

CARMINE SCARPITTA



THE ITALIAN
EDUCATION
& RESEARCH
NETWORK



SDN Architecture for IPv6 Segment Routing Enabled Networks

GIORNATA DI INCONTRO
BORSE DI STUDIO GARR
"ORIO CARLINI"
GIOVEDÌ 6 DICEMBRE 2018
ROMA

Roma Consortium GARR, 06/12/2018

Borsisti Day 2018



Università degli Studi Roma
Tor Vergata





Outline

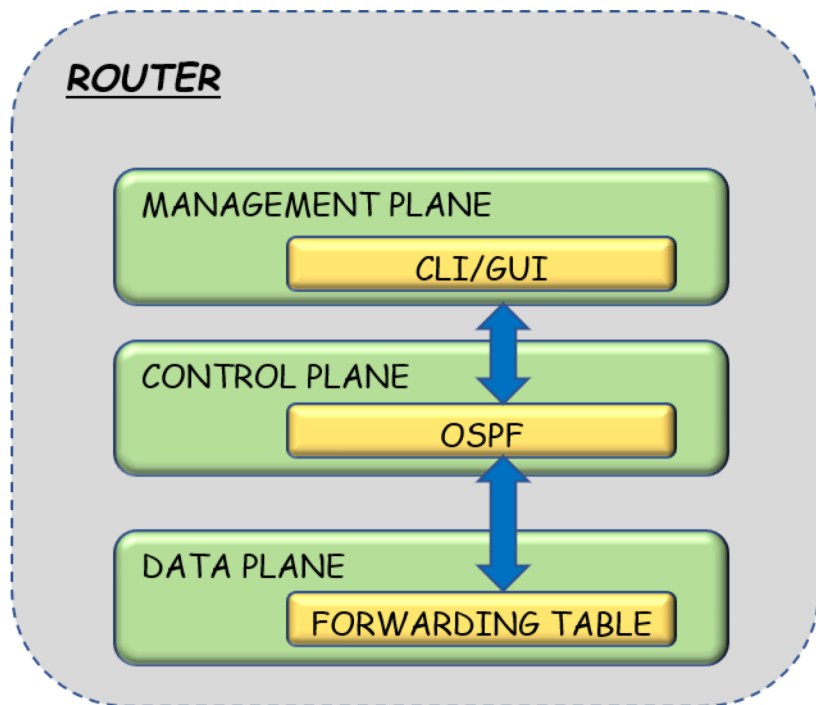
- ❑ Introduction to **SDN**
- ❑ Introduction to **SRv6**
- ❑ **Goals** and **Objectives**
- ❑ **We are working on...**
 - SDN-SRv6 Architecture
 - SRv6-based VPN
- ❑ **Our plans for the future**
 - VPN-TE
 - SRv6-OpenStack integration





Introduction to SDN (1/2)

From **traditional networks**...

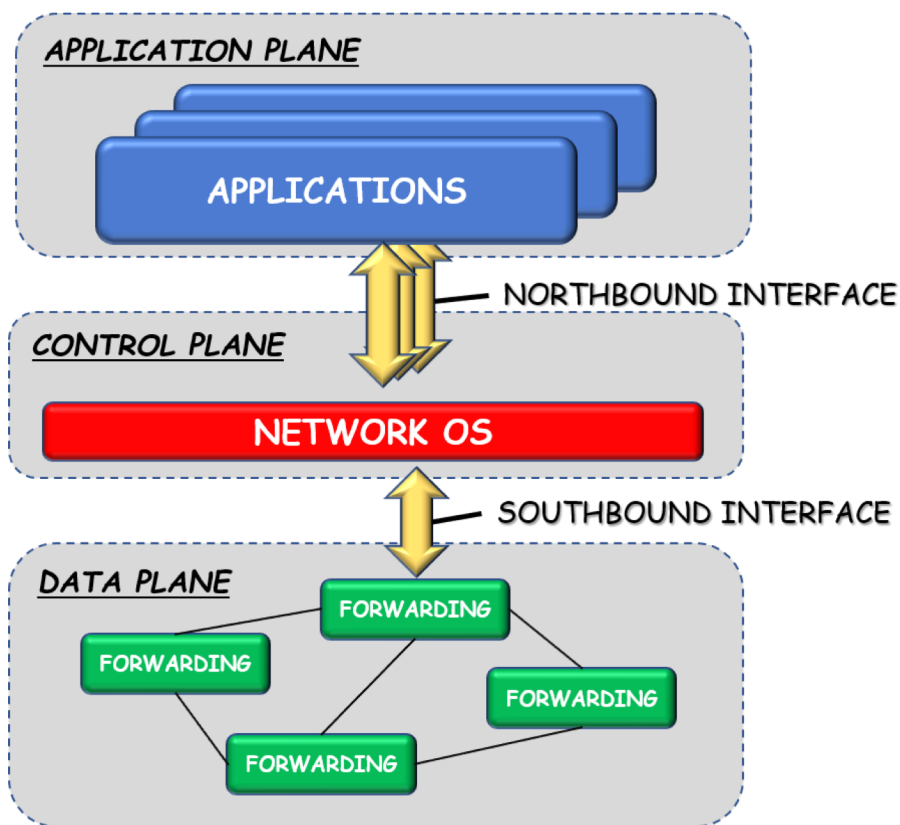


- ☐ **Integration** control plane / data plane
 - Control plane: routing protocols (e.g. OSPF)
 - Data plane: forwarding mechanisms
- ☐ Specialized **expensive hardware**
- ☐ Architecture **not much flexible**



Introduction to SDN (2/2)

...to **SDN paradigm**



- ❑ Control plane / data plane **separation**
- ❑ Several **advantages**
 - More flexibility
 - More services
 - Costs reduction



Introduction to SRv6

- ❑ **Segment routing** is a form of **source routing**
- ❑ It can use **MPLS** or **IPv6** as **dataplane technology**
 - In our project we consider IPv6 dataplane (**SRv6**)
- ❑ In SRv6 **segments** are IPv6 addresses
- ❑ The instructions are encoded in the packet header (**SRH**)
- ❑ High **scalability**
- ❑ Provides a **new way** to implement SDN paradigm



Goals and Objectives

1. Defining a **SDN architecture based on SRv6**

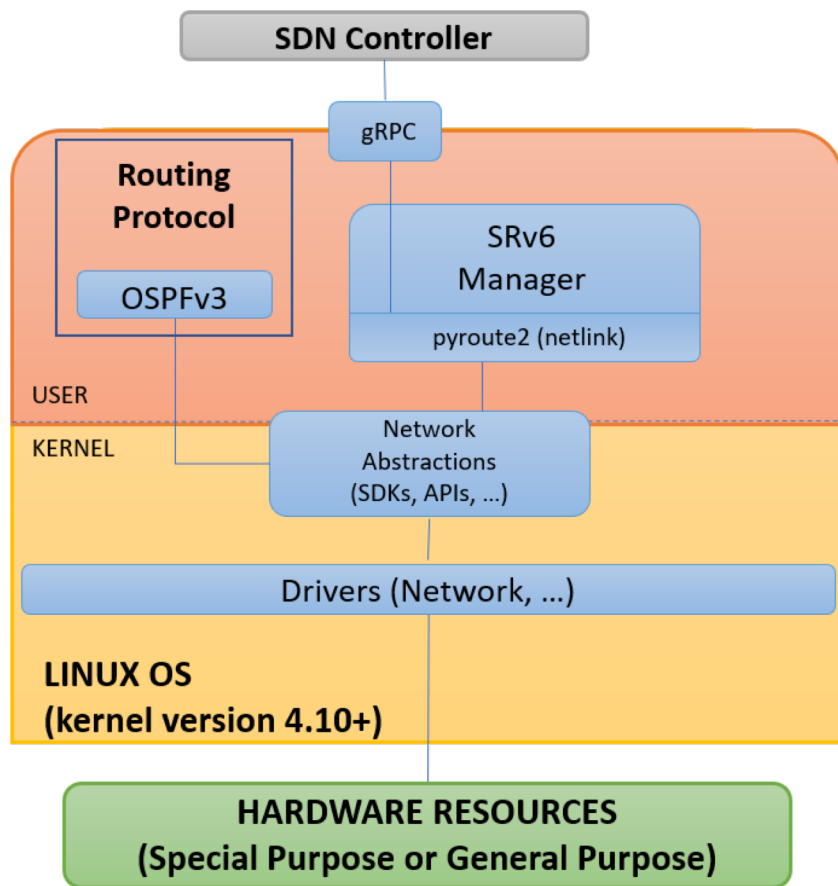
- ☐ Realization of a **SRv6 enabled node**
- ☐ Realization of a **Southbound protocol** based on gRPC

2. Study of two use cases

- ☐ Traffic Engineered Virtual Private Networks (**VPN-TE**)
- ☐ **Integration** within a **cloud orchestrator** (e.g. OpenStack)



SDN-SRv6 Architecture (1/3): SRv6 enabled node



Our implementation uses

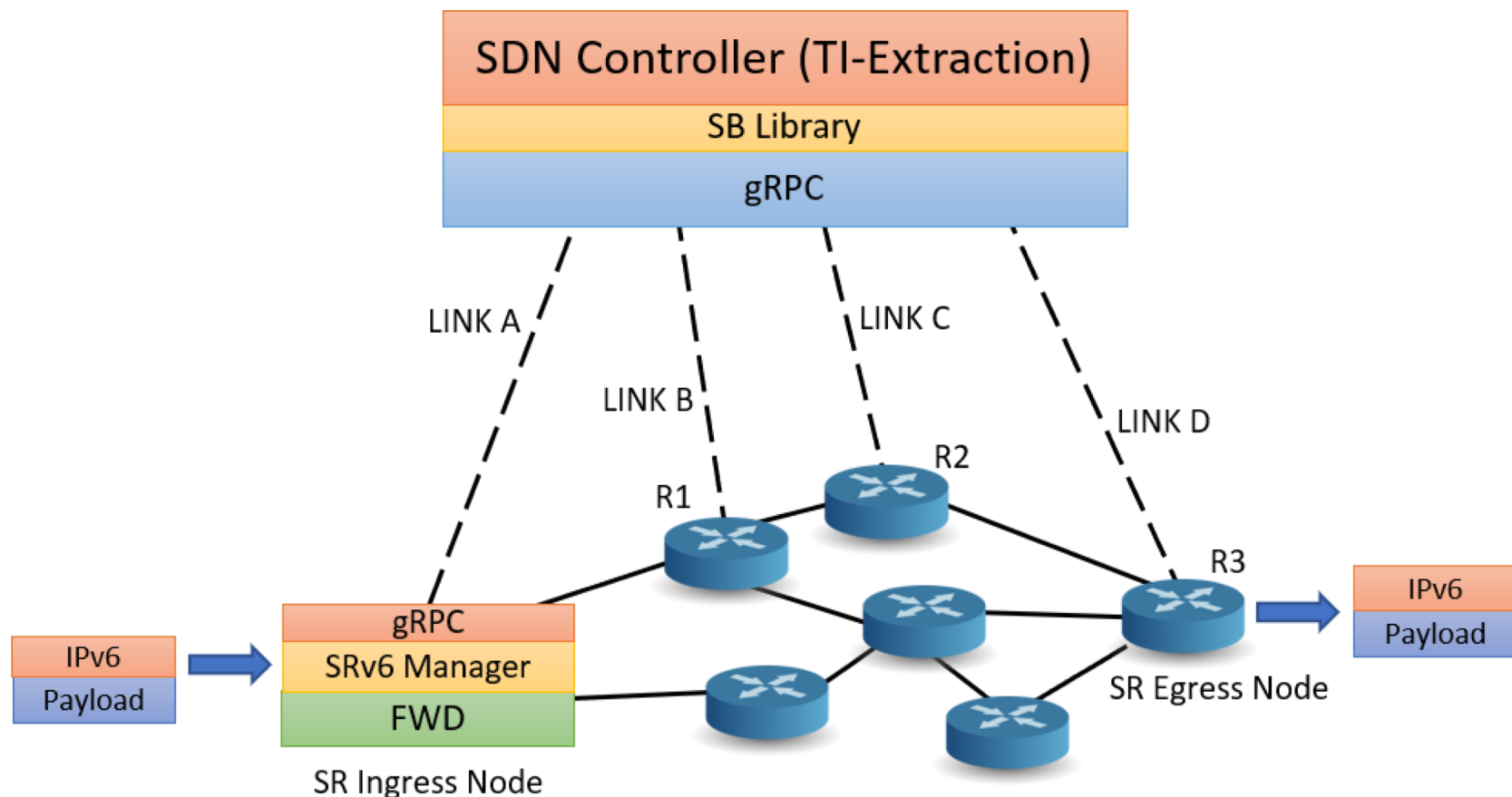
- ❑ **Commodity hardware**
- ❑ Linux Kernel **general purpose** distribution
- ❑ Quagga implementation of **OSPFv3**

We are developing

- ❑ **Southbound protocol** based on gRPC
- ❑ **SRv6 Manager** written in Python



SDN-SRv6 Architecture (2/3): overview





SDN-SRv6 Architecture (3/3): TI extraction example

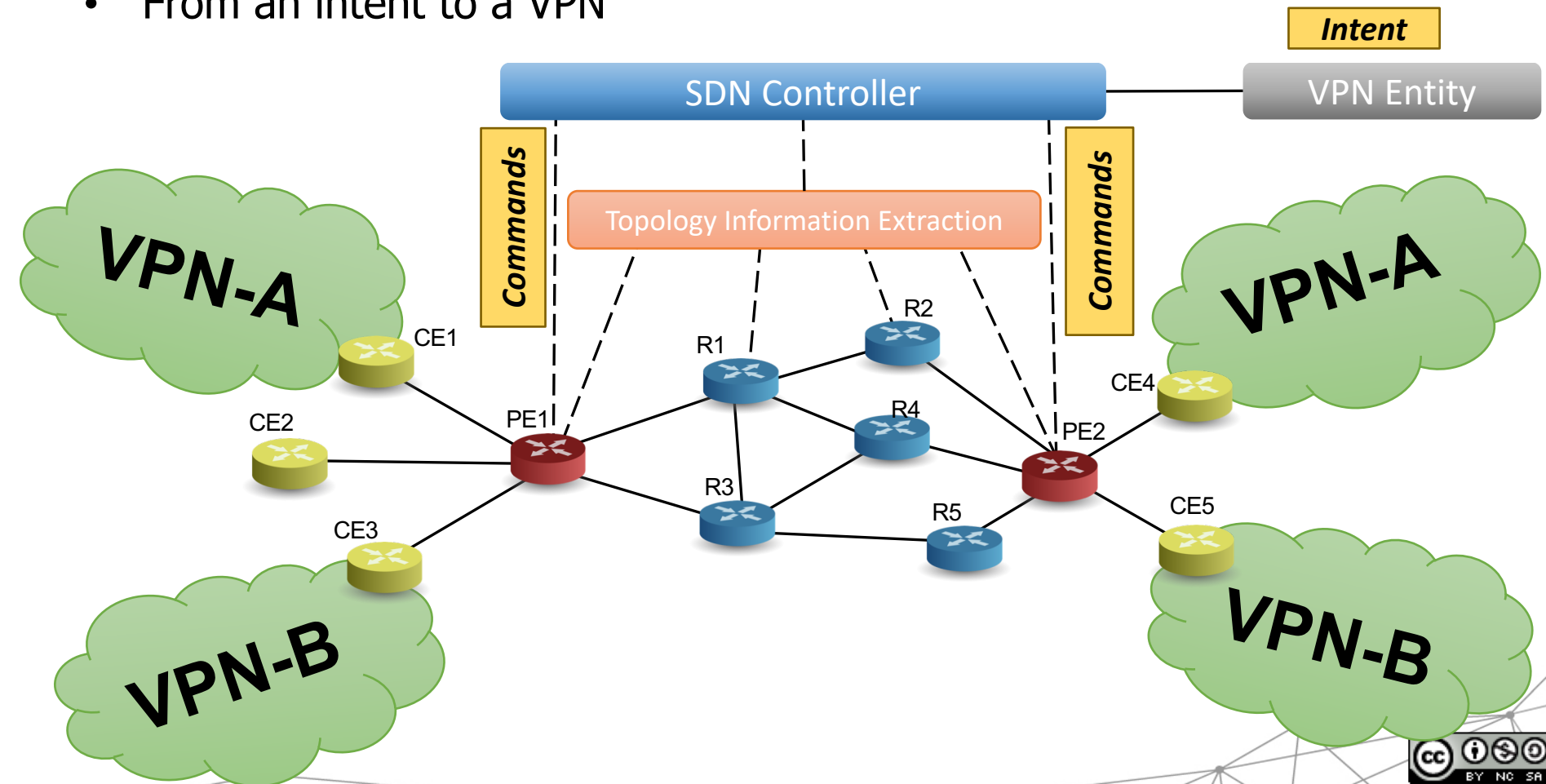
```
"nodes": [  
  {  
    "id": "fdf0:0:0:6::/64",  
    "type": "stub_network"  
  },  
  ...  
  {  
    "id": "0.0.0.1",  
    "interfaces": {  
      "ads1-eth1": "fdf0::1/64",  
      "ads1-eth2": "fdf0:0:0:1::1/64",  
      "ads1-eth3": "fdf0:0:0:3::1/64",  
      "ads1-eth4": "fdf0:0:0:6::1/64",  
      "lo": "fdff::1/128"  
    },  
    "loopbackip": "fdff::1/128",  
    "mgmtip": "2000::1",  
    "type": "router"  
  },  
  ...  
]
```

```
"links": [  
  {  
    "lhs_intf": "ads1-eth1",  
    "lhs_ip": "fdf0::1/64",  
    "net": "fdf0::/64",  
    "rhs_intf": "ads2-eth1",  
    "rhs_ip": "fdf0::2/64",  
    "source": 6,  
    "source_id": "0.0.0.1",  
    "target": 8,  
    "target_id": "0.0.0.2"  
  },  
  ...  
]
```



SRv6-based VPN (1/3): a use case overview

- From an intent to a VPN



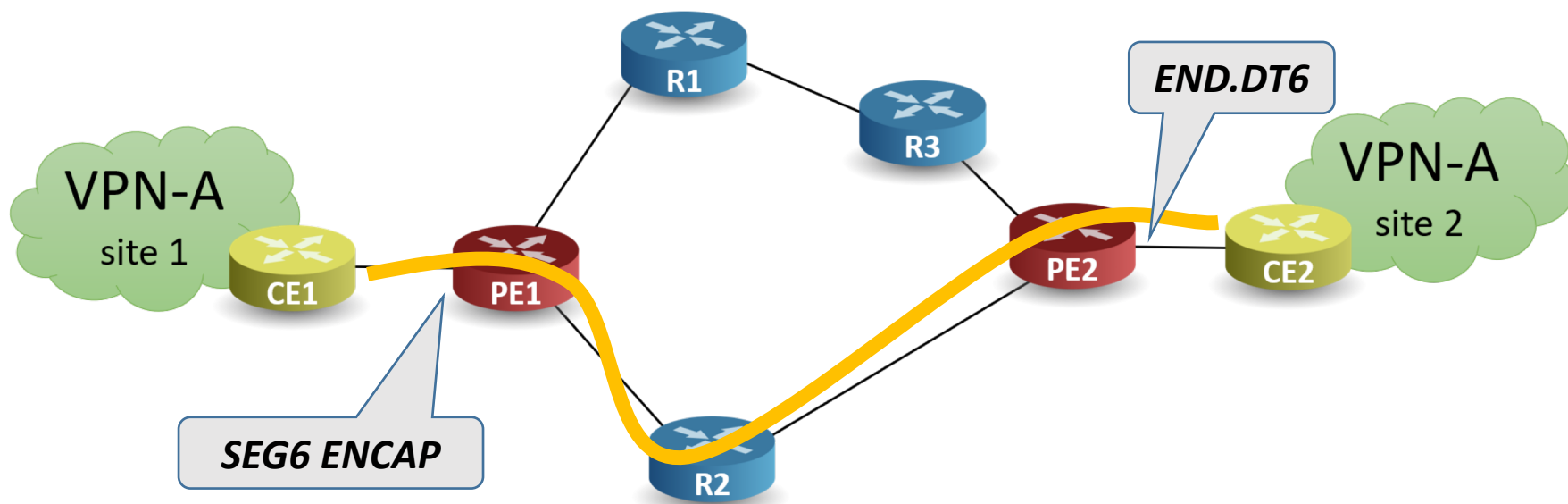


SRv6-based VPN (2/3): implementation details

- ❑ Our solution is based on **Linux VRFs**
- ❑ Nothing stored in the **core routers**
- ❑ **Edge routers** store interfaces belonging to the VPN
 - Local interfaces
 - Remote interfaces
- ❑ The **SDN controller** stores VPNs informations
 - Name
 - Interfaces
 - Table ID
 - SID
 - Tenant ID



SRv6-based VPN (3/3): an example





Intent based emulation

❑ Two types of **emulation**

- **Mininet emulation**
- **distributed emulation**
(VM or physical nodes)

❑ 4 types of nodes:

- SRv6 enabled **routers**
- **terminal nodes**
- **VNF** nodes
- **SDN controller**

❑ Emulation environment

- a **web GUI** to design the topology to be emulated
- **deployment scripts** to configure the nodes
- the possibility to **run experiments** over the emulated topology



We are working on...

- ❑ Design and realization of a **SRv6 enabled node**
- ❑ Design and realization of a **Southbound protocol** based on gRPC
 - Management of SRv6 policies
 - Management of VPN policies
- ❑ Design and realization of a **SRv6-based VPN**
- ❑ Contribute to the opensource project **Pyroute2**
 - Added seg6local support
- ❑ Development of **tools for the emulation** of a SRv6 enabled network

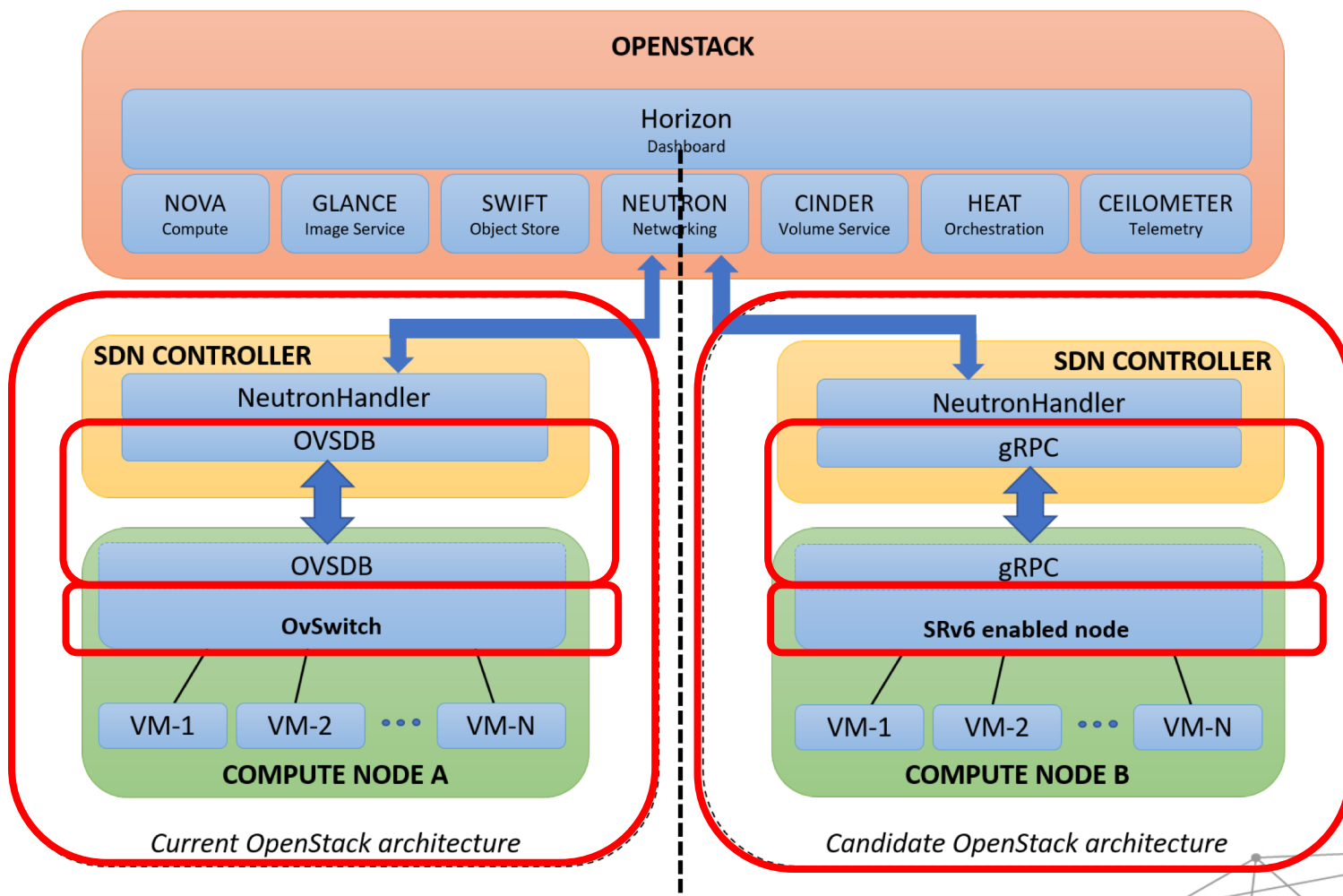


Our plans for the future

- ❑ Development of a **northbound interface** for VPNs
 - **gRPC**
 - Command Line Interface (**CLI**)
- ❑ More complex VPN services, such as
 - Traffic Engineered VPNs (**VPN-TE**)
- ❑ Implementation of Service Function Chaining (**SFC**) scenarios
- ❑ **Integration** within a **cloud orchestrator**



Integration within a cloud orchestrator: OpenStack



Thanks for your attention!
Any Questions or Comments?

