SRv6 demo proposal

Segment Routing (SR) is an architecture based on the source routing paradigm that seeks the right balance between distributed (network-wide) intelligence and centralized (controller-based) programmability. This architecture gives applications and network devices complete control over the forwarding paths by combining simple network instructions. Commercial deployments of SR using the MPLS forwarding plane have been very successful, and Cisco is now moving forward with the IPv6 instantiation of SR, SRv6. SRv6 enhances the properties of network simplification, strict SLA enforcement, automation and scaling (stateless fabric) pioneered by SR-MPLS. SRv6 extends the implementation of network programming by allowing virtually any computable behavior to be bound to a segment; for example: forwarding and user-defined instructions. SRv6 use-cases expand into Network Function Virtualization (NFV), Service Chaining and verticals such as the Internet of Things (IoT), security and group-based policy enforcement. The support of SRv6 in open source projects (Linux Mainline and FD.io Vector Packet Processing (VPP) platform) provides researchers a powerful tool for the validation of new proposals and for sharing their findings with the networking community. SRv6 brings untapped potential and innovation to the network layer that translates into impactful research opportunities. As a result, SR has been gaining quite a bit of traction in the research community (http://www.segment-routing.net/scientific-papers/).

This demo showcases some of the main behaviors brought by SRv6, using Cisco products, Barefoot Tofino, and open source implementations on Linux and FD.io VPP. It runs on a small network topology formed of two non-SRv6 edge domains located on both sides of an IPv6 core domain (see below diagram). Each edge domain consists of two Linux machines, one IPv4 only and the other IPv6 only. In the core domain, node 1 is running P4 implementation of SRv6 on Barefoot Tofino, nodes 2 and 4 are SRv6-enabled Cisco routers, node 3 is non-SR router, node 5 is an SRv6-enabled Linux machine and node 6 is a regular Linux machine running FD.io VPP. Two non-SR services, Netfilter and Snort, are running respectively behind node 5 and 6.

In this demo, HTTP traffic flows between the IPv4 client in one edge domain to the IPv4 server in the other edge domain, and similarly between the IPv6 client and server. The IPv4 traffic coming from the client and headed towards the server is steered on node 1 into an SRv6 policy that encapsulates the IPv4 packet with an IPv6 header and a Segment Routing header of 4 segments:
- a traffic engineering segment steering the traffic through node 2,
- two service segments, for Netfilter and Snort, which integrate a proxy behavior that removes the SR header before the traffic is sent to the service, and re-adds it when it comes back,
- a VPN segment that decapsulates the traffic and sends the inner IPv4 packet to the right server.
Similarly, node 1 steers the traffic coming from the IPv6 client into another SRv6 encapsulation policy, which has a last VPN segment that decapsulates and sends the inner IPv6 traffic to the IPv6 server. For the returning traffic, two SRv6 policies are defined on node 4 to send IPv4 and IPv6 traffic straight back to the right client on the default path.