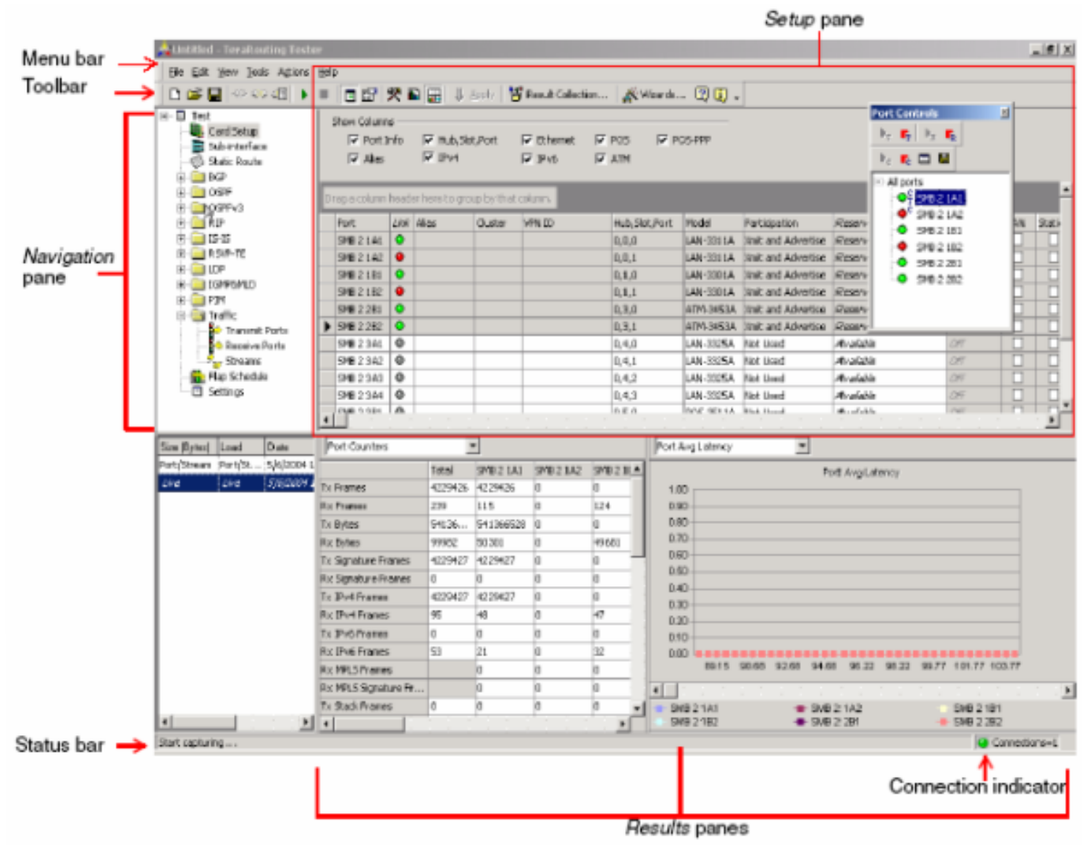


图5 TRT用户界面

端口设置

端口参数

点击导航界面的Card Setup，可以看到设置界面出现如下表格。

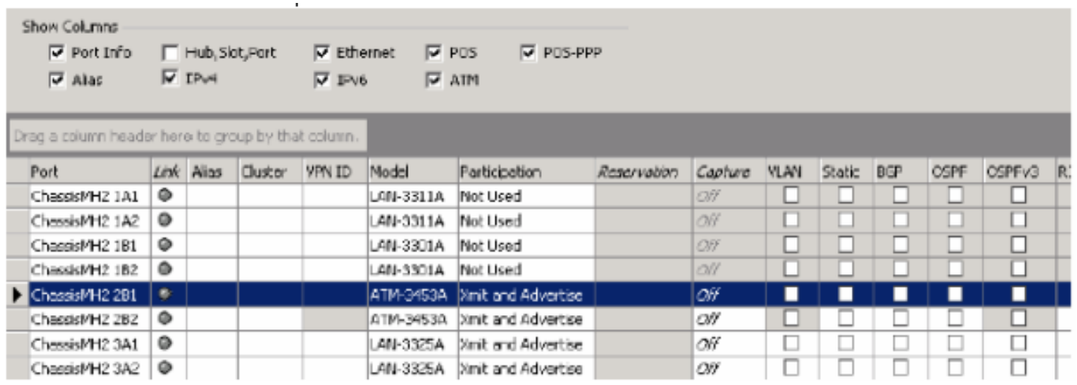


图6 Card Setup界面

我们挑选部分重要的参数加以说明：

Participation：端口参与测试的方式。Xmit Only表示端口仅发送流量，Advertise Only表示仅通告路由，Xmit and Advertise表示同时发送流量和通告路由。

Reservation：端口占用情况，In Use表示有其他人使用，Available表示无人使用，Reserved表示自己占用。

VLAN/RIP/OSPF/BGP… …：选择端口支持的协议。需要测试何种路由协议勾选即可。

SPEED/DUPLEX/FLOW CONTROL… …：以太网端口状态。

IPv4 ADDRESS/ IPv4 GATEWAY… …：端口地址配置。

子接口

以太网端口和ATM端口允许配置多个子接口，以太网端口利用VLAN划分子接口（所以配置子接口必须先勾选对应端口的VLAN。），ATM端口利用VPI/VCI分配子接口。

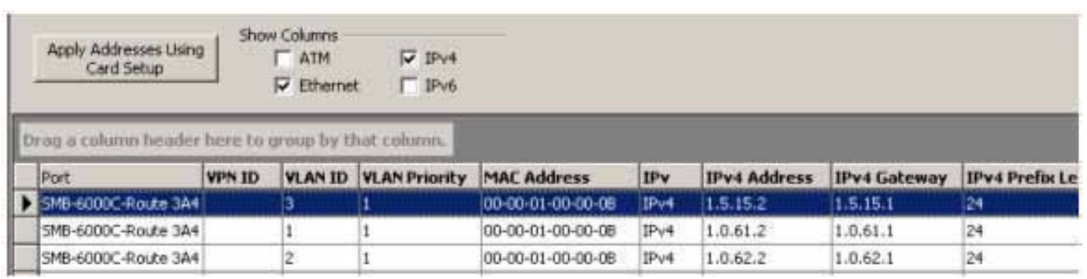


图7 子接口配置界面

这里举了一个子接口配置的例子，注意，和上面的配置界面图的配置没有关系。我们仅仅通过示意图来理解测试仪的一个端口模拟多个子接口的模型。DUT上同时存在多个VLAN时，我们需要模拟每个VLAN的路由邻居。

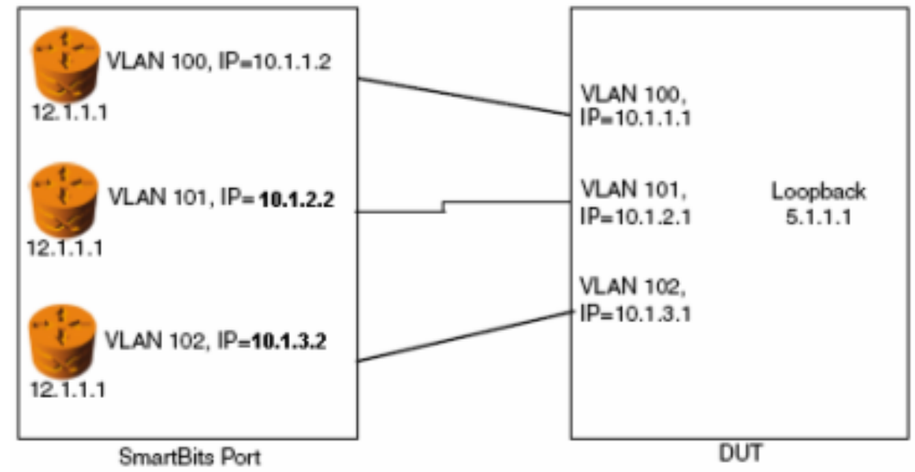


图8 Smartbits端口模拟多个以太网子接口

我们只需要右击配置界面就可以方便的增加或删除子接口，并在配置界面上修改子接口的各种参数。当需要添加较多子接口时，选择Duplicate即可。当我们批量修改有序变化的参数时，TRT提供了比较自由和省力的修改方式，如下图，我们选定需要变化的所有行，并且右击其中最上面一行的指定列（我们需要修改的那一列），可以看见弹出的菜单中包含如下项目：

Copy Down：被右击的这个单元格以下所有的行在该列的值，全部与该单元格取相同值。

Fill Increment：该列所有单元格的数值按照一定规律增长。我们看到以IP地址为例，我们有8种增长方式。其他参数增长方式可能略有不同。

Fill Decrement：该列所有单元格的数值按照一定规律减少。

Fill Random：该列所有单元格的数值随机变化。

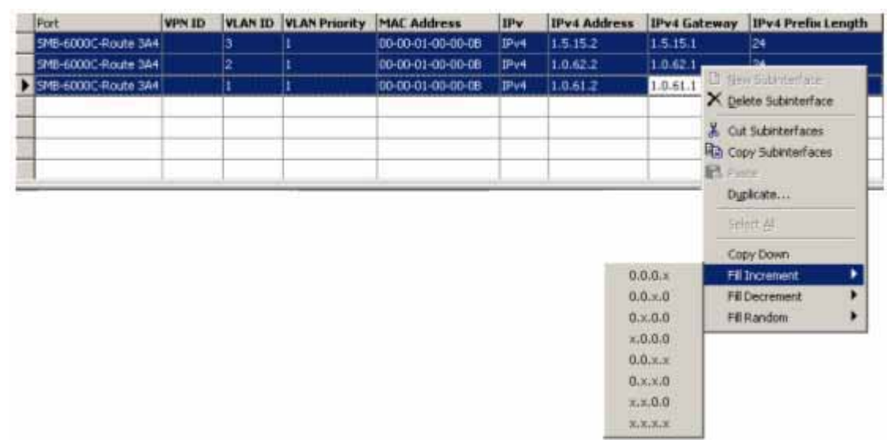
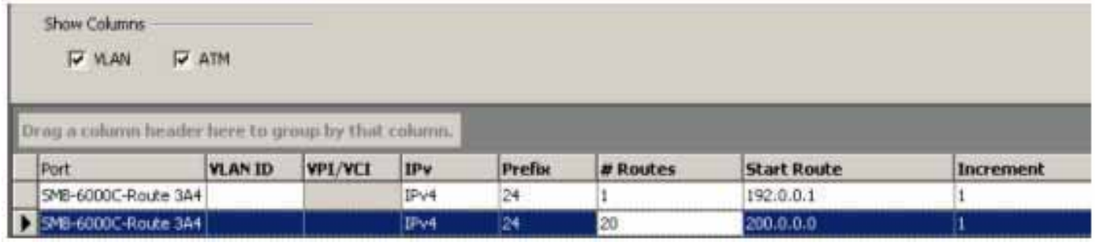


图9 批量修改参数

这种批量修改参数的方法很多地方都适用。后面将不再赘述。另外一种比较便捷的方式是添加多个子接口后，点击Apply Addresses Using Card Setup，TRT会自动配置子接口参数。

静态路由



Port	VLAN ID	VPI/VCI	IPv	Prefix	# Routes	Start Route	Increment
SMB-6000C-Route 3A4			IPv4	24	1	192.0.0.1	1
SMB-6000C-Route 3A4			IPv4	24	20	200.0.0.0	1

图10 静态路由配置界面

我们可以在一个端口上添加静态路由以辅助流量发送和接收，个人认为这个功能似乎没有什么必要，实际测试中也确实很少用到。

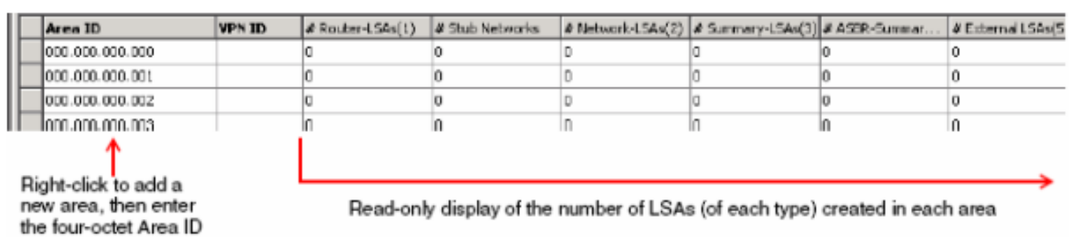
OSPF路由测试

测试设置概述

OSPF测试的设置分为三类：区域设置、邻居设置、LSA设置。通常配置也是按照这个顺序进行的。

设置区域

如上节所说，OSPF路由测试的第一步是建立和设置区域。



Area ID	VPN ID	# Router-LSAs(1)	# Stub Networks	# Network-LSAs(2)	# Summary-LSAs(3)	# ASBR-Summar...	# External LSAs(5
000.000.000.000		0	0	0	0	0	0
000.000.000.001		0	0	0	0	0	0
000.000.000.002		0	0	0	0	0	0
000.000.000.003		0	0	0	0	0	0

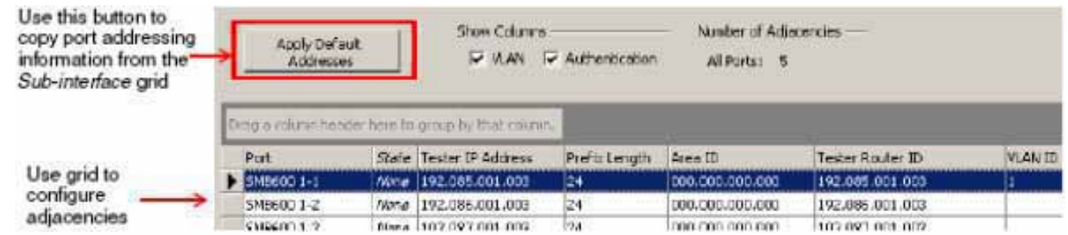
图11 设置OSPF区域

上图是一个区域设置的界面，右击此界面选择NEW可以建立新的区域，并且每添加一个区域，该区域ID可以自动增加。使用“Duplicate”可以批量添加区域。

设置邻居

可以使用两种方法添加邻居：使用上图的邻居设置界面或者使用图形化的拓扑编辑器。拓扑编辑器将在后面详细介绍。在邻居设置界面添加和删除邻居的方法同区域操作相同。

点击Apply Default Addresses按钮能够自动填写邻居的IP和VLAN设置。遵循一定的填写规则，如果没有配置子接口，板卡设置的IP地址等会填写到邻居列表中。我们注意到，自动填写的IP地址并不是我们在板卡设置或子接口配置时填入的地址，不过这并不影响我们的测试。



Port	State	Tester IP Address	Prefix Length	Area ID	Tester Router ID	VLAN ID
SMB600 1-1	None	192.085.001.003	24	000.000.000.000	192.085.001.003	1
SMB600 1-2	None	192.086.001.003	24	000.000.000.000	192.086.001.003	

图12 邻居设置界面

简单描述一下邻居的主要参数：

State：邻居当前状态，分为两个部分，前一部分是接口状态，后一部分是邻居状态，此栏只读。用于跟踪测试状态。

Tester IP Address：我们所模拟的邻居的接口地址，有两种方法指定这个地址，一个是手工填写，另一个是上面提到的Apply Default Addresses按钮。

Prefix Length：指定相应的子网掩码。默认值为24，有效值1—32。同样的，我们可以手工填写或者使用Apply Default Addresses按钮拷贝板卡设置的掩码。

Area ID：指定所属的区域，只能在已经配置好的区域中选取。默认为0。

Tester Router ID：我们所模拟的路由器的Router ID。如果不是手工指定，TRT通常使用Tester IP Address来作为路由器的Router ID。

VLAN ID：路由器的VLAN ID。取值1—4094，可以指定或自动获取，自动获取时使用子接口相应配置的值。

Hello Interval：发送Hello报文的间隔，默认10秒，取值范围0—65535。

Router Priority：路由器优先级，在选举中决定谁来充当DR和BDR。取值范围0—255，默认值0。注意这个默认值会使得TRT所模拟的路由器处于DR/Other状态。

Options：OSPF运行选项这里提供8个值，0表示Tos，1表示External Routing，2表示Multicast，3表示NSSA，4表示External Attribute，5表示Demand Circuit，6表示Opaque LSA，7表示Unused。默认值是1。

Advertise：路由器通告那些LSA，如果是Reachable LSAs，路由器将仅仅通告通过其可达的拓扑部分的LSA，如果是ALL LSAs，那么路由器将通告全部的本区域LSA，不管其是否可达，不管其是否与本路由器相连。比如下图，路由器1.1.12.3通告LSA时，如果选择Reachable LSAs，那么它将仅仅通告自己发出的LSA，但是如果选择ALL LSAs，那么TRT所模拟的所有本区域路由器发出的LSA将全部被路由器1.1.12.3通告，包括始发于1.5.14.3、1.1.13.3、1.1.11.2、1.1.11.3、1.1.11.4、1.1.11.5、1.1.11.6、1.11.11.2的LSA。如果图中两个不相连的孤立路由器1.11.11.6、1.11.11.7都与1.1.12.3同属一个区域的话，始发于这两台路由器的LSA也将被1.1.12.3通告。

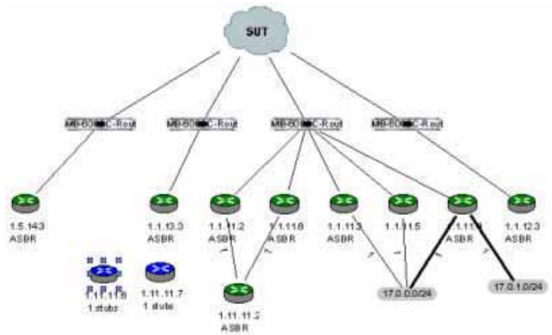


图13 Advertise选项说明

Network Type：路由器接口连接的网络类型。PtP指点对点链路，Broadcast表示广播链路，Native在以太网接口上表示广播链路，在POS或ATM口上表示PtP链路。
右击一个或多个路由器，在右键菜单选择“Start Router”即可启动路由模拟。

设置LSA

TRT可以模拟不同类型的路由器，所以同时允许模拟不同类型的LSA。TRT提供三种方法添加和修改LSA：在配置界面手工修改、使用LSA Generator和使用图形化拓扑编辑器，拓扑编辑器我们后面专门介绍。

LSA配置界面

下图是LSA配置界面，上方有按照区域查看LSA、显示选项和LSA Generator。按照区域查看LSA允许单独查看属于不同区域的LSA，默认显示0域。显示选项由两个复选框构成：Edit Links和Show All Header Fields，Edit Links仅对1、2类LSA的配置生效，它允许编辑路由器间的连接关系和连接类型等，Show All Header Fields允许查看LSA首部信息比如老化时间、序列号、校验和，这个选项对所有LSA生效。

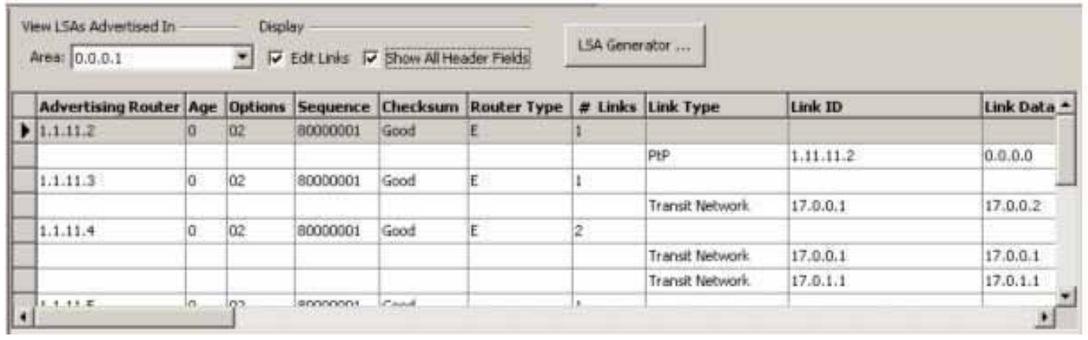


图14 LSA配置界面

同添加子接口或邻居路由器相同，添加或删除LSA也是使用右键菜单。

使用LSA GENERATOR

LSA Generator通过收集用户的输入数据，也就是我们指定的参数来自动生成每个路由器的LSA，用户可以根据需要修改其自动生成的LSA。
LSA Generator的配置对话框如下图。这个对话框共有六个标签，分别是Topology、Stub Networks、Summary Routes、External Routes、Opaque/TE、Other。一般我们按照这个顺序来填写参数，即首先设置Topology来指定区域的参数以确定拓扑。然后设置后面的四项Stub Networks、Summary Routes、External Routes、Opaque/TE决定产生怎样的LSA。最后使用Other确定IP地址范围即其他选项。

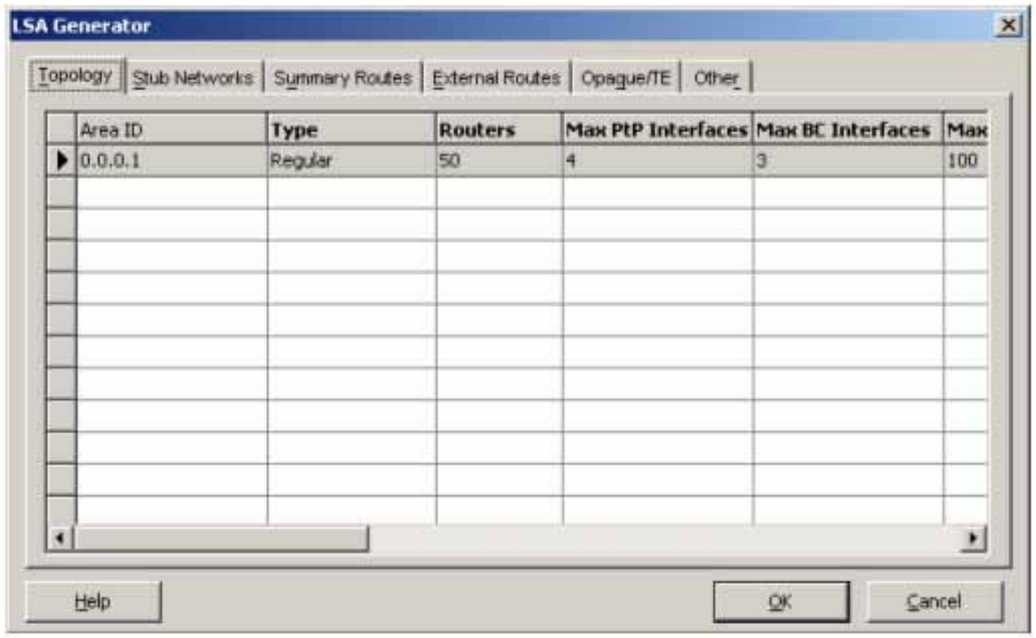


图15 LSA Generator配置对话框

在Topology部分，可以配置的参数有：
- Area ID：区域ID。这部分只读，由前面的邻居设置决定。
- Type：区域类型。
- Routers：本区域模拟的路由器的数量。
- Max PtP Interfaces：每台路由器最多可以拥有的PtP接口数量。
- Max BC Interfaces：每台路由器最多可以拥有的广播接口数量。
- Max Transit Networks：本区域最多可以拥有的穿越网络的数量。
- Max Routers per Transit Networks：每个穿越网络最多可以拥有的路由器数量。

Stub Network部分包含如下可配置参数：
- Area ID：区域ID。这部分只读，由前面的邻居设置决定。
- #Stub Networks：Stub网络个数。
- Distribution：只读，表示掩码分布方式。

Min Prefix Length：最短掩码长度。

Max Prefix Length：最长掩码长度。

Summary Routes包含如下参数：

Area ID：区域ID。这部分只读。

#Summary LSAs：Summary LSA的数量。

Distribution：只读，表示掩码分布方式。只能取Even，表示均匀分布。

Min Prefix：最短掩码长度。

Max Prefix：最长掩码长度。

%Duplicated：由不同端口模拟的重复的LSA占LSA总数的百分比。

External Routes包含如下参数：

Area ID：区域ID。这部分只读。如果选择区域类型为Stub（No Summary）或者Stub这部分配置将会被禁止。如果区域类型为NSSA（No Summary）或者NSSA的话，将使用7类代替5类LSA。

#External－Route LSAs：产生的外部路由数量。默认10000。

Distribution：掩码分布方式。可选两种方式，Even和Internet。Even表示Prefix Length在最大和最小值之间均匀分布，如果剩余部分路由不能均匀分布，则全部分布到24位。Internet使用指数方式，即先取一半路由给最长掩码，再取一半给次长掩码，如此重复至最短掩码，如果剩余部分路由则分配给最长掩码。默认使用Internet方式。

Min Prefix：均匀分布时最短掩码长度。默认1，可取1－32。

Max Prefix：均匀分布时最长掩码长度。默认32，可取1－32。

%Duplicated External－Route LSAs：由不同端口模拟的重复的外部LSA占外部LSA总数的百分比。默认10，可取0－100。

Other标签页与上述几个不同，它包含一个Options组和一个表示通告网段的IP Range组。

Prevent aggregation by assigning non-sequential networks to ports：选中表示测试仪端口模拟的路由条目以步长2增长，以避免DUT将TRT通告的路由聚合起来，减轻DUT和测试仪压力。不选表示路由以步长1增长。默认勾选。

Create loopback routes：选中表示TRT将在发布的Router LSA加上一个32位掩码的Stub网络，IP地址与Advertising Router字段相同。发送流量时TRT将忽略环回路由，所以勾选这个选项不会增加测试仪负担。

Create numbered point-to-point links：选中表示创建编号的点对点连接。

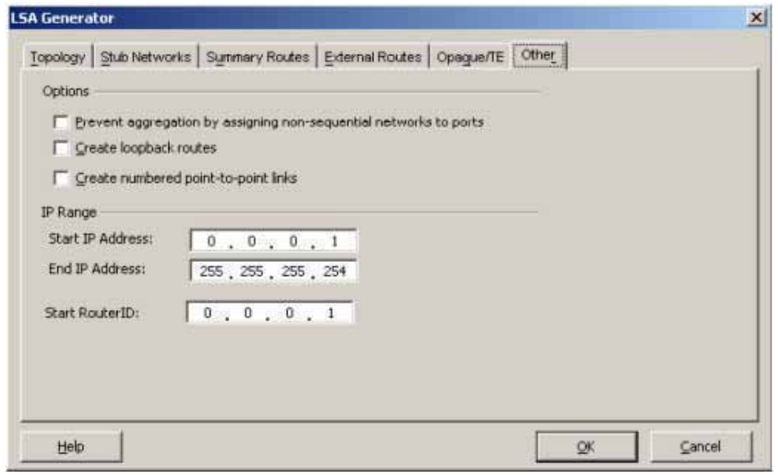


图15 LSA Generator配置对话框

我们填好了所有的参数，只需要点击OK就可以生成环境了。不过在生成环境的同时，TRT也会进行一系列的检查，防止不适当的参数导致错误。首先会检查Max值是否大于等于Min值，End值是否大于Start值。其次TRT还要求点对点连接和广播连接数量之和大于等于2小于等于100，如果点对点连接数小于2那么穿越网络的数量必须大于0。如果检查不通过，TRT会报错。如果配置中原有一些LSA，TRT会提示是否删除。最后TRT显示拓扑摘要，点击“OK”确认你的配置。

那么LSA Generator是按照怎样的规则建立这个测试拓扑的呢？我们来结合实例说明一下。比如我们输入了下表的参数希望LSA Generator使用两个Smartbits端口自动生成拓扑，拓扑中包含18台路由器每台路由器的点对点接口不超过3个，广播接口不超过1个，总共区域内的穿越网络不超过2个，每个穿越网络中不超过3台路由器。设置8个Stub网络，拓扑中共有8个Summary LSA，其中50%重复（这个地方可能会有歧义，我们放在后面单独解释）。

参数类型	参数	取值
Topology		
	Type	Regular
	# Routers	18
	Max PtP Interfaces	3 per emulated router
	Max BC interfaces	1 per emulated router
	Max Transit Networks	2 per area
	Max Routers Per Transit Network	3
Stub Network		
	Stub Networks	8
	Distribution	Even
	Min Prefix Length	24
	Max Prefix Length	24
Summary Routes		
	# of Summary LSAs	8
	% Duplicated	50
	Distribution	Even
	Min Prefix Length	24
	Max Prefix Length	24
External Routes		
	# of External LSA	0
Other		
	Start IP Address	1.0.0.1
	End IP Address	255.255.255.254
	Start Router ID	50.0.0.1
	Prevent aggregation...	unchecked

图16 Other Tab标签

当我们填写好参数点击OK之后，TRT会按照如下步骤开始生成拓扑。

第一步，为每端口生成根节点。如下图，每个端口都模拟一个和DUT直连的根路由器。

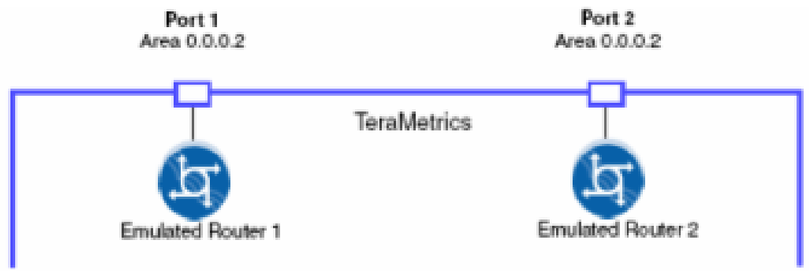


图17 生成根节点

第二步，为每一个根节点添加子节点。TRT使用轮询的方式向同一个区域每个端口添加子节点，目的是平衡节点数量。同时将保证根节点的各类接口数量不能超过限制，当穿越网络的数量到达最大值的时候，我们只能继续添加点对点接口。

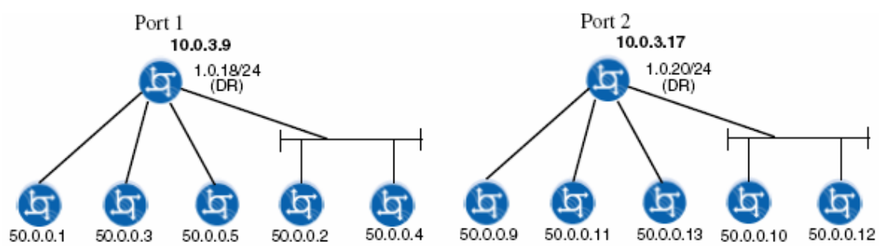


图18 添加子节点

添加子节点的过程将持续到如下两种情况发生：已添加的路由器总数到达最大值；根节点的接口数量到达最大值。本例中根节点会因为接口数量分别到达3和1而停止添加这一级子节点。此时18个路由器中的12个已经被添加到拓扑中，还有6个怎么办呢？我们可以继续添加。

第三步，添加下一级子节点。同样的这个过程也是均匀的添加到各个端口，至此，路由器的数量到达了指定的18台，我们不需要继续添加路由器了。同时广播接口和穿越网络也不允许添加了。还缺少什么？8个Stub网络。

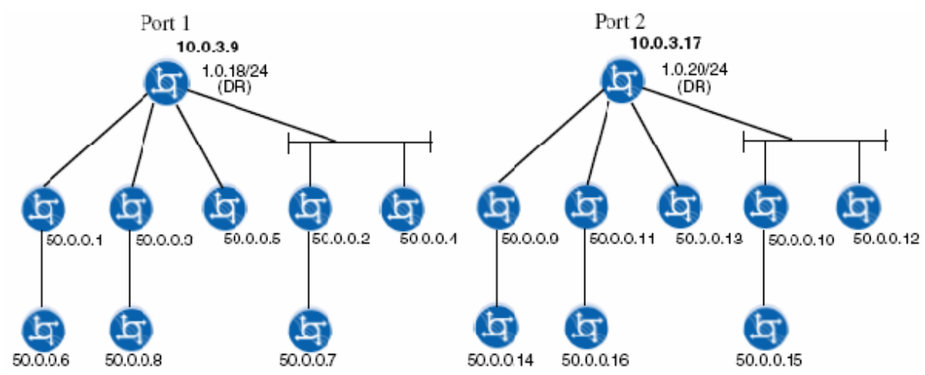


图19 添加下一级子节点

第四步，添加Stub网络。TRT只将Stub网络添加到没有子节点的路由器上，仍然以轮询的方式均匀添加到每个测试端口。

添加Stub网络时不仅物理位置均匀，而且我们还要考虑到我们设置的参数Min Prefix Length和Max Prefix Length以及分布方式，这些参数要求Stub网络的掩码也必须均匀分布，也就是每添加一个Stub网络，其掩码长度需要顺序变化，当然在本例中，由于Min Prefix Length和Max Prefix Length相等，所以只需使用24位掩码即可。

请注意，物理位置均匀指的是按照测试仪端口均匀分布，而不是按照路由器均匀分布。

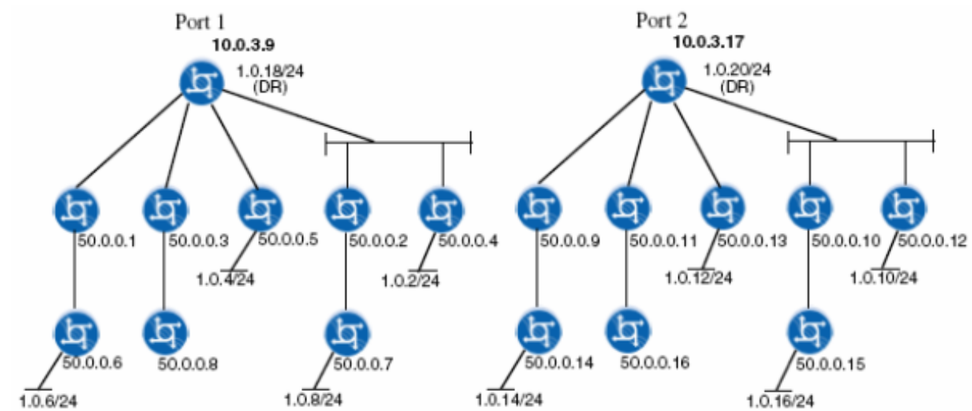


图20 添加Stub网络

最后一步是添加8条Summary LSA，同样遵循按照端口均匀添加的原则，通常添加在根节点。有几点需要注意：

产生Summary LSA的路由器属性将被设置为ABR。

如果参与测试的端口只有1个，那么TRT无法产生重复的External LSA或者Summary LSA，并且将%Duplicated参数视为0。

重复部分的LSA也将均匀分布。

路由器	Summary LSA Routes
10.0.3.9	Unique: 1.0.22/24, 1.0.24/24 Primary: 1.0.26/24, 1.0.28/24 Secondary: 1.0.34/24, 1.0.36/24
10.0.3.17	Unique: 1.0.30/24, 1.0.32/24 Primary: 1.0.34/24, 1.0.36/24 Secondary: 1.0.26/24, 1.0.28/24

表2 生成Summary LSA路由

我们看到全部的Summary LSA包括重复的那50％的LSA都按照规则均匀的分布在不同的端口上。

使用图形化拓扑编辑器

图形化的拓扑编辑器是一个能够帮助我们正确构造我们所需拓扑的很方便的工具，同时也具有预览自动生成的拓扑的功能。通过观察下图的拓扑编辑器界面，我们可以看到我们所模拟的环境可以被非常直观的显示出来。

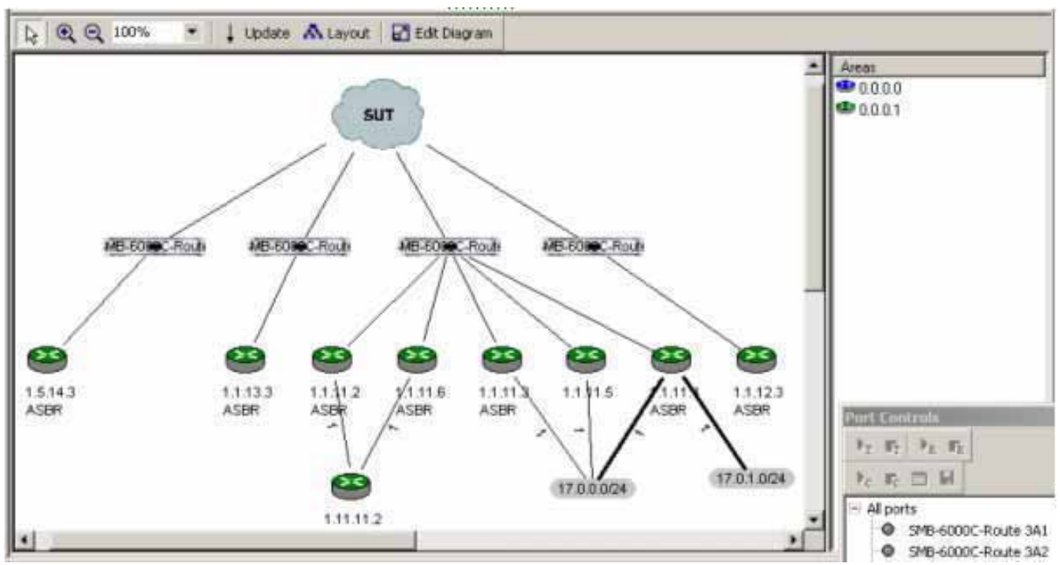


图21 拓扑编辑器

下面以表格形式对于拓扑编辑器的重要界面元素加以说明：

窗口元素	说明
Update	更新视图。之后视图变为只读属性。不能添加删除。
Layout	重新排列图标。
Edit Diagram	进入手工编辑界面。

表3 图形化拓扑编辑器窗口说明

点击“Edit Diagram”可以进入手工编辑界面，如下图：

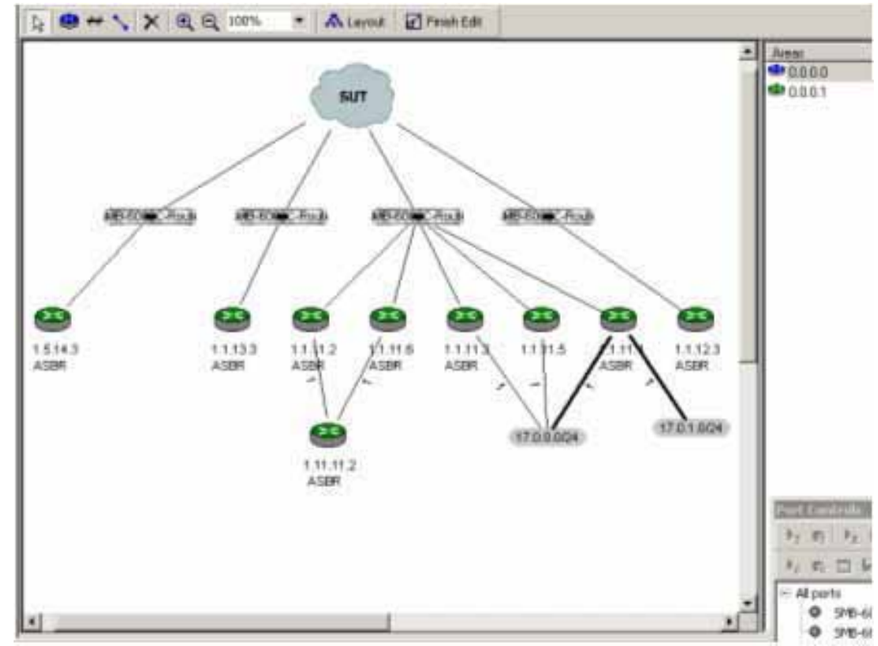


图22 手工编辑界面

手工编辑界面的窗口发生了一点变化，说明如下：

窗口元素	说明
	添加路由器，路由器的颜色由所属区域决定。
	添加网络。
	添加连接。
	删除图标。
Finish Edit	结束编辑。

表4 手工编辑界面说明

我们可以利用这些按钮灵活添加、删除和编辑拓扑图，拓扑图中重要包含如下元素。

图标	说明
SUT	SUT，待测系统。
SMB6000 1A1	Smartbits的Tera卡端口。
	路由器。
1.0.5.0/24	网络。
	没有DR的PtP连接。
	DR与网络的连接。

表5 拓扑图例

我们可以双击这些图形来设置路由器、网络和连接的属性。当然使用编辑器的前提是必须设置好区域和邻居。

流量设置

导航界面上点击Transmit Ports可以看到发送端口的配置，在表格中修改相应参数即可完成发送端口的配置。这部分仅仅是端口级的配置，流量的具体参数配置还需要Stream部分完成。

Drag a column header here to group by that column.

Port	Link	Transmit	Tx Mode	Seconds	# Bursts	Burst Size	Frame Size	Load
SMB-6000C-Route 3A1			Continuous			1	128	Per Stream
SMB-6000C-Route 3A2			# Seconds	10		1	128	10 %
SMB-6000C-Route 3A3			# Bursts		1000	1	128	10 %
SMB-6000C-Route 3A4			Continuous			1	128	Per Stream

图23 Transmit Ports设置

简要说明我们可以配置的参数：

Tx Mode：端口的流量发送方式。可取Continuous、#Seconds、#Bursts三种方式。

Continuous表示流量将持续发送直到用户手动停止。#Seconds表示流量将按照指定的持续时间发送流量，发送时间到达指定的时长即停止。#Bursts表示用户可以指定发送的Burst数，并按照这个数量决定继续发送或是停止。

Seconds：发送时间。只在选择#Seconds方式时有效，默认值为1。

Bursts：发送的Burst数。仅在选择#Bursts方式有效，默认值为1。

Burst Size：每个Burst包含的报文数。默认为1。

Frame Size：发送报文的长度，包含CRC。可以按照两种方式指定，Per Port和Per Stream。Per Port表示整个端口的所有流使用统一包长设置。Per Stream表示端口所包含的每一条流量单独配置包长。

Load：发送报文的速率。同样可以按照Per Port和Per Stream的方式设置。

导航界面上点击Streams可以编辑流量。Streams表格上方有3个按钮和一个状态栏，其中Traffic Wizard可以启动流量生成向导，Traffic Groups可以创建流量组，Edit用于编辑流量报文，状态栏显示当前数据流数量。

Traffic Configuration — Traffic Wizard... Traffic Groups...
Streams — Edit... Total Streams: 2 Active Streams: 2

Active	Name	Rate			Rx Port(s)	Fill Patter
SMB-6000C-Route 3A1						
☑	Stream1	1000 fps	Eth	IPv4 DA 114.0.0.14-114.0.62.126/28,SA...	Any	All Zeros
	Total:1					
SMB-6000C-Route 3A2						
SMB-6000C-Route 3A3						
SMB-6000C-Route 3A4						
☑	Stream2	1000 fps	Eth	IPv4 DA 111.0.0.14-111.0.62.126/28,SA...	Any	All Zeros
	Total:1					

图24 Streams配置界面

添加数据流可以使用两种方法：右击表格，使用右键弹出菜单添加并且直接在表格上编辑参数；或者使用Traffic Wizard来生成数据流。如果需要进一步修改报文字段，需要使用Edit功能。

点击Traffic Wizard按钮即可启动此向导第一步是配置数据流的发送端和接收端，非常方便的是，TRT允许我们根据各个端口所通告的网段来配置数据的起始终止点。我们只需要简单的勾选好一对端口并且添加到下面的Pairs列表中。同样可以配置每路由发送的Flow数，TTL，帧长，测试结果跟踪选项，以及IP地址合并选项。使用IP地址合并选项可以选择目的去往相邻网络的流是否合并，怎样合并，主要有三种方式：不合并、合并所有、按照路由通告的端口合并。

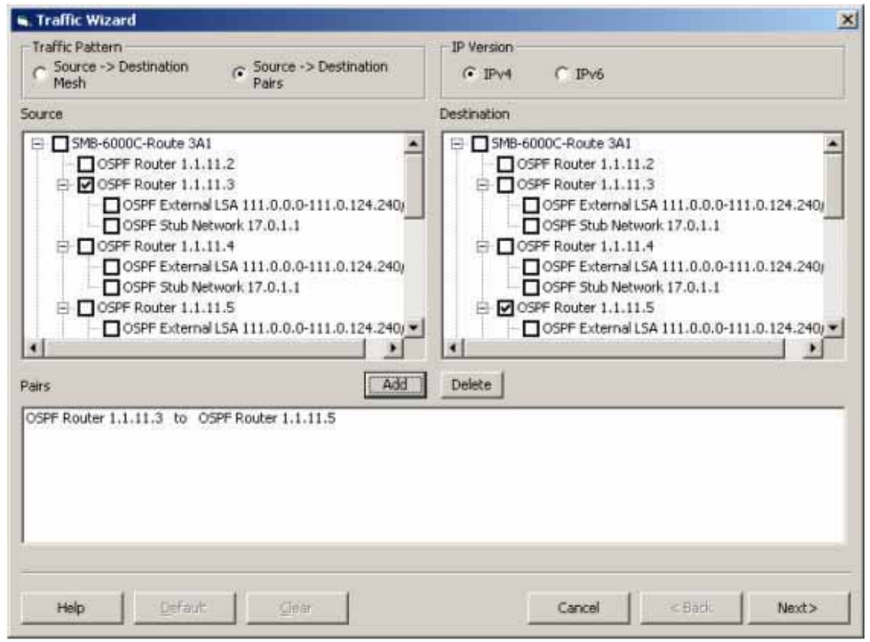


图25 Traffic Wizard配置界面一

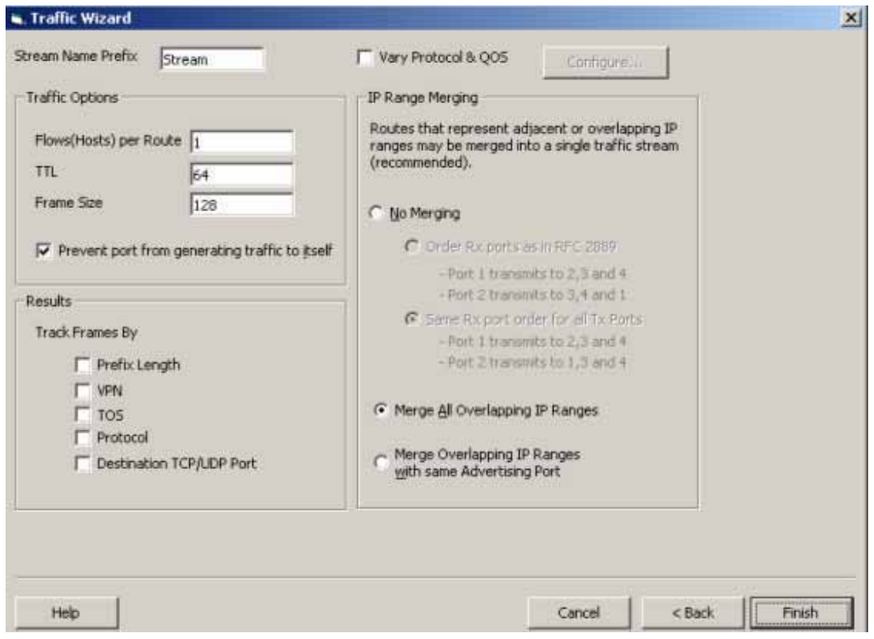


图26 Traffic Wizard配置界面二

完成这两步的设置，点击Finish即可看到生成的数据流。

当然，你可能对这些数据流并不完全满意，没关系，使用Stream Editor可以继续我们的工作。前面已经介绍过如何启动Editer，启动后的界面如下。

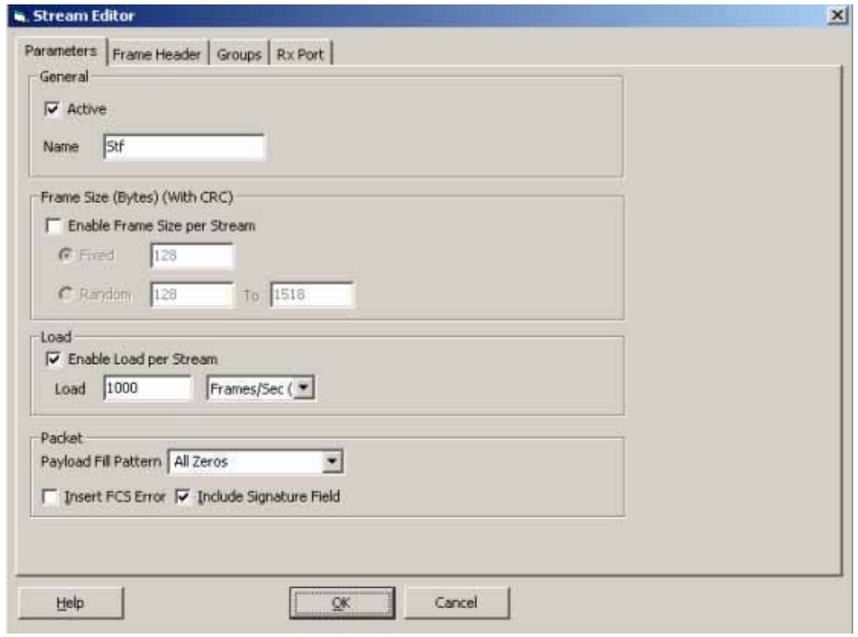


图27 Stream Editor配置界面

这个编辑器共包含4个标签页面：Parameters、Frame Header、Groups、Rx Port。

其中Parameters包含了这条流的基本属性，包括长度、速率和填充字符，这些同样可以在表格中修改。

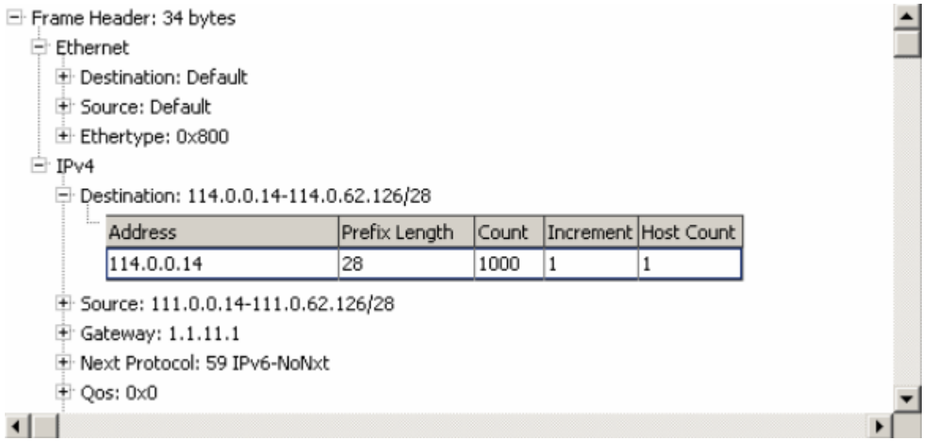


图28 报文编辑界面

Frame Header是编辑器的主要功能，所以特意截了上图辅助说明一下。很多字段都可以在这里很方便的被修改，并且大多数字段能够设置循环变化。
流量设置成功之后，右击发送端口，在弹出的菜单中选择Start Traffic即可发送流量，或者也可以在端口控制界面点击发送按钮。

路由震荡设置

实际测试当中我们往往需要模拟路由的不断变化来考验设备的性能和健壮性，TRT可以帮助我们自动实现路由的震荡。
通常编辑震荡需要至少3步：1、添加步骤；2、添加事件；3、修改参数（包含两步之间的间隔、震荡步骤的重新安排、震荡事件的修改等等）。TRT为我们设计了丰富的震荡事件。

测试结果

测试结果概述

我们查看测试结果界面，可以看到TRT为我们提供了丰富的测试结果统计，让我们能够比较全面的了解测试的运行情况。通过这个结果界面我们可以看到TRT能够提供Port Counter、Port Rate Statistics、Receive Rate、Average Latency、Tx/Rx Statistics、Traffic Stream Statistics、Traffic Group Statistics、Routing Protocol Statistics、Event Log、Throughput Test、Capturing Frames等信息。

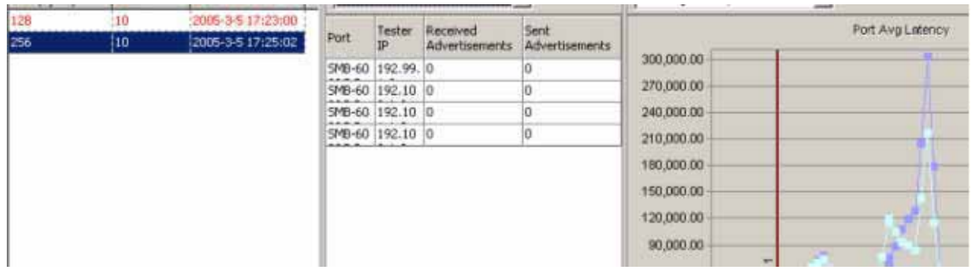


图29 结果界面

图表信息

TRT拥有比较强大的图表显示功能，相对于表格，图表能够提供更加直观的结果，而且可以记录每一时刻的流量状况。我们这里使用Port Rx Rate为例看看TRT如何以图表形式反映测试结果。

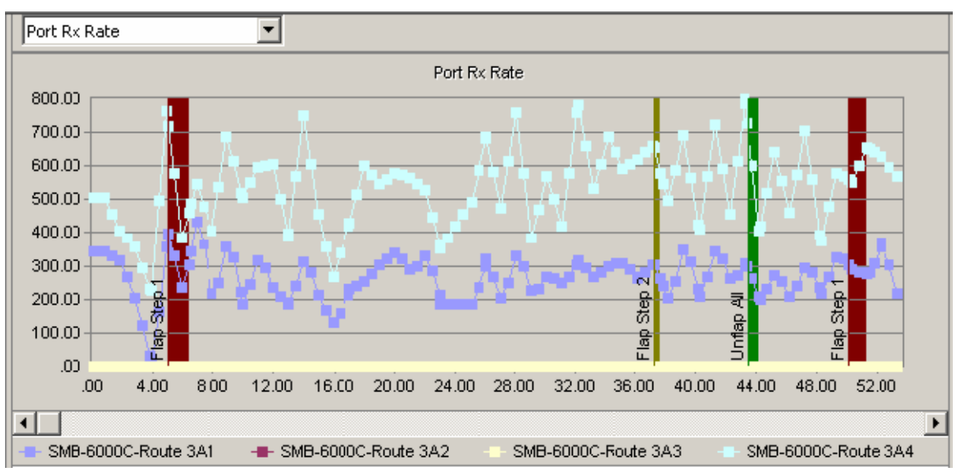


图30 Port Rx Rate图表

我们看到图表的X轴为时间轴，此处Y轴表示速率，以此TRT构建了对于每一个时间点的实时流量收发情况记录。通过使用不同颜色区分曲线，TRT在图表中可以同时显示多个端口的流量曲线。我们设置的路由震荡也被准确的标注在对应的时间点上，这样非常的便于我们分析流量异常的原因以及我们预置的动作对于测试结果的影响。美中不足的是，TRT不能够存贮比较长的流量记录，这可能会让我们在测试时有所限制。
右击图表，可以看到下拉菜单中还有很多让人感兴趣的功能，不过大多是增强图表显示的锦上添花的功能，和我们的实际测试关系不大。

报文捕获

通过报文捕获的方法来分析问题是我们经常使用的一种方法，TRT同样提供，不过TRT只能捕捉报文，却不能解析报文，如果查看报文内容反而需要借助其他工具，比较令人费解。

范例

测试描述

下面我们具体讲解一个比较简单的路由切换测试的例子。
测试要求：向DUT的两个接口通告同样的外部路由，但是由于路径开销值不同或者通告者的Router ID不同，所以存在一条优选路径。测试试图得到当优选路径的下一跳路由器失效后以及再次恢复时流量转发路径切换的时间（秒级）。
规定参数：外部路由数量设定为2000条，类型为E1，业务流量速率设置为10M，其他参数没有规定。

测试过程

确定测试拓扑

在满足测试要求的前提下，我们设计使用下面的拓扑来完成测试：

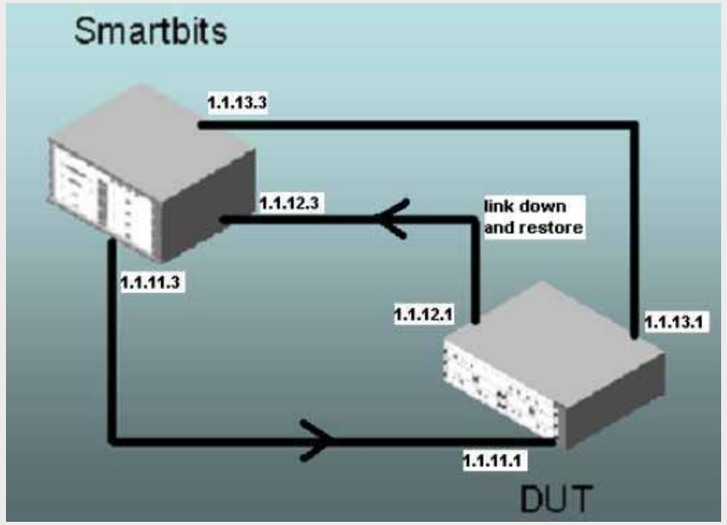


图31 测试拓扑图

DUT配置

在DUT端作如下配置：

```
interface Ethernet0/0/0
 ip address 1.1.11.1 255.255.255.0
interface Ethernet0/0/1
 ip address 1.1.12.1 255.255.255.0
interface Ethernet2/0/1
 ip address 1.1.13.1 255.255.255.0
ospf 1
 area 0.0.0.1
  network 1.1.11.0 0.0.0.255
  network 1.1.12.0 0.0.0.255
  network 1.1.13.0 0.0.0.255
```

TRT配置

首先是Smartbits的板卡设置，选定3个端口参与测试：端口3A1－3A3。3A1设定参与方式为Xmit and Advertise或者Xmit，3A2和3A3的参与方式为Xmit and Advertise或者Advertise，这里为方便起见，全部使用了Xmit and Advertise。勾选OSPF协议，配置接口的IP地址分别为1.1.11.3、1.1.12.3、1.1.13.3，网关为1.1.11.1、1.1.12.1、1.1.13.1，速率和双工方式为自协商，自协商设置为强制使用。

Port	Link	C	V	N	Participation	Reservation	C	S	E	OSPF	C	R	I	F	L	I.	P	IPv	IPv4 Addr...	IPv4 Ga...	IP...	MAC Address
SMB-6000C-Route 2B4	●			L	Not Used	Available	C			☐								IPv4	192.98.1.2	192.98....	24	00-00-01-00-00-07
SMB-6000C-Route 3A1	●			L	Xmit and Advert	Reserved	C			☑								IPv4	1.1.11.2	1.1.11.1	24	00-00-01-00-00-08
SMB-6000C-Route 3A2	●			L	Xmit and Advertise	Reserved	C			☑								IPv4	1.1.12.2	1.1.12.1	24	00-00-01-00-00-09
SMB-6000C-Route 3A3	●			L	Xmit and Advertise	Reserved	C			☑								IPv4	1.1.13.2	1.1.13.1	24	00-00-01-00-00-0A
SMB-6000C-Route 3A4	●			L	Not Used	Reserved	C			☐								IPv4	1.5.14.2	1.5.14.1	24	00-00-01-00-00-0B

图32 Card Setup设置

然后展开OSPF的配置，在Areas中建立区域1。建立区域1是因为DUT的接口属于区域1，没有其他特别的原因。

在Adjacencies中可以看到已经存在三个默认的邻居，我们只需要点击Apply Default Addresses按钮即可使得邻居的接口IP和Router ID改为我们指定网段的值，如果不是我们期望的IP地址，我们还可以手动修改。注意此处应设置邻居的区域ID为1。

Port	State	Tester IP Addr...	Pref...	Area ID	Tester Ro...	VPI/V...	Interf...	TE ...	TE ...	Hello Interval	Router Priority
SMB-60...	Backup/Full	1.1.11.3	24	0.0.0.1	1.1.11.3		1	0		10	100
SMB-60...	DR/Full	1.1.12.3	24	0.0.0.1	1.1.12.3		1	0		10	100
SMB-60...	DR/Full	1.1.13.3	24	0.0.0.1	1.1.13.3		1	0		10	100

图33 Adjacencies设置

为了使这三个邻居能够通告外部路由，我们必须设置他们为ASBR。可以通过两种方式做到这一点：使用拓扑编辑器双击路由器图标，在对话框中勾选“ASBR”；或者编辑邻居对应的Router LSA的Options选项为2。注意查看或编辑LSA时应确认已经选择了正确的区域。

接下来我们开始设置邻居路由器通告的外部路由，点击External－LSAs添加外部路由，我们这里选用了112.0.0.0-112.0.124.240/28共计2000个网段，邻居路由器1.1.12.3和1.1.13.3同时通告，但是通告的LSA的Metric值分别为1和10以确保3A2通告的路由较优。

Active	Advertis...	Age	Opt...	Sequence	Check...	L...	Pr...	# LSAs	Start LSA	End LSA	In...	Metric ...	Metric	Forwar...	Rout...
☑	1.1.11.3	0	02	80000001	Good	A...	28	2000	111.0.0.0	111.0.124.240	1	1	1	0.0.0.0	0
☑	1.1.12.3	0	02	80000001	Good	A...	28	2000	112.0.0.0	112.0.124.240	1	1	1	0.0.0.0	0
☑	1.1.13.3	0	02	80000001	Good	A...	28	2000	112.0.0.0	112.0.124.240	1	1	10	0.0.0.0	0

图34 External－LSAs设置

对于流量的设置，为了简便起见，我们使用Traffic Wizard建立流量勾选流量的源IP地址（范围）和目的IP地址（范围）即可。当然也可以使用添加流并且修改参数的方式做到，但是目的IP地址一定要是112.0.0.0-112.0.124.240/28，RX Port应该为Any Ports，流量速率为要求的10M。

路由振荡设置时应首先选中Enable Flapping复选框，本测试中我们需要模拟两种情况：一是3A2链路失效，二是3A2链路恢复。链路失效对应的事件为Break Physical Link，链路恢复的事件为Restore Physical Link，我们设置时间间隔为30秒以保证流量恢复。

此处为了考验设备在反复振荡情况下的稳定性，我们设置振荡模式为Continues即可。

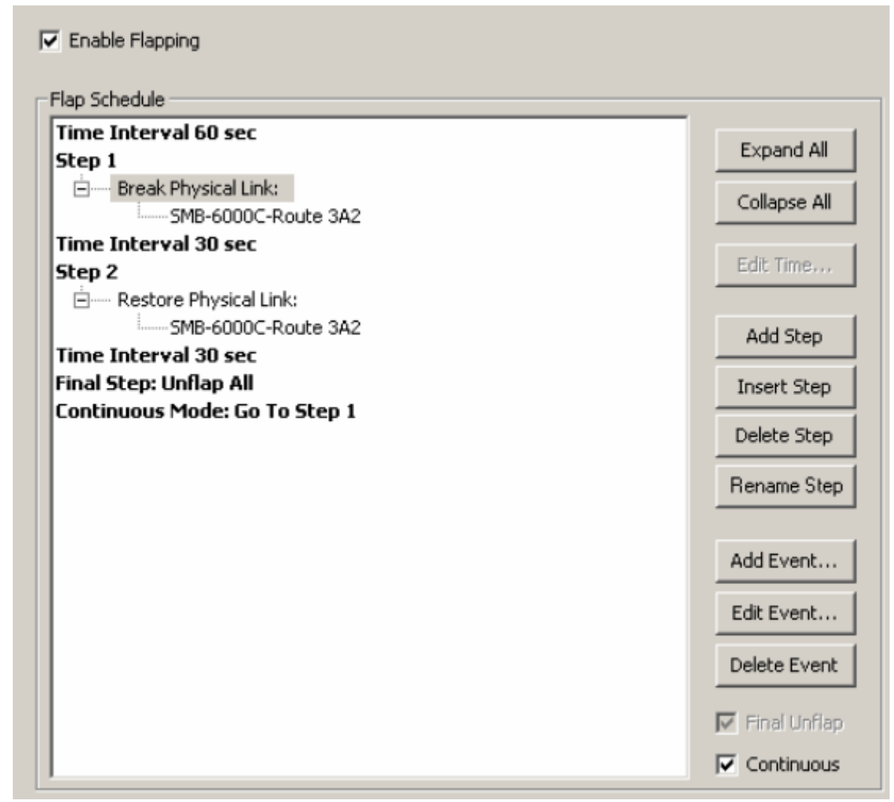


图35 Flap Schedule设置

至此我们完成了全部的配置。

测试运行和结果查看

点击RUN按钮开始测试， Test Configuration配置界面可以不必修改，直接使用默认配置。

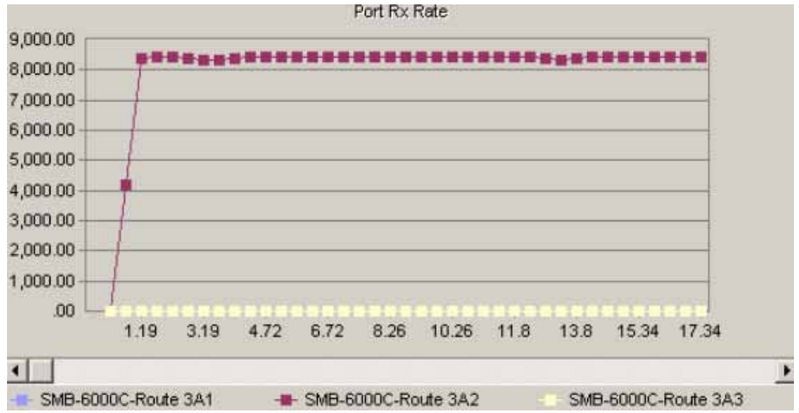


图36 流量接收状况一

如果运行过程中我们始终监控Port RX Rate图表我们可以清楚的看到设备在链路状态和路由表发生变化时的转发状态。上图就是设备路由学习完毕，测试仪开始发送流量时的流量接收情况，我们看到，此时3A2接收全部流量，3A3没有收到任何数据报文，因为我们设置的优选路径为3A2。流量速率为每秒8400个包左右，包长128，相当于10M左右流量。

我们看到在第59秒左右，发生了路由振荡事件1，3A2链路失效造成流量迁移。

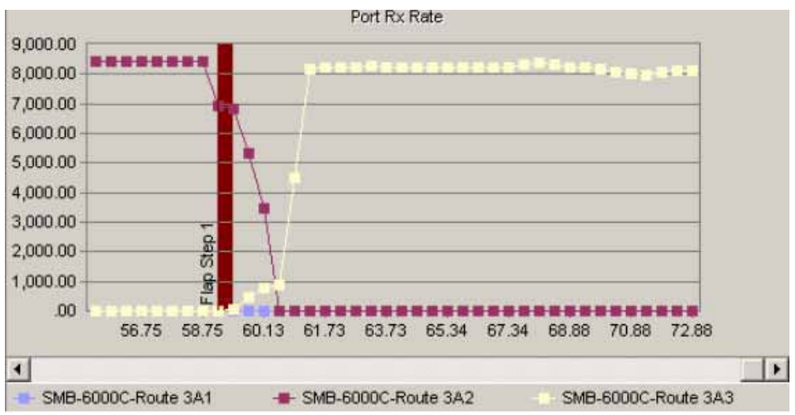


图37 流量接收状况二

3A2接收的流量变为0，3A3接收流量增加。从振荡事件发生到流量切换完毕（3A3流量速率达到峰值）大约耗时2秒左右。但是我们观察图62注意到切换后3A3接收到的流量比切换前3A2接收到的流量减少了，说明可能部分路由没有及时更新。

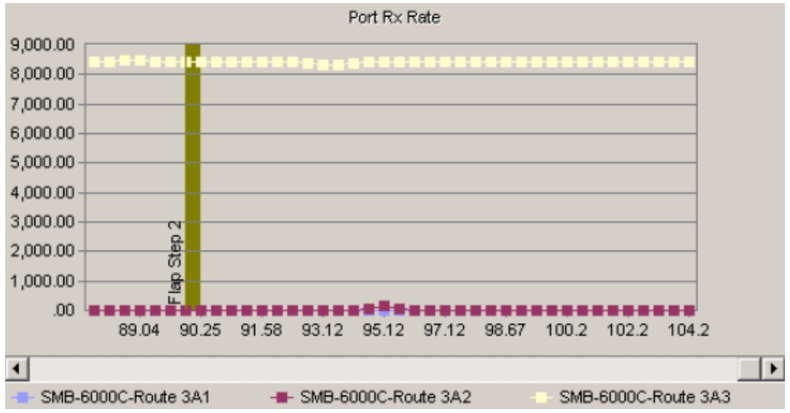


图38 流量接收状况三

在第90秒左右，TRT按照计划发生了振荡事件2，即3A2链路恢复，但是我们看到流量并没有很快的切换回来，仍然是由3A3链路转发报文。大约在第105秒也就是15秒后（见图64）3A2开始接收报文，第108秒左右3A2流速达到峰值。从事件发生到切换完毕，大约耗时18秒。

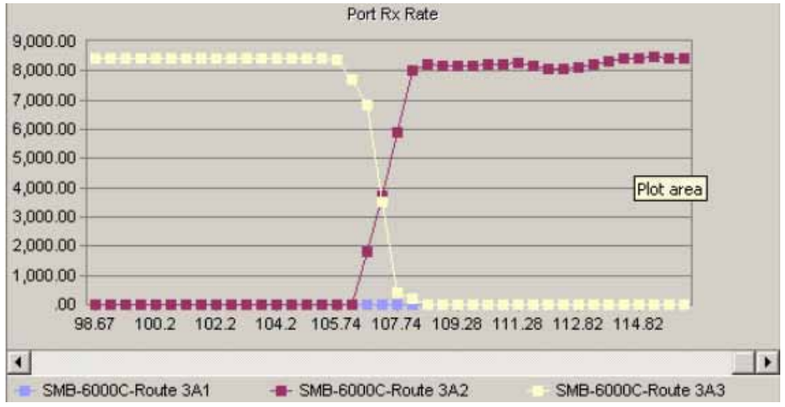


图39 流量接收状况四

以上就是我们的测试过程。如果需要更多本次测试的详细数据，我们可以在测试结束后查看TX/RX Statistics、Port Counters、Port Avg Latency等。

结束语

TRT的OSPF配置和测试通常建立在对于OSPF协议本身的足够了解的基础上，而TRT本身仅仅是一个比较好用的工具。本人水平有限，文中可能会出现一些模糊或者错误的描述，欢迎大家指正，您的宝贵意见对于以后本文档的升级和完善一定会有很大的帮助。

OSPF链路状态数据库解析

鉴定测试中心 蔡金龙

路由器之间通过Unnumbered串口连接的情况

- 网络拓扑图

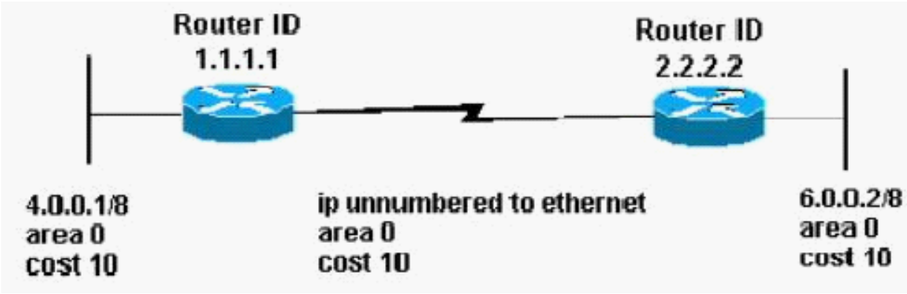


图1

- 相关配置

Router 1.1.1.1	Router 2.2.2.2
Current configuration: hostname r1.1.1.1 interface Loopback0 ip address 1.1.1.1 255.0.0.0 interface Ethernet2/0/0 ip address 4.0.0.1 255.0.0.0 interface Serial2/1/0 ip unnumbered Ethernet2/0/0 router ospf 1 network 4.0.0.0 0.255.255.255 area 0 end	Current configuration: hostname r2.2.2.2 interface Loopback0 ip address 2.2.2.2 255.0.0.0 interface Ethernet0/0/4 ip address 6.0.0.2 255.0.0.0 interface Serial2/1/0 ip unnumbered Ethernet0/0/4 router ospf 2 network 6.0.0.0 0.255.255.255 area 0 end

- 链路状态数据库解析

通过使用Show ip ospf database命令，我们可以看到在该网络类型中，链路状态数据库的情况。如下所示：

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID       ADV Router      Age      Seq#         Checksum       Link count
1.1.1.1        1.1.1.1        254     0x8000001A    0xA6FA              2
2.2.2.2        2.2.2.2        253     0x80000017    0x4858              2
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 279
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的
Router ID.
Advertising Router: 1.1.1.1
这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1
LS Seq Number: 8000001A
Checksum: 0xA6FA
Length: 48
Number of Links: 2
Link connected to: another Router (point?to?point)
这一行显示公布该LSA的路由器在point?to?point网络上的邻居为2.2.2.2
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 0.0.0.12
当接口配置为Unnumbered方式时，我们通常使用 MIB II IfIndex 值该地址的特点是以0作为起始位
Number of TOS metrics: 0
TOS 0 Metrics: 64
这行显示了连接两台路由器间链路的COST值为64
Link connected to: a Stub Network
这行显示这台路由器连接的STUB NETWORK
(Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
这行显示了从公布这条LSA的路由器到这个STUB NETWORK的COST值
r2.2.2.2#show ip ospf database router 2.2.2.2
```

```
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 295
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000017
Checksum: 0x4858
Length: 48
Number of Links: 2
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address: 0.0.0.10
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

- 计算最短路径

我们从路由器2.2.2.2 的链路状态数据库的显示来看一下最短路径树在这种网络环境下是如何计算的。首先，路由器2.2.2.2发现在他自己的链路状态数据库里发现路由器1.1.1.1是它的邻居。这时，它会检查路由器1.1.1.1发送过来的router LSA，如果在1.1.1.1发送的router LSA中发现路由器1.1.1.1也把自己作为邻居，两台路由器互为邻居，则此时路由器认为彼此可达，它们会把邻居发送的router LSA中的stub network记录到自己的路由表中。

在上面这个例子中，路由器2.2.2.2会把stub network 4.0.0.0记录到自己的路由表中，因为在路由器1.1.1.1发送的LSA中列出了该条目。

我们可以通过命令show ip route ospf，来看计算结果，如下所示：

```
r2.2.2.2#show ip route ospf\
O        4.0.0.0/8 [110/74] via 4.0.0.1, 00:06:01, Serial0/1/0
r1.1.1.1#show ip route ospf
O        6.0.0.0/8 [110/74] via 6.0.0.2, 00:06:16, Serial2/1/0
```

路由器之间通过分配地址的串口P TO P连接的情况

- 网络拓扑图

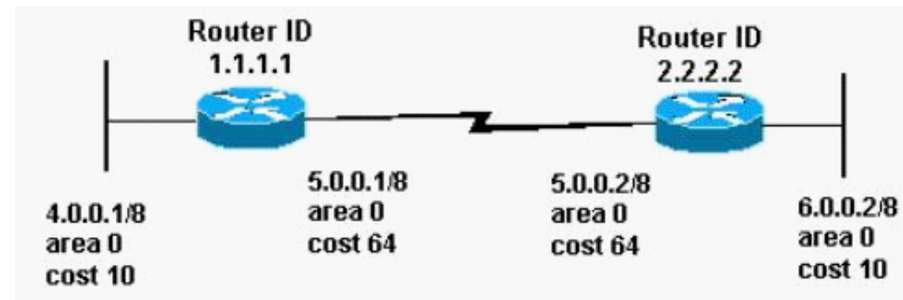


图2

- 相关配置

Router 1.1.1.1	Router 2.2.2.2
Current configuration: hostname r1.1.1.1 interface Loopback0 ip address 1.1.1.1 255.0.0.0 interface Ethernet2/0/0 ip address 4.0.0.1 255.0.0.0 interface Serial2/1/0 ip address 5.0.0.1 255.0.0.0 router ospf 1 network 4.0.0.0 0.255.255.255 area 0 network 5.0.0.0 0.255.255.255 area 0 end	Current configuration: hostname r2.2.2.2 interface Loopback0 ip address 2.2.2.2 255.0.0.0 interface Ethernet0/0/4 ip address 6.0.0.2 255.0.0.0 interface Serial2/1/0 ip address 5.0.0.2 255.0.0.0 router ospf 2 network 6.0.0.0 0.255.255.255 area 0 network 5.0.0.0 0.255.255.255 area 0 end

链路状态数据库解析

通过使用Show ip ospf database命令，我们可以看到在该网络类型中，链路状态数据库的情况。如下所示：

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID ADV Router Age Seq# Checksum Link count
1.1.1.1 1.1.1.1 522 0x80000002 0x96D3 3
2.2.2.2 2.2.2.2 401 0x80000003 0x9FC2 3
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 540
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
```

这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1

```
LS Seq Number: 80000002
Checksum: 0x96D3
Length: 60
Number of Links: 3
Link connected to: another Router (point?to?point)
```

这一行显示公布该LSA的路由器在point?to?point网络上的邻居为2.2.2.2

```
 (Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 5.0.0.1
```

这行显示路由器与邻居2.2.2.2所连接的接口地址为5.0.0.1

```
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

这行显示了连接两台路由器间链路的COST值为64

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK

```
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

这行显示了连接两台路由器间链路的COST值为64

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8

```
(Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

这行显示了从公布这条LSA的路由器到这个STUB NETWORK的COST值

```
r2.2.2.2#show ip ospf database router 2.2.2.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 424
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000003
Checksum: 0x9FC2
Length: 60
Number of Links: 3
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address: 5.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

计算最短路径

我们从路由器2.2.2.2 的链路状态数据库的显示来看一下最短路径树在这种网络环境下是如何计算的。首先，路由器2.2.2.2发现在他自己的链路状态数据库里发现路由器1.1.1.1是它的邻居。这是，它会检查路由器1.1.1.1发送过来的LSA，如果在1.1.1.1发送的LSA中发现路由器1.1.1.1也把自己作为邻居，两台路由器互为邻居，则此时路由器认为彼此可达，每一台路由器同时也会检查它们的本地邻居表，（我们可以通过show ip ospf neighbor命令来看相关信息）以确认它的接口地址和邻居的接口地址是否在同一个子网，如果在同一个子网，它们会把邻居发送的LSA中的stub network记录到自己的路由表中在上面这个例子中，路由器2.2.2.2会把stub network 4.0.0.0记录到自己的路由表中，因为在路由器1.1.1.1发送的LSA中列出了该条目。同时路由器1.1.1.1也将5.0.0.0/8做为stub network列在LSA中，但是路由器2.2.2.2不会将该子网加到OSPF路由表中，因为这是它的直连子网。（在路由器看来直连子网永远要比动态路由要好）

```
Router 2_2_2_2#show ip route ospf
O        4.0.0.0/8 [110/74] via 5.0.0.1, 00:09:26, Serial0/1/0
Router 1_1_1_1#show ip route ospf
O        6.0.0.0/8 [110/74] via 5.0.0.2, 00:00:49, Serial2/1/0
```

路由器之间通过P TO MP网络类型连接的情况

- 网络拓扑图

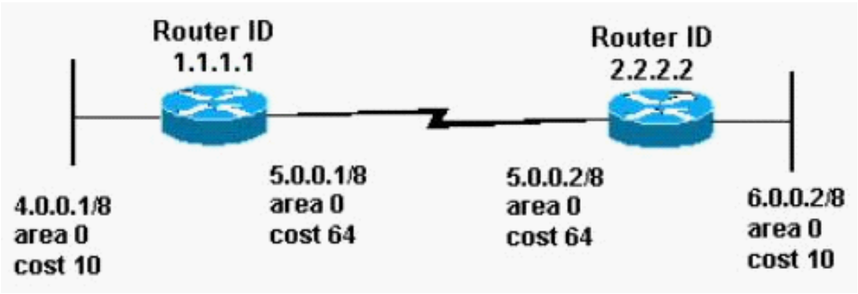


图3

- 相关配置

Router 1.1.1.1	Router 2.2.2.2
Current configuration: hostname r1.1.1.1 interface Loopback0 ip address 1.1.1.1 255.0.0.0 interface Ethernet2/0/0 ip address 4.0.0.1 255.0.0.0 interface Serial2/1/0 ip address 5.0.0.1 255.0.0.0 ip ospf network point-to-multipoint router ospf 1 network 4.0.0.0 0.255.255.255 area 0 network 5.0.0.0 0.255.255.255 area 0 end	Current configuration: hostname r2.2.2.2 interface Loopback0 ip address 2.2.2.2 255.0.0.0 interface Ethernet0/0/4 ip address 6.0.0.2 255.0.0.0 interface Serial2/1/0 ip address 5.0.0.2 255.0.0.0 ip ospf network point-to-multipoint router ospf 2 network 6.0.0.0 0.255.255.255 area 0 network 5.0.0.0 0.255.255.255 area 0 end

- 链路状态数据库解析

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum Link      count
1.1.1.1         1.1.1.1         206      0x8000000A      0x158C             3
2.2.2.2         2.2.2.2         206      0x8000000B       0x791             3
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 224
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的
Router ID
Advertising Router: 1.1.1.1
这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1
LS Seq Number: 8000000A
Checksum: 0x158C
Length: 60
Number of Links: 3
Link connected to: another Router (point?to?point)
这一行显示公布该LSA的路由器在P TO MP网络上的邻居为2.2.2.2，注意CISCO在此显示的并不是P TO
MP，而是P TO P。
 (Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 5.0.0.1
这行显示路由器与邻居2.2.2.2所连接的接口地址为5.0.0.1
Number of TOS metrics: 0
TOS 0 Metrics: 64
这行显示了连接两台路由器间链路的COST值为64
Link connected to: a Stub Network
这行显示了路由器在P TO MP网络上的接口地址子网
(Link ID) Network/subnet number: 5.0.0.1
(Link Data) Network Mask: 255.255.255.255
注意这里使用的子网，只有接口的地址被公布出去
Number of TOS metrics: 0
TOS 0 Metrics: 0
此处显示为0，是因为在此只公布了接口而没有公布整个子网！
Link connected to: a Stub Network
这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8
 (Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
```

```
Number of TOS metrics: 0
TOS 0 Metrics: 10
```
这行显示了从公布这条LSA的路由器到这个STUB NETWORK的COST值10

```
r2.2.2.2#show ip ospf database router 2.2.2.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 253
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 8000000B
Checksum: 0x791
Length: 60
Number of Links: 3
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address: 5.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.2
(Link Data) Network Mask: 255.255.255.255
Number of TOS metrics: 0
TOS 0 Metrics: 0
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

计算最短路径

我们从路由器2.2.2.2 的链路状态数据库的显示来看一下最短路径树在这种网络环境下是如何计算的。首先，路由器2.2.2.2发现在他自己的链路状态数据库里发现路由器1.1.1.1是它的邻居。这时，它会检查路由器1.1.1.1发送过来的LSA，如果在1.1.1.1发送的LSA中发现路由器1.1.1.1也把自己作为邻居，两台路由器互为邻居，则此时路由器认为彼此可达，每一台路由器同时也会检查它们的本地邻居表，（我们可以通过show ip ospf neighbor命令来看相关信息）以确认它的接口地址和邻居的接口地址是否在同一个子网，如果在同一个子网则它们会把邻居发送的LSA中的stub network记录到自己的路由表中

在上面这个例子中，路由器2.2.2.2会把stub network 4.0.0.0记录到自己的路由表中，因为在路由器1.1.1.1发送的LSA中列出了该条目。同时路由器1.1.1.1也公布了5.0.0.1/32这个子网，这是该路由器在P TO MP网络上的接口子网，路由器2.2.2.2也会将该子网加到自己的路由表中！（这和P TO P类型是不同的，回忆一下，PTOP网络是将接口的子网5.0.0.0/8公布出去，而不是仅接口地址！！）

```
Router 2.2.2.2#show ip route ospf
O      4.0.0.0/8 [110/74] via 5.0.0.1, 00:09:26, Serial0/1/0
O      5.0.0.1/32 [110/64] via 5.0.0.1, 00:09:26, Serial0/1/0
Router 1.1.1.1#show ip route ospf
O      6.0.0.0/8 [110/74] via 5.0.0.2, 00:00:49, Serial2/1/0
O      5.0.0.2/32 [110/64] via 5.0.0.2, 00:00:49, Serial2/1/0
```

路由器之间通过多路访问网络类型连接的情况

网络拓扑图

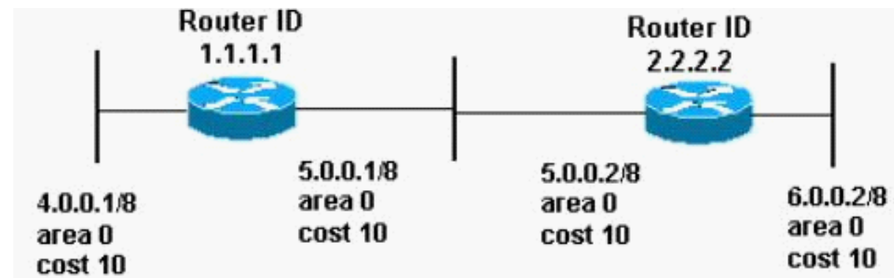


图4

相关配置

Router 1.1.1.1	Router 2.2.2.2

Current configuration:
 	
hostname r1.1.1.1
 	
interface Loopback0
  ip address 1.1.1.1 255.0.0.0
 	
interface Ethernet2/0/0
  ip address 4.0.0.1 255.0.0.0
 	
interface Ethernet2/0/2
  ip address 5.0.0.1 255.0.0.0
 	
router ospf 1
  network 4.0.0.0 0.255.255.255 area 0
  network 5.0.0.0 0.255.255.255 area 0
 	
end
	Current configuration:
	
	hostname r2.2.2.2
	
	interface Loopback0
	  ip address 2.2.2.2 255.0.0.0
	
	interface Ethernet0/0/4
	  ip address 6.0.0.2 255.0.0.0
	
	interface Ethernet0/0/2
	  ip address 5.0.0.2 255.0.0.0
	
	router ospf 2
	  network 6.0.0.0 0.255.255.255 area 0
	  network 5.0.0.0 0.255.255.255 area 0
	
	end

链路状态数据库解析

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID       ADV Router       Age        Seq#       Checksum       Link count
1.1.1.1         1.1.1.1         107      0x80000018    0x7966            2
2.2.2.2         2.2.2.2         106      0x80000015    0x6770            2
Net Link States (Area 0)
Link ID           ADV Router         Age        Seq#          Checksum
5.0.0.2            2.2.2.2           102      0x80000004      0x7E9D
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 147
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
```

这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1

```
LS Seq Number: 80000018
Checksum: 0x7966
Length: 48
Number of Links: 2
Link connected to: a Transit Network
```

这里显示路由器连接着一个Transit Network，这意味着需要选举DR、BDR

```
(Link ID) Designated Router address: 5.0.0.2
```

这行显示DR路由器连接到该网络的接口地址为5。0。0。2

```
(Link Data) Router Interface address: 5.0.0.1
```

这行显示路由器1。1。1。1连接到该网络的接口地址为5。0。0。1，所以很明显它不是DR

```
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

这行显示了连接两台路由器间链路的COST值为10

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8

```
 (Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

这行显示了从公布这条LSA的路由器到这个STUB NETWORK的COST值10

```
r2.2.2.2#show ip ospf database router 2.2.2.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 162
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000015
Checksum: 0x6770
Length: 48
Number of Links: 2
Link connected to: a Transit Network
(Link ID) Designated Router address: 5.0.0.2
```

这行显示DR路由器连接到该网络的接口地址为5.0.0.2

```
(Link Data) Router Interface address: 5.0.0.2
```

这行显示路由器1.1.1.1连接到该网络的接口地址为5.0.0.2，所以它就是DR

```
Number of TOS metrics: 0
TOS 0 Metrics: 10
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
r2.2.2.2#show ip ospf database network 5.0.0.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Net Link States (Area 0)
Routing Bit Set on this LSA
LS age: 182
Options: (No TOS?capability, DC)
LS Type: Network Links
Link State ID: 5.0.0.2 (address of Designated Router)
```

这行显示的是DR的接口地址，注意这里不使用ROUTER ID

```
Advertising Router: 2.2.2.2
```

这行显示的是产生这条LSA的路由器ROUTER ID

```
LS Seq Number: 80000004
Checksum: 0x7E9D
Length: 32
Network Mask: /8
```

这一行和上面的Link State ID一起表示该子网为5。0。0。0/8

```
Attached Router: 2.2.2.2
Attached Router: 1.1.1.1
```

计算最短路径

我们从路由器1.1.1.1 的链路状态数据库的显示来看一下最短路径树在这种网络环境下是如何计

算的。路由器1.1.1.1查找它自己的LSA，看它自己是否有一条链路连接到这个transit network。如果有则查找DR发布的NETWORK LSA，查找其中的attached routers列表。在这需要注意的是，所有的路由器通过该transit network能够互通。当路由器1.1.1.1看到自己的ROUTER ID出现在列表中时，它就可以开始计算通过这些邻居路由器所能到达的路由了！它查询路由器2.2.2.2的LSA，将子网6.0.0.0/8的路由加到它的路由表中，因为这个子网被路由器2.2.2.2的LSA公布。

```
r1.1.1.1#show ip route ospf
O        6.0.0.0/8 [110/20] via 5.0.0.2, 00:03:35, Ethernet2/0/2
r2.2.2.2#show ip route ospf
O        4.0.0.0/8 [110/20] via 5.0.0.1, 00:03:18, Ethernet0/0/2
```

OSPF的链路状态数据库在广播型多路访问网络和非广播型多路访问网络中是相同的。它们的主要差别在于邻居发现的状态机制不同，广播型多路访问网络邻居的发现是通过组播HELLO报文来实现的，而非广播型多路访问网络，邻居是静态配置的，并且是通过发送单播HELLO报文来维持邻居间的邻接关系。

路由器之间通过非广播型多路访问网络类型连接的情况

- 网络拓扑图

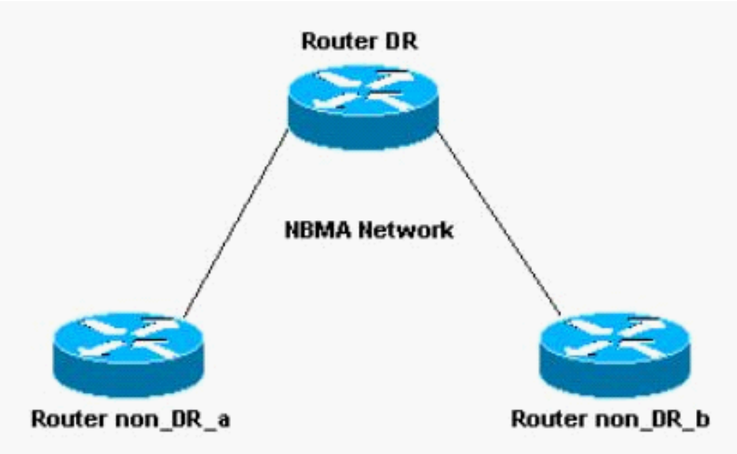


图5

- 实例分析

我们通过分析这个例子来要研究在这种网络类型下OSPF的工作方式。在该实例中的transit network内有三台路由器(Routers non_DR_a, non_DR_b, and DR)，他们通过NBMA类型的介质Frame Relay、ATM、X.25以hub and spoke网络拓扑进行互连。

在这种拓扑结构下，当路由器non_DR_a计算通过路由器non_DR_b可到达的路由时，它将non_DR_b的接口地址作为下一条地址，但是这两台路由器之间没有直接的虚链路，它们之间是不能互相通信的，这样OSPF装载到路由表里的路由的下一跳是不可达的，所以路由无效。我们可以通过静态配置frame?relay map解决这个问题，这样使得所有的邻居都可以通过VC访问到DR。如下所示：

```
interface Serial0
frame?relay map ip 1.1.1.1 700 broadcast
```

这一行是做到DR的映射

```
frame?relay map ip 1.1.1.2 700 broadcast
```

这一行是做到非DR邻居的映射

在这我们对OSPF与ISIS协议在该功能上做一个比较。ISIS协议中如果下一跳不是直连的邻居它不会将路由加到自己的路由表中，这意味着ISIS不能很好的工作在非广播型多路访问网络中，除非网络是全互连的。但是OSPF确可以通过MAP命令来实现。

OSPF的区域间路由

- 网络拓扑图

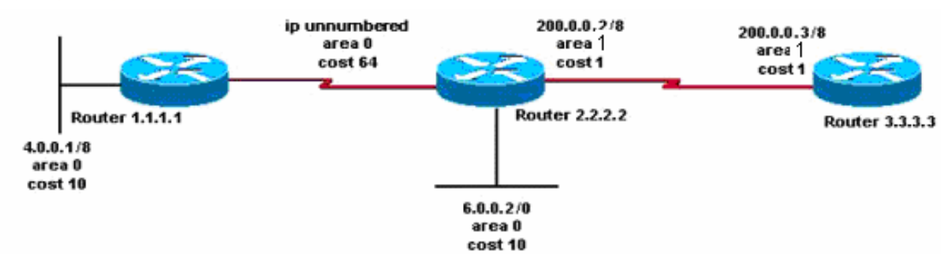


图6

- 相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Ethernet2/0/0
 ip address 4.0.0.1 255.0.0.0

interface Serial2/1/0
 ip unnumbered Ethernet2/0/0

router ospf 1
 network 4.0.0.0 0.255.255.255 area 0

end
```

Router 2.2.2.2

```
hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Ethernet0/0/4
 ip address 6.0.0.2 255.0.0.0

interface Serial0/1/0
 ip unnumbered Ethernet0/0/4

interface ATM1/0.20 point-to-point
 ip address 200.0.0.2 255.0.0.0

router ospf 2
 network 6.0.0.0 0.255.255.255 area 0
 network 200.0.0.0 0.255.255.255 area 1

end
```

Router 3.3.3.3

```
Current configuration:

hostname r3.3.3.3

interface Loopback0
 ip address 3.3.3.3 255.0.0.0

interface ATM2/0.20 point-to-point
 ip address 200.0.0.3 255.0.0.0

router ospf 2
 network 200.0.0.0 0.255.255.255 area 1

end
```

链路状态数据库解析

在此案例中路由器2.2.2.2为ABR路由器，在他的链路状态数据库里有两个区域的信息，所以我们把它作为最佳分析对象。

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID          ADV Router          Age         Seq#       Checksum     Link count
1.1.1.1          1.1.1.1             697         0x80000040 0x5A21       2
2.2.2.2          2.2.2.2             696         0x80000045 0xEE82       2
Summary Net Link States (Area 0)
Link ID          ADV Router          Age         Seq#       Checksum
200.0.0.0        2.2.2.2             352         0x80000001 0x2546
Router Link States (Area 1)
Link ID          ADV Router          Age         Seq#       Checksum     Link count
2.2.2.2          2.2.2.2             351         0x8000000B 0xCA9D       2
3.3.3.3          3.3.3.3             354         0x80000006 0x71F7       2
Summary Net Link States (Area 1)
Link ID          ADV Router          Age         Seq#       Checksum
4.0.0.0          2.2.2.2             689         0x80000001 0xFFE6
6.0.0.0          2.2.2.2             700         0x80000001 0x63C1
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 773
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
```

这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1

```
LS Seq Number: 80000040
Checksum: 0x5A21
Length: 48
Number of Links: 2
Link connected to: another Router (point?to?point)
```

这里显示的是路由器1.1.1.1和路由器2.2.2.2互为邻居

```
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 0.0.0.12
```

接口为Unnumberd所以这里显示接口的地址为0开头的地址

```
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

这行显示了连接两台路由器间链路的COST值为64

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8

```
(Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
r2.2.2.2#show ip ospf database router 2.2.2.2
```

```
OSPF Router with ID (2.2.2.2) (Process ID
2)
Router Link States (Area 0) ?> following is
the router LSA for 2.2.2.2 in area0 LS age:
789
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000045
Checksum: 0xEE82
Length: 48
Area Border Router
这里显示路由器2.2.2.2为ABR路由器
Number of Links: 2
有两条链路连接到AREA0
Link connected to: another Router (point?to
?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address:
0.0.0.10
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
Router Link States (Area 1)
下面显示的是AREA1的LSA
LS age: 445
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 8000000B
Checksum: 0xCA9D
Length: 48
Area Border Router
Number of Links: 2
Link connected to: another Router (point?to
?point)
(Link ID) Neighboring Router ID: 3.3.3.3
(Link Data) Router Interface address:
200.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 200.0.0.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
r2.2.2.2#show ip ospf database router
3.3.3.3
OSPF Router with ID (2.2.2.2) (Process ID
2)
Router Link States (Area 1)
LS age: 465
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 80000006
Checksum: 0x71F7
Length: 48
Number of Links: 2
Link connected to: another Router (point?to
?point)
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address:
200.0.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 200.0.0.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
ABR路由器使用summary LSA在区域间公布，我们
可以通过命令show ip ospf database summary
<link?state id> 来查看该类型的LSA.
r2.2.2.2#show ip ospf database summary
200.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 0)
LS age: 487
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 200.0.0.0 (summary Network
Number)
由ABR将 200.0.0.0/24子网公布到AREA 0
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0x2546
Length: 28
Network Mask: /24
TOS: 0 Metric: 1
r2.2.2.2#show ip ospf database summary
4.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 1)
LS age: 840
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 4.0.0.0 (summary Network
Number)
由ABR将 4.0.0.0/8子网公布到AREA 1
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0xFFE6
Length: 28
Network Mask: /8
TOS: 0 Metric: 74
r2.2.2.2#show ip ospf database summary
6.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 1)
LS age: 861
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0 (summary Network
Number)
由ABR将 6.0.0.0/8子网公布到AREA 1
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0x63C1
Length: 28
Network Mask: /8
TOS: 0 Metric: 10
```

计算最短路径

我们以路由器3.3.3.3为例来分析一下最短路径是如何计算的。

首先路由器3.3.3.3查看它自己的路由表，发现路由器2.2.2.2是它的一个邻居，然后它查看路由器2.2.2.2的LSA，如果发现路由器2.2.2.2也把自己作为邻居，则路由器3.3.3.3认为它们彼此可达。同时每台路由器也会检查它本地的邻居表以确认邻居的接口地址是否和自己的接口地址在相同的子网。（注意：如果接口是Unnumbered的，那不会进行地址检查）如果接口都在相同的子网里面，则路由器会将邻居发布的ROUTER LSA中列出的stub networks记录到自己的路由表里面。在本例中路由器2.2.2.2的LSA中只公布了200.0.0.0/24这一条stub networks信息，而该子网是路由器3.3.3.3的直连子网，所以不会记录到路由表中，取而代之的是一条直连路由。检查完所有在AREA1 中的ROUTER LSA后，路由器开始检查链路状态数据库中的summary LSAs，发现有子网4.0.0.0/8 和6.0.0.0/8的两条summary LSAs，如果此时路由器知道如何到达公布这条LSA的路由器的接口，则它会把这些子网加到自己的路由表里面，因为公布这条LSA的路由器是2.2.2.2，所以路由器3.3.3.3知道如何到达，因此会将这两条路由加到路由表里面。路由的度量值是路由器到达公布此LSA的路由器的度量与LSA中公布的度量值之和。

下面的输出显示了根据上面所述方法生成的各个路由器的路由表：

```
r1.1.1.1#show ip route ospf
O IA      200.0.0.0/24 [110/65] via 6.0.0.2, 00:09:00, Serial2/1/0
O          6.0.0.0/8 [110/74] via 6.0.0.2, 00:14:41, Serial2/1/0
r2.2.2.2#show ip route ospf
O          4.0.0.0/8 [110/74] via 4.0.0.1, 00:09:16, Serial0/1/0
r3.3.3.3#show ip route ospf
```

```
O IA      4.0.0.0/8 [110/75] via 200.0.0.2, 00:09:27, ATM2/0.20
O IA      6.0.0.0/8 [110/11] via 200.0.0.2, 00:09:27, ATM2/0.20
```

OSPF如何将外部路由引入到多个区域

- 网络拓扑图

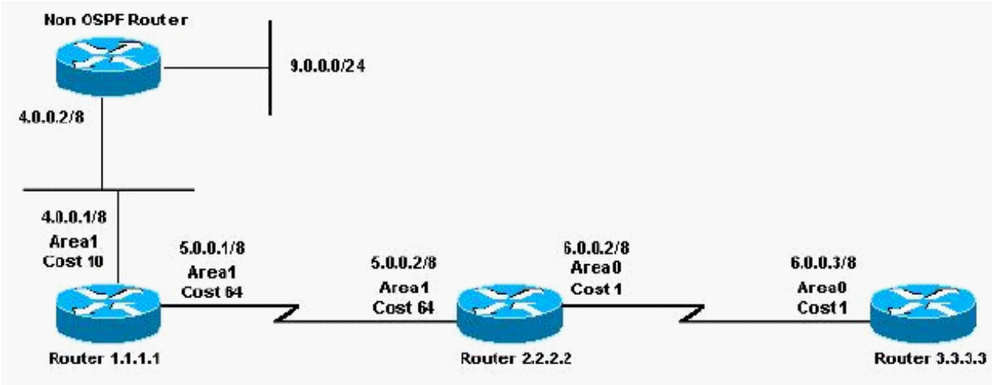


图7

- 相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Serial2/1/0
 ip address 5.0.0.1 255.0.0.0

interface Ethernet2/0/0
 ip address 4.0.0.1 255.0.0.0

router ospf 4
 redistribute static metric 5 metric-type 1
 network 5.0.0.0 0.255.255.255 area 1

ip route 9.0.0.0 255.0.0.0 4.0.0.2

end
```

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 network 6.0.0.0 0.255.255.255 area 0

end
```

Router 3.3.3.3

```
Current configuration:

hostname r3.3.3.3

interface Loopback0
 ip address 3.3.3.3 255.0.0.0

interface ATM2/0.20 point-to-point
 ip address 6.0.0.3 255.0.0.0

router ospf 2
 network 6.0.0.0 0.255.255.255 area 0

end
```

链路状态数据库解析

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)

Router Link States (Area 0)
Link ID         ADV Router      Age     Seq#       Checksum    Link count
2.2.2.2         2.2.2.2         93      0x80000020  0xCD0B      2
3.3.3.3         3.3.3.3         1225    0x8000000D  0x9057      2

Summary Net Link States (Area 0)
Link ID         ADV Router      Age     Seq#       Checksum
4.0.0.0         2.2.2.2         73      0x80000001  0xFFE6
5.0.0.0         2.2.2.2         1651    0x80000006  0x8466

Summary ASB Link States (Area 0)
Link ID         ADV Router      Age     Seq#       Checksum
1.1.1.1         2.2.2.2         74      0x80000001  0x935C

Router Link States (Area 1)
Link ID         ADV Router      Age     Seq#       Checksum    Link count
1.1.1.1         1.1.1.1         89      0x80000011  0xFF59      3
2.2.2.2         2.2.2.2         88      0x80000033  0x2130      2

Summary Net Link States (Area 1)
Link ID         ADV Router      Age     Seq#       Checksum
6.0.0.0         2.2.2.2         94      0x8000001F  0xCC43
Type?5 AS External Link States
Link ID         ADV Router      Age     Seq#       Checksum    Tag
9.0.0.0         1.1.1.1         135     0x80000001  0x3AE8      0

r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 1)
Routing Bit Set on this LSA
LS age: 109
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
```

这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1

```
LS Seq Number: 80000011
Checksum: 0xFF59
Length: 60
AS Boundary Router
```

这一行表示由该路由器发起外部路由

```
Number of Links: 3
```

在AREA1里有3个连接

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8

```
 (Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
```

OSPF在以太网段的COST值

```
Link connected to: another Router (point?to?point)
```

这一行显示路由器1.1.1.1与路由器2.2.2.2互为邻居

```
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 5.0.0.1
```

路由器1.1.1.1与路由器2.2.2.2连接的接口地址为5.0.0.1

```
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

两台路由器之间链路的COST值

```
Link connected to: a Stub Network
```

这一行显示路由器1.1.1.1连接了5.0.0.0/8 Stub Network

```
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
```

串行链路的COST值为64

```
r2.2.2.2#show ip ospf database router 2.2.2.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 135
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000020
Checksum: 0xCD0B
Length: 48
Area Border Router
Number of Links: 2
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 3.3.3.3
(Link Data) Router Interface address: 6.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
Router Link States (Area 1)
LS age: 130
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000033
Checksum: 0x2130
Length: 48
Area Border Router
Number of Links: 2
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address: 5.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
r2.2.2.2#show ip ospf database router 3.3.3.3
OSPF Router with ID (2.2.2.2) (Process ID 2)
```

```
Router Link States (Area 0)
LS age: 1280
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 8000000D
Checksum: 0x9057
Length: 48
Number of Links: 2
Link connected to: another Router (point?
to?point)
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address:
6.0.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
To advertise routes from one area into
another, the ABR creates (type 3) summary
LSAs.
r2.2.2.2#show ip ospf database summary
4.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 0)
LS age: 184
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 4.0.0.0 (summary Network
Number)
```

这一行显示子网4.0.0.0由ABR路由器2.2.2.2公布到AREA0

```
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0xFFE6
Length: 28
Network Mask: /8
TOS: 0 Metric: 74
r2.2.2.2#show ip ospf database summary
5.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 0)
LS age: 1768
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 5.0.0.0 (summary Network
Number)
```

这一行显示子网5.0.0.0由ABR路由器2.2.2.2公布到AREA0

```
Advertising Router: 2.2.2.2
LS Seq Number: 80000006
Checksum: 0x8466
Length: 28
Network Mask: /8
TOS: 0 Metric: 64
r2.2.2.2#show ip ospf database summary
6.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Summary Net Link States (Area 1)
LS age: 216
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0
```

子网6.0.0.0/8 由ABR路由器2.2.2.2公布到AREA1

```
 Advertising Router: 2.2.2.2
LS Seq Number: 8000001F
Checksum: 0xCC43
Length: 28
Network Mask: /8
TOS: 0 Metric: 1
```

要发布外部路由，由ASBR路由器创建类型5的LSA。

```
r2.2.2.2#show ip ospf database external
9.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID
2)
Type?5 AS External Link States
Routing Bit Set on this LSA
LS age: 286
Options: (No TOS?capability, DC)
LS Type: AS External Link
Link State ID: 9.0.0.0 (External Network Number )
```

该行显示外部路由9.0.0.0由ASBR路由器1.1.1.1公布

```
Advertising Router: 1.1.1.1
LS Seq Number: 80000001
Checksum: 0x3AE8
Length: 36
Network Mask: /8
Metric Type: 1 (Comparable directly to link state metric)
TOS: 0
Metric: 5
Forward Address: 0.0.0.0
```

在这里FORWARD ADRESS被设置为0，说明在路由器1.1.1.1的以太网上没有相应的OSPF邻居

```
External Route Tag: 0
```

为了使得所有的路由器都能够访问到ASBR路由器，ABR路由器同时还会公布类型4的LSA

```
r2.2.2.2#show ip ospf database asbr?summary 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Summary ASB Link States (Area 0)
LS age: 266
Options: (No TOS?capability, DC)
LS Type: Summary Links(AS Boundary Router)
Link State ID: 1.1.1.1 (AS Boundary Router address)
```

ABR路由器公布如何到达ASBR路由器的信息

```
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0x935C
Length: 28
Network Mask: /0
TOS: 0 Metric: 64
```

由ABR路由器到达ASBR路由器的COST

之所以要产生一条类型4的LSA，是因为ABR路由器2.2.2.2从ASBR路由器学习到了外部LSA并将它泛洪到了AREA0，然而AREA0的路由器并不知道如何到达ASBR路由器，所以需要有一条类型4的LSA用于通告ASBR路由器的信息！

计算最短路径

我们以路由器3.3.3.3为例分析一下计算过程。

首先路由器3.3.3.3查看它自己的路由表，发现路由器2.2.2.2是它的一个邻居，然后它查看路由器2.2.2.2的LSA，如果发现路由器2.2.2.2也把自己作为邻居，则路由器3.3.3.3认为它们彼此可达。同时每台路由器也会检查它本地的邻居表以确认邻居的接口地址是否和自己的接口地址在相同的子网。（注意：如果接口是Unnumbered的，那不会进行地址检查）如果接口都在相同的子网里面，则路由器会将邻居发布的ROUTER LSA中列出的stub networks记录到自己的路由表里面。在本例中路由器2.2.2.2的LSA中只公布了6.0.0.0/8这一条stub networks信息，而该子网是路由器3.3.3.3的直连子网，所以不会记录到路由表中，取而代之的是一条直连路由。检查完所有在AREA1 中的ROUTER LSA后，路由器开始检查

完所有在AREA1 中的ROUTER LSA后，路由器开始检查链路状态数据库中的summary LSAs，发现在子网4.0.0.0/8 和5.0.0.0/8的两条summary LSAs如果此时路由器知道如何到达公布这条LSA的路由器接口，则它会把这些子网加到自己的路由表里面，因为公布这条LSA的路由器是2.2.2.2，所以路由器3.3.3.3知道如何到达，因此会将这两条路由加到路由表里面。路由的度量值是路由器到达公布此LSA的路由器的度量与LSA中公布的度量值之和。当检查完所有的内部LSA后，路由器开始检查外部LSA。检查到一条由ASBR1.1.1.1公布的外部LSA9.0.0.0/8，然后检查是否可以到达ASBR路由器，由于ABR路由器2.2.2.2公布了一条类型4的LSA，所以路由器3.3.3.3知道如何到达ASBR路由器，因此路由器3.3.3.3会将该路由加入自己的路由表中。

```
r3.3.3.3#show ip route ospf
O IA 4.0.0.0/8 [110/75] via 6.0.0.2, 00:07:59, ATM2/0.20
O IA 5.0.0.0/8 [110/65] via 6.0.0.2, 00:07:59, ATM2/0.20
O E1 9.0.0.0/8 [110/70] via 6.0.0.2, 00:07:59, ATM2/0.20
r2.2.2.2#show ip route ospf
O 4.0.0.0/8 [110/74] via 5.0.0.1, 00:06:55, Serial0/1/0
O E1 9.0.0.0/8 [110/69] via 5.0.0.1, 00:06:55, Serial0/1/0
r1.1.1.1#show ip route 9.0.0.0
Routing entry for 9.0.0.0/8
Known via "static", distance 1, metric 0
Redistributing via ospf 4
Advertised by ospf 4 metric 5 metric?type 1
Routing Descriptor Blocks:
* 4.0.0.2
Route metric is 0, traffic share count is 1
```

OSPF的虚连接

网络拓扑图

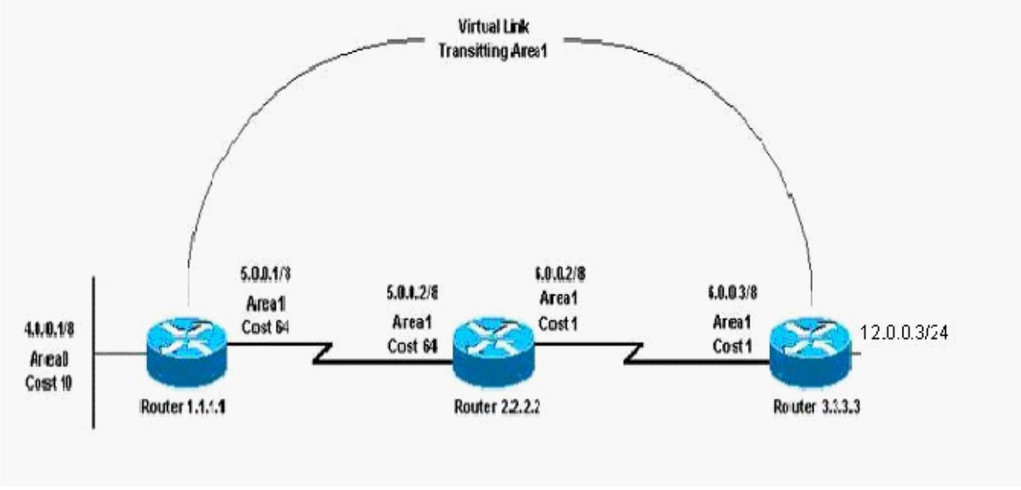


图8

相关配置

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20 point-to-point
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 6.0.0.0 0.255.255.255 area 1
 network 5.0.0.0 0.255.255.255 area 1

end
```

Router 3.3.3.3

```
Current configuration:

hostname r3.3.3.3

interface Loopback0
 ip address 3.3.3.3 255.0.0.0

interface Ethernet0/0
 ip address 12.0.0.3 255.0.0.0

interface ATM2/0.20 point-to-point
 ip address 6.0.0.3 255.0.0.0

router ospf 2
 network 12.0.0.0 0.255.255.255 area 2
 network 6.0.0.0 0.255.255.255 area 1
 area 1 virtual-link 1.1.1.1

end
```

链路状态数据库解析

```
r1.1.1.1#show ip ospf database
OSPF Router with ID (1.1.1.1) (Process ID 2)
Router Link States (Area 0)
Link ID      ADV Router      Age        Seq#         Checksum
1.1.1.1      1.1.1.1         919        0x80000003   0xD5DF
3.3.3.3      3.3.3.3         5 (DNA)    0x80000002   0x3990
Summary Net Link States (Area 0)
Link ID      ADV Router      Age        Seq#         Checksum
5.0.0.0      1.1.1.1         1945       0x80000002   0xAA48
5.0.0.0      3.3.3.3         9 (DNA)    0x80000001   0x7A70
6.0.0.0      1.1.1.1         1946       0x80000002   0xA749
6.0.0.0      3.3.3.3         9 (DNA)    0x80000001   0xEA3F
12.0.0.0     3.3.3.3         9 (DNA)    0x80000001   0xF624
Router Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
1.1.1.1      1.1.1.1         1946       0x80000005   0xDDA6
2.2.2.2      2.2.2.2         10         0x80000009   0x64DD
3.3.3.3      3.3.3.3         930        0x80000006   0xA14C
Summary Net Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
4.0.0.0      1.1.1.1         1947       0x80000002   0x9990
4.0.0.0      3.3.3.3         911        0x80000001   0xEBF5
12.0.0.0     1.1.1.1         913        0x80000001   0xBF22
12.0.0.0     3.3.3.3         931        0x80000001   0xF624
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
1.1.1.1      1.1.1.1         1988       0x80000005   0xDDA6
2.2.2.2      2.2.2.2         50         0x80000009   0x64DD
3.3.3.3      3.3.3.3         969        0x80000006   0xA14C
Summary Net Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
4.0.0.0      1.1.1.1         1988       0x80000002   0x9990
4.0.0.0      3.3.3.3         950        0x80000001   0xEBF5
12.0.0.0     1.1.1.1         955        0x80000001   0xBF22
12.0.0.0     3.3.3.3         970        0x80000001   0xF624
r3.3.3.3#show ip ospf database
OSPF Router with ID (3.3.3.3) (Process ID 2)
Router Link States (Area 0)
Link ID      ADV Router      Age        Seq#         Checksum
1.1.1.1      1.1.1.1         6 (DNA)    0x80000003   0xD5DF
3.3.3.3      3.3.3.3         977        0x80000002   0x3990
Summary Net Link States (Area 0)
Link ID      ADV Router      Age        Seq#         Checksum
5.0.0.0      1.1.1.1         1027 (DNA) 0x80000002   0xAA48
5.0.0.0      3.3.3.3         986        0x80000001   0x7A70
6.0.0.0      1.1.1.1         1027 (DNA) 0x80000002   0xA749
6.0.0.0      3.3.3.3         987        0x80000001   0xEA3F
12.0.0.0     3.3.3.3         987        0x80000001   0xF624
Router Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
1.1.1.1      1.1.1.1         2007       0x80000005   0xDDA6
2.2.2.2      2.2.2.2         68         0x80000009   0x64DD
3.3.3.3      3.3.3.3         987        0x80000006   0xA14C
Summary Net Link States (Area 1)
Link ID      ADV Router      Age        Seq#         Checksum
4.0.0.0      1.1.1.1         2007       0x80000002   0x9990
4.0.0.0      3.3.3.3         967        0x80000001   0xEBF5
12.0.0.0     1.1.1.1         973        0x80000001   0xBF22
12.0.0.0     3.3.3.3         987        0x80000001   0xF624
Router Link States (Area 2)
Link ID      ADV Router      Age        Seq#         Checksum
3.3.3.3      3.3.3.3         987        0x80000003   0xCF5
Summary Net Link States (Area 2)
Link ID      ADV Router      Age        Seq#         Checksum
4.0.0.0      3.3.3.3         968        0x80000001   0xEBF5
5.0.0.0      3.3.3.3         988        0x80000001   0x7A70
6.0.0.0      3.3.3.3         988        0x80000001   0xEA3F
```

上面我们可以看到由虚连接发起的LSA的AGE字段为（DNA---DO NOT AGE）

```
r1.1.1.1#show ip ospf database router 1.1.1.1
OSPF Router with ID (1.1.1.1) (Process ID 2)
Router Link States (Area 0)
LS age: 1100
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
```

这里显示的是公布这条LSA的路由器的Router ID为1.1.1.1

```
LS Seq Number: 80000003
Checksum: 0xD5DF
Length: 48
Area Border Router
```

此行显示这台路由器是ABR
Number of Links: 2
在AREA0 有两个连接
Link connected to: a Virtual Link
(Link ID) Neighboring Router ID: 3.3.3.3
虚连接另一端路由器的ROUTER ID 为3.3.3.3k
(Link Data) Router Interface address:
5.0.0.1
这是用来向对端发送数据的端口
Number of TOS metrics: 0
TOS 0 Metrics: 65
这是到达邻居的COST值ATM链路的COST值为1，串口链路的COST值为64.
Link connected to: a Stub Network
这行显示这台路由器连接的STUB NETWORK
4.0.0.0/8
(Link ID) Network/subnet number: 4.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
Router Link States (Area 1)
LS age: 122
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
Advertising Router: 1.1.1.1
LS Seq Number: 80000006
Checksum: 0xDBA7
Length: 48
Area Border Router
Number of Links: 2
在AREA1中有两个连接
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address:
5.0.0.1
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64

r1.1.1.1#show ip ospf database router
2.2.2.2
OSPF Router with ID (1.1.1.1) (Process ID 2)
Router Link States (Area 1)
LS age: 245
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000009
Checksum: 0x64DD
Length: 72
Number of Links: 4
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 3.3.3.3
(Link Data) Router Interface address:
6.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address:
5.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
r1.1.1.1#show ip ospf database router
3.3.3.3
OSPF Router with ID (1.1.1.1) (Process ID 2)
Router Link States (Area 0)
Routing Bit Set on this LSA
LS age: 5 (DoNotAge)
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0x3990
Length: 36
Area Border Router
Number of Links: 1
Link connected to: a Virtual Link
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address:
6.0.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 65
Router Link States (Area 1)
Routing Bit Set on this LSA
LS age: 1137
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 80000006
Checksum: 0xA14C
Length: 48
Area Border Router
Number of Links: 2
Link connected to: another Router (point?to?point)
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address:
6.0.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
路由器3.3.3.3认为自己是一台ABR，因为它通过虚连接与AREA0相连，所以它会生成一条12.0.0.0的汇总LSA发布到AREA1和AREA0。
r3.3.3.3#show ip ospf database summary
12.0.0.0
OSPF Router with ID (3.3.3.3) (Process ID 2)
Summary Net Link States (Area 0)
LS age: 1779
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 12.0.0.0 (summary Network Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000001
Checksum: 0xF624
Length: 28
Network Mask: /8
TOS: 0 Metric: 10
Summary Net Link States (Area 1)
LS age: 1766
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 12.0.0.0 (summary Network Number)
Advertising Router: 1.1.1.1
LS Seq Number: 80000001
Checksum: 0xBF22
Length: 28
Network Mask: /8
TOS: 0 Metric: 75
LS age: 1781
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 12.0.0.0 (summary Network Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000001
Checksum: 0xF624
Length: 28
Network Mask: /8
TOS: 0 Metric: 10
我们可以注意到路由器3.3.3.3将它从区域1和区域3学到的信息，为区域2生成了汇总LSA。
r3.3.3.3#show ip ospf database summary self?originate
OSPF Router with ID(3.3.3.3)(Process ID 2)
Summary Net Link States (Area 0)

```
LS age: 155
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 5.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0x7871
Length: 28
Network Mask: /8
TOS: 0 Metric: 65
LS age: 155
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xE840
Length: 28
Network Mask: /8
TOS: 0 Metric: 1
LS age: 156
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 12.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xF425
Length: 28
Network Mask: /8
TOS: 0 Metric: 10
Summary Net Link States (Area 1)
LS age: 157
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 4.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xE9F6
Length: 28
Network Mask: /8
TOS: 0 Metric: 75
LS age: 165
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 12.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xF425
Length: 28
Network Mask: /8
TOS: 0 Metric: 10
Summary Net Link States (Area 2)
LS age: 167
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 4.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xE9F6
Length: 28
Network Mask: /8
TOS: 0 Metric: 75
LS age: 168
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 5.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0x7871
Length: 28
Network Mask: /8
TOS: 0 Metric: 65
LS age: 168
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0 (summary Network
Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xE840
Length: 28
Network Mask: /8
TOS: 0 Metric: 65
LS age: 168
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0 (summary Network Number)
Advertising Router: 3.3.3.3
LS Seq Number: 80000002
Checksum: 0xE840
Length: 28
Network Mask: /8
TOS: 0 Metric: 1
```

下面我们来看一下虚连接是如何工作的。最初，虚连接是DOWN的，这是因为路由器1.1.1.1不知道如何到达它的虚连接对端3.3.3.3，只有当AREA1中所有的OSPF路由器全部泛洪自己的LSA，并且所有的路由器进行了SPF计算得出了到达彼此的路由，此时路由器1.1.1.1才会知道如何到达路由器3.3.3.3。当路由器可以互相到达后，虚连接路由器试图建立虚连接的邻接关系，虚连接之间的OSPF报文不是组播报文，而是通过虚连接构建的隧道报文。当虚连接邻接关系建立完成，这时候路由器3.3.3.3便认为自己是ABR路由器，因为它有到达AREA0的连接，所以路由器3.3.3.3创建12.0.0.0/8的汇总LSA并泛洪到AREA0和AREA1。

计算最短路径

我们以路由器2.2.2.2为例分析一下计算过程。

首先路由器2.2.2.2查看它自己的路由表，发现路由器3.3.3.3是它的一个邻居，然后它查看路由器3.3.3.3的LSA，如果发现路由器3.3.3.3也把自己作为邻居，则路由器2.2.2.2认为它们彼此可达。同时每台路由器也会检查它本地的邻居表以确认邻居的接口地址是否和自己的接口地址在相同的子网。（注意：如果接口是Unnumbered的，那不会进行地址检查）如果接口都在相同的子网里面，则路由器会将邻居发布的ROUTER LSA中列出的stub networks记录到自己的路由表里面。在本例中路由器3.3.3.3的LSA中只公布了6.0.0.0/8这一条stub networks信息，而该子网是路由器2.2.2.2的直连子网，所以不会记录到路由表中，取而代之的是一条直连路由。路由器3.3.3.3对路由器1.1.1.1做与路由器2.2.2.2所做相同的检查，但是并没有在路由器1.1.1.1的LSA中找到有效的stub networks信息。

在检查完所有的AREA1中邻居路由器的ROUTER LSA后，路由器2.2.2.2查找自己数据库中的summary LSA，它发现有两条到达子网12.0.0.0./8的LSA，因此会进行COST计算，以找到一条COST值最低的路径放进自己的路由表中。计算方法是summary LSA的COST值加上到公布这条LSA的路由器的路径COST值。

路由器 2.2.2.2 通过路由器1.1.1.1到达子网12.0.0.0./8的cost为64 + 75 = 139 ??
路由器 2.2.2.2 通过路由器3.3.3.3到达子网12.0.0.0./8的cost为1 + 10 = 11 ??
所以路由器2.2.2.2将通过路由器3.3.3.3到达子网12.0.0.0/8的路由添加到自己的路由表中

```
r1.1.1.1#show ip route ospf
O        6.0.0.0/8 [110/65] via 5.0.0.2, 00:38:12, Serial2/1/0
O IA    12.0.0.0/8 [110/75] via 5.0.0.2, 00:38:02, Serial2/1/0
r2.2.2.2#show ip route ospf
O IA     4.0.0.0/8 [110/74] via 5.0.0.1, 00:38:08, Serial0/1/0
O IA    12.0.0.0/8 [110/11] via 6.0.0.3, 00:38:12, ATM1/0.20
```

```
r3.3.3.3#show ip route ospf
O        4.0.0.0/8 [110/75] via 6.0.0.2, 00:38:18, ATM2/0.20
O        5.0.0.0/8 [110/65] via 6.0.0.2, 00:38:28, ATM2/0.20
```

OSPF NSSA区域类型7及类型5的LSA解析

- 网络拓扑图

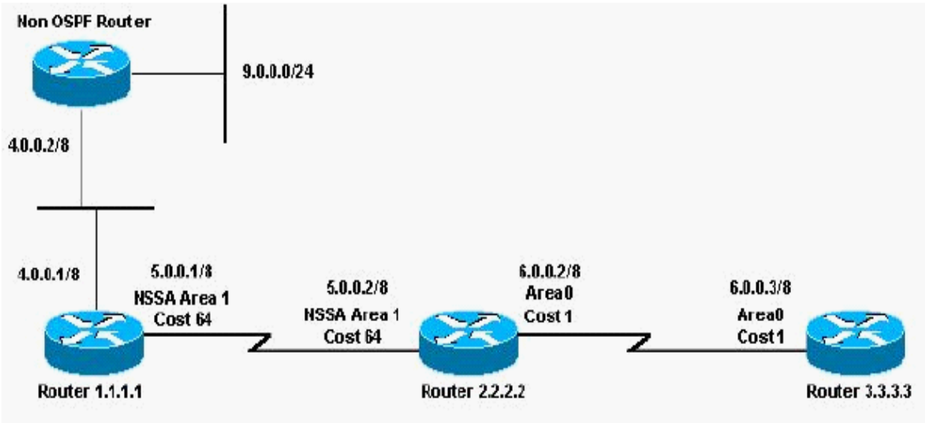


图9

- 相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Serial2/1/0
 ip address 5.0.0.1 255.0.0.0

interface Ethernet2/0/0
 ip address 4.0.0.1 255.0.0.0

router ospf 4
 redistribute static metric 5 metric-type 1
 network 5.0.0.0 0.255.255.255 area 1
 network 4.0.0.0 0.255.255.255 area 1
 area 1 nssa

ip route 9.0.0.0 255.0.0.0 4.0.0.2

end
```

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 network 6.0.0.0 0.255.255.255 area 0
 area 1 nssa

end
```

Router 3.3.3.3

```
Current configuration:

hostname r3.3.3.3

interface Loopback0
 ip address 3.3.3.3 255.0.0.0

interface ATM2/0.20 point-to-point
 ip address 6.0.0.3 255.0.0.0

router ospf 2
 network 6.0.0.0 0.255.255.255 area 0

end
```

链路状态数据库解析

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID        ADV Router       Age       Seq#       Checksum     Link count
2.2.2.2         2.2.2.2         1235   0x8000001D    0xD9FF            2
3.3.3.3         3.3.3.3         1100   0x8000000B    0x9455            2
Summary Net Link States (Area 0)
Link ID        ADV Router       Age       Seq#       Checksum
4.0.0.0        2.2.2.2          1979   0x80000002    0xFDE7
5.0.0.0        2.2.2.2          1483   0x80000004    0x8864
Router Link States (Area 1)
Link ID        ADV Router       Age       Seq#       Checksum     Link count
1.1.1.1        1.1.1.1          319    0x8000000C    0xAFA8            3
2.2.2.2        2.2.2.2          220    0x8000002F    0xD478            2
Summary Net Link States (Area 1)
Link ID        ADV Router       Age       Seq#       Checksum
6.0.0.0        2.2.2.2          1483   0x8000001C    0x7894
Type?7 AS External Link States (Area 1)
Link ID        ADV Router       Age       Seq#       Checksum         Tag
9.0.0.0        1.1.1.1           334    0x80000005    0xD738           0
Type?5 AS External Link States
Link ID        ADV Router       Age       Seq#       Checksum         Tag
9.0.0.0        2.2.2.2          1725   0x80000004    0x50C6           0
```

在NSSA区域公布外部路由是通过ASBR路由器发布的类型7LSA

```
r2.2.2.2#show ip ospf database nssa?external 9.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Type?7 AS External Link States (Area 1)
Routing Bit Set on this LSA
LS age: 381
Options: (No TOS?capability, Type 7/5 translation, DC)
```

这一行表示类型7的LSA可以由ABR路由器转换成类型5的路由

```
LS Type: AS External Link
Link State ID: 9.0.0.0 (External Network Number )
```

ASBR路由器公布9.0.0.0/8的外部路由

```
Advertising Router: 1.1.1.1
```

这一行显示公布该路由的ASBR的ROUTER ID

```
LS Seq Number: 80000005
Checksum: 0xD738
Length: 36
Network Mask: /8
Metric Type: 1 (Comparable directly to link state metric)
TOS: 0
Metric: 5
Forward Address: 4.0.0.1
```

Forward Address 被分配为ASBR的接口地址

ABR路由器将类型7的LSA转换成类型5的LSA并泛洪到正常区域.

```
r2.2.2.2#show ip ospf database external 9.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Type?5 AS External Link States
LS age: 1782
Options: (No TOS?capability, DC)
LS Type: AS External Link
Link State ID: 9.0.0.0 (External Network Number )
```

路由器2.2.2.2公布 9.0.0.0/8的外部LSA

```
Advertising Router: 2.2.2.2
```

当类型7被转换为类型5的LSA后，公布该LSA的路由器变为2.2.2.2，因为是由路由器2.2.2.2向外公布类型5的外部LSA

```
LS Seq Number: 80000004
Checksum: 0x50C6
Length: 36
Network Mask: /8
Metric Type: 1 (Comparable directly to link state metric)
TOS: 0
Metric: 5
Forward Address: 4.0.0.1
External Route Tag: 0
r2.2.2.2#show ip ospf database router 1.1.1.1
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 1)
Routing Bit Set on this LSA
LS age: 426
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 1.1.1.1
```

在Router LSA中，Link State ID是和Advertising Router的值是相同的，即发布这条LSA的路由器的Router ID

```
Advertising Router: 1.1.1.1
LS Seq Number: 8000000C
Checksum: 0xAFA8
Length: 60
AS Boundary Router
```

这一行显示该路由器公布类型5的外部LSA

```
Number of Links: 3
```

在AREA1有三个邻居1

```
Link connected to: a Stub Network
```

这行显示这台路由器连接的STUB NETWORK 4.0.0.0/8

```
(Link ID) Network/subnet number: 4.0.0.0
```

(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 10
OSPF在以太网段的COST值
Link connected to: another Router (point?to ?point)
这一行显示路由器1.1.1.1与路由器2.2.2.2互为邻居
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 5.0.0.1
该路由器连接邻居2.2.2.2的接口地址为5.0.0.1
Number of TOS metrics: 0
TOS 0 Metrics: 64
两台路由器之间链路的COST值为64
Link connected to: a Stub Network
这行显示该路由器连接了一个Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
串行链路的COST值为64
从下面的显示我们可以看到，即使路由器2.2.2.2没有任何的引如操作，但是它依然是作为ASBR，因为它将类型7的LSA转换成类型5的LSA，并向其他区域泛洪.
r2.2.2.2#show ip ospf database router 2.2.2.2
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 1361
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 8000001D
Checksum: 0xD9FF
Length: 48
Area Border Router
这一行显示这台路由器为ABR
AS Boundary Router
同时这台路由器还是ASBR
Number of Links: 2
该路由器在AREA0中有两个邻居
Link connected to: another Router (point?to ?point)
(Link ID) Neighboring Router ID: 3.3.3.3
(Link Data) Router Interface address: 6.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
Router Link States (Area 1)
LS age: 346
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 8000002F
Checksum: 0xD478
Length: 48
Area Border Router
AS Boundary Router
Number of Links: 2
Link connected to: another Router (point?to ?point)
(Link ID) Neighboring Router ID: 1.1.1.1
(Link Data) Router Interface address: 5.0.0.2
Number of TOS metrics: 0
TOS 0 Metrics: 64
Link connected to: a Stub Network
(Link ID) Network/subnet number: 5.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 64
r2.2.2.2#show ip ospf database router 3.3.3.3
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
LS age: 1245
Options: (No TOS?capability, DC)
LS Type: Router Links
Link State ID: 3.3.3.3
Advertising Router: 3.3.3.3
LS Seq Number: 8000000B
Checksum: 0x9455
Length: 48
Number of Links: 2
Link connected to: another Router (point?to ?point)
(Link ID) Neighboring Router ID: 2.2.2.2
(Link Data) Router Interface address: 6.0.0.3
Number of TOS metrics: 0
TOS 0 Metrics: 1
Link connected to: a Stub Network
(Link ID) Network/subnet number: 6.0.0.0
(Link Data) Network Mask: 255.0.0.0
Number of TOS metrics: 0
TOS 0 Metrics: 1
To advertise routes from one area into another, the ABR creates summary LSAs (type 3).
r2.2.2.2#show ip ospf database summary 4.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Summary Net Link States (Area 0)
LS age: 172
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 4.0.0.0 (summary Network Number)
ABR路由器向AREA0公布子网4.0.0.0
Advertising Router: 2.2.2.2
LS Seq Number: 80000003
Checksum: 0xFBE8
Length: 28
Network Mask: /8
TOS: 0 Metric: 74
r2.2.2.2#show ip ospf database summary 5.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Summary Net Link States (Area 0)
LS age: 1687
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 5.0.0.0 (summary Network Number)
ABR路由器向AREA0公布子网5.0.0.0
Advertising Router: 2.2.2.2
LS Seq Number: 80000004
Checksum: 0x8864
Length: 28
Network Mask: /8
TOS: 0 Metric: 64
r2.2.2.2#show ip ospf database summary 6.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Summary Net Link States (Area 1)
LS age: 1697
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 6.0.0.0 (summary Network Number)
ABR路由器向AREA1公布子网6.0.0.0
Advertising Router: 2.2.2.2
LS Seq Number: 8000001C
Checksum: 0x7894
Length: 28
Network Mask: /8
TOS: 0 Metric: 1
在这个例子中我们不需要类型4的LSA，因为ABR在这里充当ASBR的角色，而它对与其他的路由器来说，都是可达的。如下是该情况下的路由表信息：
r1.1.1.1#show ip route 9.0.0.0
Routing entry for 9.0.0.0/8
Known via "static", distance 1, metric 0
Redistributing via ospf 4
Advertised by ospf 4 metric 5 metric?type 1
Routing Descriptor Blocks:
* 4.0.0.2

```
Route metric is 0, traffic share count is 1
r2.2.2.2#show ip route ospf
O 4.0.0.0/8 [110/74] via 5.0.0.1, 01:10:13, Serial0/1/0
O N1 9.0.0.0/8 [110/79] via 5.0.0.1, 01:07:20, Serial0/1/0
R3.3.3.3#show ip route ospf
O IA 4.0.0.0/8 [110/75] via 6.0.0.2, 02:11:14, ATM2/0.20
O IA 5.0.0.0/8 [110/65] via 6.0.0.2, 03:10:41, ATM2/0.20
O E1 9.0.0.0/8 [110/80] via 6.0.0.2, 02:08:11, ATM2/0.20
```

OSPF如何向普通区域引入默认路由

网络拓扑图

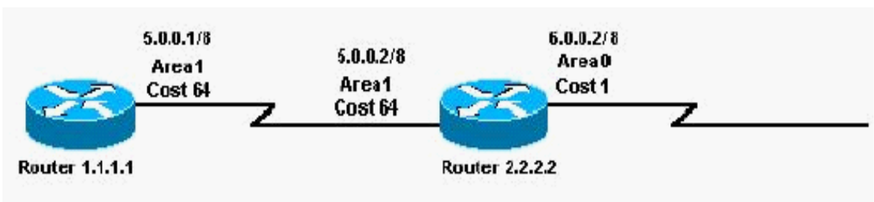


图10

相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Serial2/1/0
 ip address 5.0.0.1 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1

end
```

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 network 6.0.0.0 0.255.255.255 area 0
 default-information originate

ip route 0.0.0.0 0.0.0.0 6.0.0.3

end
```

链路状态数据库解析

这个例子向我们展示了如何将默认路由引入到OSPF的普通区域。默认路由可以通过任何一台OSPF路由器引入到OSPF普通区域内。默认情况下运行OSPF的路由器是不会在OSPF域内自动生成或引入默认路由的，我们需要使用default?information originate命令来实现。有两种方法可以实现该功能，一种是路由器上配置了默认路由，这是通过default?information originate命令即可实现，另一种方法是不管在路由器上是否配置了默认路由，OSPF路由器都在OSPF域内公布，这需要在default?information originate命令后加上“always”参数

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID ADV Router Age Seq# Checksum Link count
2.2.2.2 2.2.2.2 600 0x80000001 0x9583 1
Summary Net Link States (Area 0)
Link ID ADV Router Age Seq# Checksum
5.0.0.0 2.2.2.2 600 0x80000001 0x8E61
Router Link States (Area 1)
Link ID ADV Router Age Seq# Checksum Link count
1.1.1.1 1.1.1.1 864 0x8000005E 0xD350 2
2.2.2.2 2.2.2.2 584 0x8000001E 0xF667 2
Summary Net Link States (Area 1)
Link ID ADV Router Age Seq# Checksum
6.0.0.0 2.2.2.2 585 0x80000004 0xA87C
Type?5 AS External Link States
Link ID ADV Router Age Seq# Checksum Tag
0.0.0.0 2.2.2.2 601 0x80000001 0xD0D8 0
```

路由器2.2.2.2通过类型5的LSA向外公布默认路由，这是因为配置了default?information originate的

缘故。

```
r2.2.2.2#show ip ospf database external 0.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Type?5 AS External Link States
LS age: 650
Options: (No TOS?capability, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0xD0D8
Length: 36
Network Mask: /0
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 0
r2.2.2.2#show ip route 0.0.0.0
S* 0.0.0.0/0 [1/0] via 6.0.0.3, 00:28:00, ATM1/0.20
r1.1.1.1#show ip route ospf
O IA 6.0.0.0/8 [110/65] via 5.0.0.2, 00:00:18, Serial2/1/0
O*E2 0.0.0.0/0 [110/1] via 5.0.0.2, 00:00:18, Serial2/1/0
```

OSPF如何向STUB及完全STUB区域引入默认路由

网络拓扑图

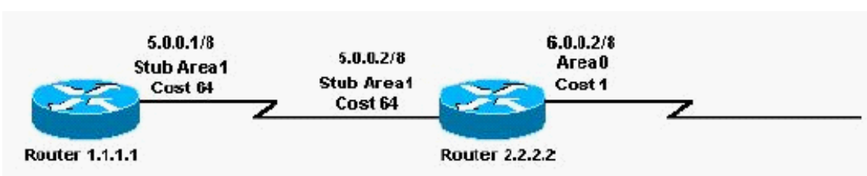


图11

相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Serial2/1/0
 ip address 5.0.0.1 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 area 1 stub

end
```

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2

interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 network 6.0.0.0 0.255.255.255 area 0
 area 1 stub

end
```

STUB区域链路状态数据库解析

ABR路由器通过类型3的LSA向STUB区域发布默认路由。

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum    Link count
2.2.2.2         2.2.2.2          19     0x80000001     0x8F8B          1
Summary Net Link States (Area 0)
Link ID         ADV Router      Age         Seq#       Checksum
5.0.0.0         2.2.2.2           9     0x80000001     0x8E61
Router Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum    Link count
1.1.1.1         1.1.1.1        1335  0x80000059       0x56DA          2
2.2.2.2         2.2.2.2           4     0x80000013     0x7FF3          2
```

```
Summary Net Link States (Area 1)
Link ID         ADV Router          Age         Seq#       Checksum
0.0.0.0           2.2.2.2            20    0x80000001       0x75C0
6.0.0.0           2.2.2.2            13    0x80000001       0x2709
r2.2.2.2#show ip ospf database summary 0.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Summary Net Link States (Area 1)
LS age: 184
Options: (No TOS?capability, DC)
LS Type: Summary Links(Network)
Link State ID: 0.0.0.0 (summary Network Number)
```

ABR路由器向STUB区域引入默认路由

```
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0x75C0
Length: 28
Network Mask: /0
TOS: 0 Metric: 1
r2.2.2.2#show ip route 0.0.0.0
% Network not in table
```

此行显示我们并没有在ABR上配置过默认路由，这条默认路由是由ABR自动生成的。

```
r1.1.1.1#show ip route ospf
O IA 6.0.0.0/8 [110/65] via 5.0.0.2, 00:04:23, Serial2/1/0
O*IA 0.0.0.0/0 [110/65] via 5.0.0.2, 00:04:23, Serial2/1/0
```

完全STUB区域链路状态数据库解析

如果我们将上面配置中的AREA1修改为完全STUB区域，ABR依然会向AREA1引入默认路由。这两种情况的不同之处在于完全STUB区域不接收类型3的LSA。将STUB区域修改为完全STUB区域只需要在ABR路由器上做如下修改：加上no?summary参数

```
area 1 stub no?summary
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID         ADV Router          Age         Seq#       Checksum       Link count
2.2.2.2           2.2.2.2           617       0x80000001     0x8F8B            1
Summary Net Link States (Area 0)
Link ID         ADV Router          Age         Seq#       Checksum
5.0.0.0           2.2.2.2           608       0x80000001     0x8E61
Summary ASB Link States (Area 0)
Link ID         ADV Router          Age         Seq#       Checksum
1.1.1.1           2.2.2.2           243       0x80000003     0x8F5E
Router Link States (Area 1)
Link ID         ADV Router          Age         Seq#       Checksum       Link count
1.1.1.1           1.1.1.1           1934      0x80000059     0x56DA            2
2.2.2.2           2.2.2.2           247       0x80000015     0x7BF5            2
Summary Net Link States (Area 1)
Link ID         ADV Router          Age         Seq#       Checksum
0.0.0.0           2.2.2.2           249       0x80000003     0x71C2
```

这行显示了只有默认路由的Summary Net被公布

```
r1.1.1.1#show ip route ospf
O*IA 0.0.0.0/0 [110/65] via 5.0.0.2, 00:04:11, Serial2/1/0
```

这行显示在完全STUB区域里只能从ABR学到默认路由。

OSPF如何向NSSA区域引入默认路由

网络拓扑图

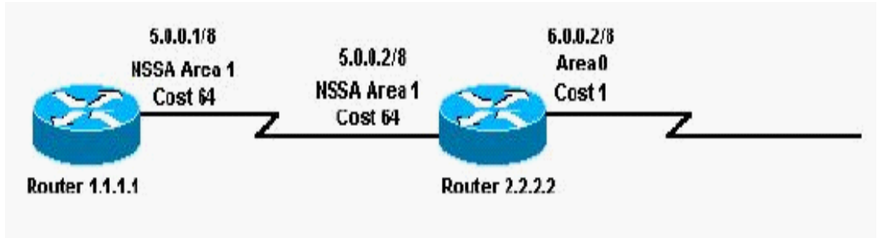


图12

相关配置

Router 1.1.1.1

```
Current configuration:

hostname r1.1.1.1

interface Loopback0
 ip address 1.1.1.1 255.0.0.0

interface Serial2/1/0
 ip address 5.0.0.1 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 area 1 nssa

end
```

Router 2.2.2.2

```
Current configuration:

hostname r2.2.2.2
interface Loopback0
 ip address 2.2.2.2 255.0.0.0

interface Serial0/1/0
 ip address 5.0.0.2 255.0.0.0

interface ATM1/0.20
 ip address 6.0.0.2 255.0.0.0

router ospf 2
 network 5.0.0.0 0.255.255.255 area 1
 network 6.0.0.0 0.255.255.255 area 0
 area 1 nssa default-information originate

end
```

NSSA区域链路状态数据库解析

```
r2.2.2.2#show ip ospf database
OSPF Router with ID (2.2.2.2) (Process ID 2)
Router Link States (Area 0)
Link ID        ADV Router        Age        Seq#        Checksum      Link count
2.2.2.2         2.2.2.2          600     0x80000001      0x9583            1
Summary Net Link States (Area 0)
Link ID        ADV Router        Age        Seq#        Checksum
5.0.0.0         2.2.2.2          600     0x80000001      0x8E61
Router Link States (Area 1)
Link ID        ADV Router        Age        Seq#        Checksum        Link count
1.1.1.1         1.1.1.1          864     0x8000005E      0xD350            2
2.2.2.2         2.2.2.2          584     0x8000001E      0xF667            2
Summary Net Link States (Area 1)
Link ID       ADV Router        Age        Seq#        Checksum
6.0.0.0         2.2.2.2          585     0x80000004      0xA87C
Type?7 AS External Link States (Area 1)
Link ID ADV Router Age Seq# Checksum Tag
0.0.0.0         2.2.2.2          601     0x80000001      0xD0D8            0
```

NSSA区域的ABR生成了一条link ID为0.0.0.0的类型7LSA，这是因为我们配置了area 1 nssa default?information?originate命令.

```
r2.2.2.2#show ip ospf database nssa?external 0.0.0.0
OSPF Router with ID (2.2.2.2) (Process ID 2)
Type?7 AS External Link States (Area 1)
LS age: 650
Options: (No TOS?capability, No Type 7/5 translation, DC)
LS Type: AS External Link
Link State ID: 0.0.0.0 (External Network Number )
Advertising Router: 2.2.2.2
LS Seq Number: 80000001
Checksum: 0xD0D8
Length: 36
Network Mask: /0
Metric Type: 2 (Larger than any link state path)
TOS: 0
Metric: 1
Forward Address: 0.0.0.0
External Route Tag: 0
```

尽管在ABR上并没有配置默认路由，但是由于area 1 nssa default?information?originate命令它依然会生成一条类型7的LSA向区域内公布！

```
r2.2.2.2#show ip route 0.0.0.0
% Network not in table
r1.1.1.1#show ip route ospf
O IA 6.0.0.0/8 [110/65] via 5.0.0.2, 00:00:18, Serial2/1/0
O*N2 0.0.0.0/0 [110/1] via 5.0.0.2, 00:00:18, Serial2/1/0
```

OSPF应用案例

鉴定测试中心 杜祥宇

概述

项目背景简介

OSPF协议作为一个自治系统内部路由协议，主要用于解决一个管理域内的路由信息自动收集和最优路径计算，为数据转发和域间路由协议提供服务。在实际的工程中，域内路由协议选择和设计不会作为一个孤立的问题提出，通常在经过网络需求分析以后，与网络结构设计、地址规划、流量模型设计等统筹考虑。

本文介绍的组网应用案例是以某省电力调度数据网的一个真实案例，由于材料中涉及部分网络细节，出于安全考虑，文中隐去了该客户的真实名称，以下章节以A客户代称。为使读者能够较好理解OSPF路由设计的背景，本文保留了一部分业务需求和相关设计描述，但限于篇幅简化和删除了与本文相关性不大的内容。

该网络是专门为电力调度生产服务的，网络承载业务主要是数据业务。调度数据网承载的业务对网络可靠性要求高，网络的可用率、实时业务的传输时延（业务应有不同的优先级）、网络的收敛时间等关键性能必须予以保证。同时，目前大多承载业务的传输频度为秒级，网络节点的业务量相对比较恒定，网络的流量不大。

A客户网存在以下多种业务，主要分为4大类：

第一类：实时监控系统

第二类：水调自动化系统

第三类：继电保护故障录波远传与信息管理系统

第四类：同步时钟系统、通信监控系统

MPLS VPN简介

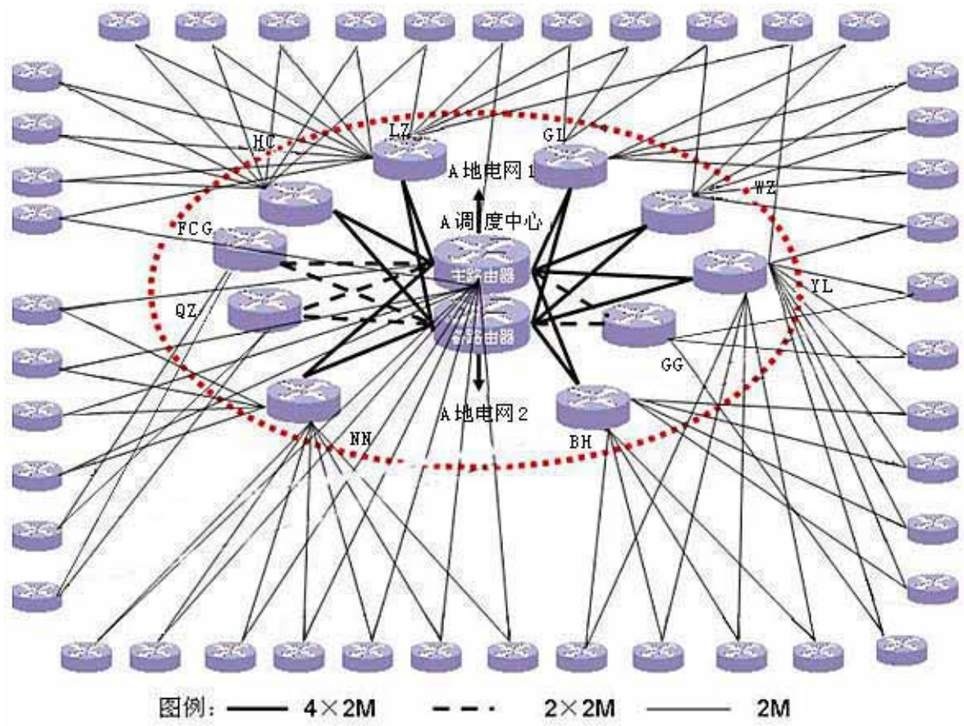
MPLS是多协议标签交换协议的简称。MPLS VPN网络中，有三种设备：CE、PE和P路由器，CE是用户直接与服务提供商相连的边缘设备，可以是路由器、交换机或者终端；PE是骨干网中的边缘设备，它直接与用户的CE相连；P 路由器是骨干网中不与CE直接相连的设备。

采用MPLS VPN技术可以把物理上单一的IP网络分解成逻辑上隔离的网络，并且每个VPN单独构成一个独立的地址空间，即VPN之间可以重用地址，在分配地址时不必考虑是否会与其他的VPN发生冲突，只需要考虑在本VPN之内不冲突即可，这样可以解决IP网络地址不足的问题，也方便网络的扩展和变更。

该工程就是采用MPLS VPN技术在A客户网上承载电力调度的各种业务。

网络详细设计

网络拓扑结构设计



A客户调度数据网络采用层次化设计，网络分为三层：核心层、汇聚层和接入层。

核心层负责整个网络的数据交换，设在A客户网调度中心（核心节点），核心节点设主备双路由器，采用Quidway NetEngine 16E。在设备上配置留有南方电网调度数据网的接入接口（暂按4×2M考虑）。接入用户业务的交换机采用多层交换机S6503。

汇聚层负责整个网络的数据汇聚，设备采用Quidway NetEngine 08E，星型双归属结构连接至核心节点。负荷较大的市地单位中继速率为4×2M；负荷较小的中继速率为2×2M。接入用户业务的交换机采用三层交换机S3526E。

接入节点共有43个，主要包括A客户电网调度的220kV变36个；500kV变7个，设备采用Quidway R4640。各接入节点按两点接入的原则，就近接入汇聚节点（部分考虑传输链路结构接入核心节点）。各接入节点的接入速率为2M。接入用户业务的交换机采用S2016。

本网络的建设相对独立，与上级电力调度数据网络设在各省的骨干节点设备背靠背连接接入上级电力调度数据网络。

路由协议配置

对于骨干层MPLS域，需要某种IGP协议与MBGP结合应用，其中MBGP主要完成私网路由标签分配、各私网路由在“公网”传递等作用，没有其他的协议可以替代。各地市CE之间的VPN路由的传递、VPN路由标签的分发，都是通过骨干区域MP-BGP协议的扩展属性实现的。

对于IGP协议，他的主要作用就是保证MBGP邻居之间的可达性，我们建议采用OSPF，因为OSPF的适应性好，功能完善，当核心层设备在20台以内的时候，我们建议只运行一个骨干域“0”，这样网络规划简单、维护方便，并且有利于今后骨干层网络的扩展。

对于PE与CE互联，可以采用的路由协议有静态路由、直连路由、RIP和BGP等多种路由协议。具体使用那种路由协议要根据情况而定，如当CE以下的网络结构比较复杂，路由条目较多的时候，PE和CE之间需要运行BGP协议，当然应用这种方案对CE的要求也是比较高的。对于PE与CE之间的路由协议类型最普遍应用的就是静态或直连路由，只要遵循上面提到的IP地址分配原则，各地的路由都可以汇聚成一条或者数目较少的几条，这时应用静态或直连路由是最简单、最高效的，而且网络的维护非常方便。

综上，在本期工程中，路由协议类型就是：骨干层MBGP＋OSPF、PE与CE互联采用静态、直连路由。这样对整个网络中的设备要求不是太高，并且很好的实现了MPLS-VPN。

BGP协议规划

AS号

采用私有的AS号，按照省公司的统一规划，AS号为64600。

路由反射器

骨干的路由器运行在同一个BGP的AS中，按照BGP协议的要求，所有这些路由器必须保证是全连通的，即：任意两台路由器之间都必须配置邻居关系。这样会导致N平方问题，为了解决这个问题，必须使用BGP反射器技术。

由于网络规模较大，所以使用二级路由反射。

路由的引入

在各地市的PE设备上，BGP的 vpnv4地址族中，不需要引入其他路由协议；BGP的IPV4地址族中，通过import 命令引入本VPN的直连和静态路由。

IGP协议规划

在MPLS 骨干区域内的IGP采用动态的、基于链路状态的OSPF协议。由于整个A客户调度数据网设备数量较多，为了保证整网的性能，考虑OSPF的分域规划，可以按地市把几个地调分到一个子域中，也可以按照每个地市单独分配一个区域，考虑到A客户调度数据网的实际情况和未来扩容的需要，此次项目采用每个地调分配一个子域的原则。由于网络拓扑结构设计的原因，每台设备都有双归属链路，因此这样导致了每个OSPF子区域都有多个ABR，在有多个ABR的子域上采用路由汇聚要保证这些子域上ABR的路由汇聚配置相同，不然会由于错误配置的原因导致网络故障。考虑到实际的运用，此次项目中的

NE08E/NE16E/AR4640都为高性能路由器，支持的路由表项容量大，而此次项目中所路由器上需要处理的路由条目数量并不大（大约1K条左右）为了简化网络配置和以后维护的工作量，此次项目不使用路由聚合技术。

区域的划分

OSPF的区域概念是基于路由器接口的，因此将A客户网中心两台NE16E和各地市10台NE08E的互联接口划分到一个 AREA 0。各地市NE08E与下带的R4640分别划分到非0 区域：
为避免路由环路，在PE上不通过OSPF 发布缺省路由。

对于ABR路由器（中调和地调路由器），她的loopback地址和网管网段（与交换机互联的接口网段）属于非骨干区域（如果存在多个非骨干区域，选择区域号小的），而不是属于骨干区域。对于厂站路由器，同时属于两个非骨干area时，发布网管网段（与交换机互联的接口网段）和loopback接口的路由时，通过import 命令发布，防止当某个区域链路中断后导致loopback路由无法发布。

路由的引入

在核心和各地调的PE设备上，通过Network命令将PE设备的loopback地址和互连端口地址引入到OSPF协议中去。为了便于控制路由的规模，不采用redistribute 命令引入静态或其他协议的路由。

定义花费值

出于链路备份和负荷分担的考虑，将OSPF链路的Cost值定义如下：

接口类型	Cost
GE	1
100M FE	10
4xE1	125
2xE1	250
1xE1	500
Loopback 接口	1

在此次项目中，对于双链路设备的两条上行链路采用如下原则：汇聚层到核心层的两条链路采用负载分担的方式；接入层到汇聚层的链路使用主备切换的方式。对于具体采用哪种方式可以通过在路由器接口模式下通过ospf cost 命令来控制，对于汇聚层到核心层的接口，汇聚层到接入层的接口，设置不同的开销（一大一小，开销小的为主链路，其值为在缺省值的基础上乘以80％，开销大的为备用链路）；采用主备用相互冗余的方式。

对于引入的直连路由，Type为1，开销为1。

NE08的链路主备用关系：GL、WZ、YL、GG、BH连接主用NE16 :ZD-H3-NE16-1的链路为主用链路。NN、QZ、LZ、FCG、HC连接备用NE16:ZD-H3-NE16-2的链路为主用链路。

AR4640的链路主备用关系：该厂站连接隶属的地调NE08的链路为主用链路，连接另一台地调NE08的链路为备用链路。

路由信息交换的认证

防止路由欺骗

OSPF路由协议在对等体（邻居）之间进行认证，可以有限防止路由欺骗，OSPF的认证方式分为明文和MD5密文认证两种方式，由于明文认证方式直接将密钥在协议报文中发送，安全性较差。

防止路由泄漏

在运行路由协议的路由器上，有些端口具有不安全的属性，最据代表性的如以太网口，该类型的接口由于需要发布本接口网段的路由信息，所以需要启动路由协议，但由于以太网的广播属性，

攻击者很容易将一台同样运行OSPF协议的路由设备连接到以太网中，并进而获得整网的路由信息及拓扑结构。所以必须在上述端口上启用Passive interface命令，启动该命令后，该接口的网段路由可以照常发布出去，但该接口不会再收发OSPF协议报文，也就不会再与任何其他路由设备建立邻居关系，从而避免了路由泄漏。
本次工程在各路由器运行OSPF的路由器的以太网（与对端设备之间并不运行OSPF）接口上启用Passive interface功能。网络安全部分，考虑到实际的应用需要，对路由协议的安全部分，只进行路由泄漏方面的安全防范，passive相关OSPF接口，不启用OSPF路由认证。

静态、直连路由规划

部分的CE（接入层）采用二层交换机，这样直接在PE上配置各业务的直连网段。

部分的CE采用三层交换机（核心层和汇聚层），这样就在PE和CE之间运行静态路由。在PE上配置网段为本地，下一跳指向CE的静态路由，然后引入到MP-BGP中，发布到其他地市；同时，在CE上配置目的网段为其他地市，下一跳为PE的静态路由。

MPLS VPN 规划

本次工程一共划分为4类VPN：（规划按照数字越小，优先级越高）
VPN1：实时监控系统
VPN2：水调自动化系统
VPN3：继电保护故障录波远传与信息管理系统
VPN4：同步时钟系统、通信监控系统。

VPN的互通和隔离

接入层本地VPN接入CE设备采用的是2层交换机，因此不同VPN可以通过2层VLAN进行本地隔离，在MPLS PE设备上为每个VPN维护一张单独的路由表，防止了不同VPN路由的相互泄漏，同一VPN的各个site可以互通，不同VPN site之间相互隔离。但汇聚和核心层VPN介入CE设备采用的是三层交换机，在三层交换机上配置VLAN虚接口后，各个VLAN是互通的，为了使得各个VPN在本地隔离，因此在汇聚层和骨干层的本地接入交换机上需要启用ACL（访问控制列表），禁止几个本地VLAN(VPN)之间的互访。

对于个别级别高的主机（信息点、业务系统），她需要访问任何一个VPN，可以通过设置一个Super VPN来解决，通过控制Route-Target属性，使其全部接受和发布全部VPN 的路由，这样使其可以访问全部的VPN（业务系统）。

不同VPN之间互通的配置方案

在本次工程中，不同的VPN之间是严禁互相访问的，但考虑到网络和业务的扩展性，日后可能会存在不同的VPN之间互相访问的需求。由于MPLS/VPN的互访是通过RT来控制的，只需要在现在配置的RT基础上进行简单的修改即可。
例如，需要将本来隔离的VPN1和VPN2之间的业务进行互访操作，可进行如下配置更改：

```
VPN1      : export: 64600:10
          : import: 64600:10
          : import: 64600:20
VPN2      : export: 64600:20
          : import: 64600:20
          : import: 64600:10
```

网络的备份和流量分担

网络的备份

网络备份可以同时通过两种方式实现：
1、通过合理的网络方案设计，实现物理链路与物理设备的备份。本次工程中，中心的两台汇聚节点是主备和负荷分担的，在正常工作的情况下，共同分担整个网络的流量，当其中一个失效后，另外一台设备能够承担起所有的流量，保证业务的正常运行；各地市到中心节点的两条广域网链路也是主备的，在主用链路中断的情况下，业务可以通过备用链路传输。对于双链路设备的两条上行链路的使用原则：汇聚层到核心层的两条链路采用负载分担的方式；接入层到汇聚层的链路使用主备切换的方式。
2、应用动态路由协议，做到网络的自动备份。OSPF和BGP 协议，均可以自动地发现两个网络节点之间的迂回路由，并在主用路由不可用时而自动使用备份链路。并且可以通过在路由器上加入策略，控制数据的流向。

- 流量的分担

1、A市的汇聚节点NE16有两个POS接口，两个POS接口实现负荷分担，第一个POS接口接入B市、F市、H市、I市；第二个POS接口接入H市、D市、C市、E市。

2、每个地市到中心节点都由两条主备链路，因此，可以通过设置不同的Cost 值，实现部分地市节点的主链路汇聚到NE16-1，而另外的节点的主链路汇聚到NE16-2，从而达到负荷分担的目的，避免网络流量不均衡。

设备清单

序号	详细描述	数量	备注
1	NE16E	2	省中心两台
2	NE08E	10	各地市一台
3	R4640	43	各接入点一台
4	S6503	2	省中心两台
5	S3526E	20	各地市两台
6	S2016	43	各接入点一台

OSPF向STUB区域产生默认路由的问题

软件开发部 郜中华

问题的提出

当一台ABR路由器在骨干区没有FULL的邻居的时候，应不应该向它所连接的STUB区域产生默认路由的3类LSA？这个问题看似细微，却可以导致网上问题。这里我们抽象出问题的本质，加以研究。

产生问题的根源，就是RFC2328关于ABR的定义不明确，根据RFC，我们可以对ABR做以下几种定义：

定义1：配置了包括骨干区在内的多个区域，骨干区内使能了至少一个接口；

定义2：配置了包括骨干区在内的多个区域，骨干区内至少有一个接口是UP的；

定义3：配置了包括骨干区在内的多个区域，骨干区内至少有一个接口的邻居状态处在FULL。

由于ABR有着很多重要的职责，因此这个问题很关键，尤其当网络拓扑发生变化时，如果ABR的身份不能随之变化的话，就会导致此设备路由计算错误、产生LSA错误等问题。

问题研究

ABR该如何定义

ABR向stub区域产生默认路由的情况

LSA的洪泛范围归纳为3类，放在LSA的LS类型域（field）。这三类是：

区域范围：LSA只在一个OSPF区域内部传播，这类LSA包括Router-LSA、Network-LSA、Inter-Area-Prefix-LSA、Inter-Area-Router-LSA和Intra-Area-Prefix-LSA；

自治系统范围：LSA在整个自治系统内传播，这类LSA只有AS-external-LSA。

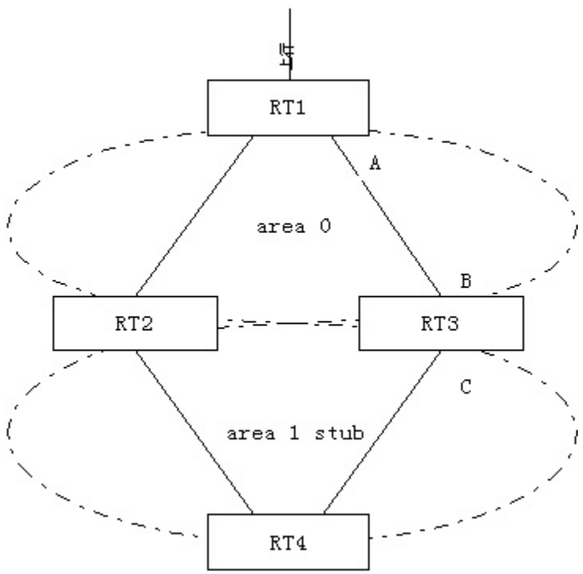


图1 ABR向stub区域产生默认路由

RFC2328规定，STUB区域的ABR要向STUB区域产生默认路由的3类LSA。

如图1所示，RT2和RT3都是ABR，正常情况下他们都向STUB区域产生默认路由的3类LSA，假设RT4优选了RT3的默认路由，那么流量都经过RT3然后到达RT1上行。

起初我们使用定义2来确认RT3的ABR身份，因此，当RT1和RT3的邻居DOWN掉，而RT3的接口B没有DOWN掉的时候，或者RT3在骨干区仍然有UP的OSPF接口（例如：LOOPBAKCK，ATM等），RT3依然执行ABR的工作，于是它仍然向STUB区域产生默认路由的3类LSA，那么RT4算出的默认路由还是指向RT3，它依然将上行流量转给RT3，导致了业务的中断。

于是，定义2造成了这种拓扑下的安全隐患，导致RT4和RT3本身都不能够很好的适应网络拓扑的变迁，路由不能自动切换。

ABR对于SUMMARY LSA的计算的案例

RFC2328里面明确规定，ABR不计算非骨干区的SUMMARY LSA（TRANSIT区域除外）。

因此，图2所示的案例中，RT1上到达1.0.0.0/8的路由是经过RT0的骨干区路由，而不是经过RT2的区域1的路由，即使经过RT2的路由可能会更近。

如果我们使用定义2来确认RT1的ABR身份，那么当RT1和RT0的邻居DOWN掉，而接口A没有DOWN掉的时候，或者RT1在骨干区仍然有UP的OSPF接口（例如：LOOPBAKCK，ATM等）的时候，RT1仍然是ABR，那么它就不计算区域1里面的SUMMARY LSA，那么它的路由就不能自动切换到RT2上，于是RT1到达1.0.0.0/8的路由就算不出来，RT0这个ASBR的路由也就算不出来，导致RT0引入的外部路由也算不出来。

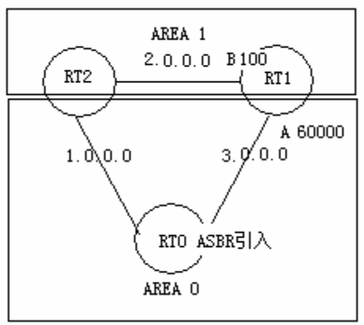
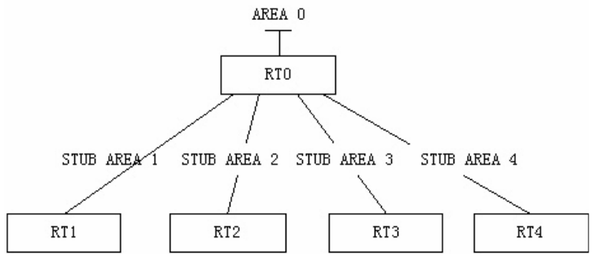


图2 ABR对于Summary LSA的计算

从上面的案例看来，很多情况下，要决定一台设备是不是ABR，它是否行使ABR的职责，都要看它在骨干区内是否有FULL的邻居。从协议角度来看，定义2和定义3都不违背RFC2328的规定，而定义3从OSPF的智能化角度来看，是更合理、更安全一些的。

随之而来的问题

然而按照定义3来定义ABR，在如图需求下，会出现一些的问题：



假设RT0能够通过BGP路由来转发上行流量，RT0以下的部分，都配置成为STUB区域，而RT0的骨干区只配置一个LOOPBACK口，期望能够由RT0向STUB区域产生默认路由的3类LSA。如果按照定义3来做的话，RT0是不会产生的，导致这个组网需求不能够满足。

这种情况可以通过配置来来规避。

方法一：将STUB区域都改成NSSA区域，然后在RT0上配置命令nssa default-route-advertise，这样RT0就会向NSSA区域强制生成默认路由的7类LSA。

方法二：将某几个STUB改成骨干区，使得RT0在骨干区内有FULL的邻居。例如将RT0盒RT1之间的链路配置到骨干区，出于安全考虑，也可以将RT0和RT2的链路也配置到骨干区，这样只有RT0和RT1、RT2的邻居都断了，RT0才不产生默认路由的3类LSA。

OSPF最新进展

软件开发部 潘凝 钱雪彪 郜中华

前言

OSPF是目前应用最为广泛的IGP协议，经过多年的发展已经非常成熟和完善，最新的进展主要在于以下几个方面：性能方面，如快速收敛、提高网络的可扩展性和稳定性；可靠性方面，包括GR(NSF)和NSR；对流量工程的支持，主要是MTR和TE扩展；在IPv6方面也就是OSPFv3的扩展。

性能优化

这部分的内容，象快速收敛、Flooding Reduction，严格意义上已经不能算是最新进展了，Cisco Networkers 2001中就已经提到了，只不过我司一直没有实现而已。

快速收敛

收敛速度是路由协议的重要指标，通过以下对OSPF的改进和增强，可以加快OSPF的收敛速度。

OSPF的HELLO报文发送时间间隔最小为1秒，这个检测速度是比较慢的，Fast Hello允许在1秒内发送多个HELLO报文，从而加快检测速度。现在又出现了一种通用快速故障检测协议BFD可以用来加快OSPF的检测速度。网络发生变化时需要重新生成LSA，OSPF协议规定重新生成一个LSA最少要间隔MinLSInterval，缺省为5秒。减小这个时间间隔可以加快LSA的生成速度加快变化的扩散，但如果时间间隔过小又可能因为频繁变化而造成网络振荡。使用指数退避算法可以很好的解决这一问题，网络比较稳定时时间间隔短，网络不稳定时时间间隔长。

LSA变化会引起SPF计算，如果SPF计算间隔时间过长会影响收敛速度，如果间隔时间过短又会因为频繁变化导致频繁计算而大量消耗系统资源。使用指数退避算法可以很好的解决这一问题，网络比较稳定时时间间隔短，网络不稳定时时间间隔长。

i-SPF（Incremental SPF）是对SPF算法的优化，当网络发生变化时，不需要整个重建SPT树，而是保留不变的部分，仅重建变化的部分，可以加快SPF计算速度。网络发生的变化距离执行计算的路由器越远，SPT树变化就越少，i-SPF计算速度就越快；同时，网络发生的变化距离越远，变化传播到该路由器的时间就越长。因此使用i-SPF可以有效的加快整网的收敛速度。

Flooding Reduction

这项技术应用于稳定的网络，通过减少协议报文流量，减少网络带宽占用，从而增强OSPF网络的可扩展性。

我们知道，OSPF协议规定，LSA如果没有变化每30分钟更新一次，如果1小时还没有被更新那么LSA将被老化。我们会发现，即使网络很稳定不发生变化，还是会产生周期更新的协议报文流量，特别是对于像ISP这样的大型网络，周期更新所消耗的带宽是不容忽视的，也限制了网络的规模。

还记得Demand Circuit吗？Flooding Reduction技术就借用其中的DoNotAge LSA来抑制周期性的flooding。通过生成LSA时置DoNotAge位来使老化失效，这样周期性的flooding就可以取消或者增大为一个可配的时间间隔，从而减少稳定网络的协议报文流量。

这个特性对应的草案为draft-pillay-esnault-ospf-flooding-07。

提高网络稳定性技术

链路故障、路由器故障、网络振荡、LSA周期更新的同步、大量外部路由的引入、Router ID变化等都可能导致同时发出大量LSU报文，产生LSA风暴。LSA风暴会使接收报文无法得到及时处理，LSAck报文得不到及时处理会导致重传，HELLO报文得不到及时处理会导致邻居DOWN，进而产生更多的LSA，形成恶性循环，造成网络不稳定。

为了避免这种情况的发生，提高网络稳定性，可以采取如下一些措施：

1. 将OSPF报文划分优先级，在接收和发送时优先处理HELLO报文和LSAck报文；或者收到任何OSPF单播报文以及点对点链路收到任何OSPF多播报文时都重置router-dead定时器。这样就不会因为HELLO报文得不到及时处理而使邻居DOWN。
2. 使用指数退避算法，包括LSA重传时间间隔和LSA发送时间间隔，都是为了避免LSU报文越来越多的恶性循环。

LSA重传时间间隔的指数退避算法：一条LSA重传次数越多，该LSA的重传时间间隔就越大。

LSA发送时间间隔的指数退避算法：当某邻居未确认的LSA数超过一定阈值，向该邻居发送LSA的时间间隔逐渐增大；当某邻居未确认的LSA数低于一定阈值，向该邻居发送LSA的时间间隔逐渐减小。

3. 限制同时正在建立的邻接关系个数，避免因数据库同步而产生大量报文。

目前这些技术的草案为draft-ietf-ospf-scalability-09。

BFD

BFD（Bidirectional Forwarding Detection）双向转发检测是一个通用的标准化的介质无关和协议无关的快速故障检测机制，可以为各上层协议如各路由协议、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，用来保护两台路由器间的双向转发路径，为上层协议服务，会话建立后如果在检测时间内没有收到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。BFD应该在转发平面实现，这样在控制平面重启时BFD还能正常工作，以便和各协议的GR结合使用。

BFD可以应用于OSPF，对邻居关系进行快速检测。实际上，我们可以把BFD看作通用的Fast Hello。

可靠性

GR

目前的分布式设备软件由控制软件和转发软件两部分组成。主控板上运行控制软件，负责整个设备的控制与管理(包括路由的发现和下刷等)；接口板上运行转发软件，根据控制软件下刷的路由信息维护独立的转发表并负责数据转发业务。它们由各自的处理器进行处理，功能完全独立。

采用控制与转发相分离的分布式结构后，当控制软件重启（由于硬件或者软件故障）或者重新加载（软件升级）时转发业务可以不中断。因为既然控制软件重启或者重新加载不影响转发软件的正常运行，只要在控制软件重启或者重新加载期间网络拓扑保持稳定，正在重启的路由器仍然在转发路线上进行数据转发是完全可行并可靠的。

目前的问题是控制软件重启或者重新加载成功后，所有的路由协议需要重新运行，

根据原有路由协议的正常流程，此时会造成周边设备与本机的邻居关系进行重建，在周边设备上会由于邻居关系的短暂中断而造成网络上的路由振荡以及转发中断问题。

为了解决这个问题，IETF针对各种路由协议（ISIS/OSPF/BGP/LDP等）分别提出了一系列增强协议，通过对原有协议流程的改造，使得在本机控制软件重启时能够通知周边设备在一定时间内将到本机的邻居关系和路由保持稳定，并且在尽量短的时间内协助本机将各种路由信息恢复到重启前的状态，在整个过程中网络路由和转发保持高度稳定，报文转发路径也没有任何改变，从整个网络的层面来看，和没有发生重启事件一样。这种增强协议被统称为GR，有时也被称为NSF。

在整个GR过程中包含以下角色：

GR Restarter： 发生控制软件重启事件的设备。

GR Helper： 和GR Restarter具有邻居关系的周边设备。

GR Restarter和GR Helper必须均具备GR能力，并事先进行好GR能力协商（通常在建立邻居关系时即进行协商）。一般把GR能力协商过程称为GR Session，而协商的内容包括双方是否都具备GR能力，GR Time是多长时间等。一旦GR能力协商通过，当某台具备GR能力的设备控制软件重启时，其通过GR协商的周边设备将在GR Time时间内保持与该台GR Restarter设备的邻居关系和路由不变。

OSPF GR的标准情况比较复杂，目前IETF关于OSPF GR的最新标准是RFC 3623，该标准由Juniper公司制定。

在Juniper的方案成为正式标准之前，Cisco也提出了自己的OSPF GR方案，并且一度成为了IETF的Draft，主要协议有：

draft-ietf-ospf-lls-00 (Cisco)

draft-ietf-ospf-oob-resync-01 (Cisco)

draft-ietf-ospf-restart-01 (Cisco)

以上协议都仅仅针对IPv4，最近针对OSPFv3（IPv6）又提出了一个新的Draft：draft-ietf-ospf-ospfv3-restart-00。

OSPFv2 GR

由于Cisco针对OSPFv2 GR提出的一系列Draft目前已经被IETF废弃，本文仅对RFC 3623进行介绍。

在RFC 3623中，用Grace-LSA来向周边邻居设备请求进入GR处理流程。

Grace-LSA采用Opaque-LSA的格式（关于Opaque-LSA的相关知识请参见RFC 2370），其中Opaque Type为3，Opaque ID为0。

下面以图1为例对RFC 3623中描述的OSPF GR流程进行分析，假设Router A和Router B通过Interface0相连，并且两台路由器在Interface0都使能了OSPF并且能正常运行。

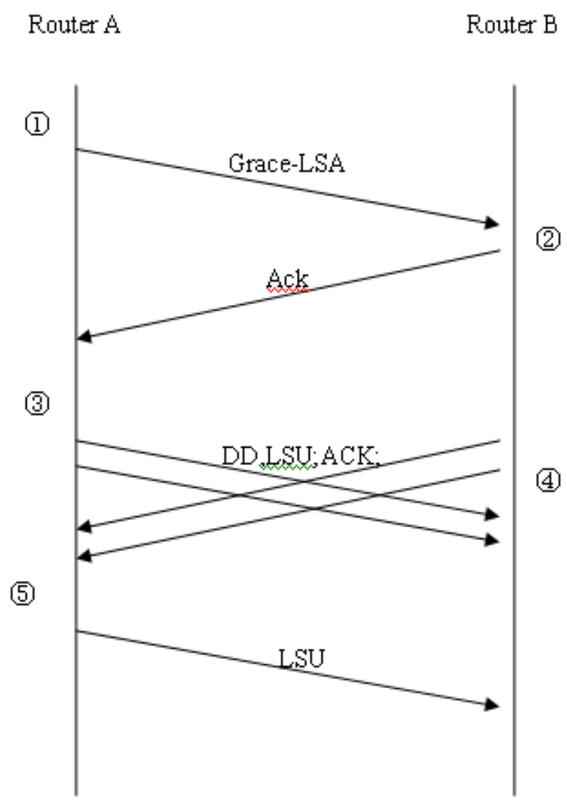


图1 RFC 3623 OSPF GR协议流程图

假设Router A和Router B原来有稳定的OSPF邻居关系，此时Router A重启，将按照以下的流程进行路由信息交互。

① 当Router A的Interface0重新UP时，在Interface0上发送Grace-LSA；

② 当Router B收到Router A发出的Grace-LSA后，回复Ack报文；

③ 当outer A在Interface0上收到Ack报文后就可以进入和传统OSPF协议完全一致的新建邻居流程，邻居状态实现Exstart->Exchange->Loading->Full的变迁。在这个过程中如果收到自己产生的LSA，直接存储下来并置上Stale标志。（这一点和传统OSPF协议的处理流程有较大差别）；

④ Router B也和传统OSPF协议流程一样进行LSDB的同步，邻居状态实现Exstart->Exchange->Loading->Full的变迁。但是在这个过程中除了要完成LSDB同步流程外，其他的所有处理与该邻居状态一直是FULL是没有任何区别的，不能因为该邻居状态的变化而重新生成router-LSA以及network-LSA；

⑤ Router A上所有邻居关系达到FULL，邻居关系和路由信息均已全部恢复，结束GR流程。重新生成并Flood LSA，并Flush那些置了Stale标志而又没有被重新生成的LSA。

OSPFv3 GR

OSPFv3 GR的Draft和RFC 3623一样，都是由Juniper提出的，因此与RFC 3623基本一致，只是由于OSPFv3采用了和OSPFv2完全不同的协议报文格式，因此在GR相关的报文格式定义上需要稍作修改。

在OSPFv3中，Grace-LSA属于link-local scope LSA（11类LSA，可以参见RFC 2740的相关描述），其LS type为0x000B。Grace-LSA中包含的TLV类型定义与RFC 3623完全一致。OSPFv3的GR协议流程也与RFC 3623的定义保持一致。

NSR

NSR（Non-Stop Routing）不间断路由是HA的一个重要特性，我司COMWARE V5版本的OSPF支持NSR特性，而Cisco不支持。

NSR是通过主用主控板在运行OSPF的时候把OSPF的各种实时状态备份到备用主控板实现的，这样当各种原因导致主备倒换以后，备板可以按照原有的OSPF状态继续运行，不影响自己和邻居的路由，主备倒换对邻居来说是透明的，从而实现不间断路由。

NSR与NSF(GR)的区别是：NSF需要邻居支持GR才能实现，而NSR完全通过设备本身的热备份机制实现，不依赖于邻居，当然实现起来也更困难。

流量工程

MPLS TE扩展

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CSPF算法为满足某条流的特定约束条件计算出一条路径，再通过信令协议RSVP-TE或CR-LDP建立LSP隧道。那么如何收集TE关心的链路信息呢？很自然的，我们就会想到链路状态路由协议。OSPF的TE扩展就是为了在区域内扩散TE关心的链路信息而对OSPF协议进行的扩展。

OSPFv2使用10类Opaque LSA来传播TE关心的链路信息，为此定义了一种新的LSA：Traffic Engineering LSA，其Opaque Type为1，Opaque ID用来标识一台路由器产生的不同TE LSA。OSPFv3定义了一种新的LSA：Intra-Area-TE LSA来传播TE关心的链路信息，其LSA Function Code为10，U bit为1，S2S1为01。

OSPFv2 TE扩展的标准是RFC 3630，OSPFv3 TE扩展的最新草案是draft-ietf-ospf-ospfv3-traffic-04。

MTR

MTR（Multi-Topology Routing）多拓朴路由对一个物理拓朴形成多个逻辑拓朴，不同的流使用不同的拓朴走不同的路径，是实现TE的一种方式。

OSPFv2的早期版本RFC1583引入了TOS路由的概念，链路对于不同的TOS有不同的metric，这样对于不同的TOS就形成不同的逻辑拓朴，是不是和MTR的概念类似呢？但是由于TOS路由一直没被应用，因此在后来的版本中被去掉了。MTR于是就又重新使用了TOS路由相关字段，1、3、4、5、7类LSA中的TOS字段被重新命名为MT-ID，TOS Metric字段被重新命名为MT-ID Metric，对于不同的MT-ID可以有不同的metric，这样对于一个物理拓朴就形成了多个逻辑拓朴，由于metric不同，因此在不同拓朴中到同一目的的路由可能是不同的。MTR和TOS路由看起来很相象，那么它们的区别在哪呢？TOS路由中不同的拓朴是与IP报文不同的TOS值相对应的，映射关系简单且固定；而MTR中不同的流与不同拓朴的映射关系非常灵活。

那么OSPFv3如何实现MTR呢？OSPFv3的LSA中没有TOS字段，因此必须想其它的方法。一个方法和下面支持多地址族的第一种方法类似，使用不同的Instance ID来实现不同的拓朴，但是这种方法对于不同的拓朴需要维护不同的邻接关系、不同的LSDB，开销比较大，并不可取。能不能象OSPFv2那样通过一个邻接关系来实现多个拓朴呢？OSPFv3现有的LSA无法满足要求，必须定义新的LSA。首先在Options字段中新增一个MT位，置位表示支持MTR。其次在LS Type字段中增加了一个T-bit，对现有LSA进行了扩展，如图2所示：

```
 0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|U |S2|S1|T |        LSA Function Code          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

图2 扩展LSA Type

对于扩展的LSA，U-bit和T-bit都要置位，因此OSPFv3目前支持的9类LSA经过扩展后其LS Type分别为0xB001、0xB002、0xB003、0xB004、0xD005、0xB006、0xB007、0x9008和0xB009。扩展的LSA都使用灵活的TLV三元组编码，这样除了支持MTR外，将来还可以很容易的支持其它扩展功能。为了支持MTR，在扩展的1、3、4、5、8、9类LSA的TLV中都包含MT-ID和MT-ID Metric字段，从而实现多拓朴。

OSPFv2 MTR目前的最新草案是draft-ietf-ospf-mt-04，OSPFv3 MTR目前的最新草案是draft-ietf-ospf-mt-ospfv3-00。

OSPFv3

多地址族支持

OSPFv3与OSPFv2最大的区别就是拓朴与路由的分离，在OSPFv3中1类和2类LSA不再携带路由信息，而纯粹是网络拓朴信息与地址族无关，因此可以很容易的通过扩展支持多地址族。因为1类和2类LSA中的拓朴信息是与地址族无关的，所以只需将其它LSA中的IPv6前缀信息替换为其它地址族的前缀信息，同时OSPFv3计算路由时是用8类LSA中的链路本地地址作为下一跳的，其它地址族需要将其替换为本地地址族的接口地址。

那么如何区分不同的地址族呢？一种方法是通过OSPFv3报文头的Instance ID区分，为不同的地址族分配不同的Instance ID范围，这样每个地址族都建立自己的邻接关系，有自己的LSDB，计算自己的路由表。另一种方法是利用OSPFv3对MTR的支持通过MT-ID区分，为不同的地址族分配不同的MT-ID范围，这样各个地址族使用同一个邻接关系，同一个LSDB，计算各自的路由表。

需要注意的是，OSPFv3是运行在IPv6之上的，使用IPv6报文封装，因此要求运行其它地址族OSPFv3的链路也必须使能IPv6。OSPFv3多地址族支持两种方法对应的草案为draft-ietf-ospf-af-alt-01和draft-ietf-ospf-mt-ospfv3-00。

其它

OSPF能力扩展

OSPF使用options字段来表示一些可选的能力，一位表示一种能力。目前OSPFv2中options字段的所有位都已经被分配出去，无法再增加新的可选能力。为此对OSPF进行了扩展，定义了Router Information（RI） LSA以支持新的能力。OSPFv2的RI LSA使用Opaque LSA（9、10、11类），Opaque type为4，Opaque ID为0。OSPFv3的RI LSA的LSA Function Code为12，U bit为1，S2S1可以是任意范围，LSID为0。路由器对于一个泛洪范围仅会生成一个RI LSA。RI LSA的内容由TLV三元组组成，其中第一个也是必须包含的TLV是Router Capabilities TLV（Type为1）用来表示路由器的能力，后面可以包含其它类型的TLV来表示路由器其它信息，目前尚未定义其它类型的TLV。RI LSA不可能包含路由器所有的信息，因为要求所有TLV能够包含在一个LSA内，因此某个特定的路由器信息是使用一个新的RI LSA TLV还是生成一种新的LSA需要仔细权衡考虑。

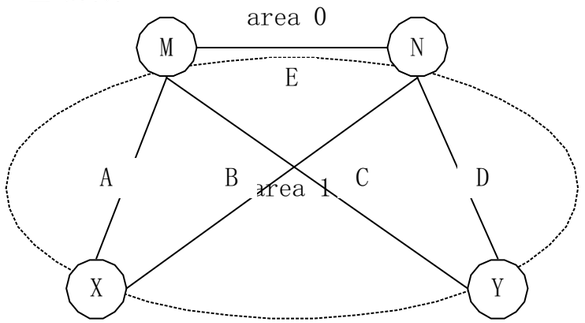
这个扩展目前的最新草案为draft-ietf-ospf-cap-06。

链路的多区域复用

链路多区域复用的目的

什么是链路的OSPF多区域复用？就是一条物理的链路，同时属于OSPF的多个区域，即在多个区域内建立接口、邻居，并达到FULL状态。

链路多区域复用的目的，简单说来，就是要让COST更近的区域间路由变成可用的区域内路由。怎么理解呢？在OSPF体系中，区域间路径不论有多近（COST有多小），总是没有区域内路径优先。因此在实际组网中常常会遇到一些困难，明明是区域间路由更近，但是都被区域内较远的路由替代，而又不能轻而易举的进行网络改造，将区域合并。



多区域复用

例如图3所示，X设备想要访问Y设备，总是要走area 1里面的区域内路径，虽然有可能A->E->D这条经过骨干区域的路径是最近的，但是很遗憾，这条路由不会被使用。这种情况下，如果能够将E这条链路也使能到area 1，让E同时并存在area 0和area 1里面，这样不影响骨干区域的配置，同时又能满足要求。

链路多区域复用的方案

draft-ietf-ospf-multi-area-adj-03提出了一种方法，就是ABR之间宣称使用了IP Unnumber建立了新的拓扑连接，而这个“IP Unnumber的拓扑连接”实际上是使用了已经被别的区域使用的物理链路。就是说，M、N设备之间会宣称建立了一条IP Unnumber的连接，而实际仍然是用了E链路。这样做的好处是，不会使得这个被复用的物理链路E的接口地址作为区域内路由存在于不同的区域，因为这样有悖于OSPF的理念：一个区域是很多子网的集合。这种宣称的实现是通过Router LSA来实现的，例如M设备在区域1的Router LSA里面就会生成如下的LINK：

Link ID = Remote's Router ID
Link Data = IfIndex

那么接口怎么建立呢？邻居怎么发现呢？接口的建立比较好办，配置的时候就将接口创建在相应的区域内即可，就好像SHAM-LINK接口、VLINK接口一样；再说邻居的发现，如果是P2P的物理链路，那么报文直接发送到组播地址AllSPFRouters即可，因为对端只可能有一个设备在运行OSPF；如果是其他物理链路，没有办法，只好配置PEER了，报文相应的也是单播发送到配置的IP地址上了。

那么OSPF原有的流程还有什么地方要修改呢？报文接收肯定要修改，而且更加热闹了，一个接口上收到的OSPF报文，要看它是不是本地接口上的，或者是VLINK接口的？或者SHAM-LINK接口的？还是Multi-Area Adjacency接口的？接口状态机、邻居状态机不需要变化。

Route to Network

策 划：刘宇 陈旭盛 刘炜刚

主 编：边江

编 委：蔡金龙 姜杏春 康媛媛 陆宇翔

Route to Network

网络之路

OSPF专题讨论

相等只是彼此不相同的，

不同一的事物之间的同一。

不相等就是不相等的事物的关系。

因此两者并非彼此毫不相干的方面或观点，

而是一方映现在另一方之中。

所以差异只是反思的差别、潜在的差别或特定的差别。

----黑格尔《小逻辑》