IETF Hackathon -
SRv6 SFC Architecture with SR-aware Functions

IETF 124
1–2 November 2025
Montreal, Canada
Wataru Mishima

Hackathon Plan

Objective: Comprehensive management of SRv6 SFC using a controller.

- SRv6 controller computes SR Policies based on network function information.
- SRv6 manager deploys network functions and configures SRv6 endpoints (End.AN).

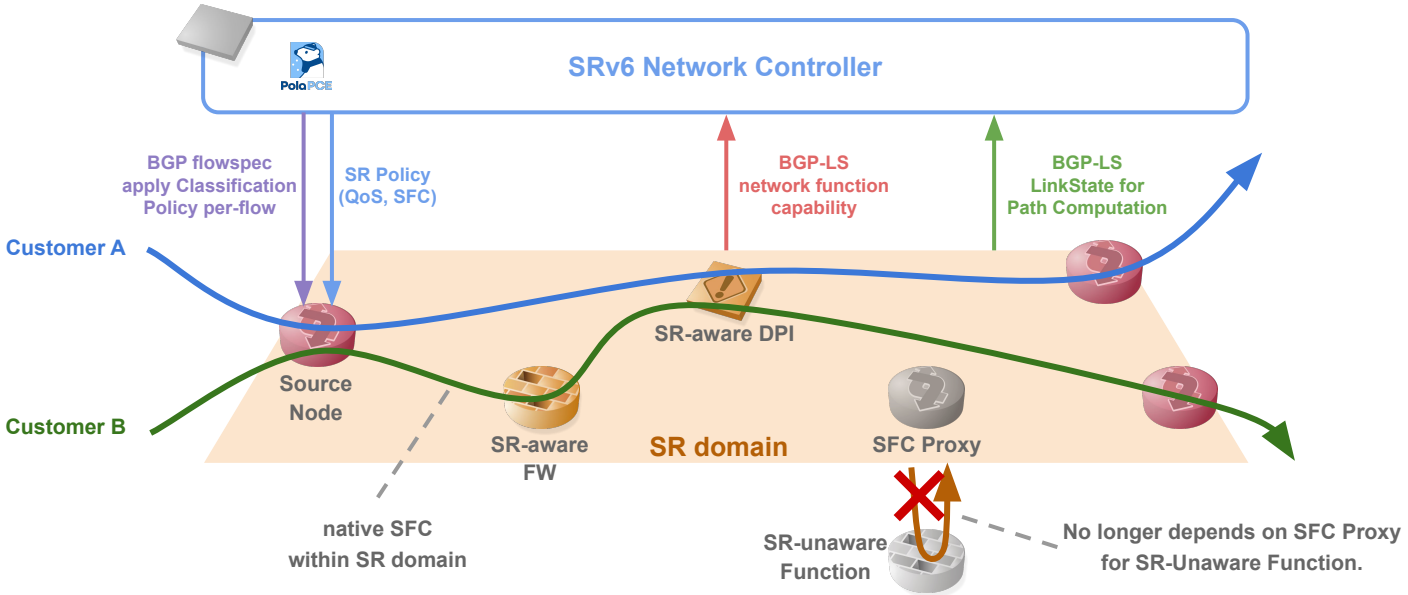
Hackathon Goal: Implement SRv6 SFC path computation in the following OSS:

- Implement loose source routing path computation for SRv6 SFC in Pola PCE.
- Update BGP-LS Service Segment API for GoBGP v4.
- Implement a manager component for initializing network functions and setting up End.AN.
- Conduct interoperability testing.

I-D: SRv6 SFC Architecture

The concept that controller manages the entire SRv6 SFC network.

A controller collects Service Segment information and applies SR Policies based on information about available Network Functions.



SRv6 SFC Architecture with SR-aware Functions

PCE Implementations (1/3)

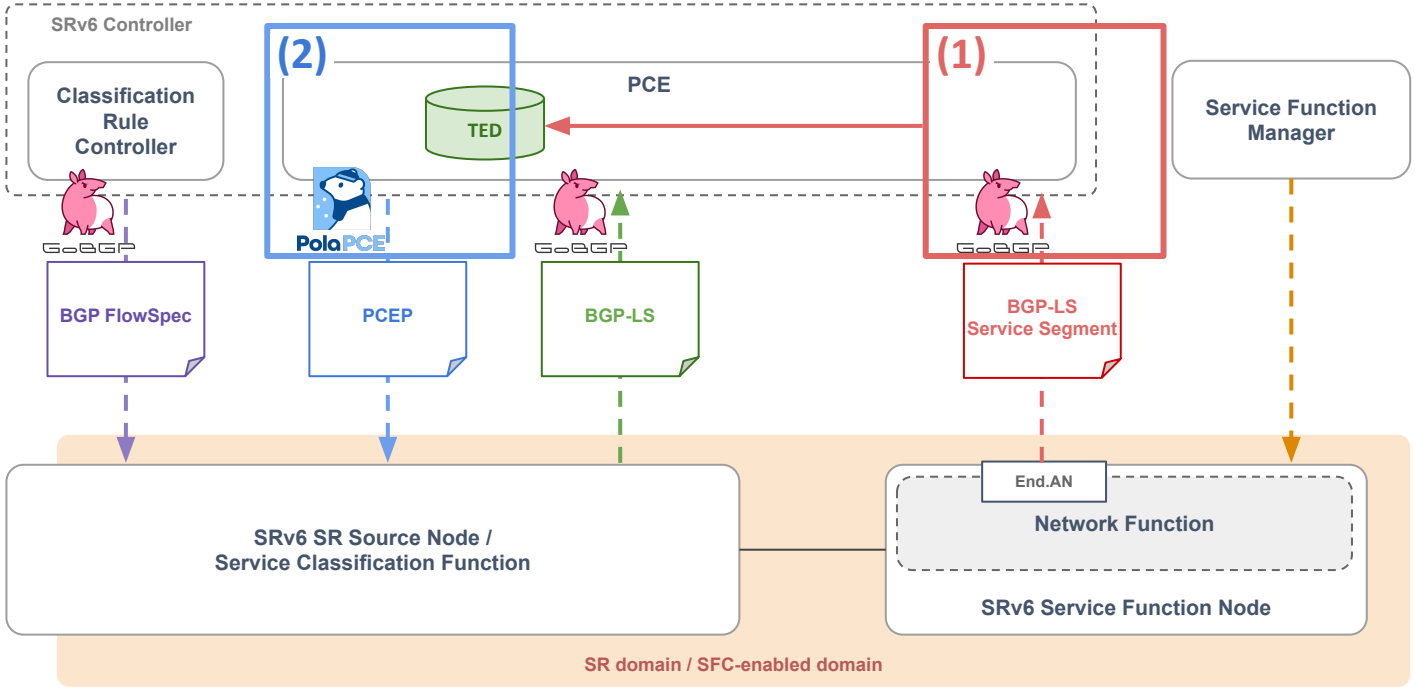
- **GoBGP** (https://github.com/osrg/gobgp)
  - An open-source BGP daemon.
  - We updated gRPC API for BGP-LS Service Segment to support GoBGP v4.

- **Pola PCE** (https://nttcom.github.io/pola/)
  - An open-source Path Computation Element developed by NTT docomo Business.
  - We added support for loose source routing path computation.
    - Enables SFC by specifying SRv6-aware network functions as waypoints.

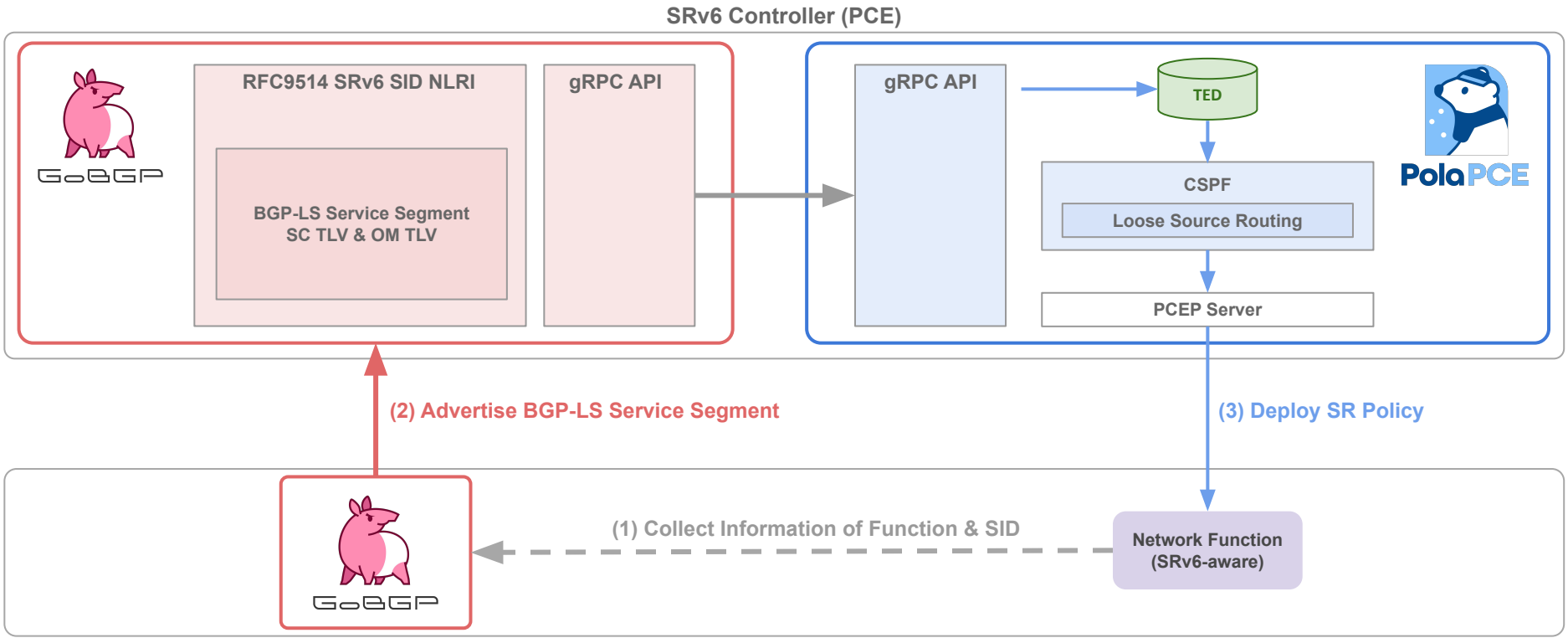
PCE Implementations (2/3)

(1) Collect network function information (including SIDs) and store it in PCE TED.

(2) Compute SR Policies with TE and SFC constraints.

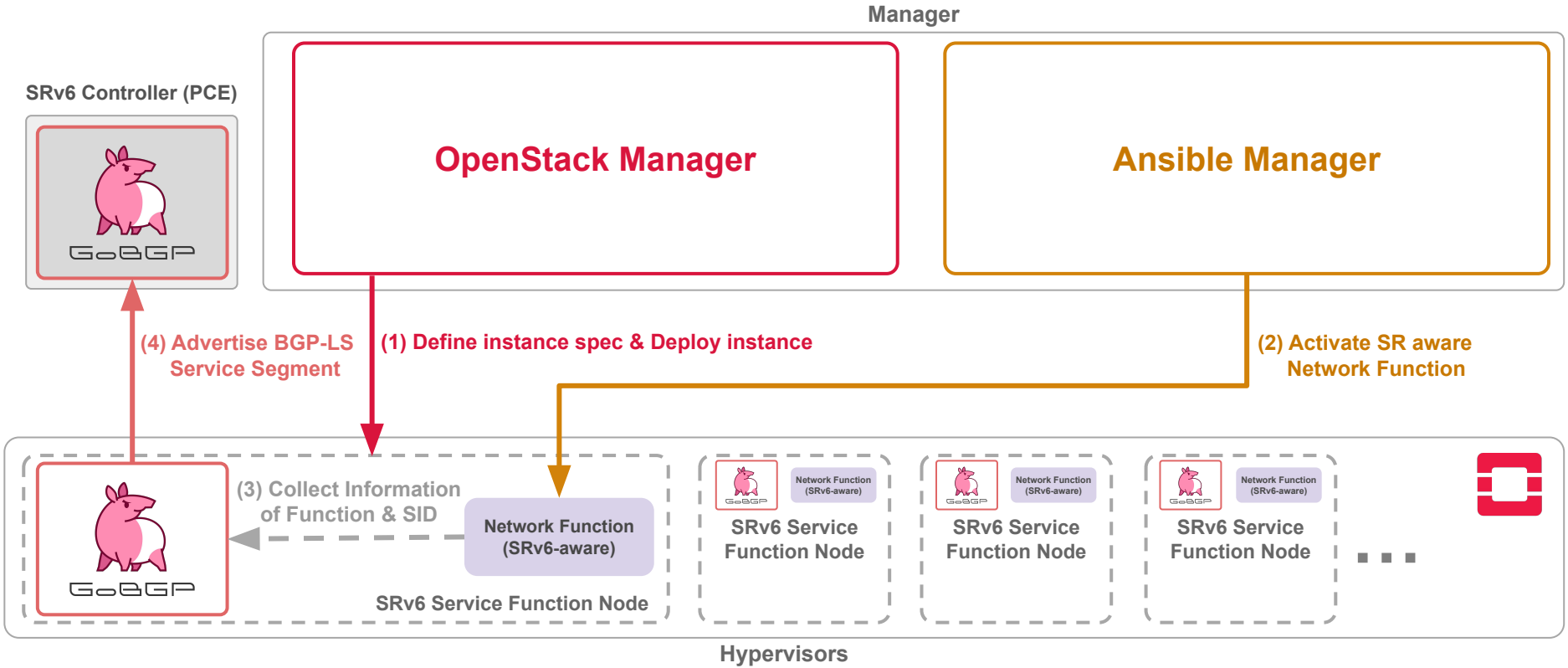


PCE Implementations (3/3)



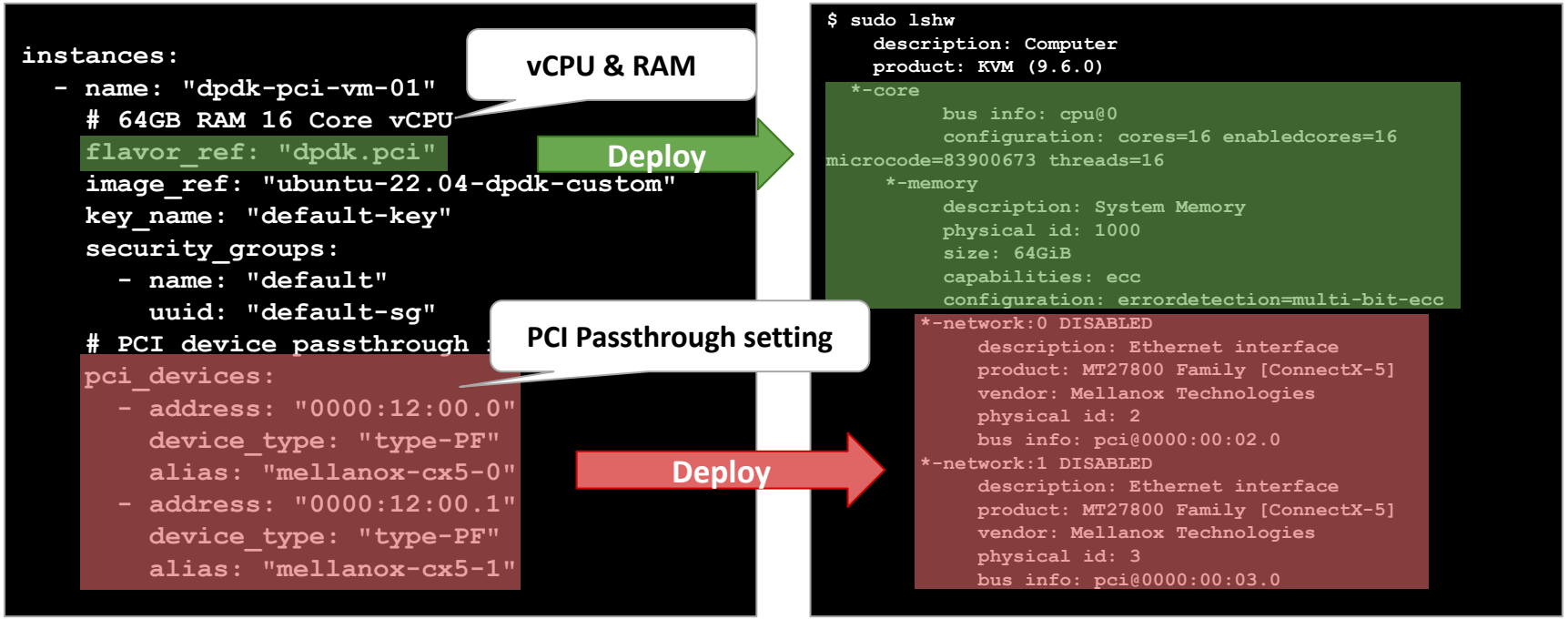
SRv6 SFC Architecture with SR-aware Functions

M-Plane Implementations



SRv6 SFC Architecture with SR-aware Functions

M-Plane Result (1/2)



Instance specification settings (excerpt)

Deployed instance information (excerpt)

M-Plane Result (2/2)

```
"srv6_end": [
    {
      "src": "end-1",
      "actions": [
        { "sid": "fd77:aa::1", "to": "swap_main_1" },
        { "sid": "fd77:aa::2", "to": "swap_main_2" },
        { "sid": "fd77:aa::3", "to": "swap_main_3" },
        { "sid": "fd77:aa::4", "to": "swap_main_4" }
      ]
    },
    {
      "src": "end-2",
      "actions": [
        { "sid": "fd77:ab::1", "to": "sub_a_1" },
        { "sid": "fd77:ab::2", "to": "sub_a_2" },
        { "sid": "fd77:ab::3", "to": "sub_a_3" },
        { "sid": "fd77:ab::4", "to": "sub_a_4" }
      ]
    },
    {
      "src": "end-3",
      "actions": [
        { "sid": "fd77:ac::1", "to": "sub_b_1" },
        { "sid": "fd77:ac::2", "to": "sub_b_2" },
        { "sid": "fd77:ac::3", "to": "sub_b_3" },
        { "sid": "fd77:ac::4", "to": "sub_b_4" }
      ]
    }
  ],
```

End.AN settings for each network function

Network function configuration (excerpt)

```
  0[||||||100.0%]   4[||||||100.0%]   8[||||||100.0%]  12[||||||100.0%]
  1[||||||100.0%]   5[||||||100.0%]   9[||||||100.0%]  13[||||||100.0%]
  2[||||||100.0%]   6[||||||100.0%]  10[||||||100.0%]  14[|          0.7%]
  3[||||||100.0%]   7[||||||100.0%]  11[||||||100.0%]  15[           0.0%]
Mem[||||||||||||||||||51.5G/62.8G]   Tasks: 32, 73 thr; 16 running
Swp[
```

Deployed Network Function (Process)

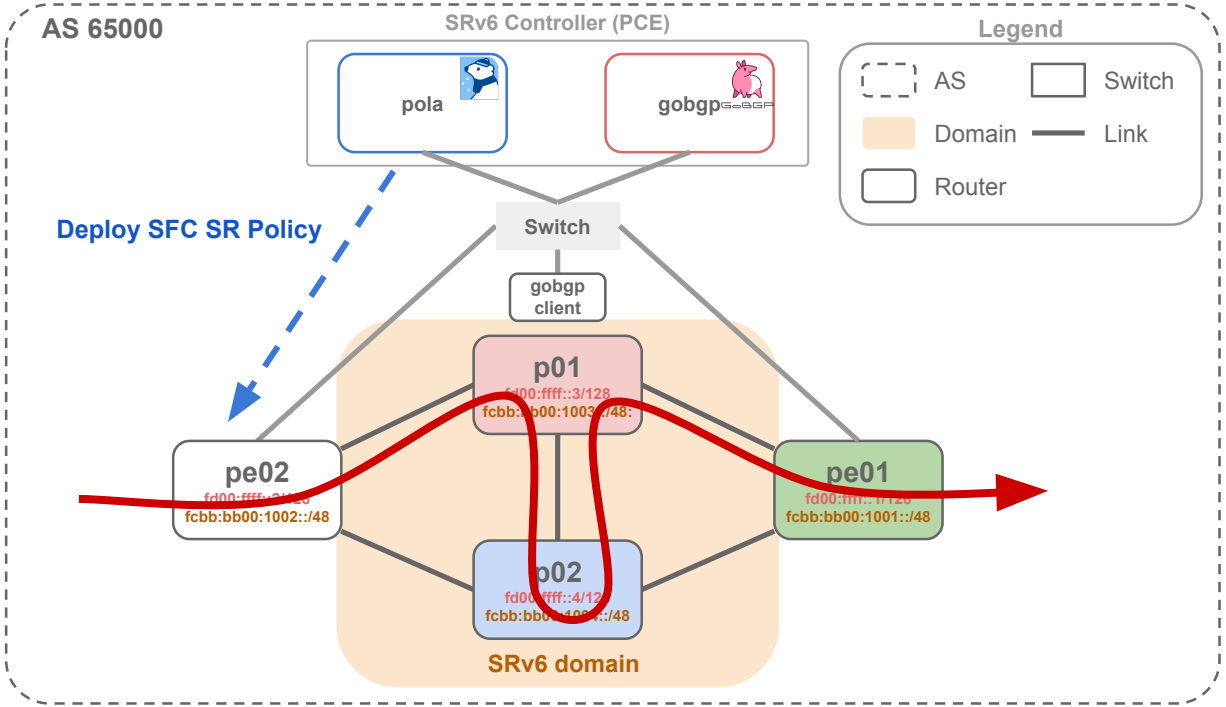
```
  PID USER      PRI  NI  VIRT   RES    S  CPU%▽MEM%   TIME+  Command
10209 root       20   0 66.6G 54508 15352 R 1400  0.1 55:33.57 ./alligator -- -
```

```
$ sudo gobgp global rib -a ls
   Network
Next Hop             AS_PATH              Age        Attrs
*  NLRI { SRv6SID { LOCAL_NODE: {ASN: 65001, BGP LS ID: 10, BGP
ROUTER ID: 163.138.211.252} SRv6_SID: {SIDs: fd77:ab::1}
MULTI_TOPO_IDs: {MultiTopoIDs: 1} } } 0.0.0.0
00:00:09    [{Origin: ?}]
*  NLRI { SRv6SID { LOCAL_NODE: {ASN: 65001, BGP LS ID: 10, BGP
ROUTER ID: 163.138.211.252} SRv6_SID: {SIDs: fd77:ac::2}
MULTI_TOPO_IDs: {MultiTopoIDs: 1} } } 0.0.0.0
00:00:08    [{Origin: ?}]
*  NLRI { SRv6SID { LOCAL_NODE: {ASN: 65001, BGP LS ID: 10, BGP
ROUTER ID: 163.138.211.252} SRv6_SID: {SIDs: fd77:aa::2}
MULTI_TOPO_IDs: {MultiTopoIDs: 1} } } 0.0.0.0
00:00:09    [{Origin: ?}]
```

RIB of GoBGP on Network Function

Network function processes & End.AN advertisement

PCE Demo Topology



Compute SFC SR Policy (p01 -> p02 -> p01 -> pe01)

PCE Result

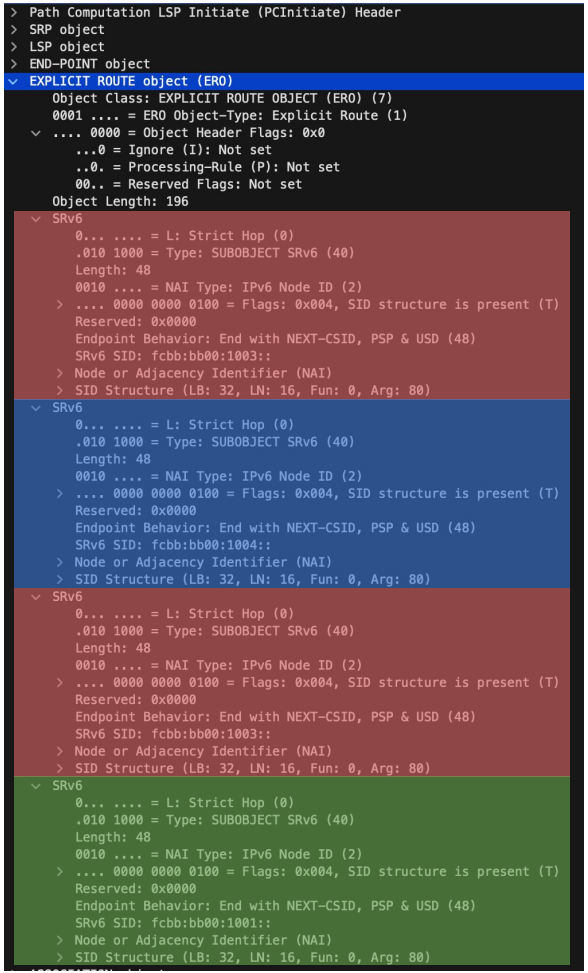
```
root@pola:/# cat /pe02-policy1.yaml
asn: 65000
srPolicy:
  pcepSessionAddr: "fd00::2"
  srcRouterID: "0000.0001.0002"
  dstRouterID: "0000.0001.0001"
  name: "DYNAMIC-POLICY"
  color: 100
  type: dynamic
  metric: igp
  waypoints:
  - routerID: "0000.0001.0003"
  - routerID: "0000.0001.0004"
  - routerID: "0000.0001.0003"
    sid: "fcbb:bb00:1003::"
root@pola:/# ./pola sr-policy add -f /pe02-policy1.yaml -p 50052
success!
root@pola:/# ./pola sr-policy list -p 50052
Session: fd00::2
  PolicyName: DYNAMIC-POLICY
    SrcAddr: fd00:ffff::2
    DstAddr: fd00:ffff::1
    Color: 100
    Preference: 0
    SegmentList: fcbb:bb00:1003:: -> fcbb:bb00:1004:: -> fcbb:bb00:1003:: -> fcbb:bb00:1001::
```

Input:
Destination Router ID

Input:
Waypoint Router IDs
& SIDs (optional)

Generated Segment List

Input YAML and generated SR Policy



PCInitiate packet (Wireshark)

What got done

- **Pola PCE**
  - **COMPLETED: Added SRv6 uSID loose source routing**
- **GoBGP**
  - **COMPLETED: Updated BGP-LS Service Segment API to v4**
- **Interoperability Testing (Pola PCE & GoBGP)**
  - **COMPLETED: Loose source routing**
  - **COMPLETED: SFC-constrained SPF (using End.AN)**
- **M-Plane**
  - **COMPLETED: Network function deployment**
  - **COMPLETED: SRv6 End.AN setup**

Our Code

- **Pola PCE**: https://github.com/watal/pola/tree/feature/ietf124-srv6-sfc
  - **Loose source routing with SFC Example (using Containerlab):** https://github.com/watal/pola/tree/feature/ietf124-srv6-sfc/examples/containerlab/srv6_usid_dynamic-path_loose_source_routing_sfc
- **GoBGP**: https://github.com/watal/gobgp/tree/feature/bgp-ls-service-segments

Members

- **Wataru Mishima**, Kanagawa Institute of Technology, mishima@nw.kanagawa-it.ac.jp
- **Taisei Tanabe**, National Institute of Informatics, tanabe@nii.ac.jp
- **Yuta Fukagawa**, NTT docomo Business / Kanagawa Institute of Technology, y.fukagawa@ntt.com

Acknowledgements

This work was partially supported by JST CRONOS, Japan, Grant Number JPMJCS24N9.