Minimal SFC Controller

Experience with a minimalistic standalone Control point

Sumandra Majee
Example F5 SFC Chain

Classification <-> Service Chain

Ingress Classifier

Classification <-> Service Chain

Controller + Manager

EM Config Manager

Lifecycle Manager

SFC Unaware

PS

Vxlan-GPE

FW

FW

FW

Vxlan-GPE

NSH

RESTy

Sd Gx

Reverse Path Classifier?

Internet

Fwding Table, Path<->NextHop

SF, Path health And availability
Minimal Configuration

• SERVICE Node: SfName, IP+[Port], In-Vlan/interface, Out-Vlan/interface, [Non]SFC, SKIP

• SERVICE Cluster: SfClusterName
  {Set of Service Node | }
  <LB Selection Method >  <SKIP>

• SFF: IP, typically tunnel endpoint, secret, SF to SFF binding info

• Finally THE SF Chain: {Sf1, Sf2Cluset, Sf3} <Symmetric or Not>

• SFC Proxy: No explicit Proxy Node
SERVICE CHAIN Types

SYMMETRIC:
Both sides of flow follow same set of services. Only ingress classifier is required

ASYMMETRIC:
Ingress and Egress has different treatment. Offload can cause that. Classifier on both side.

SKIP:
Skip this service if unavailable
Controller

1. Forwarding table Programming:

Would like to see a common API/interface definition

2. Capability exchange

- Local instance selection/LB Select
- Type: Hash, Weighted, fastest, RoundRobin
- Monitor service health

Again, Would like to see a common API/interface definition
Service Selection

24bit of pathid oh my..., MASK advertised by controller

- Local Selection
- Select the actual flow path during classification
  - Every Sfnode is represented in forwarding graph, may not have enough bits
  - Represented as slice of a service

```
<table>
<thead>
<tr>
<th>Slice#</th>
<th>Chain#</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3^2</td>
</tr>
</tbody>
</table>
```
Minimal Controller Requirement

- Standardized Service, SFF, SFF Topology. Perhaps we augment the existing yang model.
- Forwarding Table Programming: Defined API, schema, cmd protocol that has wide support and implementer friendly. Writing custom plugin is not a good answer.
  - OF Model
  - RESTy, JSON... we love that..
- API for capability discovery.
- GOOD to have service and path health information. This is bit tricky.
- Controller shouldn’t worry about classification rules.
Next Step

• Do you like to see a draft?
• Love to hear from other

Q & A