新一代 Segment Routing 流量工程体系 - SR Policy

2012 年以来，思科一直引领着 Segment Routing 技术的发展，并在一些领先运营商的支持下领导标准化工作。

2019 年 5 月，由 Clarence Filsfils 等思科专家所著的《 Segment Routing 详解（第二卷）流量工程》出版，为了尽快与国内读者分享全新的技术，笔者在短短的 3 个月内，快速完成了翻译与审校工作。精致的中文版于 2019 年 9 月与读者见面，由人民邮电出版社出版。

本文作者:

钟　庆 思科系统工程师

苏远超 思科首席工程师

蒋治春 腾讯资深架构师

摘要：本文介绍新一代 Segment Routing 流量工程（SR-TE）体系 - SR Policy。SR Policy 是全新设计的一套 SR-TE 体系架构，完全不同于传统的基于隧道接口的实现方式。基于 SR Policy 之上的一系列创新，例如按需下一跳（ODN）、自动引流、灵活算法（Flex-Algo）、原生算法等，极大地拓展了 SR-TE 的适用范围、简化了部署、优化了性能。基于 SR Policy 的 SR-TE 已得到业界的广泛接受，将在 5G、多云、物联网中得到广泛的应用。

一、 概述

随着 5G、多云、物联网的发展以及行业数字化进程的深入，网络需要服务的范围（从 5G 承载网的接入、汇聚、核心再到骨干网、云数据中心、虚拟化/容器化网元的调度）、规模（海量物联网终端）和颗粒度（区分同一租户的不同应用）都需要提升，同时网络需要能用一种更灵活的方式被上层应用所使用（或者叫驱动）。网络比以往任何时候都需要成为一个平台。

传统网络要应对这些新的需求很困难，网络需要一种创新的传送技术来统一各个不同的域，从而打破孤岛，提供一致的 SLA，并通过统一的接口供上层调用。业界公认的技术就是 Segment Routing，如下图所示：

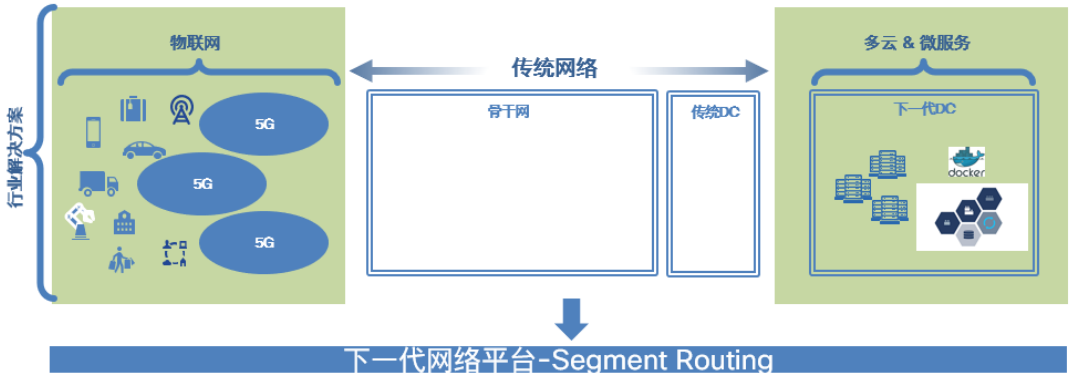


图 1 Segment Routing 使能下一代网络平台

显然，下一代网络平台必须能提供大规模、细颗粒、端到端的 SLA，而这是通过 Segment Routing 流量工程（以下简称 SR-TE）来实现的。那么怎样才是 SR-TE 的正确做法呢？

二、 流量工程回顾

IP 网络设计人员采用两种方式提供 SLA：网络工程和流量工程。

网络工程是设计网络来满足业务流量需求。这需要充分了解流量如何在网络中传送，进行适当的容量规划，采用适当的设备、链路、互联拓扑和路由策略。网络工程包括规划设计、采购、实施等一系列流程，通常的周期以月计。网络工程是基础，但如果只是依赖于网络工程提供 SLA，那么时效性、灵活性无法满足业务发展需求。

流量工程是使特定流量按照优化目标经由网络中特定路径（通常是非 IGP 最短路径）转发。流量工程支持即时部署、即时生效。流量工程不一定是隧道，事实上 BGP 流量工程、IGP 多平面设计甚至包括策略路由都是常见的流量工程手段。但必须指出的是，基于 Native IP 的流量工程，一般只能实现单跳的控制。因此纯 IP 的流量工程难以实现下一代网络平台所需要提供的端到端 SLA 路径。

2.1 RSVP-TE 的不足

如何在 IP/MPLS 网络上提供 SLA 路径？这长久以来是一个挑战。笔者在 2004 年设计建设中国电信 CN2 时，MPLS 流量工程就是重点关注的内容，并且最终在 CN2 上部署了基于 RSVP-TE 的 MPLS FRR（并未使用 RSVP-TE 疏导流量）。

RSVP-TE 已经出现了 20 年，在 SR-TE 出现之前，RSVP-TE 一直是 IP/MPLS 网络上可用于提供 SLA 路径的最主要流量工程手段。但 RSVP-TE 被设计出来的时候，IP 并非像今天这样一统天下，事实上“电路交换”(ATM/帧中继)仍然是当时的主流，因此 MPLS 设计时考虑了很多如何兼容 ATM/帧中继的功能（MPLS 运行在 ATM/帧中继交换机上，而不是运行在 IP 设备上），或者换个角度说，是如何用 MPLS 模拟电路交换。MPLS 标准里面定义的封装如下图所示，可见 IP over MPLS over Ethernet，在 20 年前并非主流，而只是若干封装方式中的一种[1]。

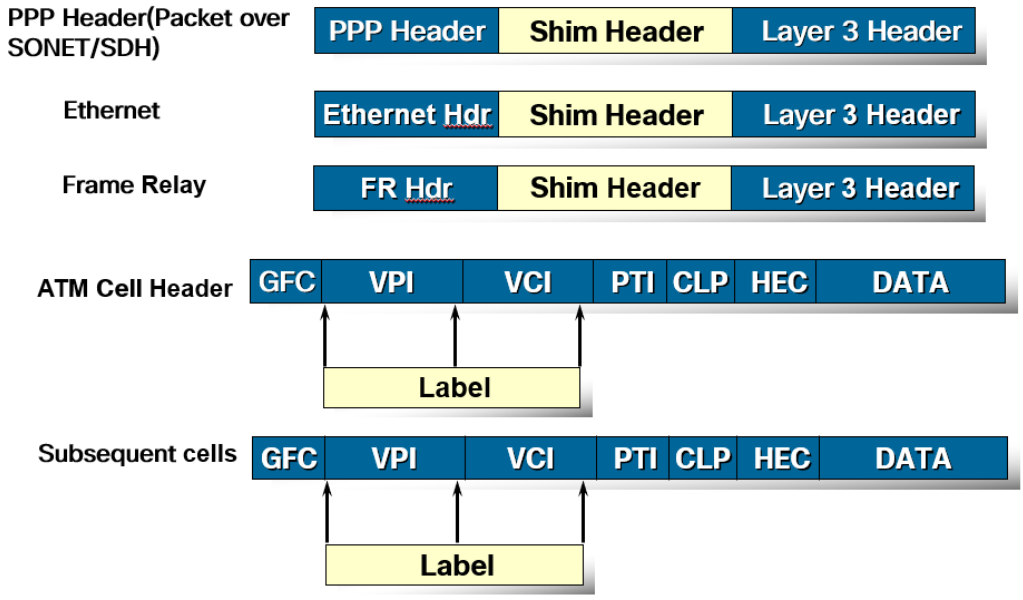


图 2 MPLS 标准定义的封装格式

[1] 目前业界主要使用 MPLS over Ethernet 这种封装方式，MPLS over PPP 只有运营商网络中还有少量使用（POS 链路），而 MPLS over ATM/帧中继都已经在现网中消失了

作为 MPLS 体系下流量工程的主要手段，RSVP-TE 无法摆脱 MPLS 体系的时代局限，因此 RSVP-TE 本质上也是模拟电路交换的思路，这使得它具有若干天生的缺陷，而这些天生的缺陷随着 IP 一统天下，愈发显得格格不入：

- RSVP-TE 本质是基于 ATM/帧中继电路的思想，用 IP 来模拟电路，而并非基于 IP 优化。其中一个典型表现是编码路径时需要把路径上沿途经过的每台设备的每跳接口的地址/标签都包括进来，而不能像 SR 一样使用 Prefix-SID，通过少数 Segment 即可编码路径。
- RSVP-TE 需要建立和维持全网状互联隧道，数量是 $k \times N^2$ 条，其中 N 为网络中节点数量，k 为等价路径数量。这是一个很严重的可扩展性问题（RSVP 的软状态协议特征，更加剧了问题的严重性），事实上这对所有 RSVP-TE 用户都造成了十足的困扰，有些用户甚至最后不得不拆除所有的 RSVP-TE 隧道。
- RSVP-TE 难以实现跨域。这极大地限制了流量工程的适用范围。
- RSVP-TE 缺少对 ECMP 的支持，必须在源和目的地之间建立多条隧道才能实现负载分担。
- RSVP-TE FRR 不能保证被保护前缀的备份路径是最优的。
- RSVP-TE 并没有解决引流的问题，需要依赖于自动路由（autoroute）、静态路由、策略路由等方式实现引流，而这些引流方式，要么会影响性能，要么颗粒度过大。

根据我们的不完全统计，自从 RSVP-TE 技术出现 20 多年以来，只有不到 10%的运营商使用了 RSVP-TE，他们中绝大多数部署 RSVP-TE 是为了使用快速重路由功能（与 CN2 情况类似），利用 RSVP-TE 进行流量调度和带宽管理控制的实际部署案例很少，运营上称得上成功的则更少。而且基本没有跨域 RSVP-TE 的实际部署案例。

基于 20 年前的 RSVP-TE 技术来建设面向未来 10 年的下一代网络平台，显然并不可行。

2.2 SR-TE 的优势

思科院士 Clarence Filsfils 在 2013 年发明了 Segment Routing（SR）技术。SR 具有源路由和状态只存在于边缘的特点，使其可以支持超大规模的流量工程，同时原生就适合与 SDN 结合，实现应用驱动的网络。

SR 其中一个关键功能是 SR-TE。SR-TE 将用户的意图（延迟、不相交路径、SRLG、带宽等）转换为 Segment 列表（每个 Segment 代表特定的操作，Segment 列表是指这些 Segment 的有序列表），然后将 Segment 列表编程到单域/跨域网络的边缘设备上，同时引导流量至 Segment 列表所对应的路径上，从而实现“基于意图的网络(IBN)”，完成传统网络向下一代网络平台的演进。

正因为 SR-TE 具有上述好处，因此在短短几年内已经得到了广泛的部署，并且成为支撑 5G、多云、物联网发展的标准传送技术。

然而早期的 SR-TE 体系存在一定的不足，需要演进至全新的 SR-TE 体系 SR Policy，下文将说明这一点。

三、 SR-TE 的两种体系-隧道接口 vs SR Policy

3.1 隧道接口

由于 RSVP-TE 已经在业界使用多年，其“隧道接口”概念被很多人所熟知，因此 SR-TE 最初的实现（包括目前大多数厂商的实现）还继续采用了隧道接口体系。

对于简单的 SR-TE 功能，基于隧道接口体系实现起来比较简单，在 SR-TE 的导入期，能满足大多数用户的需要。其引流方式也沿用 RSVP-TE 的方式，用户也比较习惯。

但是，也正是由于隧道接口体系继承了 RSVP-TE 的实现，使得这种体系下的 SR-TE 实现存在着明显不足：

- 隧道接口和引流两者是分开实现的，引流方式往往非常麻烦且造成性能损失；
- 往往需要预先配置隧道，在无法明确隧道终点的情况下，只能是部署全网状的隧道，造成可扩展性问题；
- 绝大多数厂商在沿用隧道接口体系的同时，也沿用了 RSVP-TE 的电路算法[2]，表现为只能用 Adj-SID 编码路径，而无法使用 Prefix-SID 编码

[2] 无论是在传统的隧道接口体系还是 SR Policy 体系下，思科的 SR-TE 实现都采用 SR 原生算法

路径，导致无法利用 IP ECMP 的能力，并且造成 Segment 列表长度过长，容易超出一些低端设备的支持能力；

- 隧道与路径一对一的关系，因此要配置多个隧道接口用于实现在多条路径上的（等价/不等价）负载均衡，配置繁琐且影响扩展性；
- 隧道接口占用了设备上的逻辑资源，使得设备能支持的 SR-TE 数量有限
- 不支持一些新的 SR 功能例如灵活算法（Flex-Algo）、性能测量（Performance Measurement）等[3]。

3.2 SR Policy

为了解决传统隧道接口体系存在的问题，并为 SR-TE 后续创新（包括 SRv6）打造更加坚实的基础，思科在 2017 年提出全新的 SR-TE 体系：SR Policy。SR Policy 完全抛弃了隧道接口的概念，是重新设计的一套 SR-TE 体系。

SR Policy 通过解决方案 Segment 列表来实现流量工程意图。Segment 列表对数据包在网络中的任意转发路径进行编码。列表中的 Segment 可以是任何类型：IGP Segment、IGP Flex-Algo Segment、BGP Segment 等。

从本质上讲，SR Policy（SR-TE）是 Segment 列表，而不是隧道接口。这是 SR 设计的初衷。

SR Policy 由以下三元组标识:

- 头端（Headend）： SR Policy 生成/实现的地方；
- 颜色（Color）： 是任意的 32 位数值，用于区分同一头端和端点对之间的多条 SR Policy；
- 端点（Endpoint）： SR Policy 的终结点，是一个 IPv4/IPv6 地址。

颜色是 SR Policy 的重要属性，通常代表意图，表示到达端点的特定方式（例如低延迟、低成本并排除 SRLG 等）。这个新的基本概念用于实现 SR-TE 的自动化。

基于 SR Policy 的 SR-TE 将 BGP 路由置于解决方案的核心，通过对业务路由进行着色实现自动生成 SR Policy 和自动引流至 SR Policy:

[3] 不同厂商实现上有所差别，这里指思科的实现情况

- 基于颜色模板和端点动态地生成 SR Policy 称为按需下一跳（On-Demand Next-hop，ODN）。这是相当重大的简化，所有的边缘节点只需要配置少量相同的模板，不再需要预先配置任何全网状互联隧道；
- 将 BGP 路由安装到 SR Policy 上称为自动引流（Autosteering）。这又是相当重大的简化，不再需要进行复杂和繁琐的引流配置。而且流量引导对转发性能没有影响，流量控制的颗粒度更为精细。

SR Policy 还集成了性能测量、OAM、计数器和遥测（用于自动生成流量矩阵）等功能。

SR Policy 体系和隧道接口体系对比如下表所示（相关 SR Policy 功能解析详见本文第四部分）：

表 1 SR Policy vs 隧道接口

项目	SR-TE（SR Policy）	SR-TE（隧道接口）
网络状态	只存在于边缘设备上	只存在于边缘设备上
生成方式	可通过 ODN、控制器自动生成，或者手工配置生成	控制器自动生成或者手工配置生成，不支持 ODN
网络中的状态数量	少，SR-TE 可以按需生成，用完拆除	多，在无法确定端点的情况下，只能是生成并维持全网状隧道
设备支持的 SR-TE 数量	高	受限，隧道接口需要消耗接口资源
流量引导	基于业务路由自动引流； 不影响转发性能；基于前缀或者前缀流分类结果引流	不支持自动引流； 使用自动路由或策略路由进行引流；转发性能受影响；引流的颗粒度基于下一跳，难以支持基于流的引流
多域支持	支持	支持
多条路径 ECMP	通过 Prefix-SID 或者为同一 SR Policy 的同一候选路径的多个不同 Segment 列表设置权重实现	需要配置多个隧道接口实现
计算算法	针对 IP 优化的 SR 原生算法，最大化利用 ECMP、最小化 Segment 列表长度	绝大多数厂商使用电路算法，只能用 Adj-SID 编码路径，不能利用 ECMP，Segment 列表长

项目	SR-TE（SR Policy）	SR-TE（隧道接口）
灵活算法 (Flex-Algo)	支持	不支持[4]
性能测量	基于链路和 SR Policy 的性能测量（时延、丢包、存活）	不支持 [5]

SR Policy 直接适用于 SR 的不同实现：SR-MPLS（MPLS 数据平面）或者 SRv6（IPv6 数据平面）。在本文中，为简单起见，我们提出的所有概念和说明都是基于 SR-MPLS 的，但是所有的概念和原则也都直接适用于 SRv6，并将在 SRv6 中支持。

SR Policy 是不忘初心、面向未来、全新设计的 SR-TE 体系，是正确的 SR-TE 做法!

四、 SR Policy 的关键创新

4.1 SR Policy 模型

1. 模型概述

如前所述，SR Policy 由（头端，颜色，端点）三元组标识。在给定的头端节点上，SR Policy 由（颜色，端点）二元组标识。

SR Policy 的候选路径代表将流量从相应 SR Policy 头端传送到端点的特定方式。每条候选路径（Candidate Path）有一个偏好值（Preference）。路径的偏好值越高则越优选。

SR Policy 具有至少一条候选路径，其中具有最高偏好值的有效候选路径是活动候选路径。

SR Policy 的 Segment 列表是其活动路径的 Segment 列表。每条候选路径可以具有一个或者多个 Segment 列表，每个 Segment 列表具有关联的负载均衡权重。引导至此路径的流量根据权重比例，在所有的有效 Segment 列表之间进行负载均衡。在 SR-MPLS 中，Segment 是 MPLS 标签，Segment 列表是 MPLS 标签栈。对于被引导至 SR Policy 的数据包，此标签栈（Segment 列表）将被压入到数据包报头中。

[4][5] 不同厂商实现上有所差别，这里指思科的实现情况

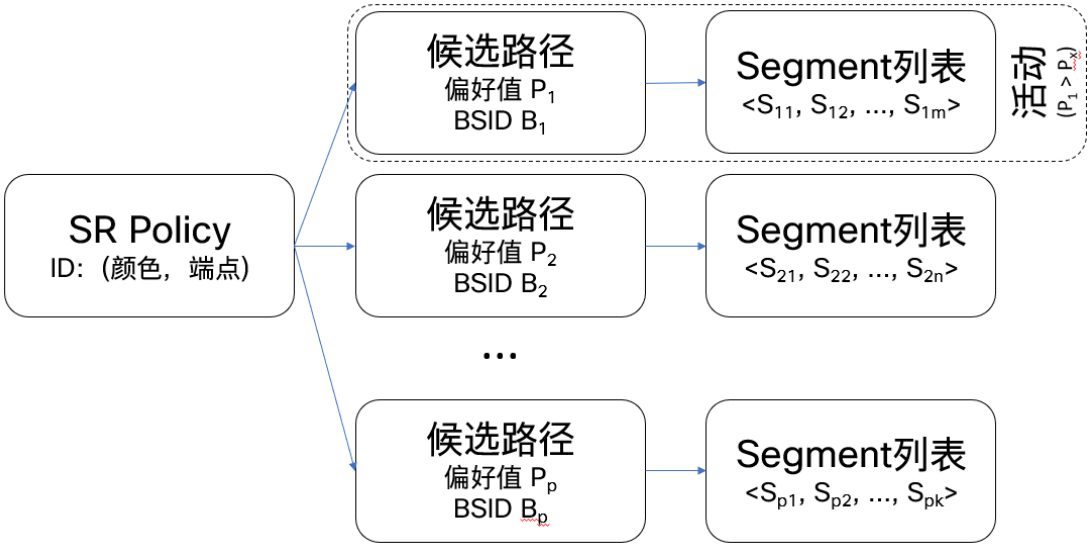


图 3 SR Policy 模型

2.候选路径

候选路径可以分为显式候选路径和动态候选路径。

显式候选路径是由操作员或者控制器计算出源路由路径，并向头端节点显式地告知要使用的路径；头端节点只需要简单地接收并使用 Segment 列表。

动态候选路径则由操作员或者应用简单地表达意图，头端节点或控制器将意图动态转换为 Segment 列表，并按需更新 Segment 列表以动态响应任意的网络变化，保证始终满足意图。计算路径需要两个必要元素：包含网络所有必要信息的数据库和在相关信息上应用算法以解决最优化问题的计算引擎。计算引擎的核心是 SR 原生优化算法，该算法以最大限度利用 ECMP 和使用尽可能少的 Segment 为宗旨。

每条候选路径可以通过不同的方式学到，例如本地配置、NETCONF、PCEP 或者 BGP。SR Policy 的活动路径根据候选路径的有效性和偏好值来选择，候选路径的来源不影响选择过程。

3.BSID（Binding-SID）

候选路径还具有 BSID 属性。在 SR-MPLS 中，这是在头端节点上绑定候选路径的 MPLS 标签。SR Policy 的 BSID 是活动候选路径的 BSID。

BSID 在 SR Policy 的引流中起着重要作用。头端将 BSID 绑定到对应的 SR Policy，并将 SR Policy 作为 BSID 标签的 MPLS 重写条目安装在转发平面

中。例如 SR Policy 的 BSID 为 B1，活动路径的 Segment 列表为<S1，S2>，则头端节点在转发表中为该 SR Policy 安装以下条目：

- 入向标签：B1；
- 标签操作：弹出 B1，压入<S1，S2>；
- 出口：S1 的出口信息（出接口和下一跳）。

4.失效及回退

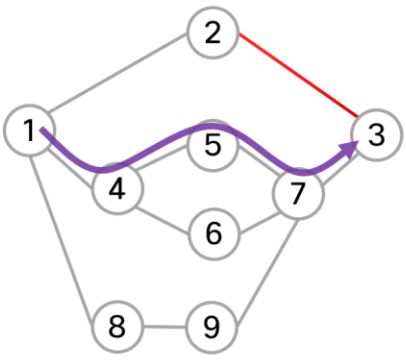
一旦 SR Policy 的候选路径不具有有效的 Segment 列表，则此候选路径变为无效；当 SR Policy 所有的候选路径都无效，则此 SR Policy 变为无效，默认情况下将删除 SR Policy 的转发表条目，流量回退到默认转发路径（通常是 IGP 最短路径）。

4.2 SR 原生算法

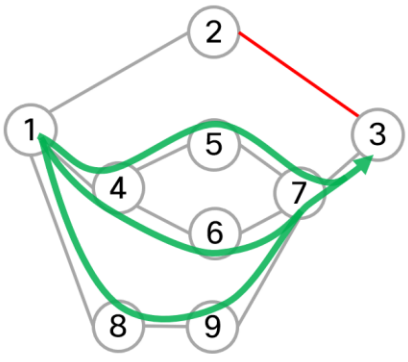
前面我们提到了 RSVP-TE 及绝大多数的基于隧道接口体系实现的 SR-TE 都是基于电路的方式计算和编码路径，而重新设计的 SR Policy 体系，是完全基于 IP 并且针对 IP 优化的，因此自然而然也会采用 SR 原生的方式计算和编码路径。

下面通过一个例子对比说明电路算法和 SR 原生算法的区别。假设要求都是计算和编码从节点 1 去往 3 且避免节点 2 到节点 3 链路的路径。下图左边是电路算法计算出的路径及流量分担情况，右边是 SR 原生算法计算出的路径及流量分担情况。

部署从节点1去往节点3、且避免节点2到节点3链路的TE路径



SR-TE出现之前的算法基于电路
CSPF => 不支持ECMP的路径
在SR-TE中重用此算法不好
Segment列表: <24014, 24045, 24057, 24073>
不支持 ECMP，Segment列表长，针对电路优化



需要SR原生TE
业界认可的创新 - SIGCOMM 2015

Segment列表: <16007, 16003>
支持ECMP，Segment列表更短，针对IP优化

图 4 SR 原生算法

如上图所示，电路算法采用路径沿途节点的 Adj-SID 编码路径，无法利用 IP 网络中大量存在的 ECMP，效率低下；而且 Segment 列表长，很多传统设备或者低端设备无法支持。相反地， SR 原生算法最大化 ECMP 且最小化 Segment 列表长度，因此大大提高了流量转发效率，也易于在现网部署。

SR-TE 需要 SR 原生算法!

4.3 自动引流

4.3.1 自动引流架构

自动引流解决方案的核心是标记的概念：出口 PE 通告 BGP 业务路由时（或者入口 PE 接收路由时），对路由进行着色，用于表示业务路由所需的 SLA。

当头端节点接收到已着色业务路由时，如果 BGP 颜色团体属性和下一跳与 SR Policy 的颜色和端点相匹配，则 BGP 安装此路由，将其解析到 SR Policy 的 BSID。

自动引流不仅支持基于目的地前缀的引流，还支持基于流的引流。基于流的引流使头端可以基于流分类的结果（例如 DSCP 值），将与同一目的业务路由匹配的不同流引导至不同的 SR Policy，实现更精细的引流。

自动引流适用于多域网络。

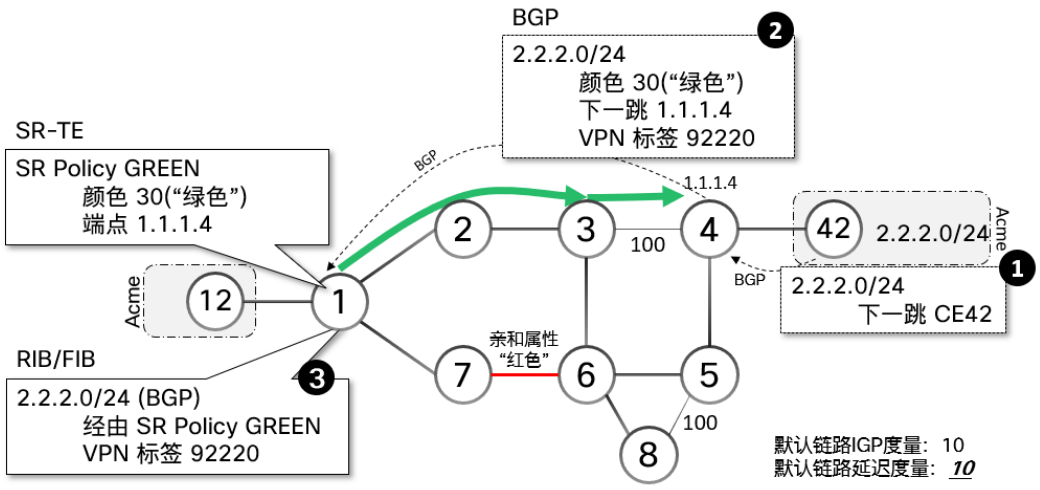


图 5 自动引流

在上图中，出口节点 4 将 BGP 路由 2.2.2.0/24 着色为“绿色”（对应的颜色值为 30），当头端节点 1 收到此业务路由后，发现此路由与本地配置的 SR Policy GREEN(“绿色”，节点 4)匹配（假设此 SR Policy 表示从节点 1 到节点 4 的低时延路径），因此在本地转发表中把此 BGP 前缀的下一跳设置为 SR Policy GREEN 的 BSID，则去往 2.2.2.0/24 的流量会被自动引导至低延迟 SR Policy GREEN。

需要说明的是，图中显示的是节点 1 和节点 4 位于相同自治域的情况，节点 1 和节点 4 位于不同自治域时，自动引流仍然适用。对于节点 1 和节点 4 位于不同自治域的情况，需要确保业务路由的下一跳在传播过程中保持不变（Nexthop Unchanged），同时需要借助控制器进行跨域路径计算。

4.3.2 基于流的自动引流

传统的流量工程都是一维的：或者是基于目的地，或者是基于业务等级。如下所示：

表 2 基于目的地的流量工程

业务等级	目的地网段 A	目的地网段 B
业务等级-金	TE 策略 1	TE 策略 2
业务等级-银		

表 3 基于业务等级的流量工程

业务等级	目的地网段 A	目的地网段 B
业务等级-金	TE 策略 1	
业务等级-银	TE 策略 2	

随着云业务的发展，基于流的流量工程越来越成为必备能力（例如为同一租户的不同应用提供不同的 SLA，又或者是实现 Overlay 和 Underlay 交接时，详见笔者文章 Linux SRv6 实战 第三篇）。

与传统的一维流量工程不同，基于流的流量工程粒度划分是二维的，即目的地+业务等级。如下表所示：

表 4 基于流的流量工程

业务等级	目的地网段 A	目的地网段 B
业务等级-金	TE 策略 1	TE 策略 2

业务等级	目的地网段 A	目的地网段 B
业务等级-银	TE 策略 3	TE 策略 4

相比于传统隧道接口体系，SR Policy 能更灵活地实现基于流的流量工程。我们来分析下两者实现基于流的流量工程时的异同。

下表是隧道接口实现基于流的流量工程时所用到的 ID/属性：

表 5 隧道接口实现基于流的流量工程

实现机制	隧道接口 ID	隧道目的地	业务等级 (service-class)
TE 策略 ID or 属性	ID	属性	属性
是否用于基于目的地的引流	否	是	否
是否用于基于业务等级的引流	否	否	是

基于隧道接口设置转发策略时的三要素包括：隧道接口 ID（tunnel interface ID）、隧道目的地（tunnel destination）和业务等级（service-class）。

为了实现基于流的流量工程，两台设备之间必须建立多个隧道组，每一个隧道组对应着一组业务目的地网段，采用单独的隧道目的地（对应于隧道尾端设备上不同的 loopback 地址），用于区分目的地；属于同一隧道组的多条隧道共享相同的隧道目的地，采用不同的隧道接口编号予以区别，每个隧道接口赋予不同的业务等级值，即一条隧道对应着一个业务等级。所以隧道接口体系下是采用不同 loopback 地址+不同隧道的方式实现基于流的流量工程。

显然，这种做法增加了地址规划、部署和运维的负担，大多数客户并不愿意在设备上配置多个 loopback 地址用于流量工程，也不愿意维护数量众多的隧道。

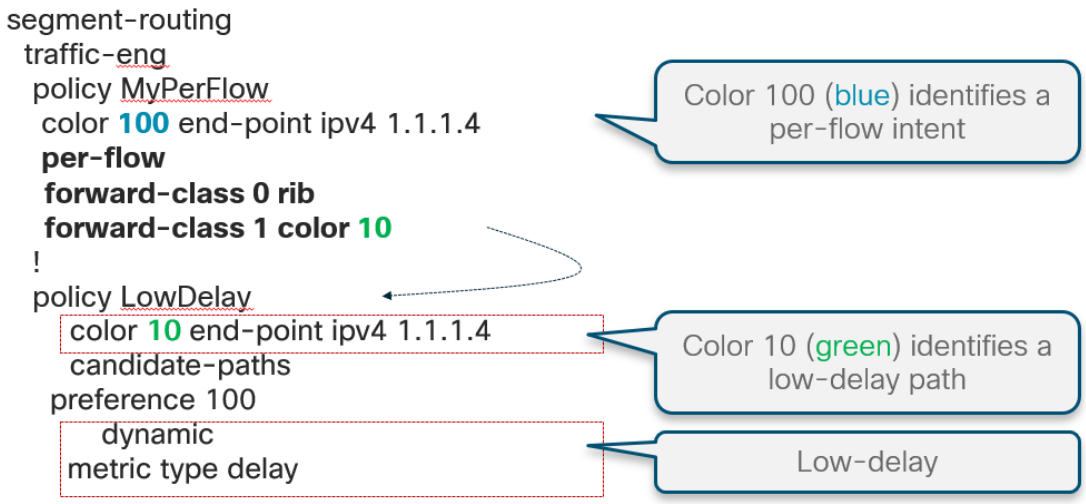
下表是 SR Policy 实现基于流的流量工程时所用到的 ID/属性：

表 6 SR Policy 实现基于流的流量工程

实现机制	SR Policy 颜色	SR Policy 端点	转发等级 (forward-class)
TE 策略 ID or 属性	ID	ID	属性
是否用于基于目的地的引流	是	是	否
是否用于基于业务等级的引流	否	否	是

SR Policy 设置转发策略时的三要素包括：SR Policy 颜色、SR Policy 端点和和转发等级。其中颜色和端点标识了 SR Policy，转发等级则是 SR Policy 的一个重要属性，其用途与隧道接口体系中的业务等级相同。

两台设备之间可以建立多组 SR Policy，每一组 SR Policy 对应着一组业务目的地网段，不同组的 SR Policy 可以采用相同的端点（不需要额外的 loopback 地址），只需要为不同目的地设置不同的颜色即可；为同一组中的多条（子）SR Policy（端点相同但颜色不同）赋予不同的转发等级，一条 SR Policy 对应着一个业务等级。所以 SR Policy 体系下是采用不同颜色+不同（子）SR Policy 的方式实现基于流的流量工程。SR Policy 实现基于流的流量工程的典型配置如下所示：



可见，正是因为 SR Policy 抛弃了传统隧道接口体系下的一维体系，建立了二维体系，才能灵活地、可扩展地支持基于流的流量工程。

4.4 按需下一跳

按需下一跳（ODN）解决方案基于头端节点上指定路径要求的模板，自动生成满足意图的 SR Policy。无需手动配置 SR Policy，也无需引入 SDN 控制器。

与自动引流一样，按需下一跳的关键也在于业务路由的着色。入口 PE 根据所需的 SLA，预先配置一组路径模板，每个模板对应一种指定 SLA 的颜色。模板中规定了所生成候选路径的特征，例如偏好值、是否动态生成、如果是动态生成需要优化哪种度量、有什么约束条件等。

出口 PE 通告业务路由时，根据 SLA 的需求，为其附加颜色扩展团体属性。

如果入口 PE 配置了颜色 C 的 ODN 模板，一旦它接收到至少一条具有颜色 C 和端点 E 的 BGP 业务路由，BGP 进程则向 SR-TE 请求生成 SR Policy（C，E）的 ODN 候选路径。如果此 SR Policy 已经存在，则 ODN 候选路径加入到 SR Policy（C，E）的候选路径中；如果尚未存在，则动态生成此 SR Policy。

一旦 SR Policy 生成了 ODN 候选路径，则按照常规执行 SR Policy 候选路径选择过程、将路由写入转发表，执行自动引流。请注意：候选路径的来源并不影响候选路径选择结果。

ODN 解决方案也适用于多域网络，此时 ODN 模板中需指定由控制器计算 ODN 路径。

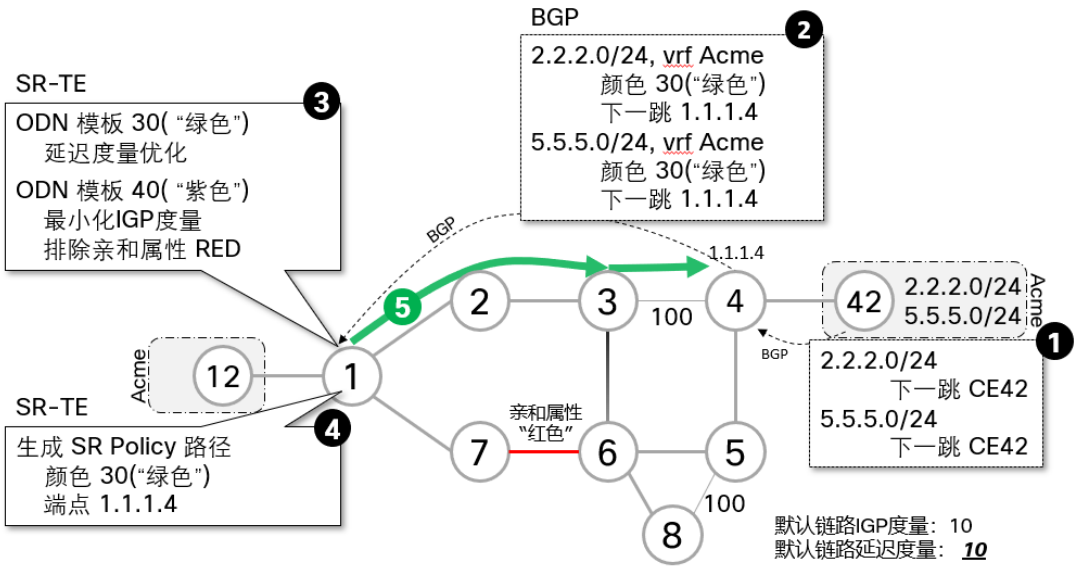


图 6 按需下一跳

在上图中，出口节点 4 将 BGP 路由 2.2.2.0/24 和 5.5.5.0/24 着色为“绿色”（对应的颜色值为 30），当头端节点 1 收到此业务路由后，发现此路由与本地配置的 ODN 模板(“绿色”)匹配（假设此模板表示低时延路径），则头端节点 1 生成低时延 SR Policy 候选路径去往 1.1.1.4。

当头端上所有与“绿色”匹配的 BGP 路由被撤销后，头端将拆除 ODN 生成的候选路径，如果此时没有其他可用的候选路径，头端还将拆除相应的 SR Policy。

这里有一个细节，头端节点 1 的 ODN 模板只指定了颜色，但并没有指定端点，头端节点是怎么知道按需生成的候选路径要去往何处呢？答案是：去往着色路由的 BGP 下一跳，在上图中，2.2.2.0/24 和 5.5.5.0/24 的 BGP 下一跳都是 1.1.1.4，因此头端节点或者控制器需要计算的路径是去往 1.1.1.4 且满足 ODN 模板所规定约束条件（低时延）的候选路径。再次强调，在跨域环境中，要使 ODN 发挥作用，需要设置 Nexthop Unchanged。

4.5 灵活算法

灵活算法(Flex-Algo)功能是 SR-TE 架构的固有组件，早在 Clarence 第一次关于 SR 的公开演讲中就谈到了 Flex-Algo 的概念。

1．Prefix-SID 算法

我们回顾一下 ISIS Prefix-SID 的格式，如下图所示：

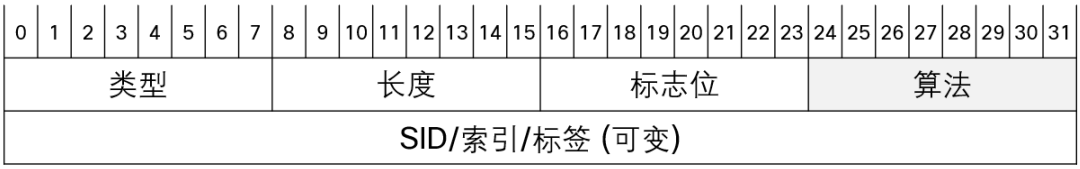


图 7 ISIS Prefix-SID 格式，包含“算法”字段

如上图所示，ISIS Prefix-SID 包含“算法”字段（OSPF 类似，不再赘述），这是在 SR 设计的第一天就确定的：SR IGP Prefix-SID 不单与前缀关联，还与算法关联。也就是说，同一个 IGP 前缀，例如同一个 loopback 地址，可关联多个 Prefix-SID，每个 Prefix-SID 对应一种算法。

“算法”字段总共 8 位，用 0-255 的数字表示不同的算法：

- 算法 0 至算法 127 保留由 IETF 进行标准化，目前在 RFC 8402 中定义了两个标准算法标识符：算法 0（基于 IGP 链路度量的 SPF 算法）和算法 1（基于 IGP 链路度量的严格 SPF 算法）。默认算法为算法 0。
- 算法 128 至 255 可以由操作员自定义，称为 SR IGP 灵活算法，简称为 Flex-Algo。

那为何之前大家都很少关注 Prefix-SID 所关联的算法呢？这是因为所有 SR 节点默认都参与算法 0，也就是常规的 SPF 算法，呈现出来的就是大家熟悉的常规 Prefix-SID 的行为：遵循去往所关联前缀的、支持 ECMP 的 IGP 最短路径转发。而 Flex-Algo，则是提供了一个手段，让意图对应于算法!

2. Flex-Algo 定义

Flex-Algo 的定义包括三个要素：计算类型、优化目标和约束条件。计算类型即用来计算路径的方法，目前已经定义两种计算类型：类型 0（SPF）和类型 1（严格 SPF）；与动态候选路径计算类似，优化目标指特定度量类型的最小化；约束条件指在计算去往 Flex-Algo 每个 Prefix-SID 的路径中必须遵守的限制。

节点针对参与的算法执行路径计算时，首先在拓扑中删除未参与此算法的节点、根据算法约束条件必须避免的资源和不具备算法所使用度量的链路，生成用于路径计算的拓扑；然后根据计算类型和优化目标计算路径。

3. Flex-Algo 好处

Flex-Algo 功能能够用单个 Segment 提供带有约束条件的动态路径，降低了节点压入标签深度的要求，并且其 TI-LFA 路径也遵循相同的约束条件和优化目标；同时沿途中间节点失效不会影响 Flex-Algo Prefix-SID 的可用性，这大大简化了 SR-TE 的高可用性设计。

SR-TE 路径计算中可以自动考虑使用特定的 Flex-Algo Prefix-SID，Flex-Algo Prefix-SID 也可以被包含在任意的 SR Policy 中。SR-TE 的 ODN 和自动引流组件也可以原生地利用 Flex-Algo。

下图显示了采用 Flex-Algo 来实现双平面不相交路径的例子:

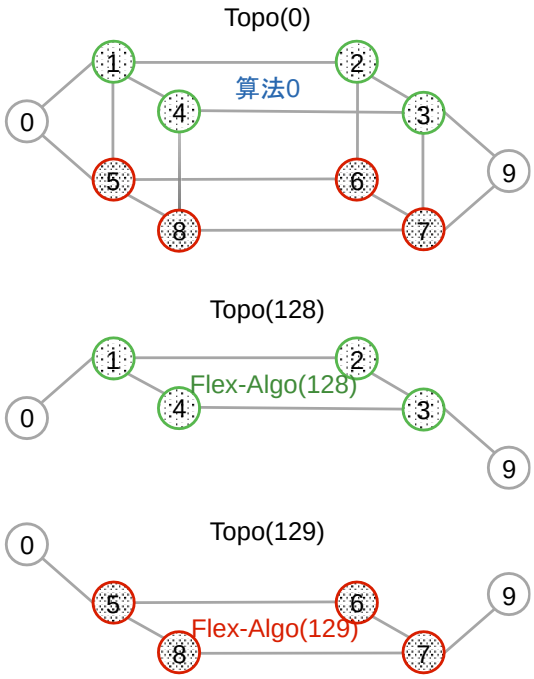


图 8 Flex-Algo 实现双平面不相交路径

图中节点 0-9 都启用了 SR，都运行算法 0。节点 0/1/2/3/4/9 运行算法 128。节点 0/5/6/7/8/9 运行算法 129。最上方的图显示了默认算法 0 Prefix-SID 路径，使用了两个平面的所有可用路径；中间是 Flex-Algo（128）Prefix-SID 路径，只使用上层平面路径；下方是 Flex-Algo（129） Prefix-SID 路径，只使用下层平面路径。如果在数据包上压入了 Flex-Algo（128）Prefix-SID，则数据包只会经由上层平面转发，即使上层平面内发生了故障，数据包的 TI-LFA 备份路径也只会在上层平面，而不会转到下层平面。

总之，Flex-Algo 是计算力和空间换取网络简化的做法。在今天以及未来，网络所连接的人、物、流程则成指数型增加，复杂性也成指数型增加，而单台的设备能力则根据摩尔定律在不断提升，因此 Flex-Algo 是完美地驾驭了这两个趋势，其价值将得到越来越多的认可。

4.6 性能测量

SR Policy 解决方案集成了基于链路和基于 SR Policy 的性能测量（Performance Measurement）。SR 性能测量功能提供了一个通用的框架，对不同网元（链路、SR Policy、节点）的各种性能特性（延迟、丢包、存活性）进行动态测量。性能测量功能可用于测量网络的实际性能度量，并作出实时动态反应。

性能测量探测数据包可使用三种方式进行编码：双向活动测量协议（TWAMP，RFC 5357）；MPLS GAL/G-Ach（RFC 6374）；IP/UDP。

性能测量度量以扩展 TE 链路度量的形式在 ISIS/OSPF/BGP-LS 中通告，并通过事件驱动遥测（Event Driven Telemetry，EDT）对外推送，实现对网络性能变化的秒级处理。

下面通过一个较为复杂的例子来说明性能测量是如何与 SR-TE 的相关组件集成的，拓扑如下图所示：

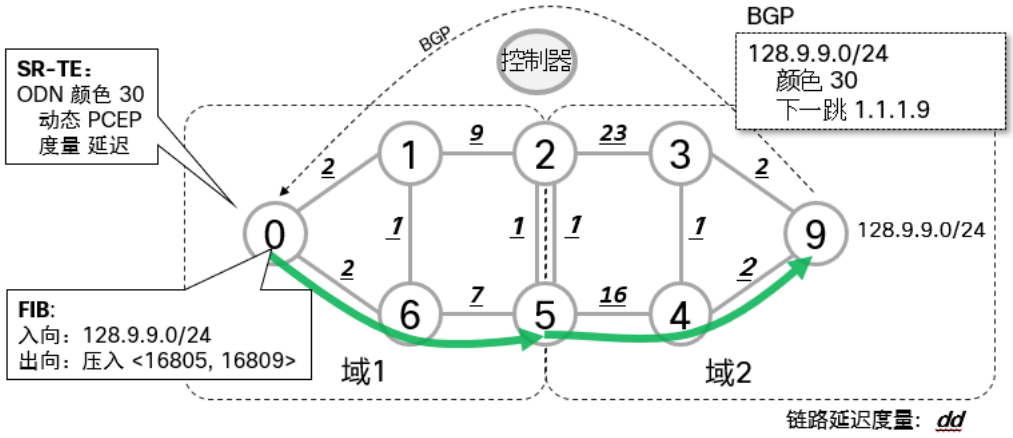


图 9 基于性能测量的数据计算跨域低时延路径

- 图中所有节点都启用了 SR 性能测量功能，用于测量链路时延；时延测量值通过 IGP/BGP-LS/Telemetry 通告给控制器；控制器具有整合的 SR-TE 数据库，包含了域 1 和域 2 的信息。
- 节点 0 启用了 ODN 功能，配置了颜色 30 的 ODN 模板，用于表示低时延的意图；当节点 0 收到节点 9 通告的着色为颜色 30 的 BGP 前缀 128.9.9.0/24 时，将触发 ODN 功能，由于节点 9 位于另外一个域，节点 0 需借助控制器计算跨域低时延路径。
- 控制器计算出跨域低时延路径，并采用域 1 的 Flex-Algo Prefix-SID 16805 和域 2 的 Flex-Algo Prefix-SID 16809 编码此路径。注意，由于使用了 Flex-Algo，因此不再需要 Adj-SID 用于跨越图中时延低但 IGP 度量高的链路。
- 自动引流功能自动将去往 128.9.9.0/24 的流量引导至此路径。

- 当性能测量结果发生变化时，控制器通过 IGP/BGP-LS/Telemetry 获知此变化，重算路径并执行路径更新，保证路径可以满足意图（低时延）。

从上述例子可以看出，SR Policy 的各个模块(SR 原生算法、自动引流、ODN、Flex-Algo)即可单独使用，也可像乐高积木一样组合起来使用，非常的灵活，从而可以适应不同的应用场景，模块化也是 SR Policy 区别于传统隧道接口体系的一个关键特征。那么 SR Policy 的这些模块实际上是如何组合起来的呢？下面将进行一个简单介绍。

五、 SR Policy 技术实现与标准体系

5.1 SR Policy 技术实现

SR-TE 进程是 SR Policy 解决方案技术实现的核心，它是可以担任不同角色的构建模块：集成在头端节点时，作为路由器的“大脑”，为本地节点提供 SR-TE 服务；在 SR PCE 服务器上时，则为网络中的其他节点提供 SR-TE 服务。头端和 SR PCE 采用相同的 SR 原生算法，它们之间的功能差异不在于计算引擎，而在于 SR-TE 数据库的内容。

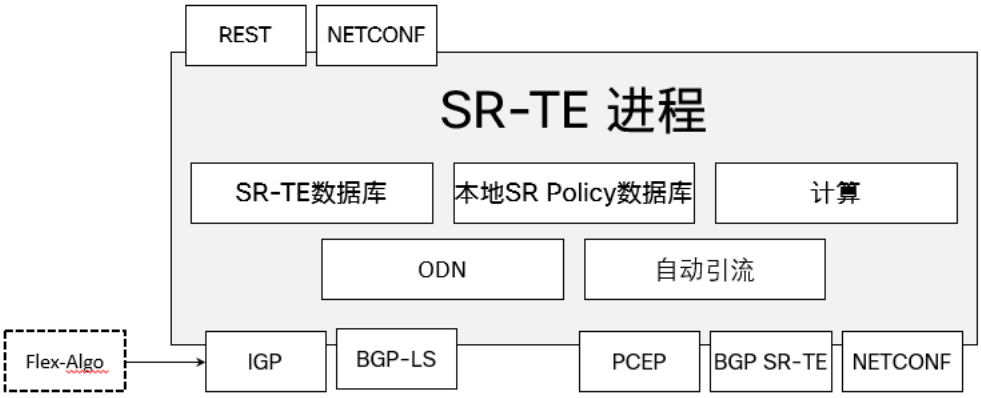


图 10 SR-TE 进程组件

SR-TE 进程的组件如上图所示，包括:

- SR-TE 数据库：保存拓扑信息、SR 信息、SR Policy 等信息；头端的 SR-TE 数据库包含本 IGP 域信息， SR PCE 的 SR-TE 数据库常常包含多域信息;
- 计算引擎：使用 SR 原生算法计算动态路径;

- 本地 SR Policy 数据库：头端节点用于维护、验证以及从不同来源选择 SR Policy 候选路径；
- ODN 模块：头端用于按需生成 SR Policy；
- 自动引流模块：头端用于自动引导流量至 SR Policy。

Flex-Algo 并不从属于 SR-TE，但它与 SR-TE 无缝集成，丰富了可用于 SR-TE 编码 SLA 路径的的 Segment 集合，并且能够与 ODN/自动引流等 SR-TE 机制完全集成。

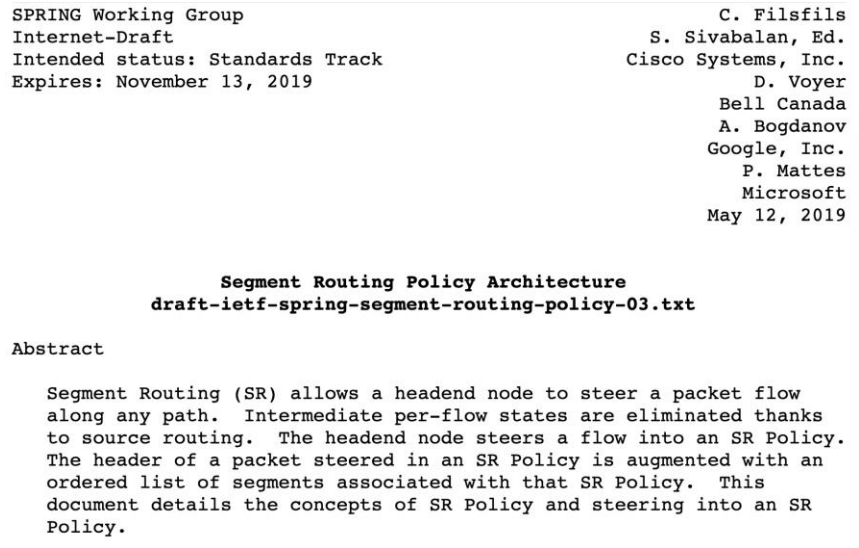
SR-TE 使用以下不同协议和接口与各种内部/外部实体进行交互：

- IGP：接收 IGP 分发的网络信息，通过 IGP 分发 TE 属性；
- BGP-LS：接收拓扑和其他网络信息，并报告 SR Policy 信息；
- PCEP：用于 SR PCE 和 SR PCC 之间的通信；
- BGP SR-TE：用于 SR PCE 和 SR PCC 之间通信的 BGP 地址族；
- NETCONF：用于 SR PCE 和 SR PCC 之间以及应用和 SR PCE 之间基于数据模型的通信；
- REST：用于应用和 SR PCE 之间的通信。

5.2 SR Policy 的标准体系

思科致力于 SR 标准化，在 IETF 发布了确保 SR-TE 互操作性（例如协议扩展）所需的全部细节，这对整个行业都至关重要。

IETF 草案 draft-ietf-spring-segment-routing-policy 是关于 SR Policy 的最主要文件。该文件详细描述了 SR Policy 整体架构及其关键概念，包括 SR Policy、BSID、候选路径类型、SR-TE 数据库、自动引流、ODN、SR Policy 的保护等。目前该草案已经成为稳定的 IETF 工作组草案，预计将在近期完成标准化，如下图所示。



SPRING Working Group
Internet-Draft
Intended status: Standards Track
Expires: November 13, 2019

C. Filsfils
S. Sivabalan, Ed.
Cisco Systems, Inc.
D. Voyer
Bell Canada
A. Bogdanov
Google, Inc.
P. Mattes
Microsoft
May 12, 2019

Segment Routing Policy Architecture
draft-ietf-spring-segment-routing-policy-03.txt

Abstract

Segment Routing (SR) allows a headend node to steer a packet flow along any path. Intermediate per-flow states are eliminated thanks to source routing. The headend node steers a flow into an SR Policy. The header of a packet steered in an SR Policy is augmented with an ordered list of segments associated with that SR Policy. This document details the concepts of SR Policy and steering into an SR Policy.

图 11 IETF 草案 draft-ietf-spring-segment-routing-policy

以下 IETF 文件引入了各种协议扩展:

- draft-filsfils-spring-sr-traffic-counters;
- draft-filsfils-spring-sr-policy-considerations;
- draft-ietf-idr-bgp-ls-segment-routing-ext;
- draft-ietf-idr-te-lsp-distribution;
- draft-ietf-idr-bgpls-segment-routing-epe;
- draft-ietf-lsr-flex-algo;
- draft-ietf-pce-segment-routing;
- draft-sivabalan-pce-binding-label-sid;
- draft-ietf-pce-association-diversity;
- draft-ietf-idr-segment-routing-te-policy;
- RFC 8491;
- RFC 8476;
- draft-ietf-idr-bgp-ls-segment-routing-msd。

5.3 多厂商互操作

2018 年以来，国际权威独立测试机构 EANTC（欧洲高级网络测试中心）连续两年在其 MPLS+SDN+NFV 多厂商互操作测试中进行了大量 SR、SR-TE 测试，发布相关的白皮书，并在巴黎 MPLS 大会上展示。

在 EANTC 测试中，涉及到用 MPLS 作为传送技术的应用场景全部采用了 SR。除非是验证与 SR 的互通性，否则不会采用 LDP 或者 RSVP-TE。SR 作为统一的传送技术，已经得到业界普通认可。从测试涉及的应用场景及互操作结果来看，设备厂商对 SR-TE 的支持不断取得进展，基本功能普遍已经支持且实现良好互通。

特别地，在 2019 年的 EANTC 测试中，除了思科以外，若干设备厂商也开始支持 SR Policy。这无疑是个积极的信号。随着 SR Policy 标准化趋向于完成，我们相信业界对 SR Policy 的支持将越来越好，有理由期待在 2020 年的 EANTC 测试中，SR Policy 会成为 SR-TE 实现的主流。

六、 SR Policy 典型应用场景

6.1 5G 网络切片

ITU-T 为 5G 业务定义了三种典型的应用场景： eMBB、uRLLC、mMTC，每种应用场景对网络有不同的需求。5G 将服务于众多的垂直行业，不同垂直行业进一步提出了差异化、多样化的业务需求。

5G 网络架构中提出了网络切片（Network Slicing）这一突破性的概念。通过网络切片，使运营商能够在通用的物理平台之上构建多个专用的、虚拟化的、互相隔离的逻辑网络，来满足不同客户对网络能力的不同要求。

SR 是实现 5G 网络传送部分切片的最佳选择。基于共享的多网络域传送网络，采用 SR Policy 和 Flex-Algo 划分多个网络切片，并在每个网络切片内用 L2/L3 VPN 进行客户和业务的隔离，即“软切片”。涉及的 VPN 和边缘设备数量可能会很多， ODN 技术能够按需自动创建 SR Policy，自动引流则确保引导业务流量到适当的网络切片。基于 SR Policy 的网络切片架构简单易操作、可扩展性超高、能为业务自动化提供端到端 SLA。

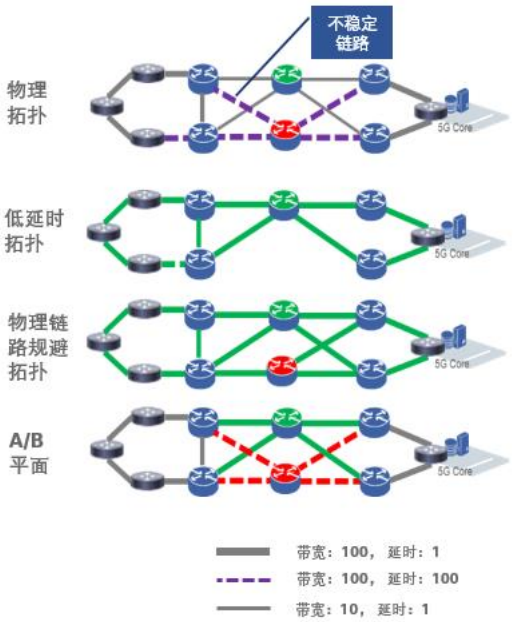


图 12 5G 网络切片

上图显示了采用 Flex-Algo 划分 5G 网络切片的例子。基于同一多域物理网络，采用 Flex-Algo 实现了三个不同的网络切片：低延迟网络切片、避免不稳定链路的网络切片和分为两个平面的网络切片。

6.2 低时延多云互联

在数字化转型过程中，很多企业出于信息安全和性价比方面的考虑，越来越倾向于混合云、多云的部署。底层网络一方面要实现多云的互联互通，另一方面要能够提供低延迟路径，满足关键业务在多云的部署和迁移。这通常并不容易，多云互联的网络配置和故障排除复杂，低延迟路径更是难于满足。

SR Policy 为实现多云互联提供了强大的技术手段。SR 性能测量实时测量每条链路的延迟，ODN 按需自动生成 SR Policy 低延迟路径，自动引流将云互联业务引导至适当的路径，结合 SDN 控制器还可实现灵活的跨域端到端动态带宽调整。

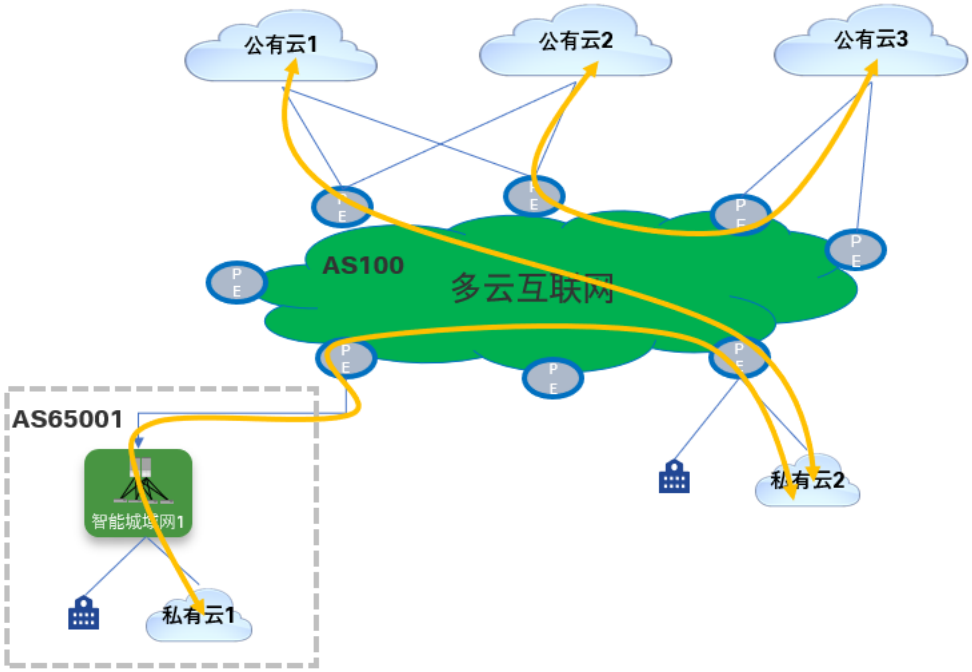


图 13 低时延多云互联

上图显示了采用 SR Policy 和性能测量实现低时延多云互联的例子，支持公有云到公有云、公有云到私有云、私有云到私有云等多种云互联场景。

七、 总结与展望

基于 SR Policy 的 SR-TE 通过简单、可扩展、自动化的方式实现单域/跨域流量工程，将意图转换为编程到网络中的 SR Policy 路径（Segment 列表），并自动引导流量至相应的 SR Policy。

SR Policy 的多项关键创新和模块化构建使其明显区别于隧道接口体系的 SR-TE，我们认为，SR Policy 是 SR-TE 的正确做法！这点也越来越得到业界的认可。

SR Policy 本身仍然在快速地发展之中，例如 BGP SR-TE 可以使得控制器通过 BGP 信令把 SR Policy 信息通告给头端，而不用借助于 PCEP 协议，今后随着 BGP-LS 具备报告 SR Policy 的能力，BGP 有很大的潜力成为控制器和设备之间统一的 SR Policy 通信协议；又例如性能测量中会加入路径测量和丢包测量功能，测量的结果同样通过 IGP/BGP-LS/Telemetry 通告给控制器，这样控制器能及时地根据不同 SLA 指标做出决策；再例如 SR Policy 与服务链结合，把网络能力和网络服务融为一体等。

需要指出的是，SR Policy 体系适用于 SRv6。事实上，在 SRv6 完全基于 IP 的框架下，SR Policy 这个完全基于 IP 优化的体系更能发挥其所长。

所有这些，都是下一代网络平台所需要的能力，是基于意图的网络的基石。

关于 SR Policy 的详细信息及最佳实践，敬请参阅由 Clarence Filsfils 等思科专家所著、由笔者翻译/审校的《Segment Routing 详解（第二卷) 流量工程》一书，本书于 2019 年 9 月人民邮电出版社出版。

【参考文献】

1. SR Policy 架构草案：https://tools.ietf.org/html/draft-ietf-spring-segment-routing-policy-03
2. SR Policy 部署实施注意事项草案：https://tools.ietf.org/html/draft-filsfils-spring-sr-policy-considerations-02
3. BGP-LS 的 SR 扩展草案：https://tools.ietf.org/html/draft-ietf-idr-bgp-ls-segment-routing-ext-12
4. BGP-LS 分发 TE 策略和状态草案：https://tools.ietf.org/html/draft-ietf-idr-te-lsp-distribution-11
5. SR BGP EPE 的 BGP-LS 扩展草案：https://tools.ietf.org/html/draft-ietf-idr-bgpls-segment-routing-epe-19
6. SR IGP Flex-Algo 草案：https://tools.ietf.org/html/draft-ietf-lsr-flex-algo-03
7. PCEP 的 SR 扩展草案：https://tools.ietf.org/html/draft-ietf-pce-segment-routing-16
8. PCEP 承载 BSID 草案：https://tools.ietf.org/html/draft-sivabalan-pce-binding-label-sid-07
9. PCEP 不相交约束条件信令扩展草案：https://tools.ietf.org/html/draft-ietf-pce-association-diversity-08
10. BGP SR-TE 扩展草案：https://tools.ietf.org/html/draft-ietf-idr-segment-routing-te-policy-07
11. ISIS 中通告 MSD，RFC 8491：https://tools.ietf.org/html/rfc8491
12. OSPF 中通告 MSD，RFC 8476：https://tools.ietf.org/html/rfc8491
13. BGP-LS 中通告 MSD 草案：https://tools.ietf.org/html/draft-ietf-idr-bgp-ls-segment-routing-msd-05

14. MPLS 网络丢包和延迟测量，RFC 6374:
https://tools.ietf.org/html/rfc6374
15. SR 网络中用 UDP 路径进行性能测量草案:
https://tools.ietf.org/html/draft-gandhi-spring-rfc6374-srpm-udp-01
16. SR 网络中用 TWAMP 进行性能测量草案:
https://tools.ietf.org/html/draft-gandhi-spring-twamp-srpm-01