

Transit router – The distributed router for OVN interconnect.

Aleš Musil

Principal Software Engineer

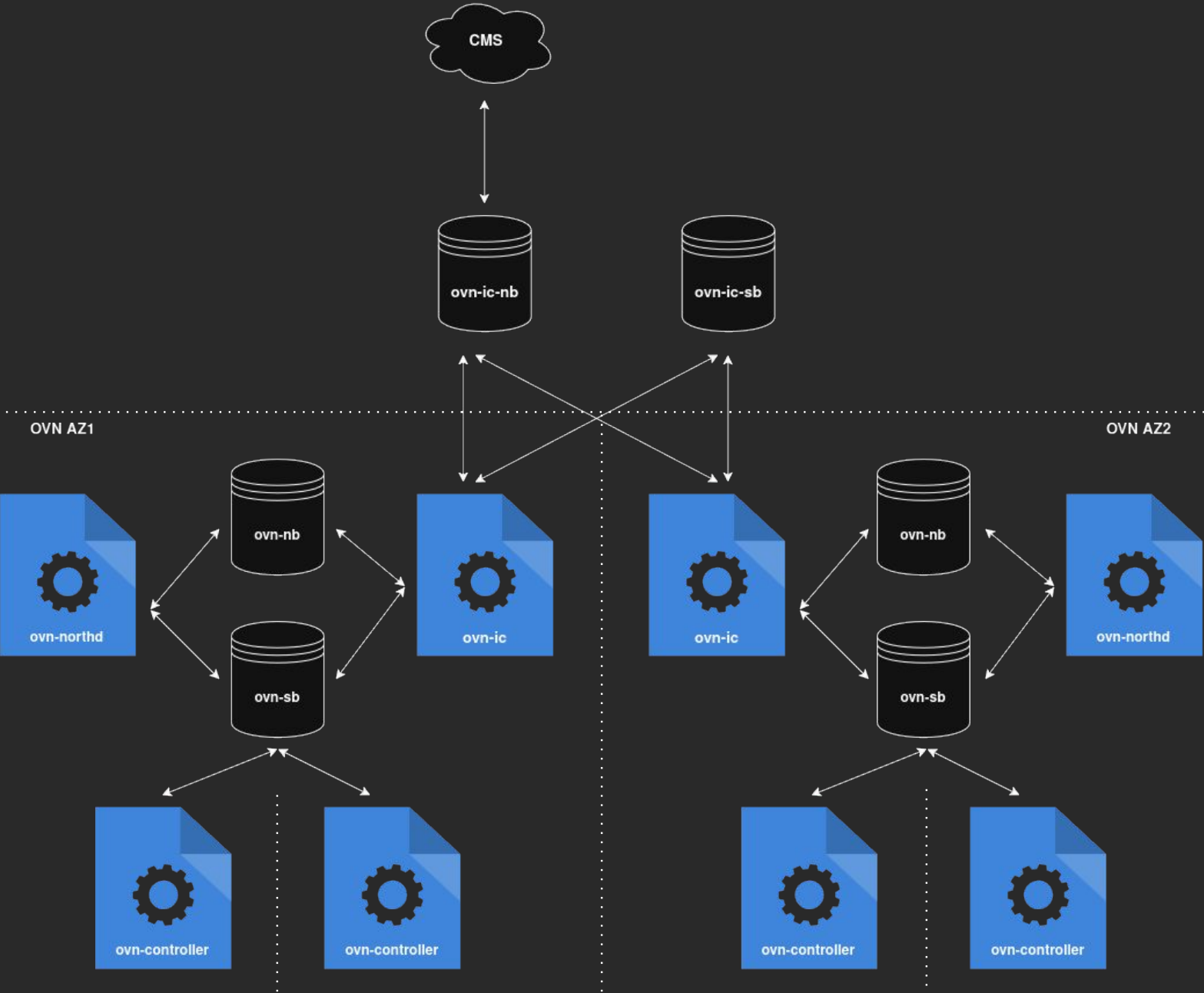
Mairtin O'Loingsigh

Senior Software Engineer

What we'll discuss today

- ▶ ovn-ic topology
- ▶ Why TR?
- ▶ Follow the traffic
- ▶ ovn-ic-nbctl
 - tr-add
 - trp-add
 - tr-del
- ▶ Demo of ovn-ic
- ▶ List of open ended questions





Why TR?

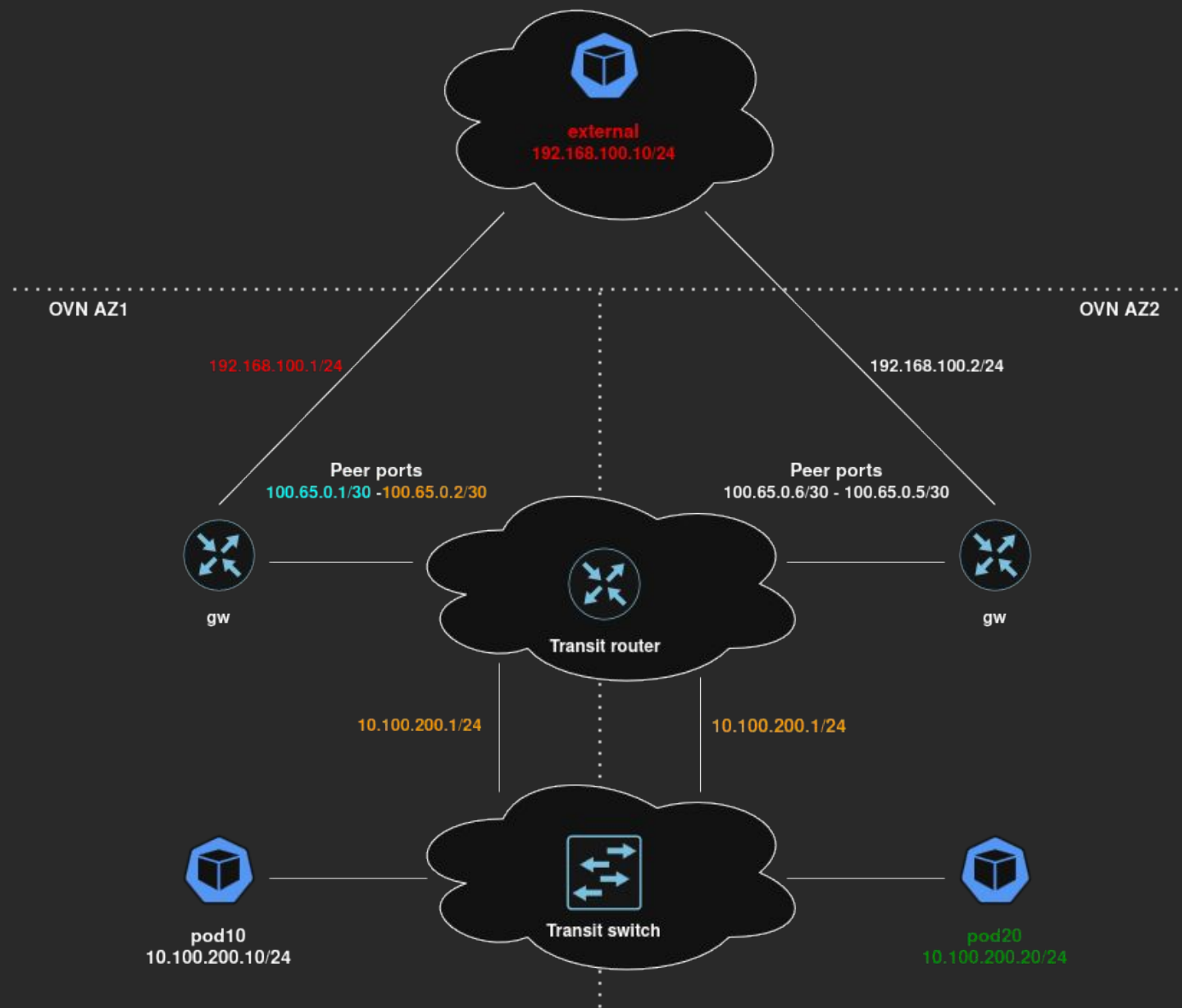
- ▶ There isn't any router equivalent in ovn-ic.
- ▶ Depending on the topology it might be tricky to achieve router like behavior within ovn-ic.
- ▶ TR behaves like distributed router in OVN.
- ▶ Allows to simplify the overall topology.



Follow the traffic...



Example topology

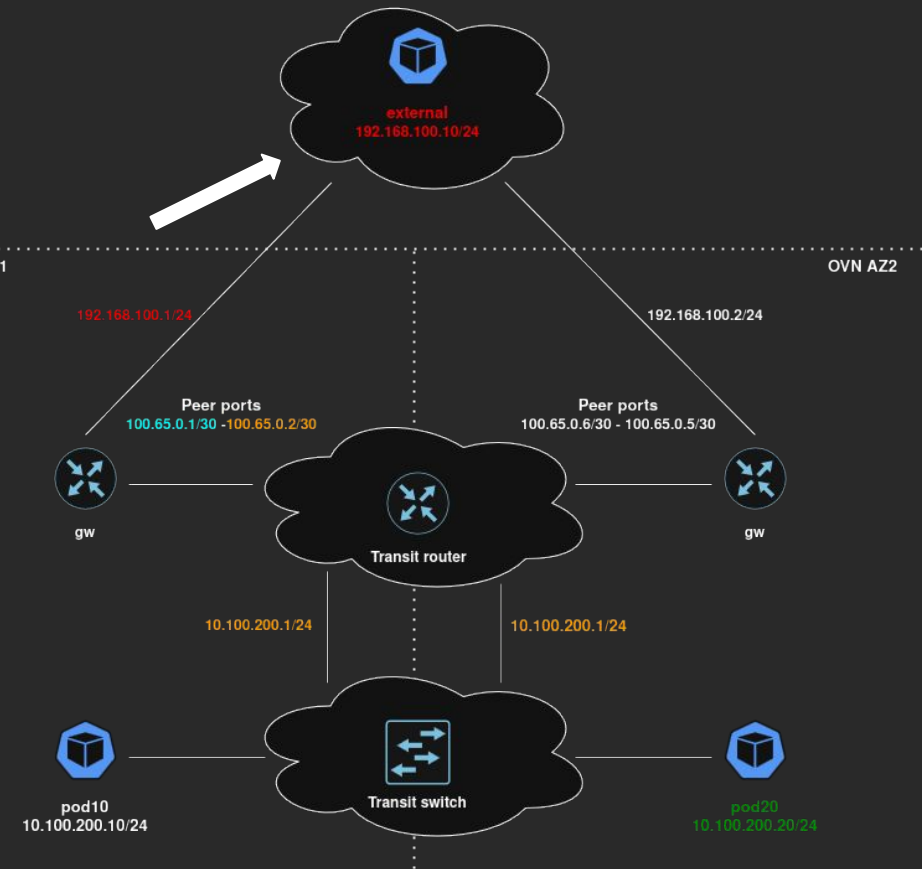


00:00:00:00:20:10 > 00:00:00:00:20:00, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 10010, offset 0, flags [DF], proto ICMP (1), length 84)

192.168.100.10 > 10.100.200.20: ICMP echo request, id 2830, seq 1, length 64

00:00:00:00:20:10 > 00:00:00:00:20:00, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 10175, offset 0, flags [DF], proto ICMP (1), length 84)

192.168.100.10 > 10.100.200.20: ICMP echo request, id 2830, seq 2, length 64



```
recirc_id(0),in_port(external-p),eth(src=00:00:00:00:20:10,dst=00:00:00:00:20:00),eth_type(0x0800),ipv4(src=192.168.100.8/255.255.255.248,dst=10.100.200.16/255.255.255.240),icmp
```

```
actions:ct(zone=2,nat),recirc(0x33)
```

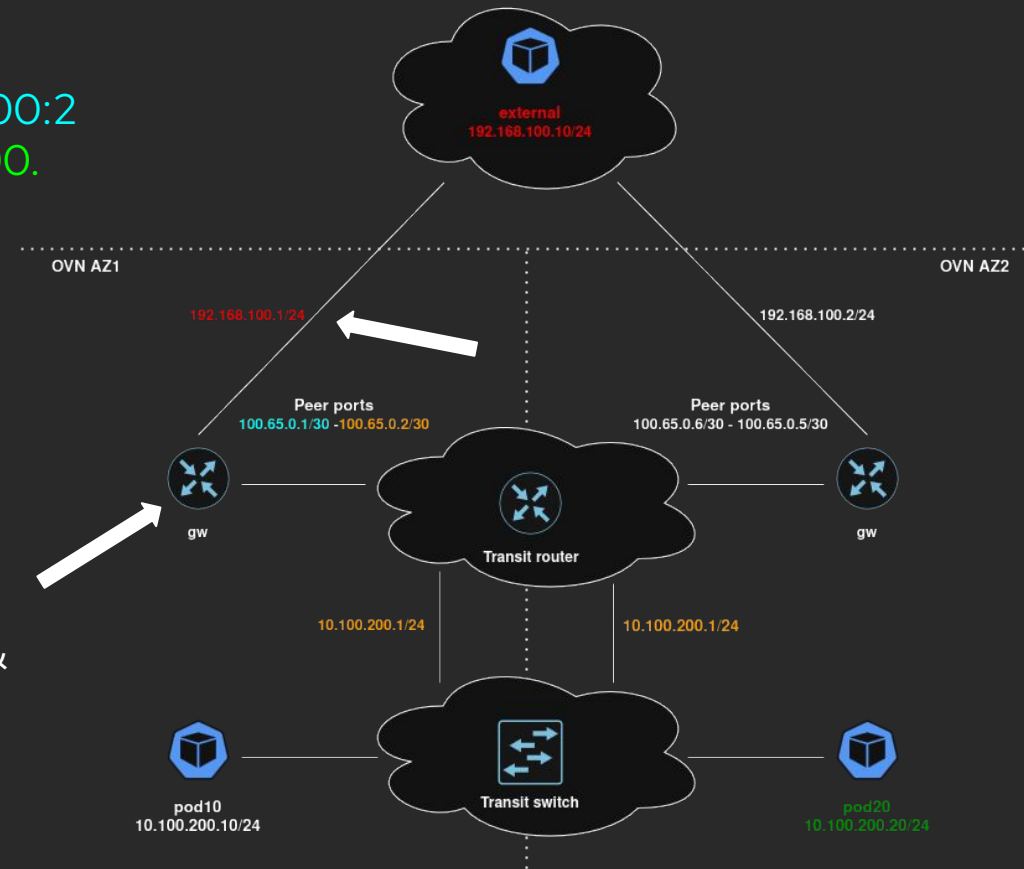
* Logical datapath: "public"

* Port Binding: logical_port "external", chassis-name "ovn-chassis-1"

* Logical datapath: "gw"

* Logical flow: table=16 (lr_in_ip_routing), priority=196, match=(reg7 == 0 && ip4.dst == 10.100.200.0/24), actions=(reg0 = 100.65.0.2; reg5 = 100.65.0.1;)

* Logical flow: table=1 (lr_out_undnat), priority=50, match=(ip), actions=(flags.loopback = 1; ct_dnat;)



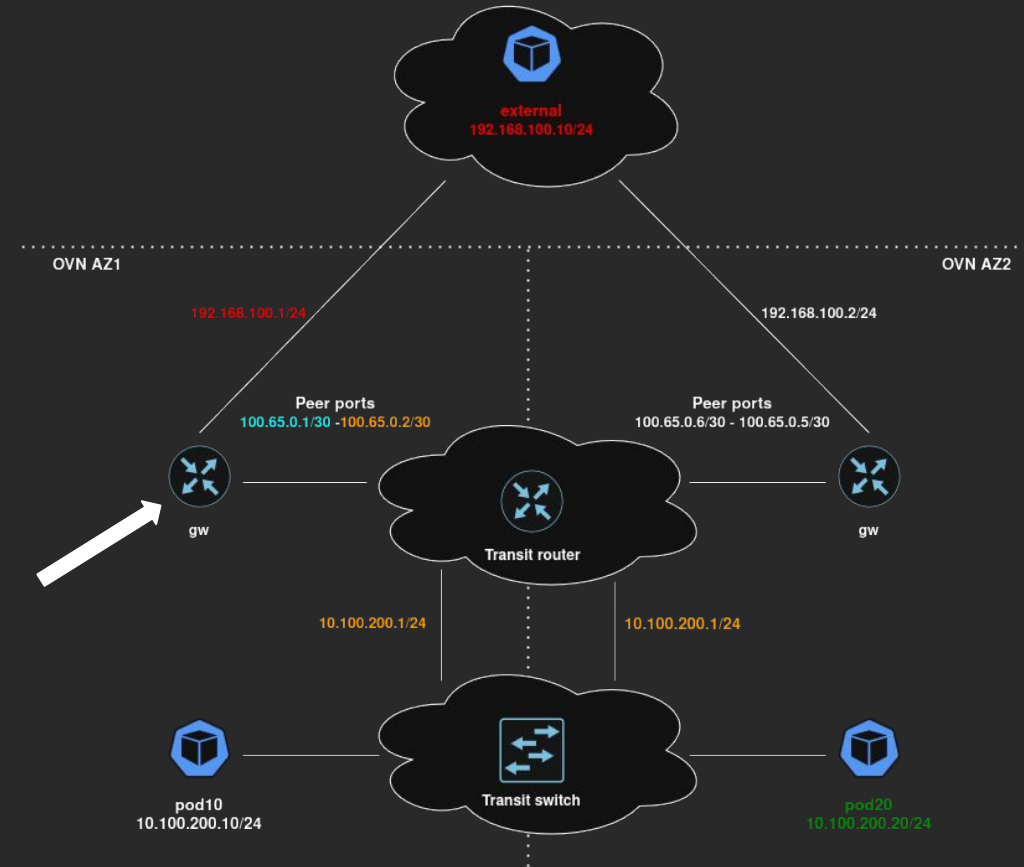
```
recirc_id(0x33),in_port(external-p)
```

```
actions:ct(commit,zone=1,nat(src=100.65.0.1),recirc(0x34)
```

* Logical datapath: "gw"

* Logical flow: table=3 (lr_out_snat), priority=25, match=(ip && ip4.src == 192.168.100.0/24 && (!ct.trk || !ct.rpl)), actions=(ct_snat(100.65.0.1);)

* NAT: external IP 100.65.0.1 logical_ip 192.168.100.0/24 type snat



```
recirc_id(0x34),in_port(external-p),ipv4(src=100.65.0.1,dst=10.100.200.20),icm
pactions:ct_clear,set(tunnel(tun_id=0xa,src=170.168.0.7,dst=170.168.0.8)),gene
v_sys_6081
```

* Logical datapath: "tr"

* Logical flow: table=16 (lr_in_ip_routing), priority=198, match=(ip4.dst == 10.100.200.0/24), actions=(reg5 = 10.100.200.1; output = "tr-ts";)

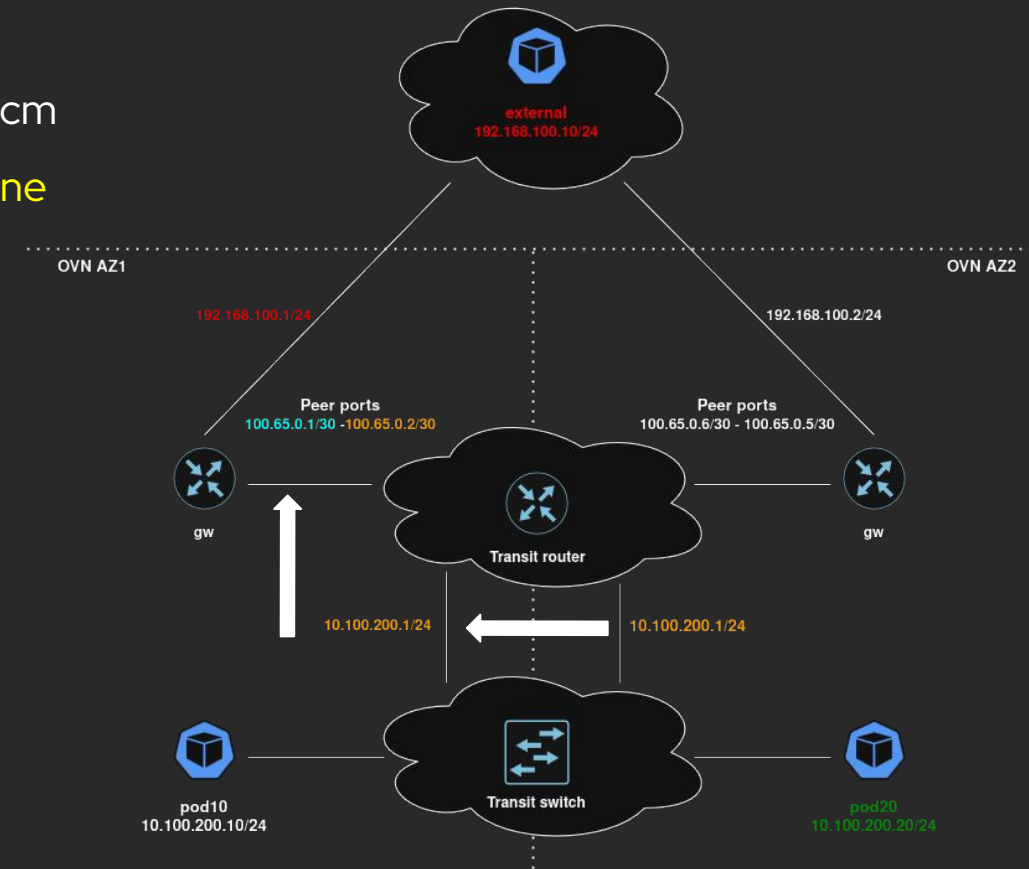
* Logical Router Port: tr-ts networks ['10.100.200.1/24', '10:200::1/64']

* Logical datapath: "ts"

* Logical flow: table=31 (ls_in_l2_lookup), priority=50, match=(eth.dst == 00:00:00:00:10:20), actions=(output = "pod20"; output;)

* Logical Switch Port: pod20 type remote (addresses ['10.100.200.20'])

* Port Binding: logical_port "pod20", chassis-name "ovn-chassis-2"

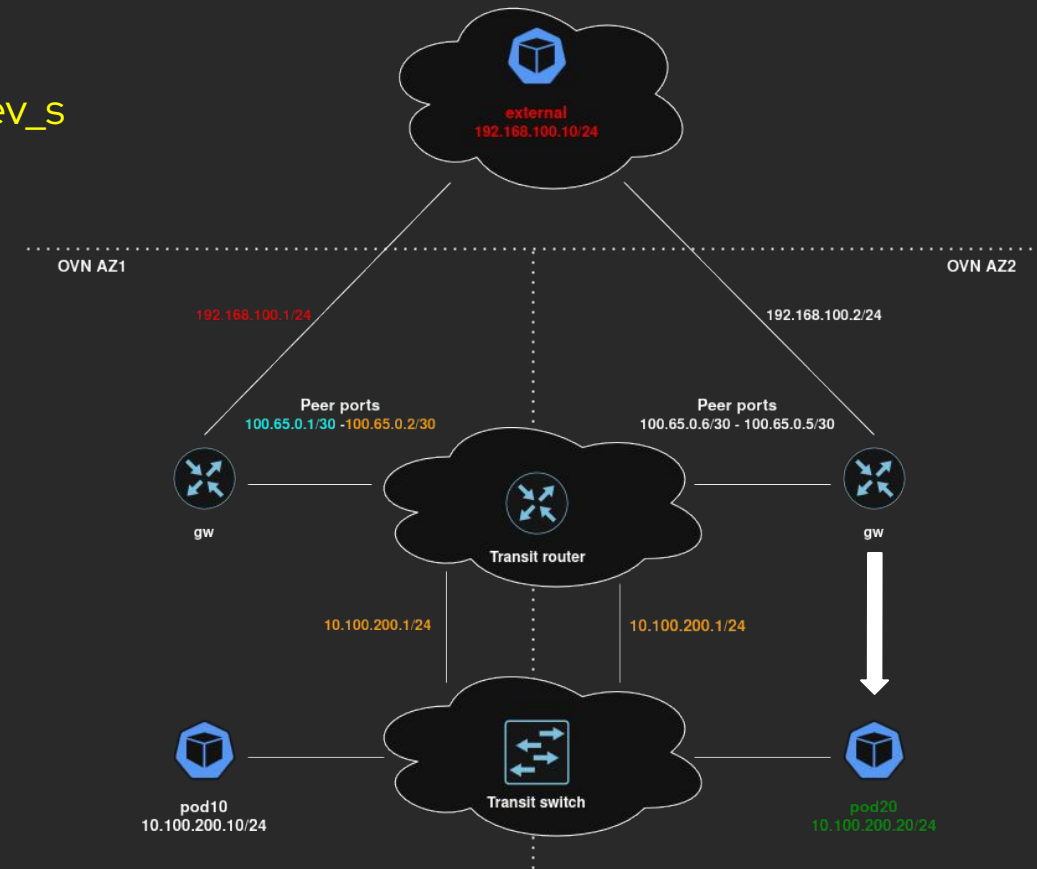


recirc_id(0),tunnel(tun_id=0xa,src=170.168.0.7,dst=170.168.0.8),in_port(genev_s
ys_6081)

actions:pod20-p

* Logical datapath: "ts"

* Port Binding: logical_port "pod20", chassis-name "ovn-chassis-2"



00:00:00:00:10:00 > 00:00:00:00:10:20, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 62, id 26895, offset 0, flags [DF], proto ICMP (1), length 84)

100.65.0.1 > 10.100.200.20: ICMP echo request, id 2830, seq 1, length 64

00:00:00:00:10:20 > 00:00:00:00:10:00, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 17854, offset 0, flags [none], proto ICMP (1), length 84)

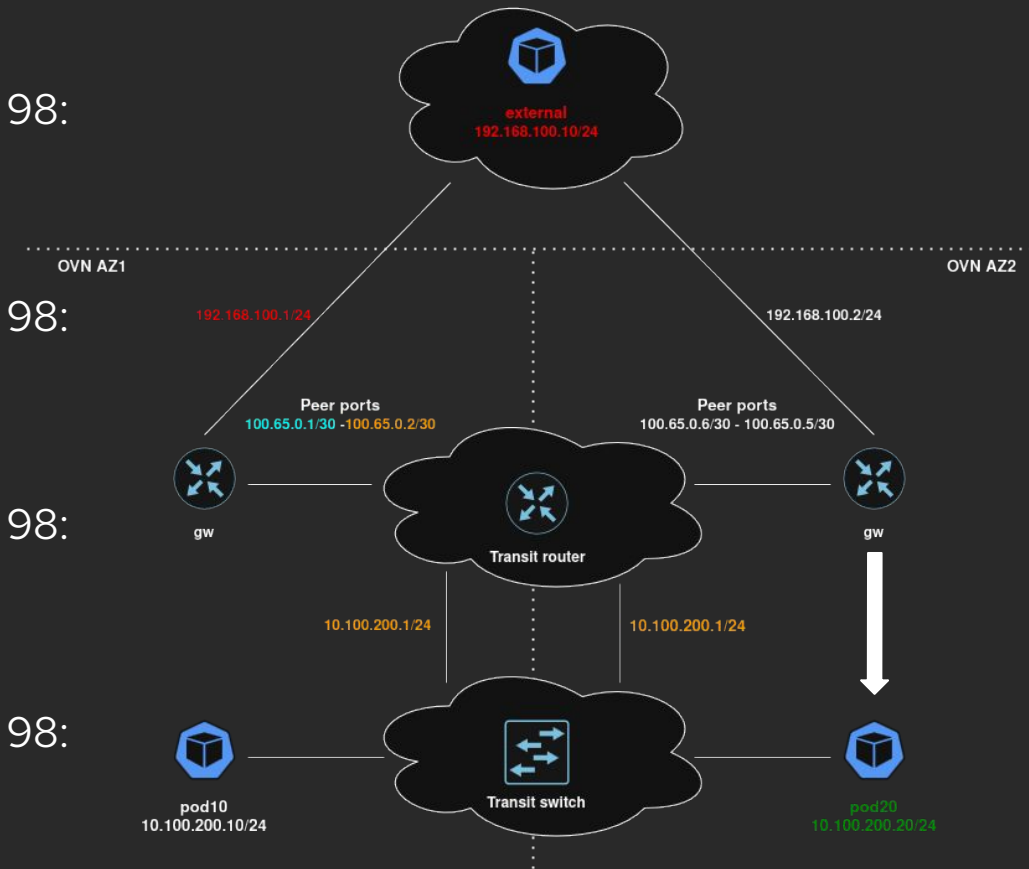
10.100.200.20 > 100.65.0.1: ICMP echo reply, id 2830, seq 1, length 64

00:00:00:00:10:00 > 00:00:00:00:10:20, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 62, id 27397, offset 0, flags [DF], proto ICMP (1), length 84)

100.65.0.1 > 10.100.200.20: ICMP echo request, id 2830, seq 2, length 64

00:00:00:00:10:20 > 00:00:00:00:10:00, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 17854, offset 0, flags [none], proto ICMP (1), length 84)

10.100.200.20 > 100.65.0.1: ICMP echo reply, id 2830, seq 2, length 64



```
recirc_id(0),in_port(pod20-p),eth(src=00:00:00:00:10:20,dst=00:00:00:00:10:00),pv4(src=10.100.200.16/255.255.255.240,dst=100.65.0.1),icmp  
actions:tunnel(tun_id=0x14,src=170.168.0.8,dst=170.168.0.7),genev_sys_6081
```

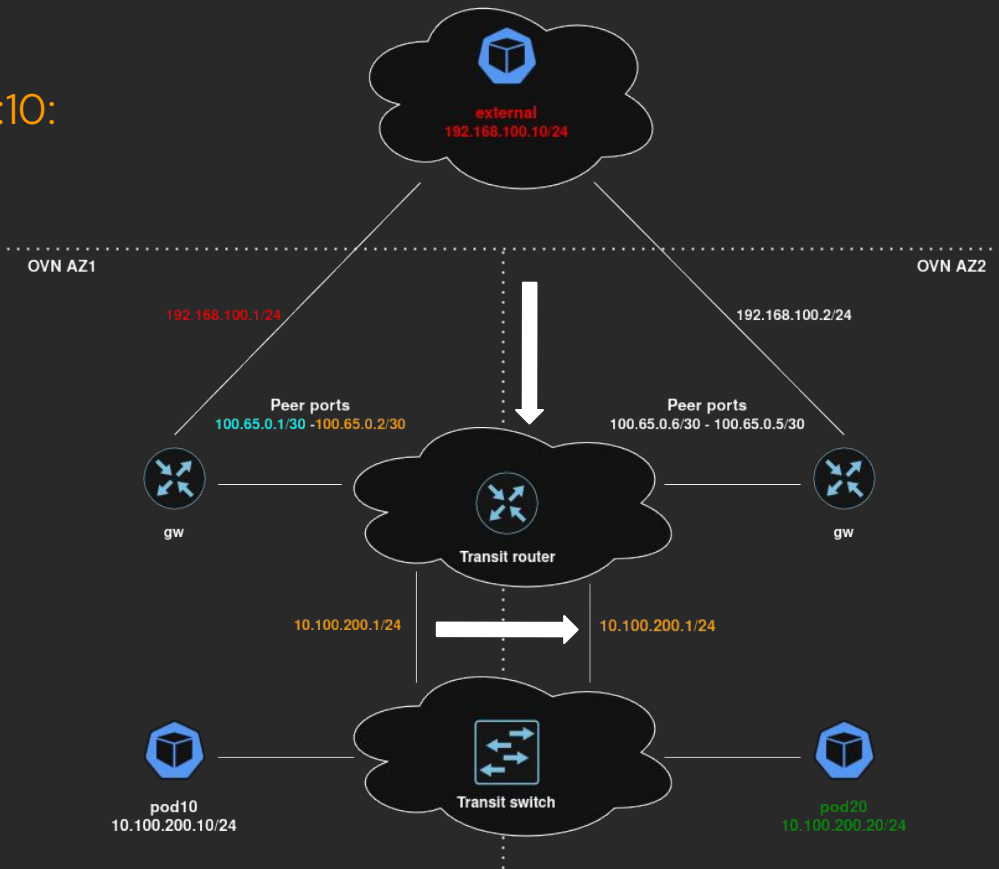
* Logical datapath: "ts"

* Port Binding: logical_port "**pod20**", chassis-name "ovn-chassis-2"

* Logical datapath: "tr"

* Logical flow: table=16 (lr_in_ip_routing), priority=246, match=(ip4.dst == **100.65.0.0/30**), actions=(ireg5 = **100.65.0.2**; outport = "tr-az1");

* Port Binding: logical_port "**tr-az1**", chassis-name "ovn-chassis-1"



```
recirc_id(0),tunnel(tun_id=0x14,src=170.168.0.8,dst=170.168.0.7,in_port(genev_s  
ys_6081),ipv4(src=10.100.200.16/255.255.255.240,dst=100.65.0.1),icmp
```

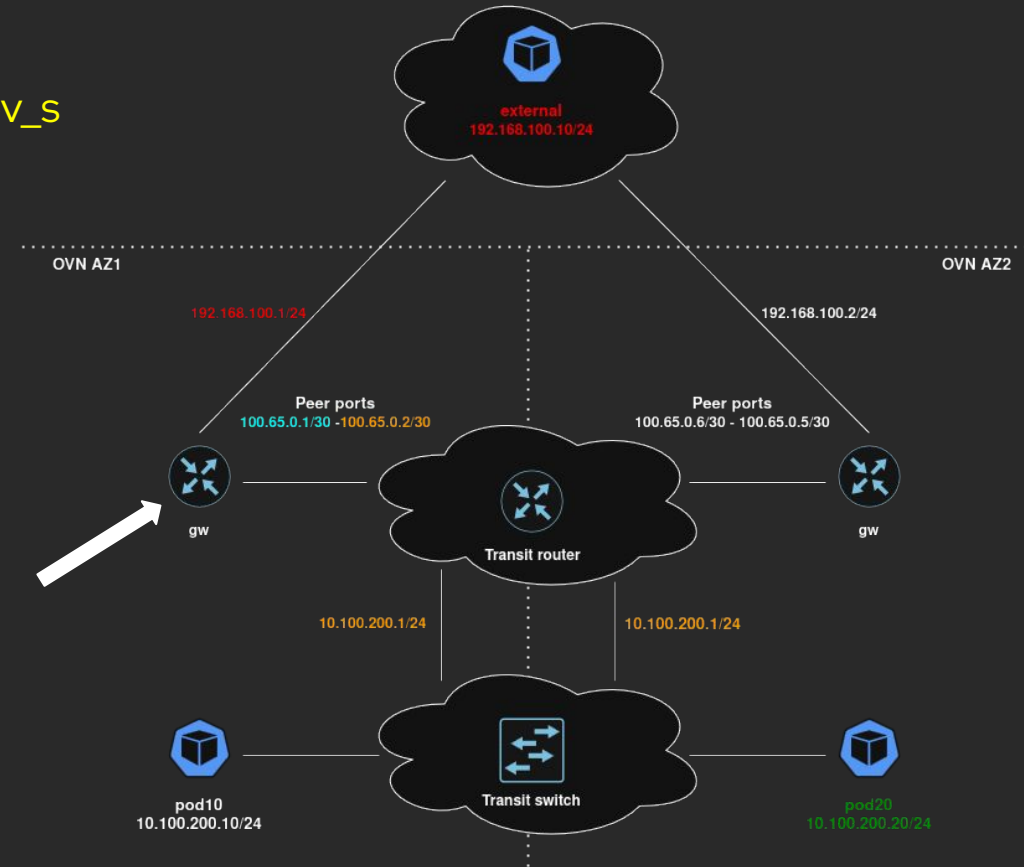
```
actions:ct(zone=1,nat),recirc(0x35)
```

- * Logical datapath: "tr"

- * Port Binding: logical_port "tr-gw"

- * Logical datapath: "gw"

- * Logical flow: table=5 (lr_in_unsnat), priority=90, match=(ip && ip4.dst == 100.65.0.1), actions=(ct_snat;)

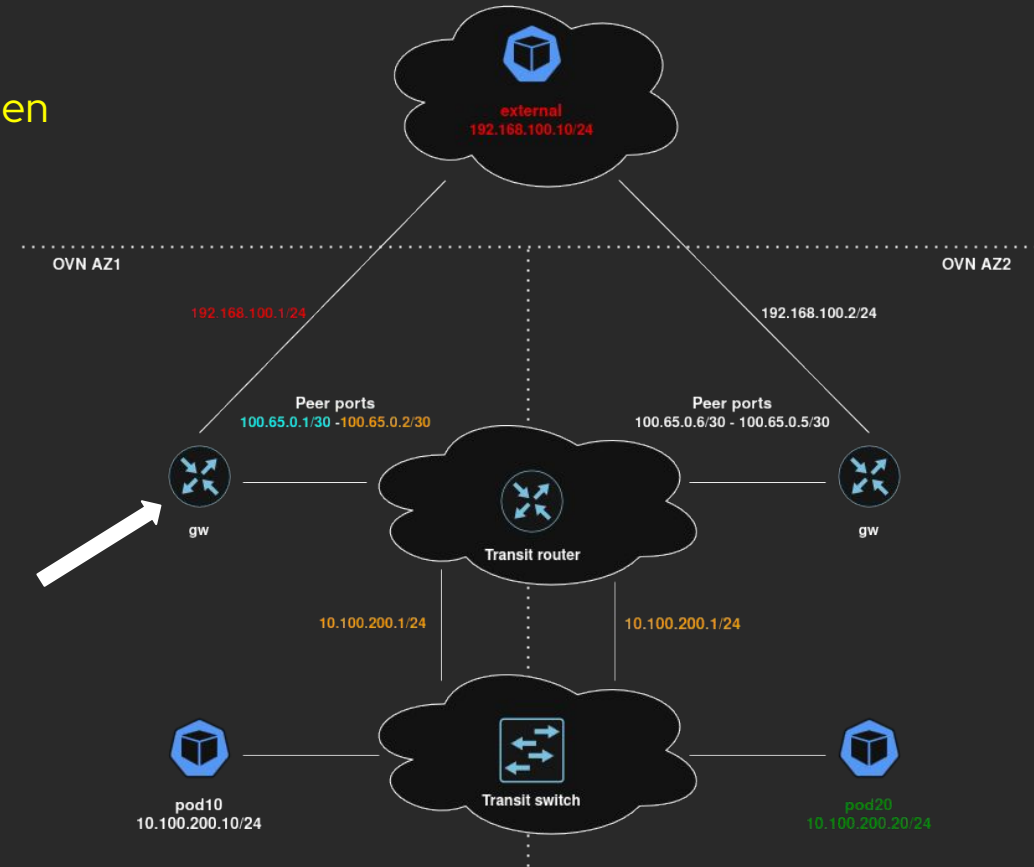


```
recirc_id(0x35),tunnel(tun_id=0x14,src=170.168.0.8,dst=170.168.0.7,in_port(gen  
ev_sys_6081),ipv4(dst=192.168.100.10),icmp
```

```
actions:ct(zone=2,nat),recirc(0x36)
```

* Logical datapath: "gw"

* Logical flow: table=1 (lr_out_undnat), priority=50, match=(ip),
actions=(flags.loopback = 1; ct_dnat;)



```
recirc_id(0x36),tunnel(tun_id=0x14,src=170.168.0.8,dst=170.168.0.7),in_port(ge  
nev_sys_6081)
```

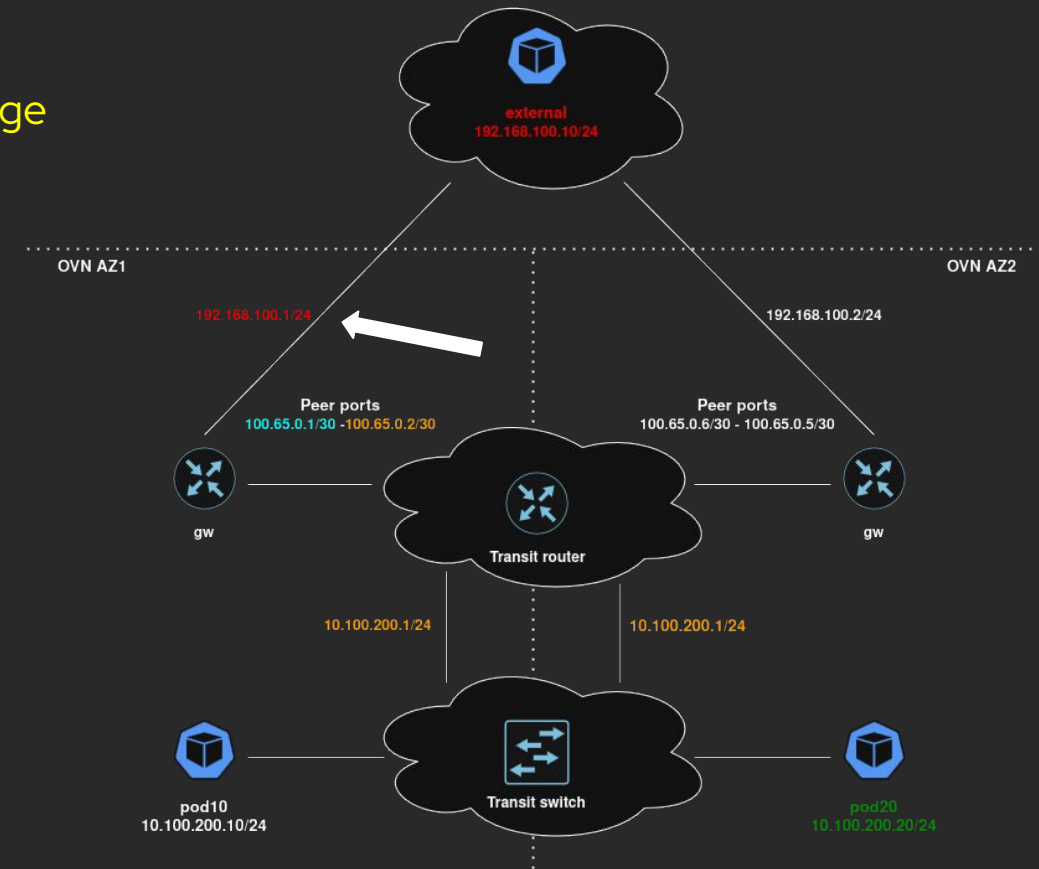
```
actions:ct_clear,external-p
```

- * Logical datapath: "gw"

- * Port Binding: logical_port "gw-public", chassis-name "ovn-chassis-1"

- * Logical datapath: "public"

- * Port Binding: logical_port "external", chassis-name "ovn-chassis-1"

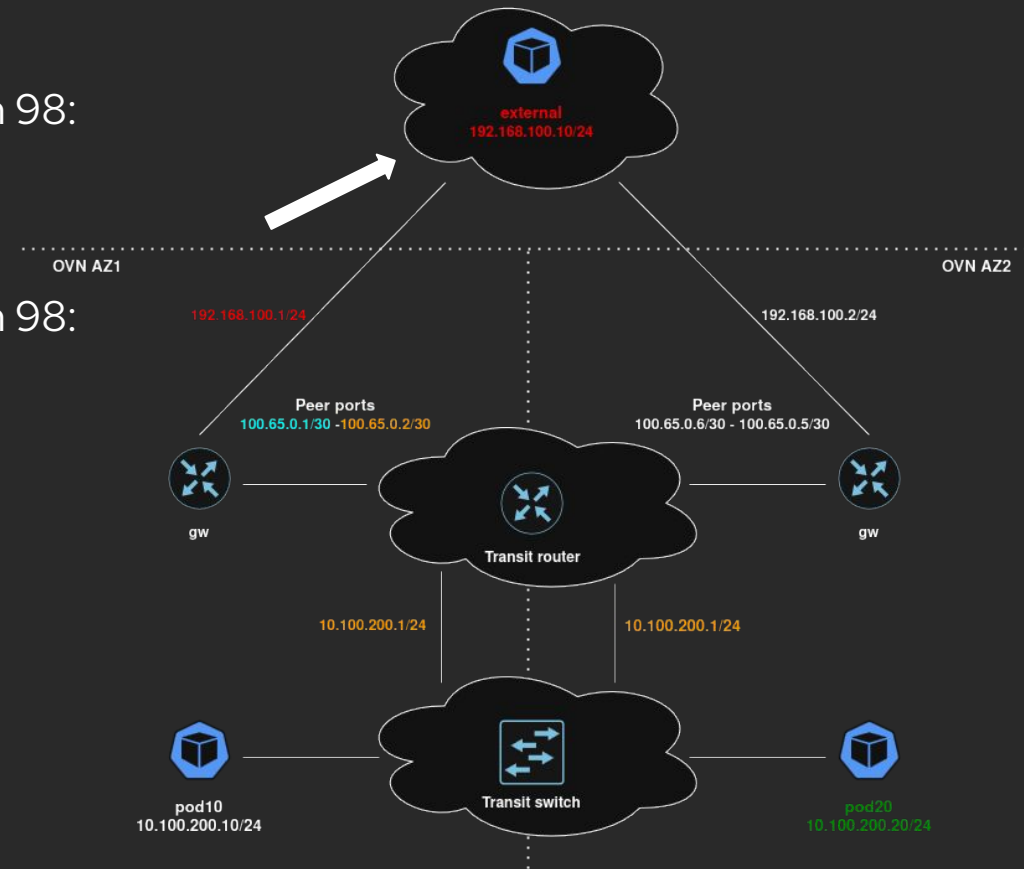


00:00:00:00:20:00 > 00:00:00:00:20:10, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 17854, offset 0, flags [none], proto ICMP (1), length 84)

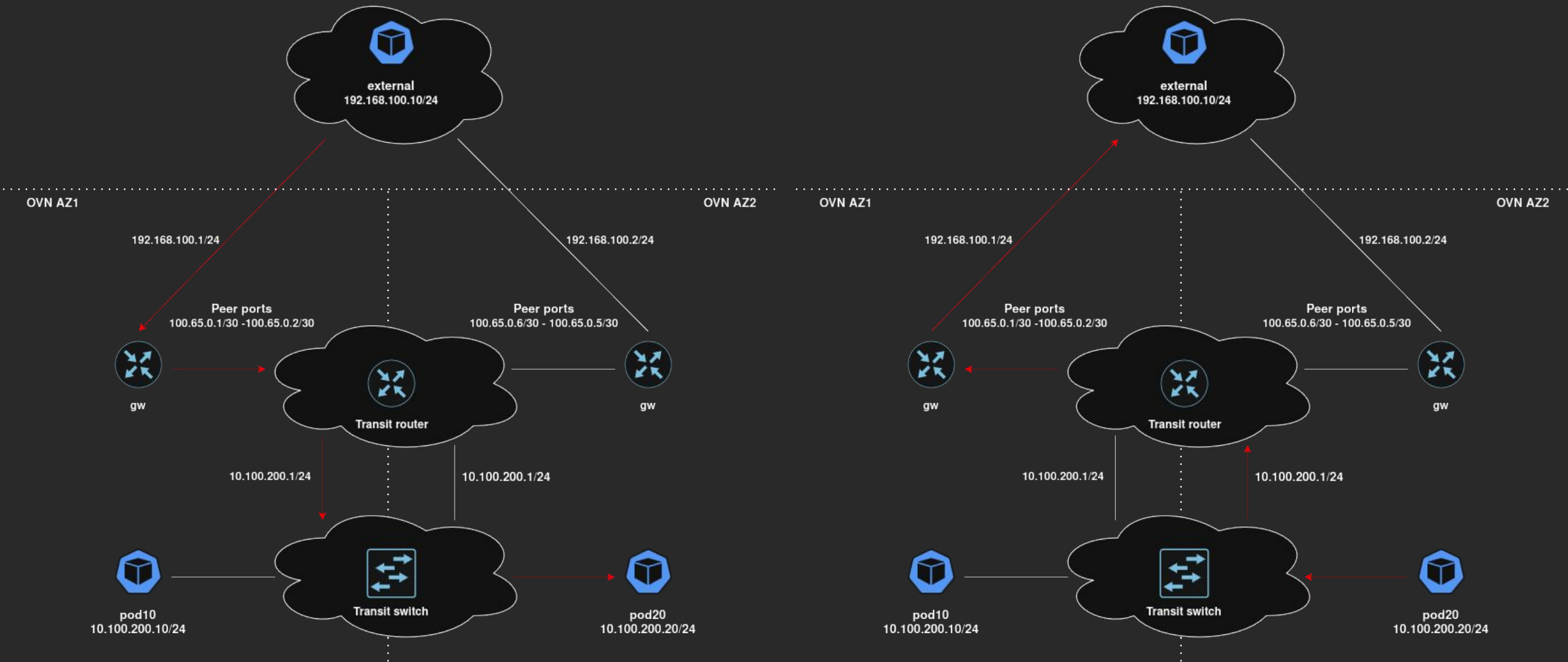
10.100.200.20 > 192.168.100.10: ICMP echo reply, id 2830, seq 1, length 64

00:00:00:00:20:00 > 00:00:00:00:20:10, ethertype IPv4 (0x0800), length 98:
(tos 0x0, ttl 64, id 17854, offset 0, flags [none], proto ICMP (1), length 84)

10.100.200.20 > 192.168.100.10: ICMP echo reply, id 2830, seq 2, length 64



Example topology



ARP flooding

- ▶ The ARP learning for LR happens in ingress pipeline.
- ▶ TR needs to learn ARPs in each AZ individually.
- ▶ ARP response between AZ needs to be flooded back.
- ▶ Specific flows are added to achieve that.



New ovn-ic-nbctl commands

- ▶ tr-add ROUTER
- ▶ trp-add ROUTER PORT MAC [NETWORK]...chassis=CHASSIS
- ▶ tr-del ROUTER
- ▶ trp-del PORT
- ▶ tr-list

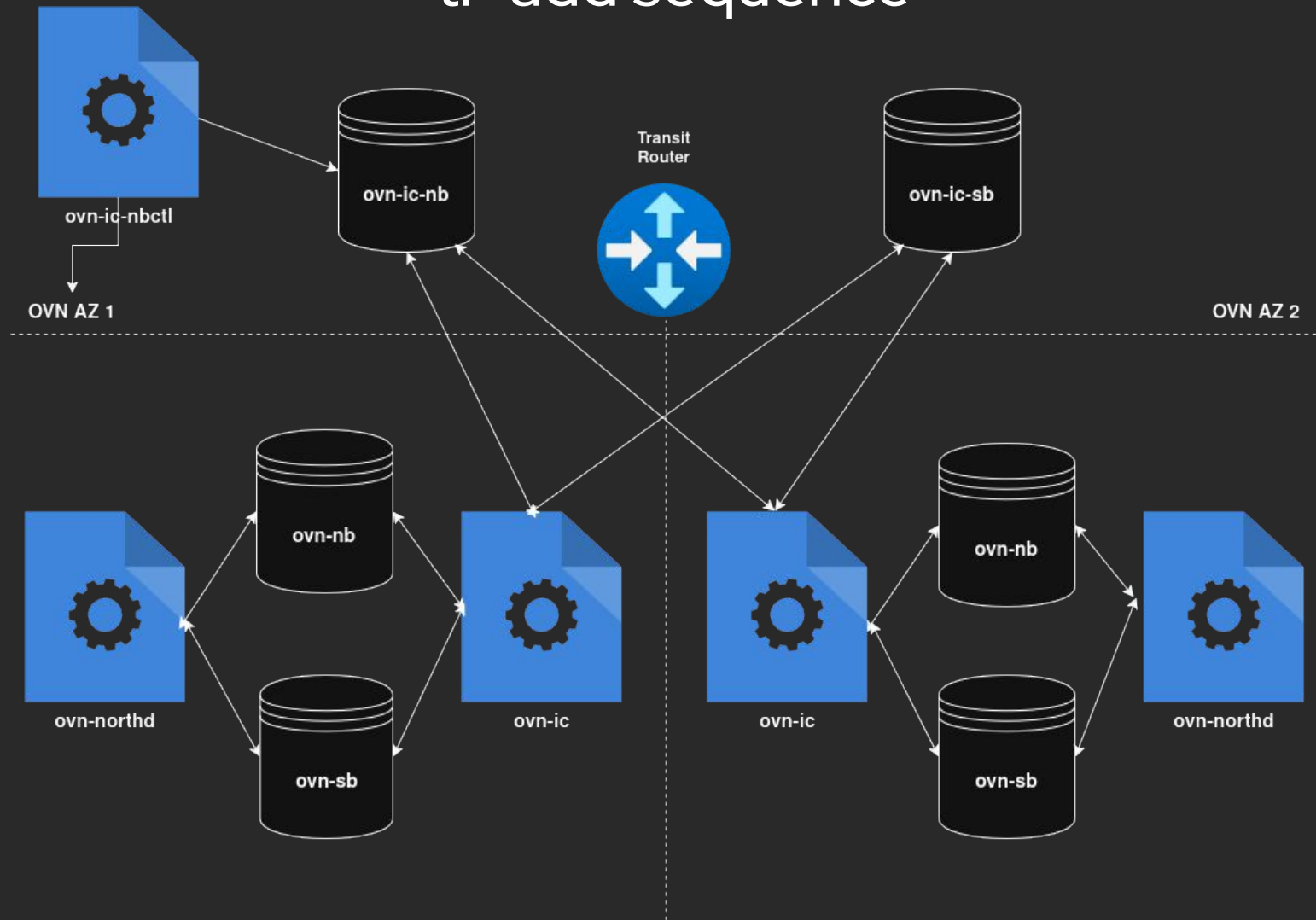


tr-add requirements

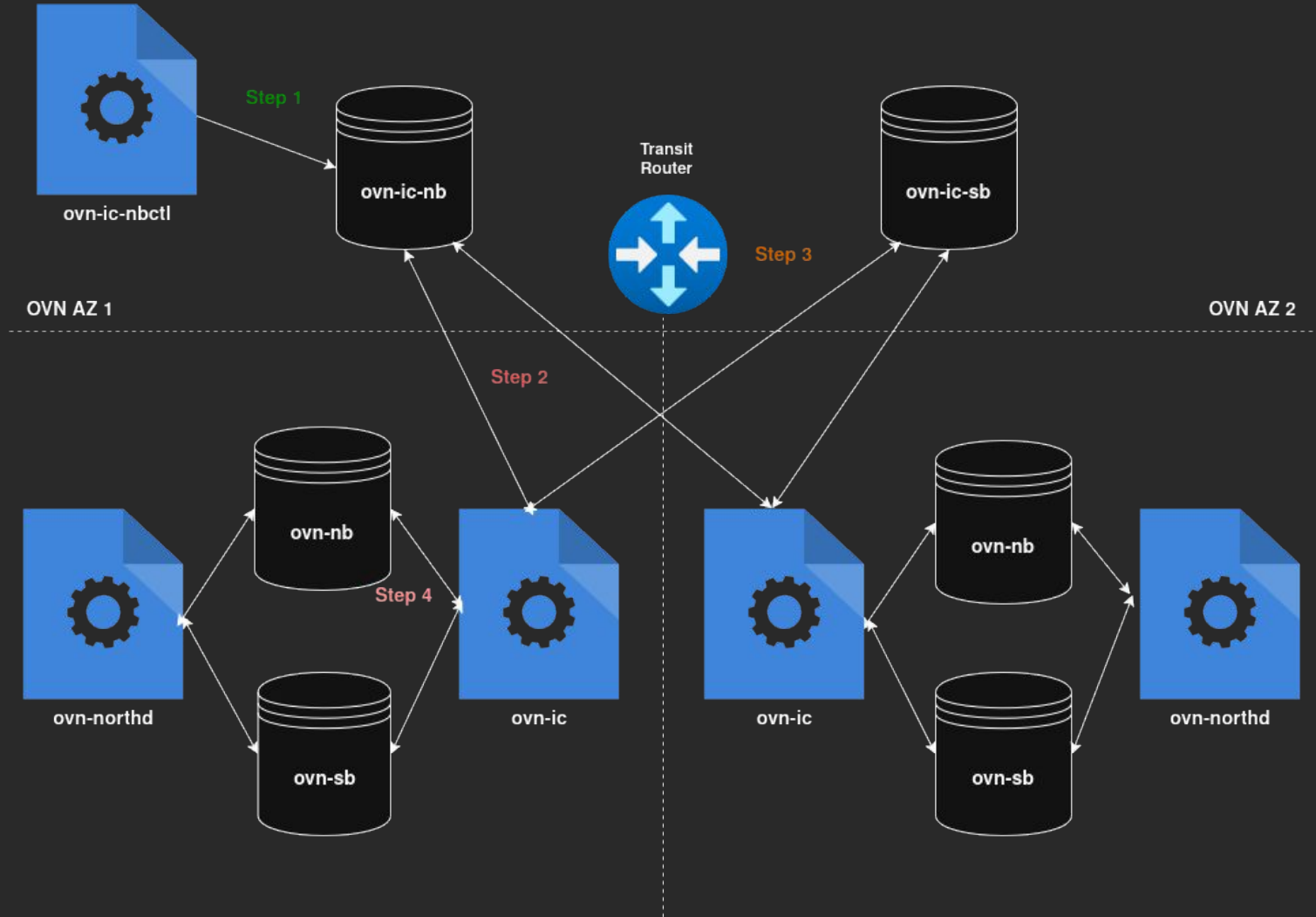
- ▶ Allocate a unique datapath tunnel key.
- ▶ Delegate creation of ICSB datapath binding.
- ▶ Create a router in each Availability Zone.



tr-add sequence



tr-add sequence

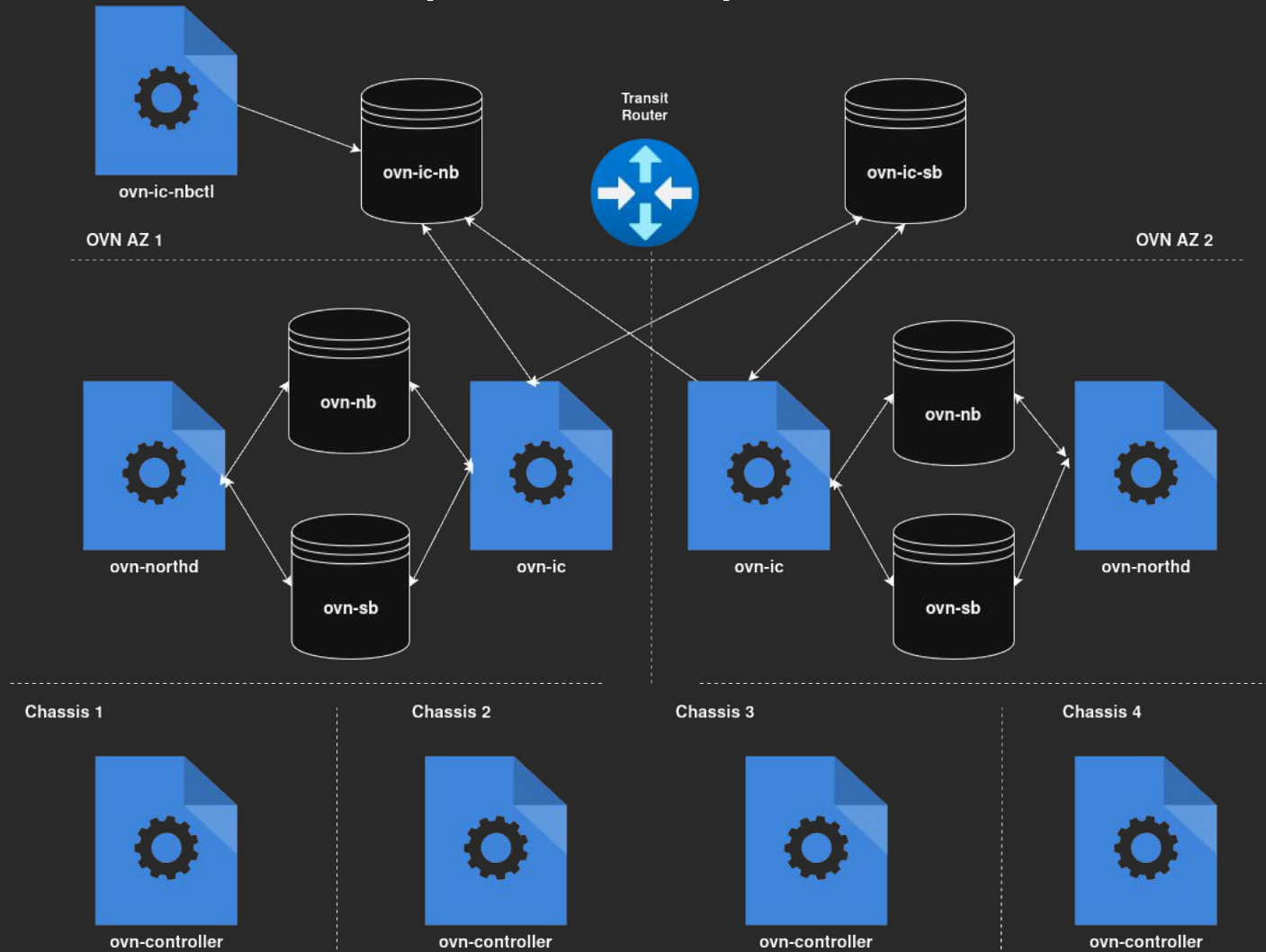


trp-add requirements

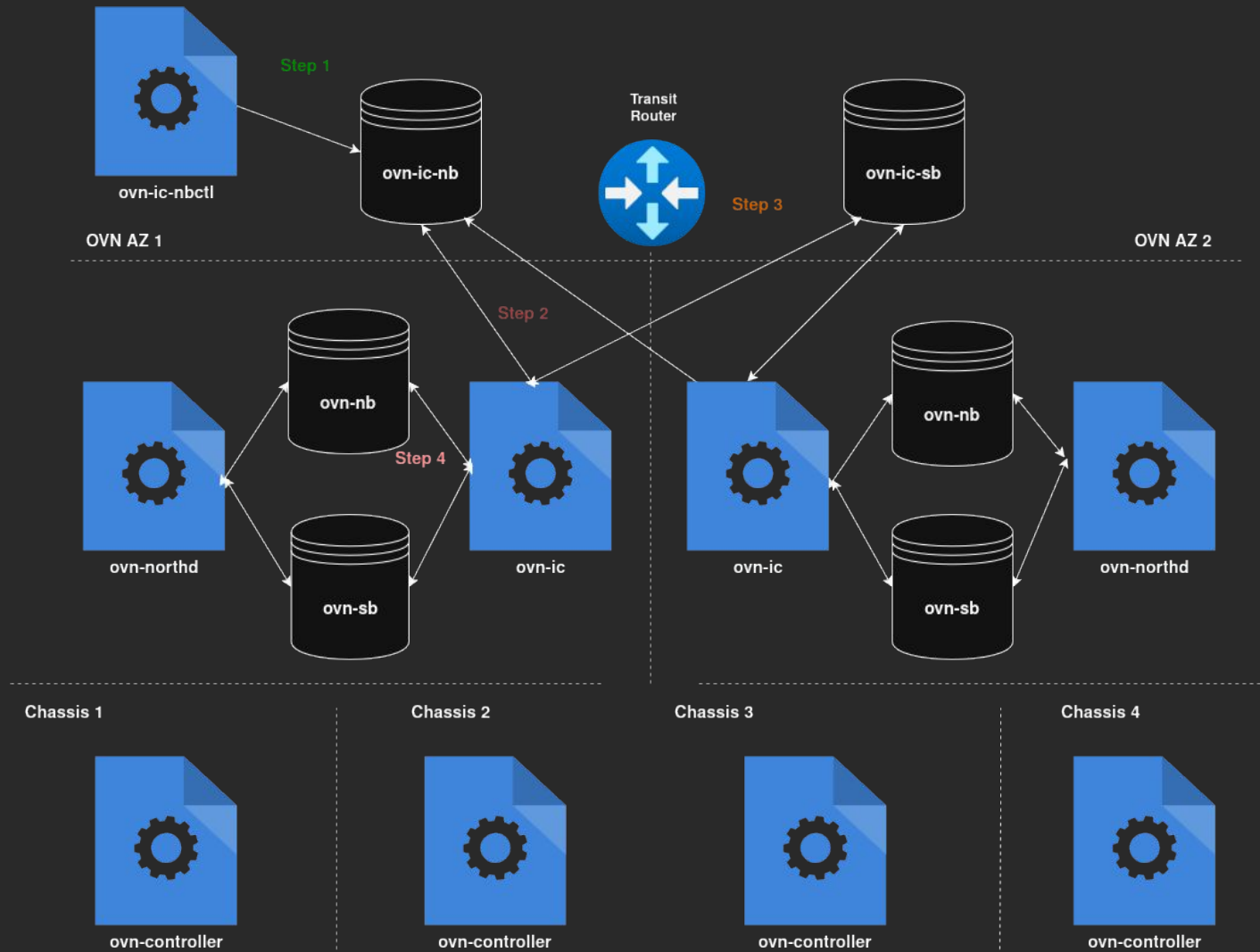
- ▶ Assign resource allocation to appropriate Availability Zone.
- ▶ Create logical router ports per Availability Zone.
- ▶ Sync port binding tunnel keys and configuration.



trp-add sequence



trp-add sequence



tr-del requirements

- ▶ Delete logical routers associated with the transit router.
- ▶ Delete transit router ports and logical router ports.
- ▶ Free global datapath tunnel id.



Open ended questions

- ▶ Definition and synchronization of routes in TR?
- ▶ Definition and synchronization of route rules in TR?
- ▶ “Local” TRP ports definition and synchronization?



Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



linkedin.com/company/red-hat



youtube.com/user/RedHatVideos



facebook.com/redhatinc



twitter.com/RedHat