OpenStack Day SP 2018

IPv6 no OpenStack

JOSÉ CASTILLO LEMA

DIRETORIA DE OPERAÇÕES E SOLUÇÕES AO CLIENTE

openstack.

mandic CLOUD SOLUTIONS + rivendel

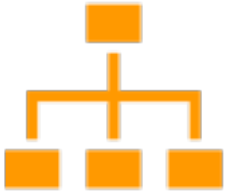
ESPECIALISTAS EM CLOUDS.

AGENDA

OPENSTACK DAY SP 2018 - IPV6 NO OPENSTACK

APRESENTAÇÃO SOLUÇÃO OPENSTACK

CRIAÇÃO DE SUBREDES IPv6



CONEXÃO COM A INTERNET

IMPLEMENTAÇÃO

MANDIC *IN A NUTSHELL*

Provedor **público** de nuvem

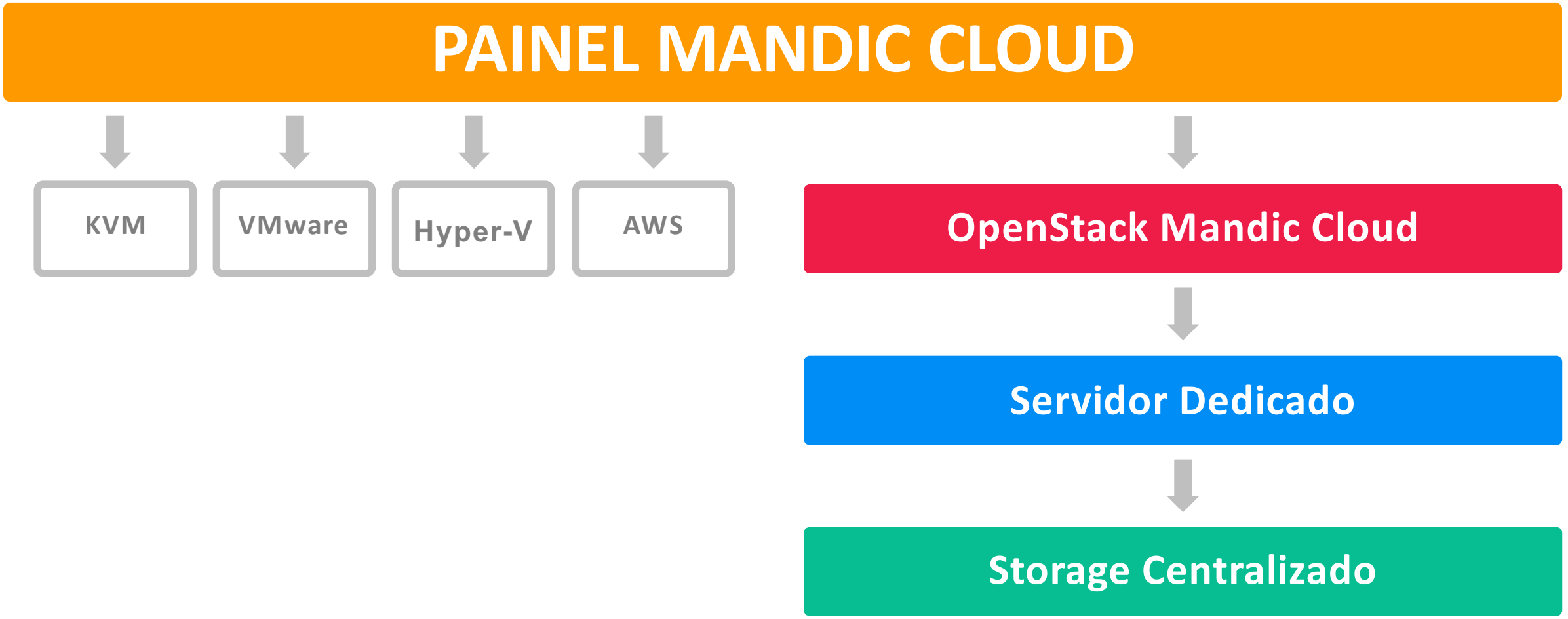
Modelo de serviço **IaaS**

Forte foco **empresarial**

mandic CLOUD SOLUTIONS + rivendel

ESPECIALISTAS EM CLOUDS.

ARQUITETURA



PARCEIROS

EQUIPE

GERD
JAKOBOVITSCH

SERGIO
DOS SANTOS OLIVEIRA

JOSE
CASTILLO LEMA

POR QUE IPv6?

+ **Abundância** de endereços IPv6
+ Elimina necessidade de *floating ips*
+ **Integração** de redes IPv6 de clientes
+ Elimina necessidade de NAT
+ Sem tráfego *broadcast*
+ Internet das Coisas

ONDE IMPLEMENTAR IPv6?

+ Servidores físicos

+ API – *endpoints*

+ **Máquinas virtuais** – *tenant networks*
⇏ provider networks

REQUISITOS DE IMPLEMENTAÇÃO

Ambiente ***dual-stack.*** Todos os projetos/clientes com:

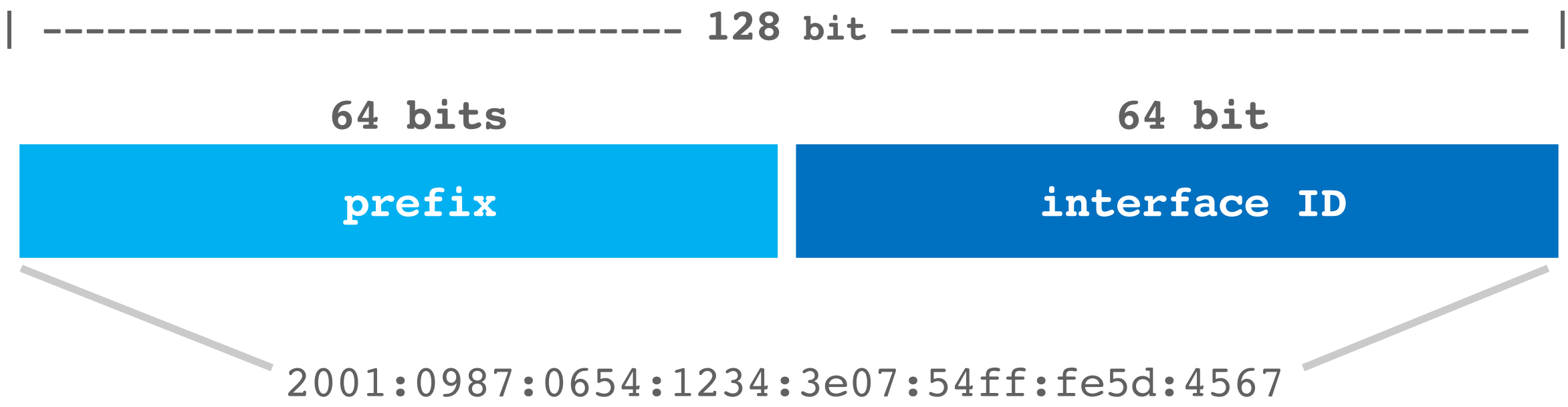
+ rede IPv4 /24 privada
+ rede **IPv6 /64** pública
+ **roteador virtual** privado
  + HA
  + Sem DVR

IPv6 NO OPENSTACK

CRIAÇÃO DE SUBREDES IPv6

IPv6 – ATRIBUIÇÃO DE ENDEREÇOS

SLAAC - Stateless Address Auto Configuration



IPv6 – ATRIBUIÇÃO DE ENDEREÇOS

DHCPv6 *stateless*

+ DNS
+ rotas

DHCPv6 *statefull*

IPv6 – ATRIBUIÇÃO DE ENDEREÇOS

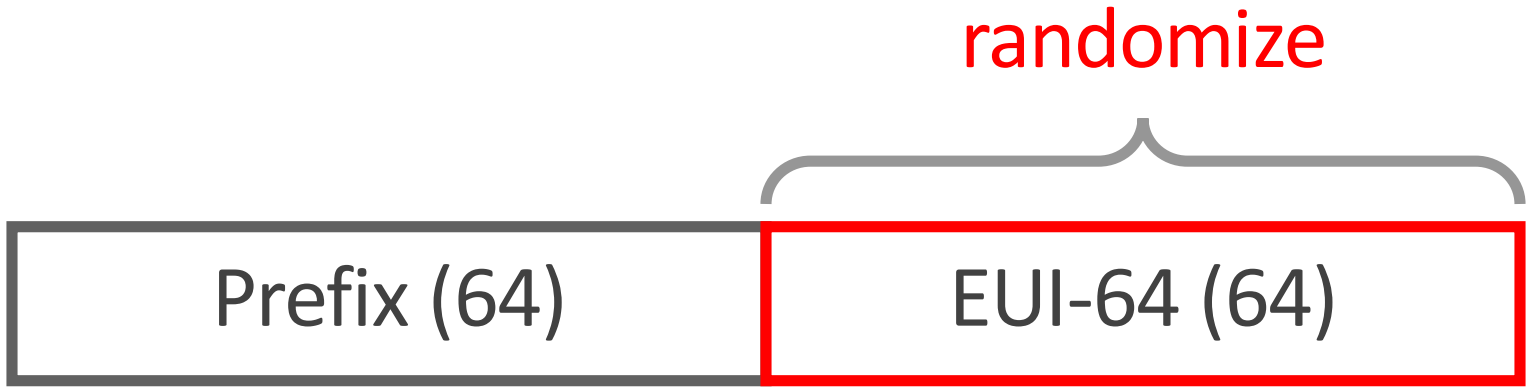
```
jose@jose-think:~$ openstack subnet show sub6-int
+-------------------+-----------------------------------------------------------+
| Field             | Value                                                     |
+-------------------+-----------------------------------------------------------+
| allocation_pools  | 2804:[redacted]::2-2804:[redacted]000:ffff:ffff:ffff:ffff |
| cidr              | 2804:[redacted]::/64                                      |
| created_at        | 2018-03-12T18:30:55Z                                      |
| description       |                                                           |
| dns_nameservers   | 2001:4860:4860::8888                                      |
| enable_dhcp       | True                                                      |
| gateway_ip        | 2804:5d4:1:1000::1                                        |
| host_routes       |                                                           |
| id                | 2b1bffe8-0624-44f3-8f43-140d4fd1736d                      |
| ip_version        | 6                                                         |
| ipv6_address_mode | dhcpv6-stateless                                          |
| ipv6_ra_mode      | dhcpv6-stateless                                          |
| name              | sub6-int                                                  |
| network_id        | 87130a10-be1a-493f-8f0f-511575ce7617                      |
| project_id        | ed5f5ffe834244479ef3bc2bbc905fad                          |
| revision_number   | 2                                                         |
| segment_id        | None                                                      |
| service_types     |                                                           |
| subnetpool_id     | 77f5924a-5058-4a5f-9a02-91d04ab131ce                      |
| tags              |                                                           |
| updated_at        | 2018-03-12T18:30:55Z                                      |
+-------------------+-----------------------------------------------------------+
```

IPv6 – ATRIBUIÇÃO DE ENDEREÇOS

```
ip_version           | 6
ipv6_address_mode    | dhcpv6-stateless
ipv6_ra_mode         | dhcpv6-stateless
```

SLAAC: WINDOWS

+ Privacy Extensions for IPv6 SLAC

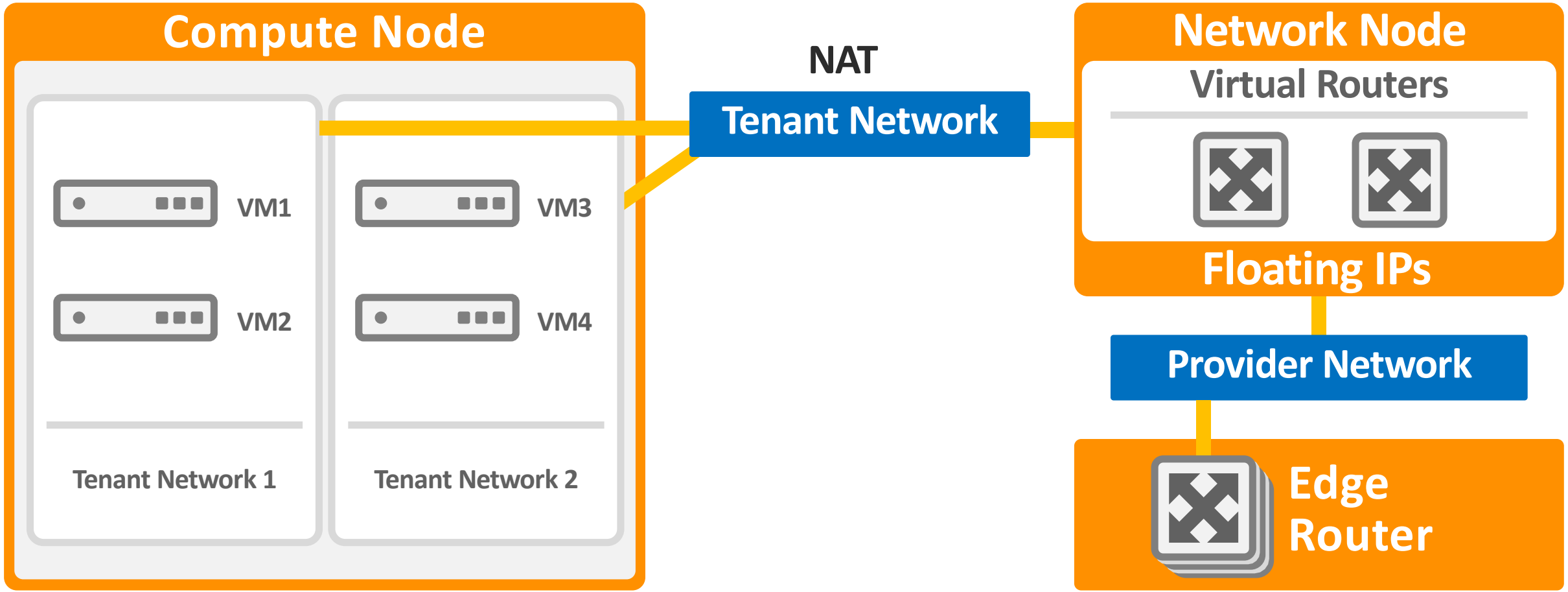


netsh interface ipv6 set global randomizeidentifiers=disabled

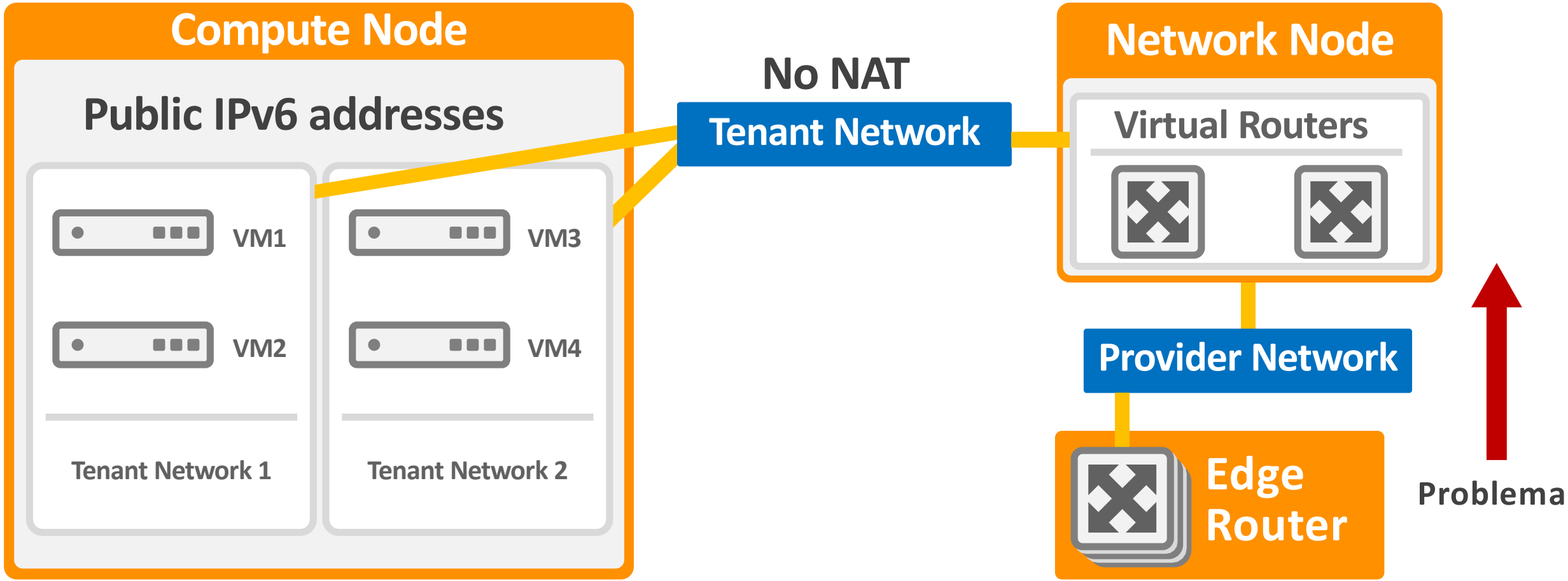
IPv6 NO OPENSTACK

CONEXÃO COM A INTERNET

ARQUITETURA IPv4



ARQUITETURA IPv6



ARQUITETURA IPv6

Diferenças:

+ sem NAT
+ os endereços IPv6 públicos estão nas vms (nos compute nodes)
+ sem *floating IPs*, endereços IPv6 públicos são tratados como *fixed IPs* dentro do OpenStack

Desafio:

+ as vms conseguem sair para a Internet via IPv6
+ o roteador *edge* não sabe para qual roteador virtual encaminhar os pacotes

MODELOS DE IMPLEMENTAÇÃO

E-BGP usando neutron-bgp-agent

+ um agente BGP rodando nos *control nodes* atua como BGP *speaker*, propagando as rotas para o roteador *edge*
+ o roteador *edge* propaga as rotas para a Internet

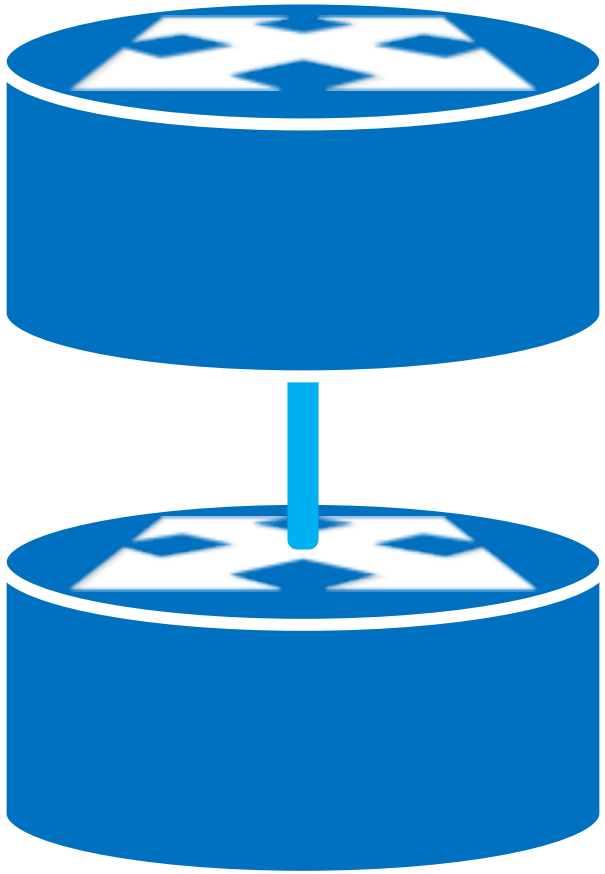
IPv6 *prefix delegation* usando neutron-pd-agent

+ o roteador *edge* atua como servidor DHCPv6
+ o roteador virtual solicita via DHCPv6 um prefixo de rede ao roteador *edge*

IPv6 PREFIX DELEGATION

Edge router

Virtual router



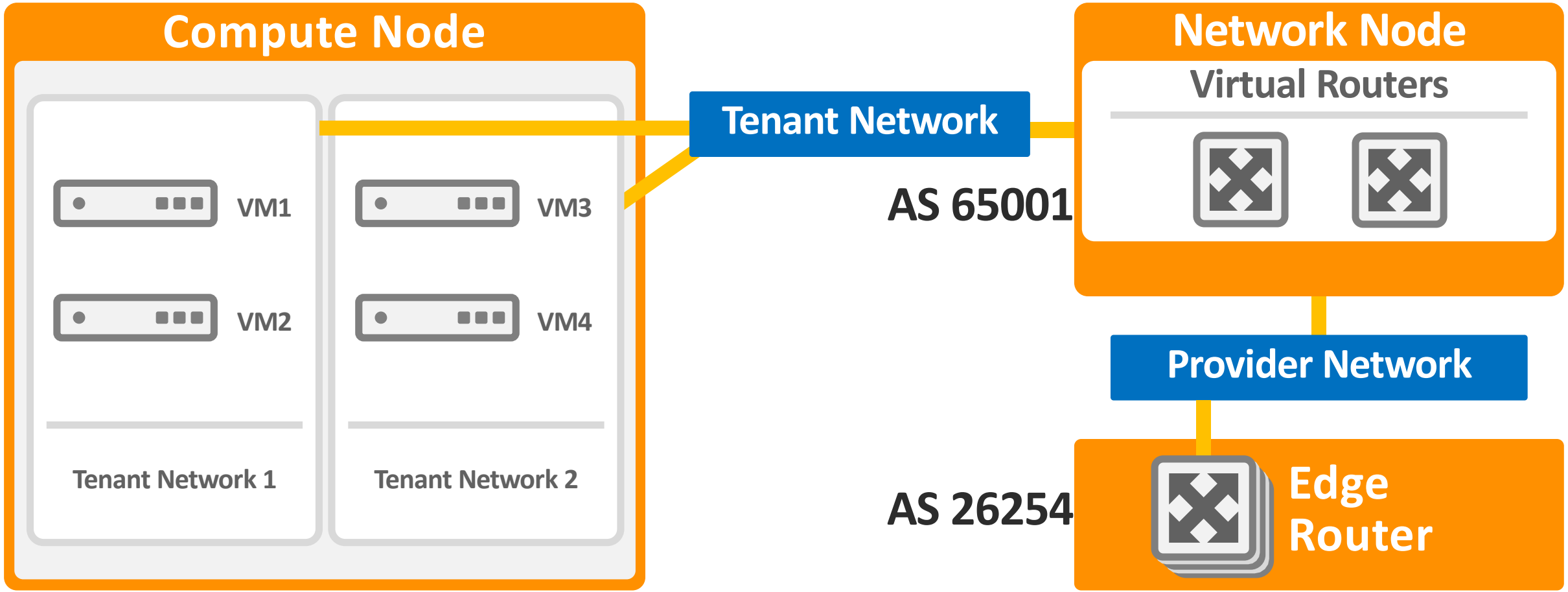
2

Sure, take
2001:db8:bad:face::/64

1

Can I have a prefix?

ROTEAMENTO


Compute Node
VM1
VM2
VM3
VM4
Tenant Network 1
Tenant Network 2
Tenant Network
AS 65001
Network Node
Virtual Routers
Provider Network
AS 26254
Edge Router

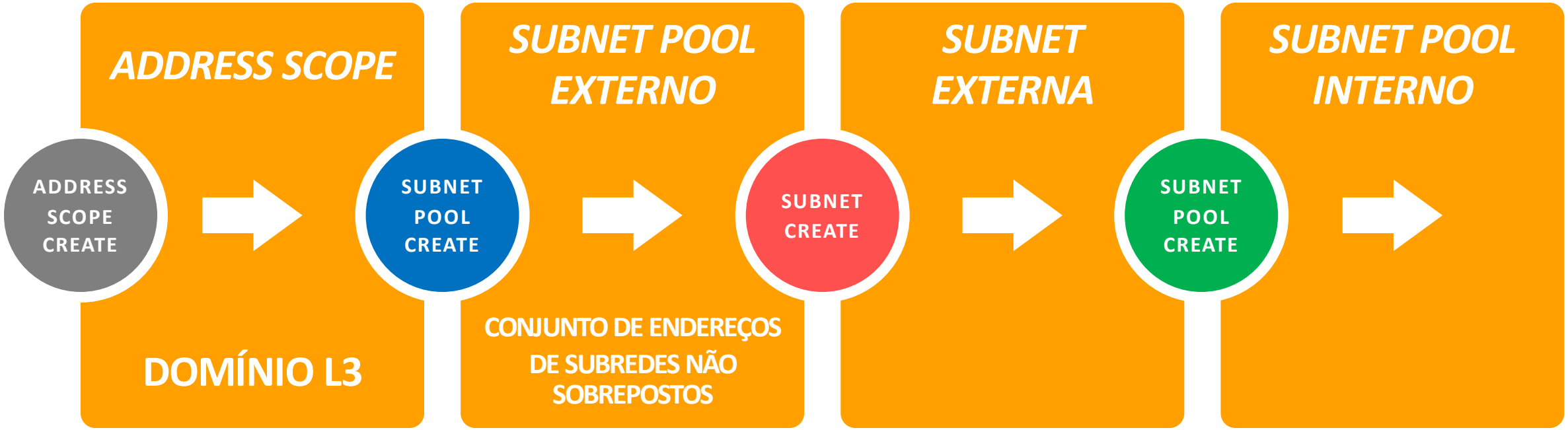
NEUTRON-BGP-AGENT

1. **Criação** do BGP *speaker* com AS privado
2. Associação com a **rede externa**
3. Definição do ***peer*** (roteador *edge*) com AS público
4. Associação do BGP *speaker* com o *peer*
5. Associação do BGP *speaker* com um *network node*

NEUTRON-BGP-AGENT

1. neutron **bgp-speaker-create** --ip-version 6 --local-as 65001 --advertise-floating-ip-host-routes false **bgp1**
2. neutron **bgp-speaker-network-add** bgp1 net-ext
3. neutron **bgp-peer-create** --peer-ip 2804:1234:1234:::1 --remote-as 262545 **bgp1-peer1**
4. neutron **bgp-speaker-peer-add** bgp1 bgp1-peer1
5. neutron **bgp-dragent-speaker-add** `openstack network agent list --agent-type bgp --host cldvcpctr001.cloudmandic.com.br -c ID -f value` bgp1

CRIAÇÃO PILHA IPv6 – ADMIN



CRIAÇÃO PILHA IPv6 – ADMIN

openstack **address scope create** --share --ip-version 6 **address-scope-ip6**

openstack **subnet pool create** --address-scope address-scope-ip6 --pool-prefix 2804:1234:1:1234::/64 --default-prefix-length 64 --max-prefix-length 64 **ext-pool-ip6**

openstack **subnet create** --network net-ext --ip-version 6 --subnet-pool ext-pool-ip6 --no-dhcp --dns-nameserver 2001:4860:4860::8888 --allocation-pool start=2804:1234:1:1234::100,end=2804:1234:1:1234:ffff:ffff:ffff:ff00 **sub6-ext**

openstack **subnet pool create** --address-scope address-scope-ip6 --share --pool-prefix 2804:1234:1:1000::/52 --default-prefix-length 64 --max-prefix-length 64 --default **default-pool-ip6**

CRIAÇÃO PILHA IPv6 – PROJETO

openstack **subnet create**

--network net-int

--ip-version 6

--ip_ra_mode dhcpv6-stateless

--ip_address_mode dhcpv6-stateless

--subnet-pool default-pool-ip6

--dns-nameserver 2001:4860:4860::8888

sub6-int

IPv6 NO OPENSTACK

IMPLEMENTAÇÃO

IMPLEMENTAÇÃO PRÓPRIA

+ WEB Service:
  Python **Flask**
+ BGP Speaker:
  RYU SDN Controller
+ Uso de alarmes via **aodh**
+ Alta disponibilidade
  2 agentes autônomos duplicados

AODH – DEFINIÇÃO DE ALARMES

/etc/ceilometer/events_definitions.yaml

```
- event_type: router.*
  traits:
    <<: *network_traits
    resource_id:
      fields: ['payload.router.id', 'payload.id']
- event_type: router.interface.*
  traits:
    <<: *network_traits
    port_id:
      fields: payload.router_interface.port_id
    subnet_id:
      fields: payload.router_interface.subnet_id
    router_id:
      fields: payload.router_interface.id
```

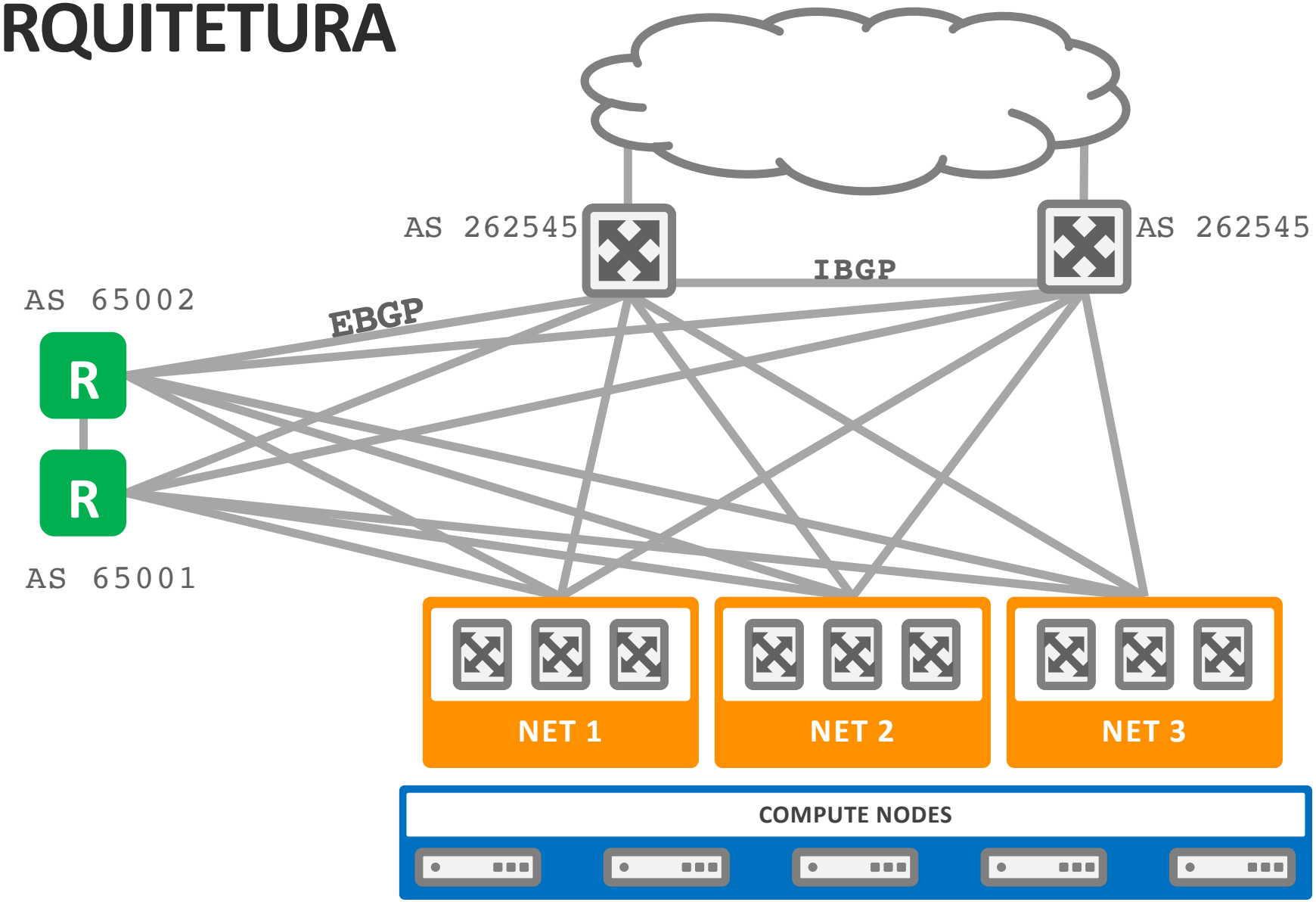
AODH – DEFINIÇÃO DE ALARMES

```
jose@jose-think:~$ openstack alarm show 991922bc-8b3b-435c-abb4-a74210c91889
+---------------------------+----------------------------------------------------------+
| Field                     | Value                                                    |
+---------------------------+----------------------------------------------------------+
| alarm_actions             | [u'http://10.[redacted].1:5125', u'http://10.[redacted].2:5125'] |
| alarm_id                  | 991922bc-8b3b-435c-abb4-a74210c91889                     |
| description               | Alarm when router.interface.* event occurred.            |
| enabled                   | True                                                     |
| event_type                | router.interface.*                                       |
| insufficient_data_actions | []                                                       |
| name                      | CD123456-routing                                         |
| ok_actions                | []                                                       |
| project_id                | 0b5930a580ea4cfc9451a1ef2d6f71af                         |
| query                     |                                                          |
| repeat_actions            | True                                                     |
| severity                  | critical                                                 |
| state                     | insufficient data                                        |
| state_timestamp           | 2018-04-06T21:43:45.750009                               |
| time_constraints          | []                                                       |
| timestamp                 | 2018-04-06T21:43:45.750009                               |
| type                      | event                                                    |
| user_id                   | 59c13f719bf24c11b66bacbb1efd839d                         |
+---------------------------+----------------------------------------------------------+
```

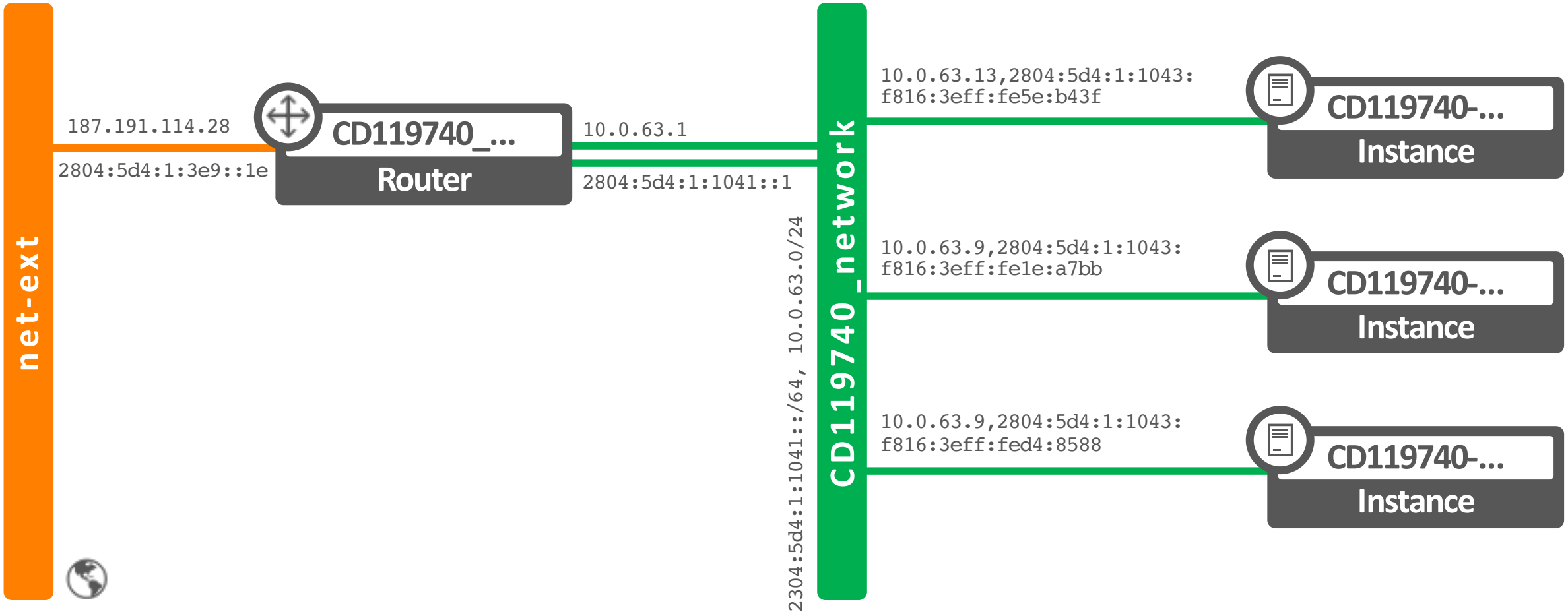
PROCESSO

1. O projeto é criado
2. É criado um alarme aodh para *router-add-subnet*
3. Um rede é criada para cada projeto
4. Um roteador virtual é criado para cada projeto
5. Duas subredes são criadas dentro da rede
    1. IPv4 a partir do *subnet pool* IPv4
    2. IPv6 a partir do *subnet pool* IPv6
6. O alarme notifica aos BGP *speakers*
7. O BGP speaker propaga a rota para os roteadores *edge*

ARQUITETURA



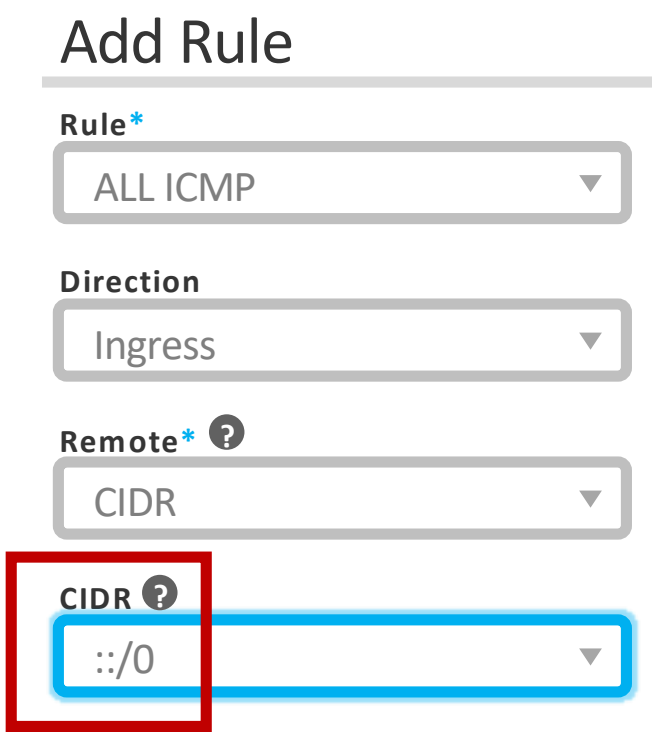
RESULTADO FINAL



RESULTADO FINAL

```
ubuntu@ub14d:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether fa:16:3e:96:e5:61 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.6/24 brd 10.0.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 2804:[redacted]:3eff:fe96:e561/64 scope global dynamic
       valid_lft 86375sec preferred_lft 14375sec
    inet6 fe80::f816:3eff:fe96:e561/64 scope link
       valid_lft forever preferred_lft forever
```

CONSIDERAÇÕES: *SECURITY GROUPS*



Manage Security Group Rules:

default (a9947ba5-522e-4708-b662-d8c5af3c9544)

Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix
Ingress	IPv6	Any	Any	-
Egress	IPv6	Any	Any	::/0

LIMITAÇÕES

+ cloud-init

PRÓXIMOS PASSOS

+ DNSaaS, DNS6,
  integração com **OpenStack Designate**

+ **VTEP** – Integração com **Cumulus**

+ **VPNaaS** usando *containers*

CONCLUSÃO

+ **Criação** de subredes IPv6 no OpenStack
  + Modos de atribuição de endereços

+ **Conexão** com a Internet
  + **IPv6 Preffix delegation**
  + **Roteamento**

+ **Implementação**
  + ***aodh***

LINKS

+ OpenStack IPv6 Networking

https://docs.openstack.org/liberty/networking-guide/adv-config-ipv6.html

+ BGP dynamic routing

https://docs.openstack.org/newton/networking-guide/config-bgp-dynamic-routing.html

+ Prefix delegation

https://docs.openstack.org/newton/networking-guide/config-ipv6.html#prefix-delegation

JOSÉ CASTILLO LEMA

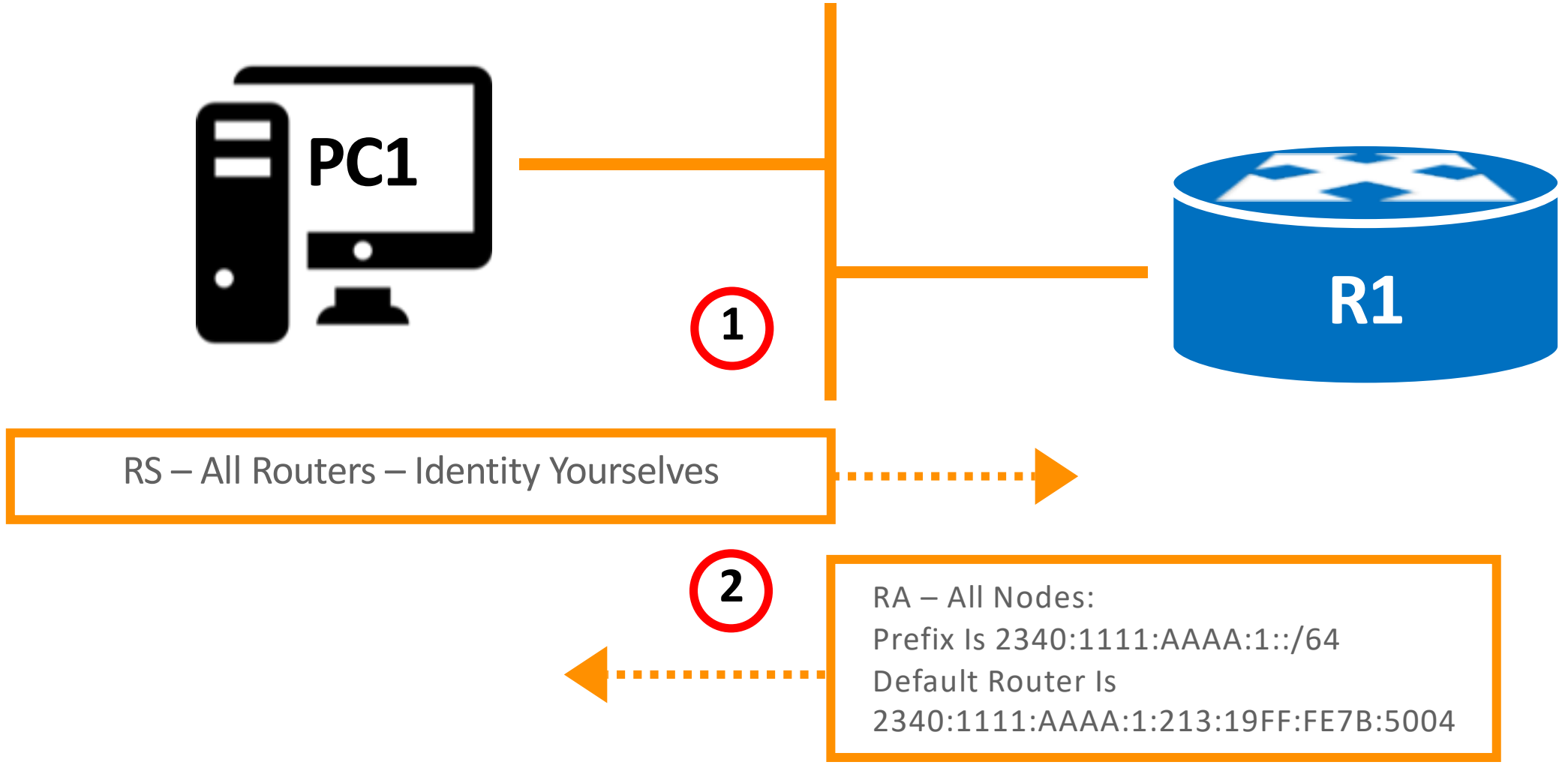
DIRETORIA DE OPERAÇÕES E SOLUÇÕES AO CLIENTE

especialistascloud@mandic.net.br

mandic CLOUD SOLUTIONS + rivendel

Especialistas em Clouds.

NEIGHBOR DISCOVERY PROTOCOL – NDP



OPENSTACK NEUTRON – PROPRIEDADES IPv6

ipv6_address_mode

+ determina como os endereços IPv6 são gerados e assignados
+ *slaac / dhcpv6-stateless / dhcpv6-stateful*

ipv6_ra_mode

+ determina como os roteadores virtuais enviam RAs
+ *slaac / dhcpv6-stateless / dhcpv6-stateful*

OPENSTACK NEUTRON – PROPRIEDADES IPv6

ipv6 ra mode	ipv6 address mode	radvd A,M,O	External Router A,M,O	Description
slaac	slaac	1,0,0	Off	Guest instance obtains IPV6 address from OpenStack managed radvd using SLAAC.
dhcpv6-stateful	dhcpv6-stateful	0,1,1	Off	Guest instance obtains IPV6 address from dnsmasq using DHCPv6 stateful and optional info from dnsmasq using DHCPv6
dhcpv6-stateless	dhcpv6-stateless	1,0,1	Off	Guest instance obtains IPv6 address from OpenStack managed radvd using SLAAC and optional info from dnsmasq using DHCPv6
Texto...	Texto...	Texto...	Texto...	Texto...