Introduction to Service Function Chaining on the OpenStack Cloud Platform

Davide Borsatti, Gianluca Davoli, Walter Cerroni, Chiara Contoli, Franco Callegati
Department of Electrical, Electronic and IT Engineering
University of Bologna, Italy

6th OSM Hackfest – Santa Clara, 16th May 2019
This presentation is an extract of the one given at CNSM – SR-SFC Workshop 2018 – Rome, 5th Nov. 2018 for the paper:


For more informations: Davide Borsatti davide.borsatti@unibo.it
A couple words on Service Function Chaining (SFC)
A couple words on Service Function Chaining (SFC)

Concatenation (*chaining*) of basic services or (virtualized) network/service functions…

…in order to obtain a composite service, spanning over the whole network domain
Service Function Chaining over an SDN Domain

Traffic steering is handled by OpenFlow-capable switches, controlled by a SDN Controller
Service Function Chaining over a generic network

Controller of the Service Provider overlay, independent of the controller of the Infrastructure Provider

Traffic steering for SFC is handled by Service Plane entities [IETF, 2016]
OpenStack - Neutron

It provides Network as a Service (NaaS), as the OpenStack users are able to create their own networks where to plug the Virtual Network Interface of their virtual instances. Neutron is able to separate the logical view of the network from the actual physical view, providing APIs to define, manage and connect virtual networks.
OpenStack - Neutron

Virtual Machines

br-int

OpenStack Node
OpenStack - Neutron

Virtual Machines

Other OpenStack Nodes

br-int  br-tun

OpenStack Node

Virtualized Ethernet Cable
OpenStack - Neutron

Virtual Machines

Other OpenStack Nodes

External networks

br-int

br-tun

br-ex

Virtualized Ethernet Cable

OpenStack Node
The extension allows for the creation of SFPs, it natively supports interaction with Open vSwitch (OvS) and it implements a flow classification mechanism.

It is composed by four main entities:

1. Flow Classifier
2. Port Pair
3. Port Pair Group
4. Port Chain
SFC-extension – Flow Classifier

Flow Classifier:
- Green: UDP traffic
- Red: HTTP traffic (TCP port 80)
- Blue: Other packets
Port Pair:  
- Same ingress and egress port
- Different ingress and egress port
Port Pair Group:

Collection of Port Pairs with same functionalities.

SFC-extension – Port Pair Group
Port Chain creation with SFC-extension

1) Flow classifier

Flow classifier

...
Port Chain creation with SFC-extension

1) Flow classifier
   Flow classifier
   ...

2) Port Pair
   Port Pair
   ...

Port Chain creation with SFC-extension

1) Flow classifier
   Flow classifier

2) Port Pair
   Port Pair
   ...

3) Port Pair Group
   Port Pair Group
   ...

1) Flow classifier

2) Port Pair
   Port Pair
   ...

3) Port Pair Group
Port Chain creation with SFC-extension

1) Flow classifier

2) Port Pair

3) Port Pair Group

4) Port Chain
The SFC-extension support two types of encapsulation mechanism to separate traffic belonging to different Port Chain:

- Multiprotocol Label Switching (MPLS)
- Network Service Header (NSH)
SFP deployed over a single physical node

Packet that matches a flow of the SFP

Source

VNF

Destination

br-int

br-tun
Packet that matches a flow of the SFP

Source

Packet that matches a flow of the SFP

VNF

Packet that matches a flow of the SFP

Destination

Packet that matches a flow of the SFP

Table:
- table=0, priority=30, icmp, in_port="sourceTap",
- nw_src=SourceIP, nw dst=DestIP
- actions=group:1
- group_id=1, type=select,
- bucket=actions=mod_dl_dst:vnf_MAC,
- resubmit(,5)
Packet with modified destination MAC address
Packet with modified destination MAC address

Table 5, ip, dl dst=vnf_MAC
actions=set mpls label(511), mod vlan vid:4, resubmit(,10)

Table 10, mpls, dl dst=vnf_MAC
mpls label=511 actions=strip vlan, pop mpls:0x0800, output:"vnf_TAP"
SFP deployed over a single physical node

The packet is forwarded to destination with normal forwarding
The packet is forwarded to destination with normal forwarding

```
table=0, priority=30, icmp,
in_port="vnf_TAP",
nw_src=SourceIP,
nw_dst=DestIP
actions=NORMAL
```
SFP deployed over a single physical node

Source

VNF

br-int

br-tun

Destination

The packet is forwarded to destination with normal forwarding.
SFP deployed over multiple nodes

Source
- br-int
- br-tun

Packet that matches a flow of the SFP

VNF
- br-int
- br-tun

Destination
- br-int
- br-tun
Packet that matches a flow of the SFP

```
table=0,priority=30,icmp,in_port=sourceTAP,
nw_src=SourceIP,nw_dst=DestIP
actions=group:1

vgroup_id=1,type=select,
bucket=actions=mod_dl_dst:vnf_MAC,
resubmit(,5)
```
SFP deployed over multiple nodes

Packet with modified destination MAC address + MPLS label + Vlan ID
Packet with modified destination MAC address + MPLS label + VLAN ID

table=5, priority=0, ip, dl_dst=vnf_MAC
actions=push_mpls:0x8847,
set_mpls_label(511),
set_mpls_ttl(255),
mod_vlan_vid:1, output:"patch-tun"
SFP deployed over multiple nodes

Source

VNF

Destination

br-int

br-int

br-int

br-int

br-tun

VXLAN encapsulation
SFP deployed over multiple nodes

Source

br-int

br-tun

VNF

VXLAN encapsulation

table=0,priority=1,in_port="patch-int"
actions=resubmit(,20)

table=20,priority=2,dl_vlan=1,dl_dst=vnf_MAC
actions=strip_vlan,
load:0x5f->NXM_NX_TUN_ID[],
output:"vxlan-Node-VNF"

Destination

br-int

br-tun

br-tun
SFP deployed over multiple nodes

Source

- br-int
- br-tun

Removal of the MPLS label and of the VLAN id

VNF

- br-int
- br-tun

Destination

- br-int
- br-tun
SFP deployed over multiple nodes

table=0, priority=1, in_port="vxlan-Node-Source"
   actions=resubmit(,4)

table=4, priority=1, tun_id=0x5f
   actions=mod_vlan_vid:4,resubmit(,10)

table=10, priority=1 actions=learn(
    table=20, hard_timeout=300, priority=1,
    NXM_OF_VLAN_TCI[0..11],
    NXM_OF_ETH_DST[]=NXM_OF_ETH_SRC[],
    load:0->NXM_OF_VLAN_TCI[],
    load:NXM_NX_TUN_ID[]->NXM_NX_TUN_ID[],
    output:OXM_OF_IN_PORT[]),
    output:"patch-int"
Removal of the MPLS label and of the VLAN id

table=0,priority=20,mpls
actions=resubmit(,10)
table=10,priority=1,mpls,dl_vlan=4,
dl_dst=vnf_MAC,mpls_label=511
actions=strip_vlan,pop_mpls:0x0800,
output:"vnf_TAP"
SFP deployed over multiple nodes

The VNF forwards the packet to destination
SFP deployed over multiple nodes

Source

br-int

br-tun

The VNF forwards the packet to destination

VNF

br-int

Destination

br-int

br-tun

br-tun

table=0, priority=30, icmp,
 in_port="vnf_TAP",
 nw_src=SourceIP,
 nw_dst=DestIP,
 actions=NORMAL

The VNF forwards the packet to destination
The packet is forwarded to the destination with normal forwarding.