SRv6-uSID Deployment

Daniel Voyer
Technical Director
May 2023

Bell

Bell Domains and services

Bell Network 3.0 is a journey to…

Transform how Bell delivers the best customer experience with seamless access to a software-driven, cloud-based ecosystem

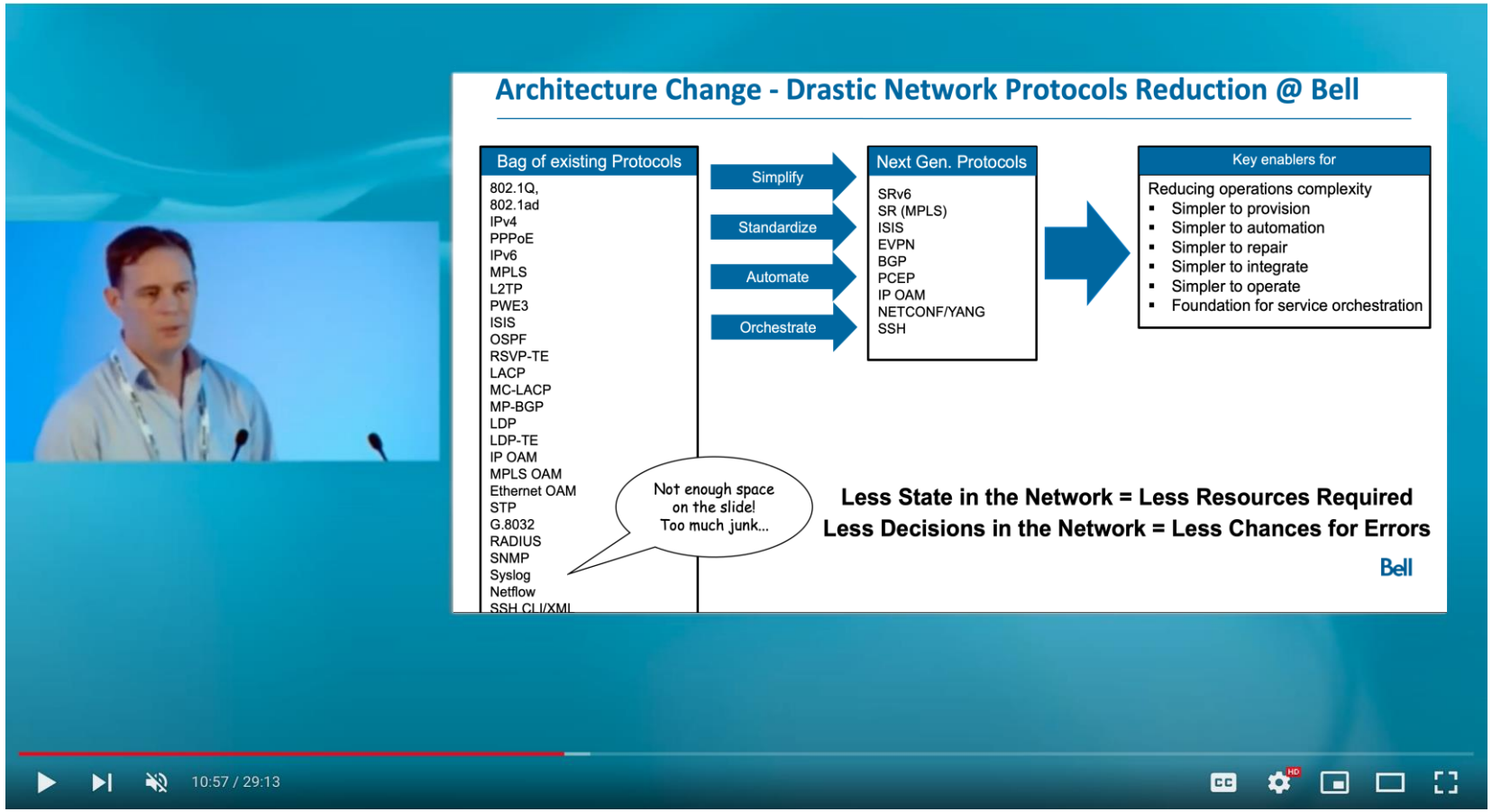
Bell deployed one of the first SR-MPLS networks

- One of the first SR-MPLS deployments
- Deployment went very well (positive tone) TI-LFA was the first benefits collected
- But then why shifting to SRv6 so quickly?
- Answer: “the power of SRv6 uSID’s”.
- Let’s review this

Bell

Unified End-to-End SRv6 uSID Dataplane

- Remove the complexity of getting MPLS to the host/socket/container
- Remove SR-MPLS/LDP /SRv6 GW at the DC & Network edge
- Simplification



Why SRv6-uSID @Bell

- Summary of the gains from Dan B's session
- + other OPEX gains,
- Not having MPLS
- Not having BGP3107 – simpler interconnect
- Not having vXLAN/MPLS gateways
- Having summarization

Reduce carrier network services costs by up to 90%
footprint by 75%
power consumption by as much as 66%

Daniel Bernier
Technical Director, Bell Canada

Jesper Eriksson,
VP Product Management, NoviFlow inc

Bell

Routing Scale

	SRv6 uSID	MPLS
Unique Nodes in the SR domain	15M-240M	0.8M
Unique Services per node	512k	0.2M
ISIS Summarization	Yes	No
BGP3107 complexity tax to scale ISIS Host Routes	No	Yes

- Available functionalities: 256 blocks (/32's). For each block, we have 16 bits space for uSID ID's. 8k are reserved for the LIB, 57k for GIB.
  - 256*57k Global ID = 15M Global ID. In the future we could go up to 4096 blocks
- If more than 8k for LIB, then 8 Wide-LIB spaces could be added for a total of 8*64K = 512k services
- More information on segment-routing.net

Bell

HW Scale

	SRv6 uSID	MPLS
Linerate steering into SR Policy of N SID's	N=26	N=~12
Number of counters associated to a remote ISIS node	1	4
Number of dataplane entries associated to remote ISIS node	1	4

- Blog: https://www.segment-routing.net/demos/26-usid-push-linerate/ with NCS5700 – Jericho2
- 1 vs 4: ip2ip, ip2mpls, mpls2ip, mpls2mpls

Bell

Other Benefits

	SRv6 uSID	MPLS
SR Domain Security	Same	Same
Optimal Load balancing	yes	no

SRv6: 20-bit rich flow entropy at fixed offset within outer IPv6 header (Flow Label)

MPLS: DPI to random location without guaranteed outcome:

- label stack walk to inner IP header fields
- label stack to Entropy Label (plus additional label stack overhead and PE complexity)

Bell

A few notes on our deployment

- We deploy SRv6 uSID with a negligeable sub-space of FD/8
  - 0.0015% of FD/8 private space (/24 out of /8 = 2^(-16))
- We have conducted many SRv6 uSID Interoperability with different vendors
  - Cisco, Ciena, Nokia, Juniper, Arrcus, FD.IO, Intel, Noviflow,

Bell

uSID Interop Description & Objectives

Today, Bell has deployed uSID on Cisco platform in productions. We are investigating interoperability with other vendors that are implemented in different part of the Bell's backbone where SRv6-uSID is required;

The **main objective for** – Bell/Nokia/Juniper – is to demonstrate that uSID can be "interoperable" with the existing SRv6 functions deployed in Bell network;

The baseline for the interop is with the L3VPN service. **Nokia & Juniper will act as an SRv6-PE** peering with a **Cisco SRv6-PE**, both using uSID F3216.

- The nodes that provide transport (LSR) for SRv6-PEs, will be Cisco acting as **LSR node when running SRv6-uSID**.
- The node acting as LSR without SRv6 (IPv6-only) could be Nokia, Cisco, Ciena, Juniper, Arrcus, etc.

LIB encoded using 0xE & 0xF in the first nibble of the uSID (e.g., "**E**XXX", "**F**XXX" --> **Local uSIDs;** Global uSIDs otherwise)

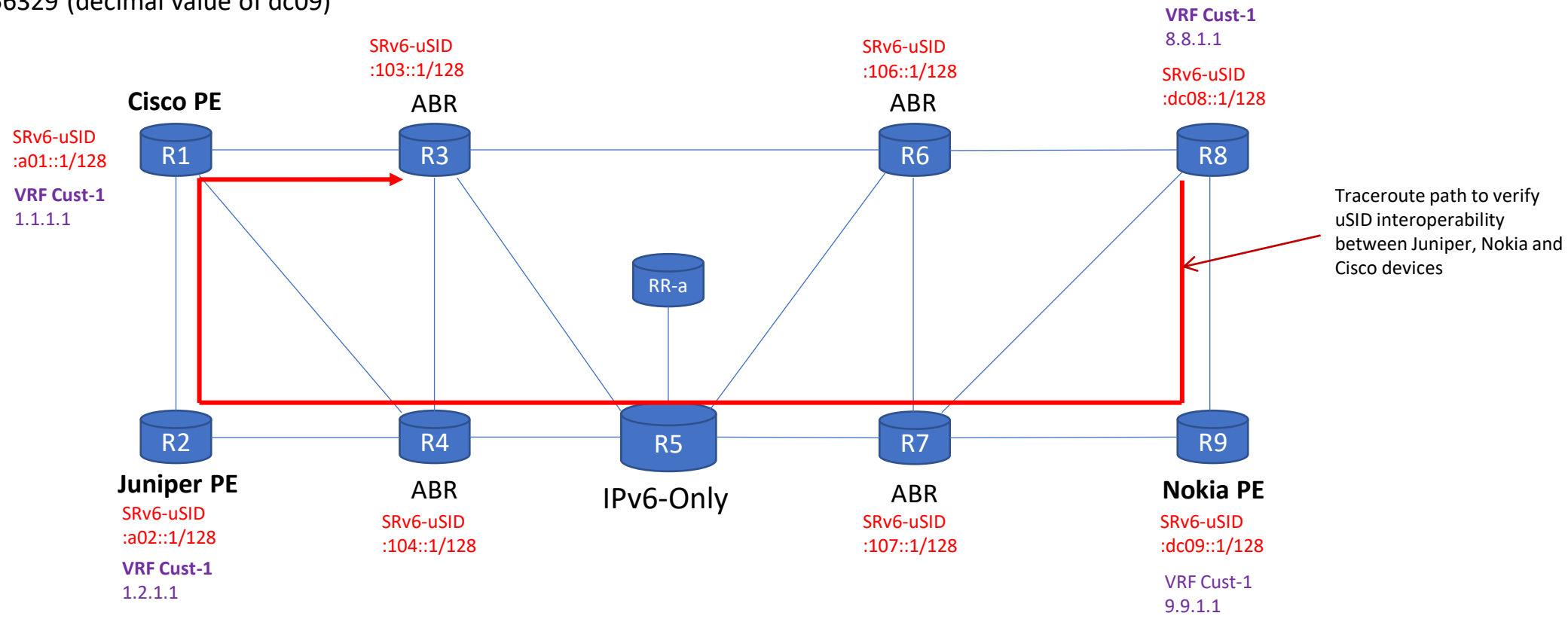
IPv6 addressing used during the test will be ULA fc::/8 range. IPv6 addressing used for point-to-point will be link-local;

Success of the test will be a demonstration that L3VPN routes are advertised from 1 PE to the other and we are able to ping and or traceroute from within the VRF interfaces.

Phase 1 – uSID interop testing

Highlights (for testing only)

1. Cisco R1 and Nokia R9 acting as PE's
2. L3VPN name: cust-1
3. IPv6 P2P – link-local
4. Nokia IPv6 loopback : fccc:cc00:dc09:1/128
5. Nokia locator0: fccc:cc03::/32 with micro-segment-locator value 56329 (decimal value of dc09)



OAM – Example Traceroute that help validating uSID

Explicit path traceroute with uSID node as reference point

Encap (SRH) traceroute from node 8 to node 3

Node 9, node 7, node 4 and then node 1
(node 4 and 1 encode different ISIS area)

Node 9 shifted by 16

```
RP/0/RP0/CPU0:r8#traceroute fccc:cc00:103:: via srv6-carriers fccc:cc00:dc09:107:104:a01::
Wed Nov 16 17:11:11.182 EST

Type escape sequence to abort.
Tracing the route to fccc:cc00:103::

 1  fccc:cc03:dcff:dc09::1
        [IP tunnel: DA=fccc:cc00:dc09:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)      ] 8 msec
        [IP tunnel: DA=fccc:cc00:dc09:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)      ] 5 msec
        [IP tunnel: DA=fccc:cc00:dc09:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)      ] 4 msec
 2  fccc:cc00:107::1                                                    Node 9 removed
        [IP tunnel: DA=fccc:cc00:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)    ] 7 msec
        [IP tunnel: DA=fccc:cc00:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)    ] 3 msec
        [IP tunnel: DA=fccc:cc00:107:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)    ] 2 msec
 3  fccc:cc00:5::1     Node 7 removed
        [IP tunnel: DA=fccc:cc00:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)        ] 14 msec
        [IP tunnel: DA=fccc:cc00:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)        ] 7 msec
        [IP tunnel: DA=fccc:cc00:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)        ] 8 msec
 4   *
    fccc:cc00:104::1
        [IP tunnel: DA=fccc:cc00:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)        ] 17 msec
        [IP tunnel: DA=fccc:cc00:104:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)        ] 3 msec
 5  fccc:cc00:a01::1 Node 4 removed
        [IP tunnel: DA=fccc:cc00:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)    ] 12 msec
        [IP tunnel: DA=fccc:cc00:a01:: SRH Stack 0 =(fccc:cc00:103::    ,SL=1)    ] 3 msec *
 6  fccc:cc00:103::1 17 msec 6 msec 4 msec
RP/0/RP0/CPU0:r8#
```

Node 7 shifted by 16

Node 4 shifted by 16

Thank You