Building multivendor network labs with Containerlab

Alejandro Guevara
@jaguevarae

Network labs

A right, not a privilege

1

Change management

2

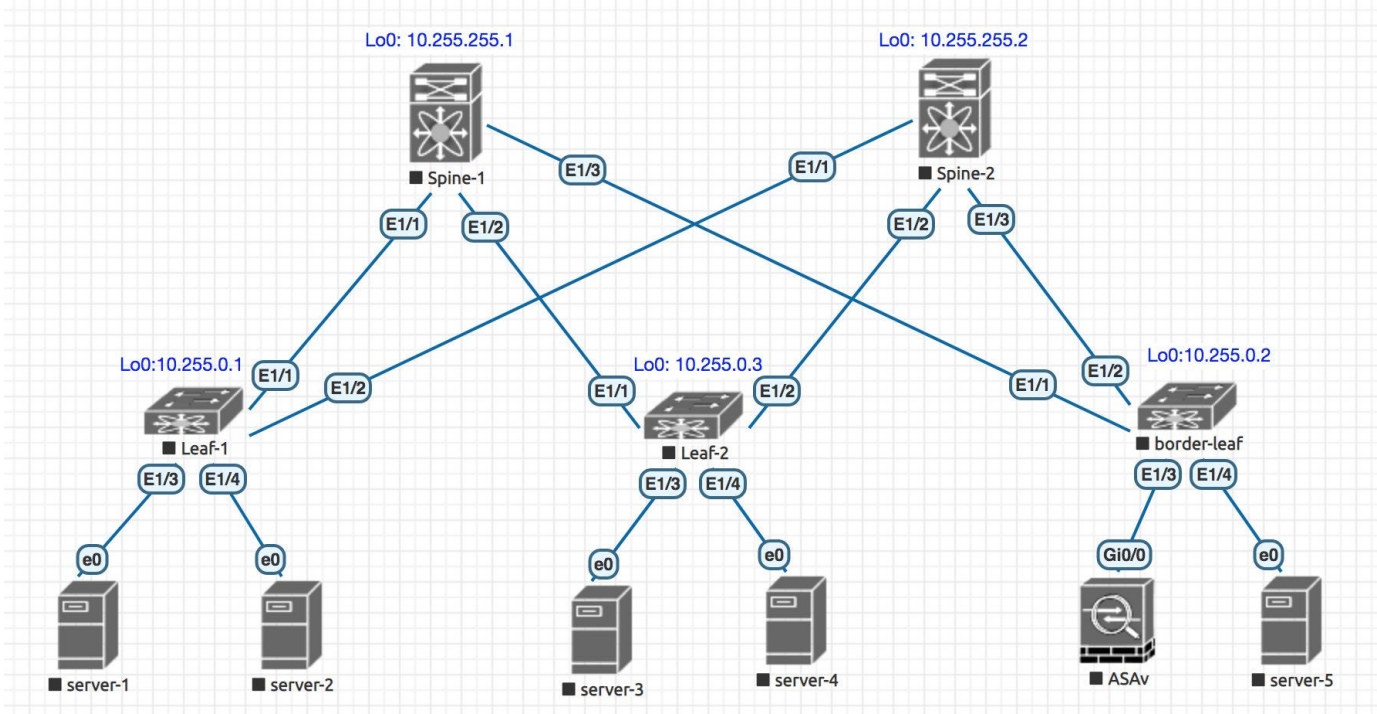
Prototyping and validation

3

Learning

Network labs

How do we typically run labs today?



Pic from https://www.reddit.com/r/networking/comments/g5fb23/eveng_lab_strage_packet_loss/

NOKIA

What else do you need?

Declarative format

What else do you need?

Declarative format

Git friendly

What else do you need?

Declarative format

Portability

Git friendly

What else do you need?

Declarative format

Portability

Git friendly

CI/CD Integration

What else do you need?

Declarative format

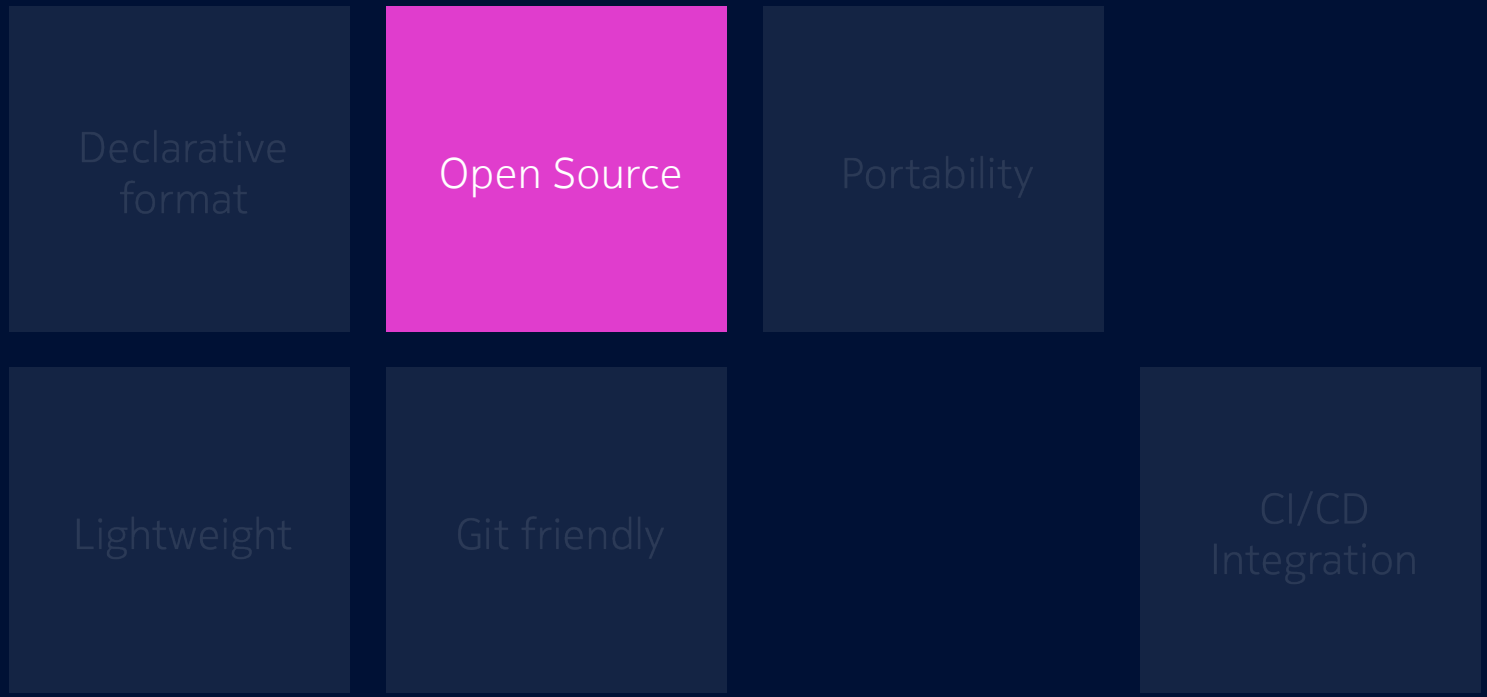
Portability

Lightweight

Git friendly

CI/CD Integration

What else do you need?



What else do you need?

Declarative format

Open Source

Portability

Lightweight

Git friendly

Easy to share

CI/CD Integration

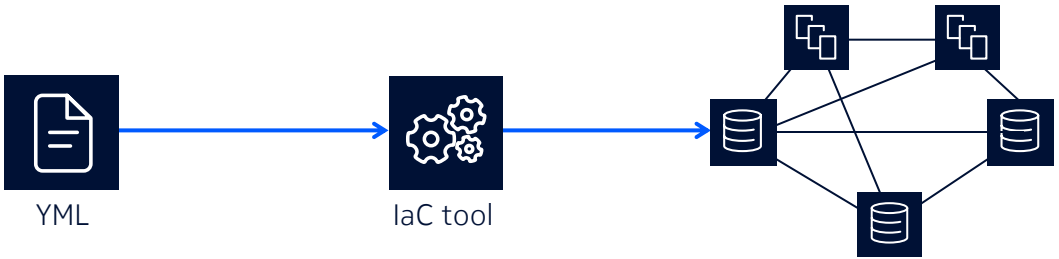
What else do you need?

Declarative format	Open Source	Portability	Container-centric
Lightweight	Git friendly	Easy to share	CI/CD Integration

Containerlab

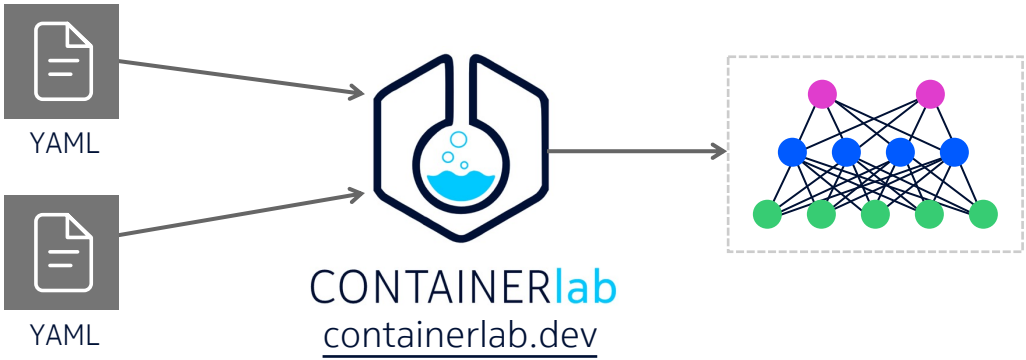
Bringing declarativeness to network labs

IT



Network labs

```
name: mylab

topology:
  nodes:
    [redacted]
    [redacted]

  links:
    - [redacted]
```



Why Containerlab if we have lab emultion tools already?

Network emulation SW

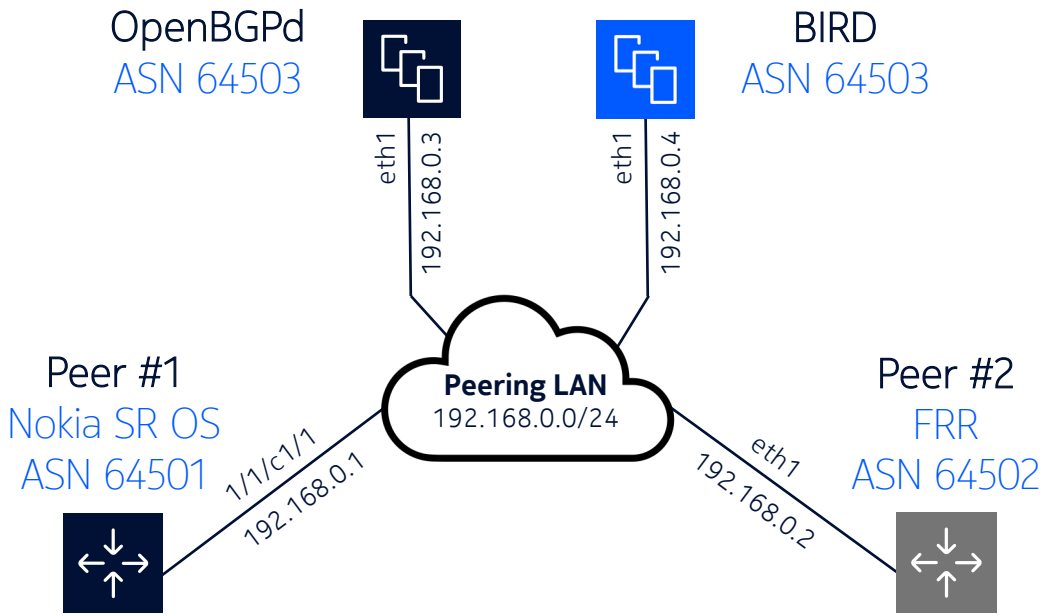
- ⊕ Purpose built and proven
- ⊕ Free versions available
- ⊕ UI
- ⊖ VM-centric weak containers support
- ⊖ Heavy and semi-open
- ⊖ UI

Containerlab

- ⊕ First class support for containerized NOSes
- ⊕ Transparent datapath
- ⊕ Git friendly and better image sharing and handling
- ⊕ Repeatable lab builds and CI friendly
- ⊕ Small footprint, open, free and fast
- ⊖ Fewer Network OSes supported
- ⊖ No UI

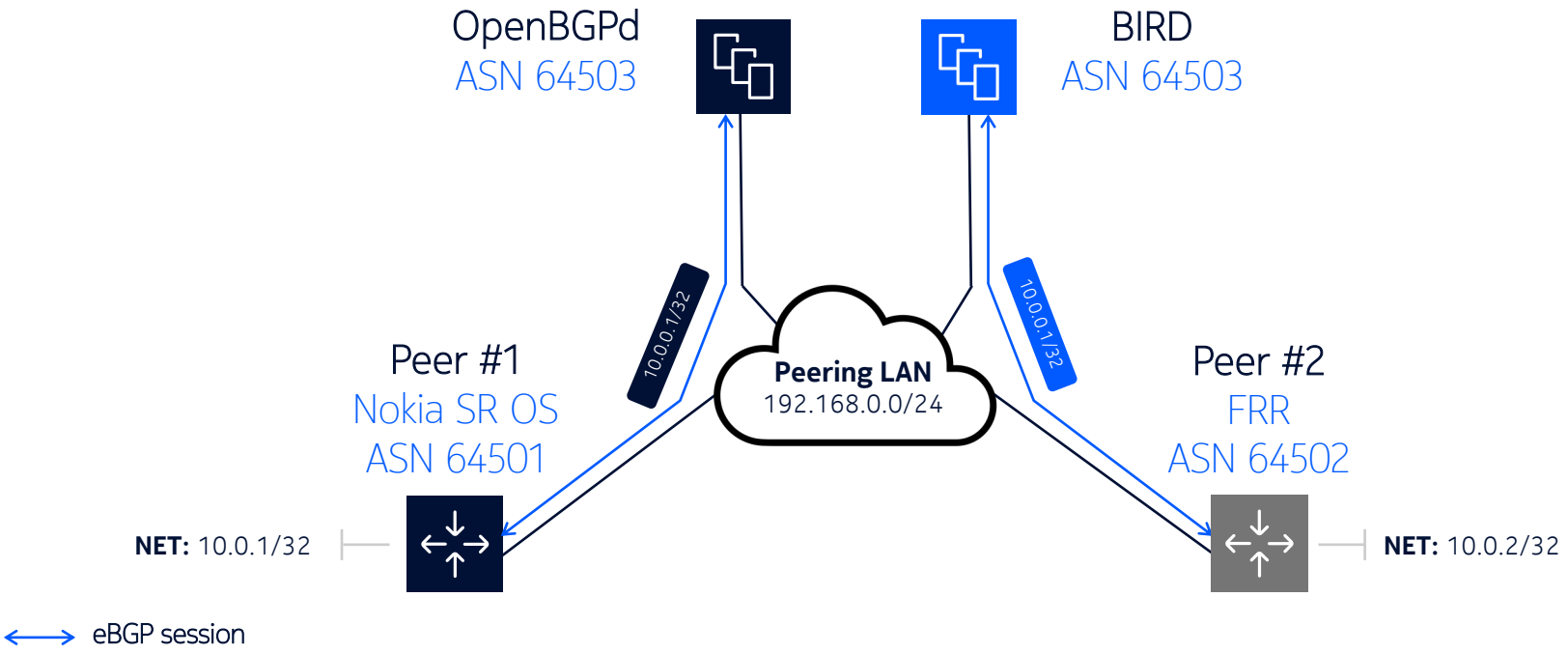
Learn by doing

Basic IXP topology with route servers



Learn by doing

Basic IXP topology with route servers



Installation

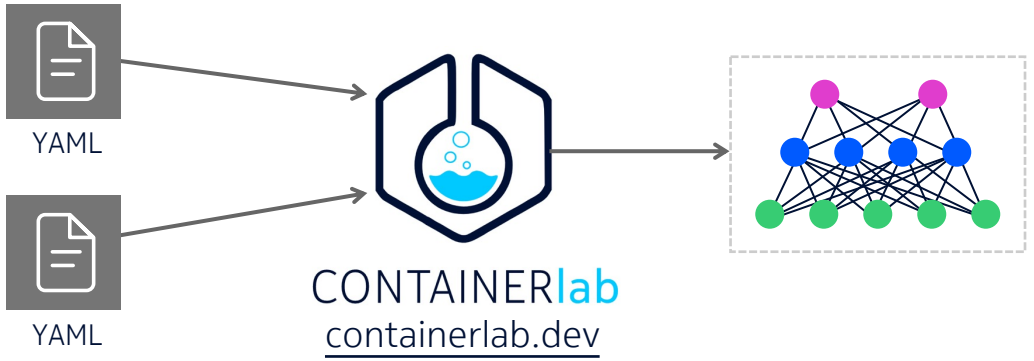
Just a single command

Other installation options:
https://containerlab.dev/install/

Topology file

Declarative way to define a lab

```
name: mylab

topology:
  nodes:
    [redacted] : [redacted]
    [redacted] : [redacted]

  links:
    - [redacted] : [redacted]
```



https://containerlab.dev

Building an IXP lab

Adding Nokia SR OS node

Topology definition

ixp.clab.yml

```yaml
name: ixp

topology:
  nodes:
    peer1:
      kind: vr-nokia_sros
      image: sros:23.3.R1
      license: license.key
```

Logical view

Building an IXP lab

Adding FRR node

Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1: {…}

    peer2:
      kind: linux
      image: quay.io/frrouting/frr:8.4.1
```

Logical view

Building an IXP lab

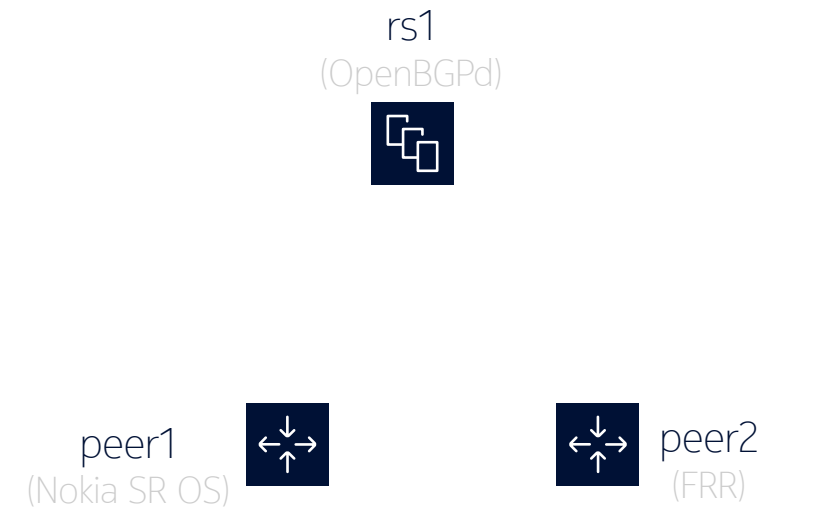
Adding route servers

Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1: {…}
    peer2: {…}

    rs1:
      kind: linux
      image: quay.io/openbgpd/openbgpd:7.9
```

Logical view



Building an IXP lab

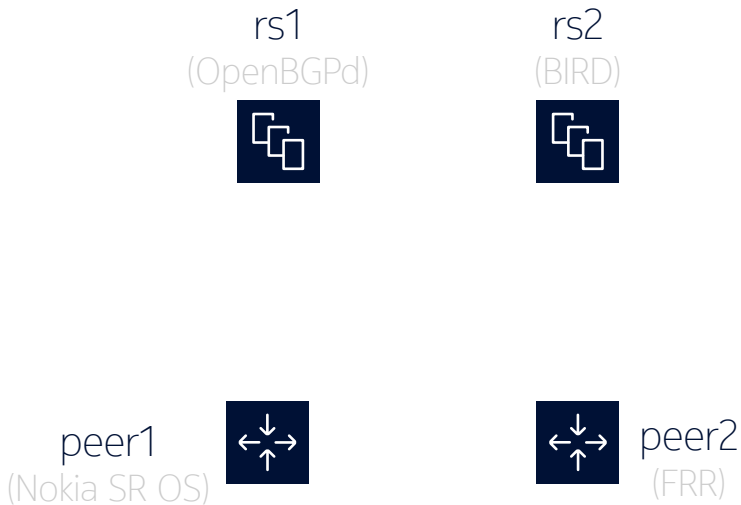
Adding route servers

Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1: {…}
    peer2: {…}
    rs1:   {…}

    rs2:
      kind: linux
      image: ghcr.io/srl-labs/bird:2.0.11
```

Logical view



Building an IXP lab

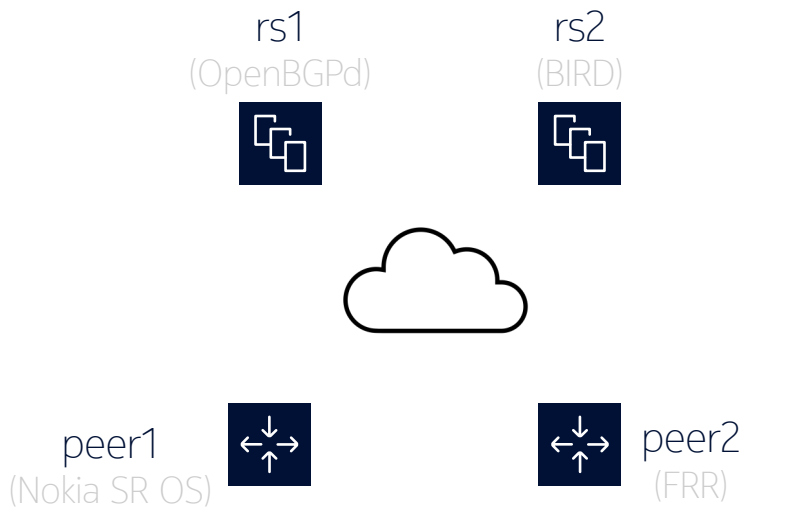
Adding peering LAN

Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1:  {…}
    peer2:  {…}
    rs1:    {…}
    rs2:    {…}

    ixp-net:
      kind: bridge
```

Logical view



Building an IXP lab

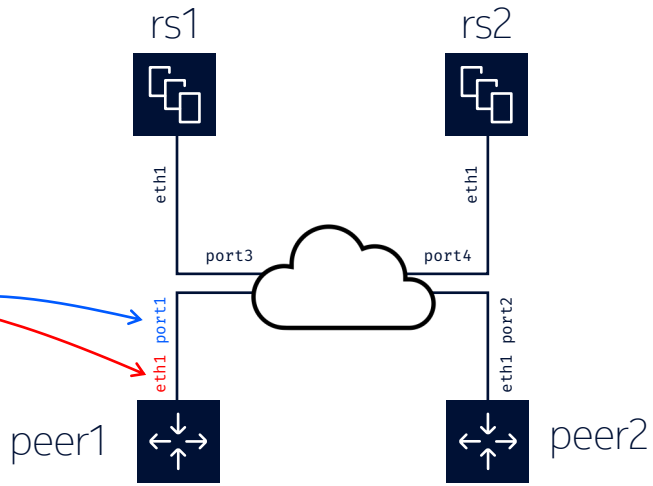
Adding links

Topology definition

ixp.clab.yml

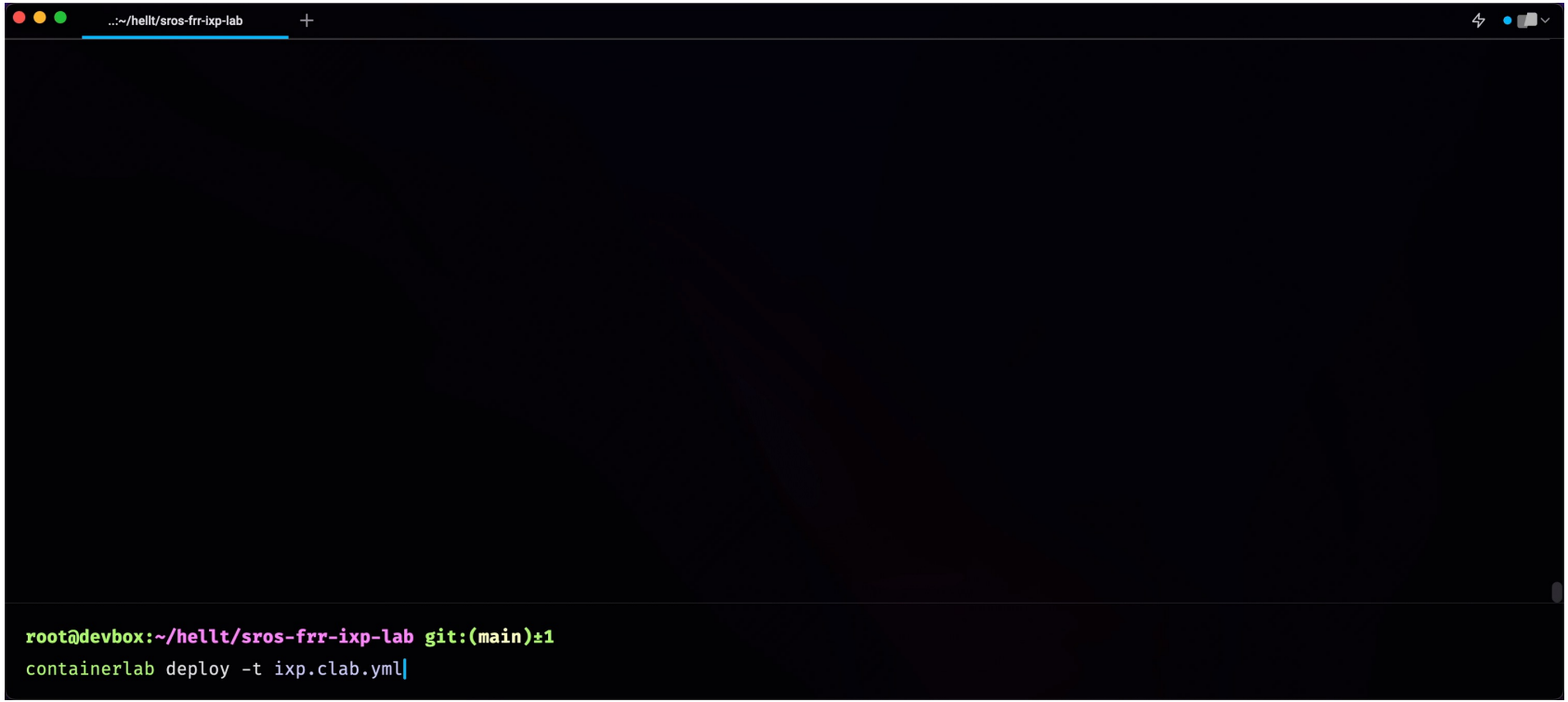
```
name: ixp

topology:
  nodes:
    peer1:    {…}
    peer2:    {…}
    rs1:      {…}
    rs2:      {…}
    ixp-net:  {…}

  links:
    - endpoints: ["peer1:eth1", "ixp-net:port1"]
    - endpoints: ["peer2:eth1", "ixp-net:port2"]
    - endpoints: ["rs1:eth1",   "ixp-net:port3"]
    - endpoints: ["rs2:eth1",   "ixp-net:port4"]
```

Logical view



*Management links are not shown

Deploying the lab



Deploying the lab

```
root@devbox:~/hellt/sros-frr-ixp-lab git:(main) (2.525s)
containerlab deploy -t ixp.clab.yml
INFO[0000] Containerlab v0.39.0 started
INFO[0000] Parsing & checking topology file: ixp.clab.yml
INFO[0000] Creating lab directory: /root/hellt/sros-frr-ixp-lab/clab-ixp
INFO[0000] Creating docker network: Name="clab", IPv4Subnet="172.20.20.0/24", IPv6Subnet="2001:172:20:20::/64", MTU="1450"
INFO[0000] Creating container: "rs2"
INFO[0000] Creating container: "rs1"
INFO[0000] Creating container: "peer2"
INFO[0000] Creating container: "peer1"
INFO[0001] Creating virtual wire: peer2:eth1 <--> ixp-net:port2
INFO[0001] Creating virtual wire: peer1:eth1 <--> ixp-net:port1
INFO[0001] Creating virtual wire: rs2:eth1 <--> ixp-net:port4
INFO[0001] Creating virtual wire: rs1:eth1 <--> ixp-net:port3
INFO[0002] Adding containerlab host entries to /etc/hosts file
+---+----------------+--------------+-------------------------------+--------------+---------+----------------+----------------------+
| # |      Name      | Container ID |             Image             |     Kind     |  State  |  IPv4 Address  |     IPv6 Address     |
+---+----------------+--------------+-------------------------------+--------------+---------+----------------+----------------------+
| 1 | clab-ixp-peer1 | 94f22546922e | sros:23.3.R1                  | vr-nokia_sros | running | 172.20.20.3/24 | 2001:172:20:20::3/64 |
| 2 | clab-ixp-peer2 | 8ba9c9bdbfce | quay.io/frrouting/frr:8.4.1   | linux         | running | 172.20.20.2/24 | 2001:172:20:20::2/64 |
| 3 | clab-ixp-rs1   | 0ac2e6518043 | quay.io/openbgpd/openbgpd:7.9 | linux         | running | 172.20.20.5/24 | 2001:172:20:20::5/64 |
| 4 | clab-ixp-rs2   | 37d7f3507b8b | ghcr.io/srl-labs/bird:2.0.11  | linux         | running | 172.20.20.4/24 | 2001:172:20:20::4/64 |
+---+----------------+--------------+-------------------------------+--------------+---------+----------------+----------------------+

root@devbox:~/hellt/sros-frr-ixp-lab git:(main)±1
```

Containerlab

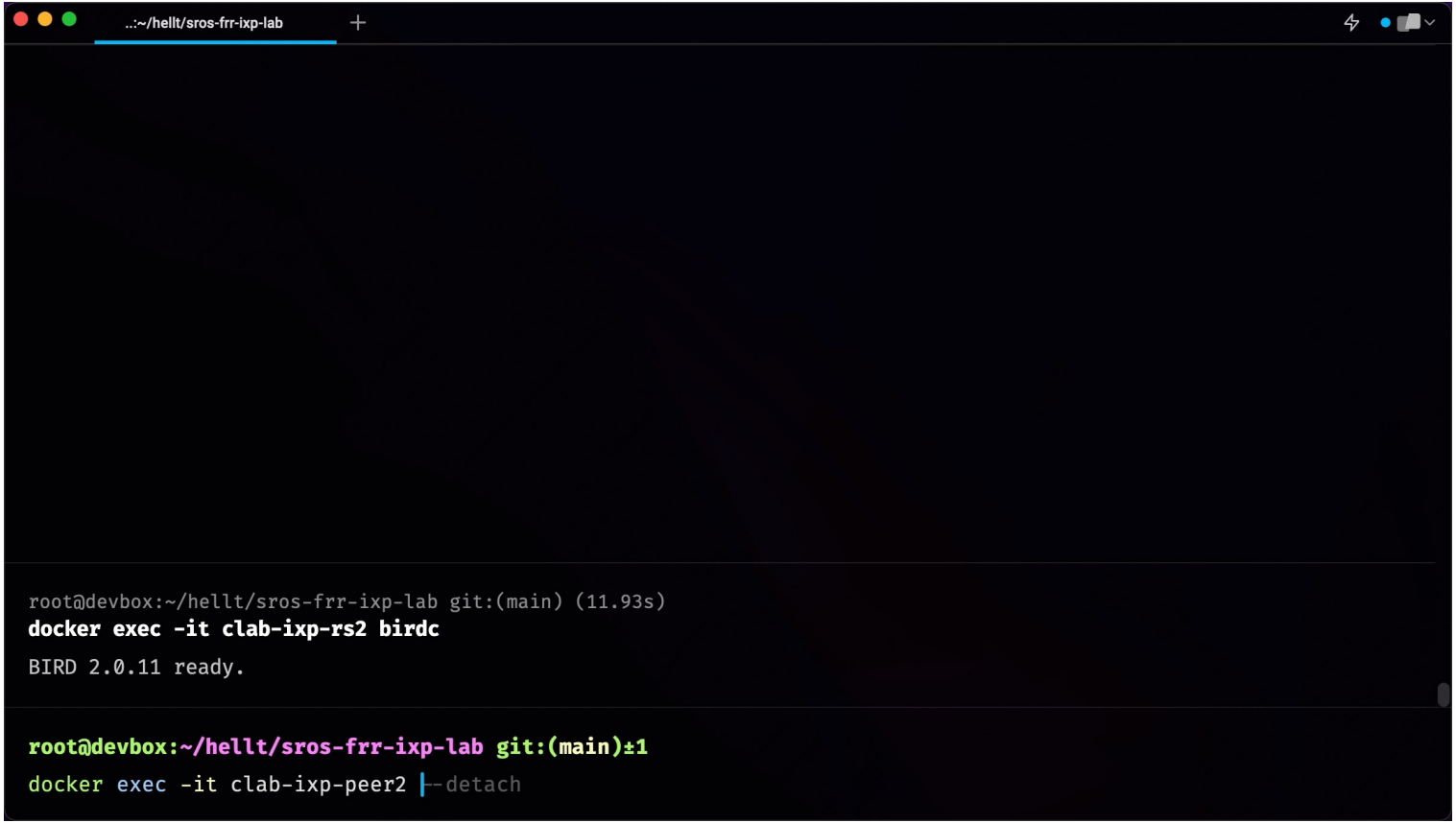
Connecting to the nodes

SSH

```
ssh admin@clab-ixp-peer1

admin@clab-ixp-peer1's password:

[/]
A:admin@peer1#
```

Docker exec

```
docker exec -it clab-ixp-rs2 birdc

BIRD 2.0.11 ready.
bird>
```



Building an IXP lab

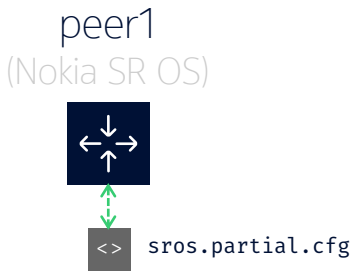
Adding startup configurations

Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1:
      kind: vr-nokia_sros
      image: sros:23.3.R1
      license: license.key
      startup-config: sros.partial.cfg
```

Logical view



Building an IXP lab

Adding startup configurations

Topology definition

ixp.clab.yml

```yaml
name: ixp

topology:
  nodes:
    peer1: {…}

    peer2:
      binds:
        - frr.conf:/etc/frr/frr.conf
        - daemons:/etc/frr/daemons
```

Logical view

Building an IXP lab

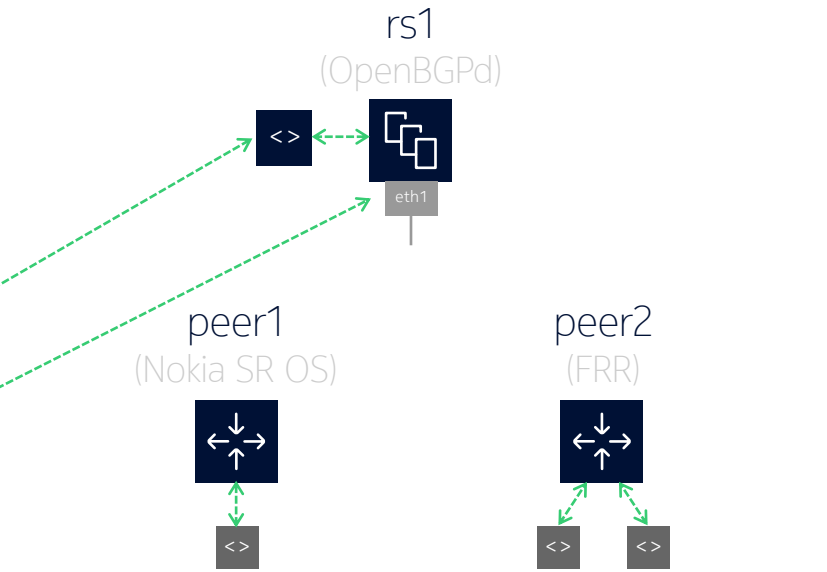
Adding startup configurations

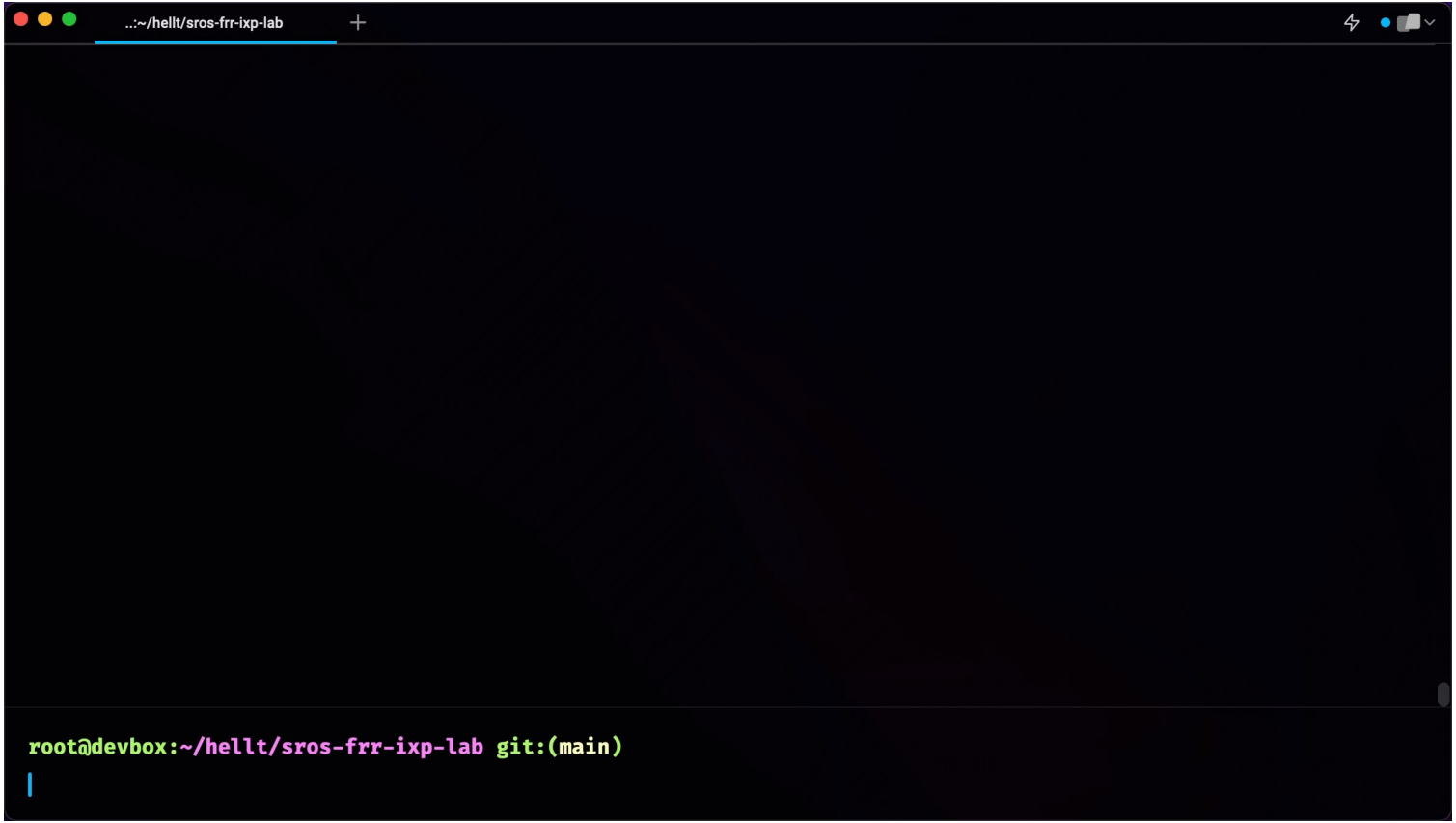
Topology definition

ixp.clab.yml

```
name: ixp

topology:
  nodes:
    peer1: {…}
    peer2: {…}

    rs1:
      binds:
        - openbgpd.conf:/etc/bgpd/bgpd.conf
      exec:
        - ip address add dev eth1 192.168.0.3/24
```

Logical view

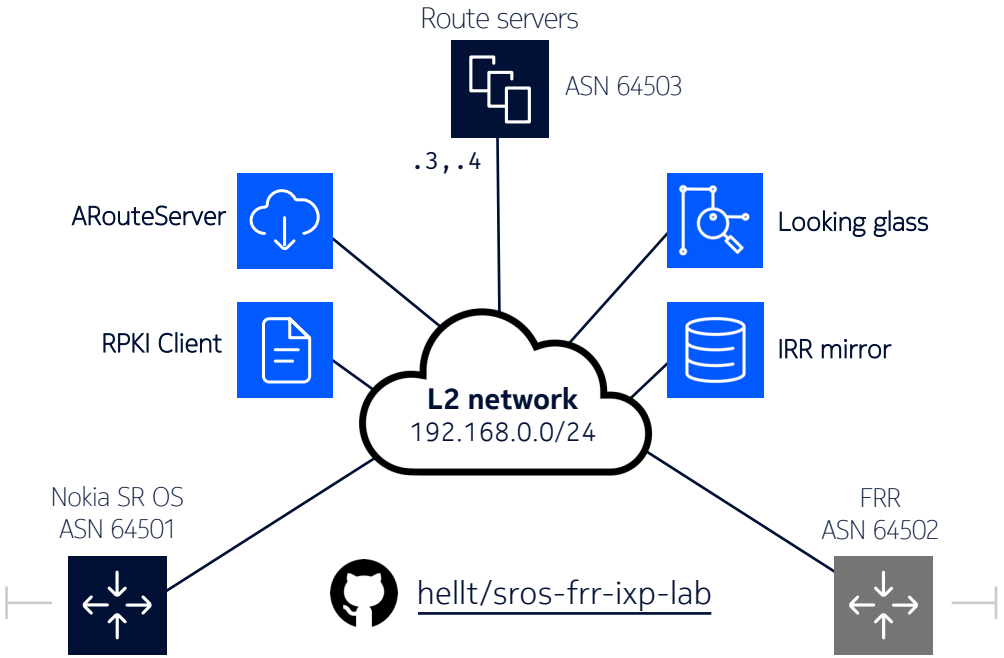




What next?

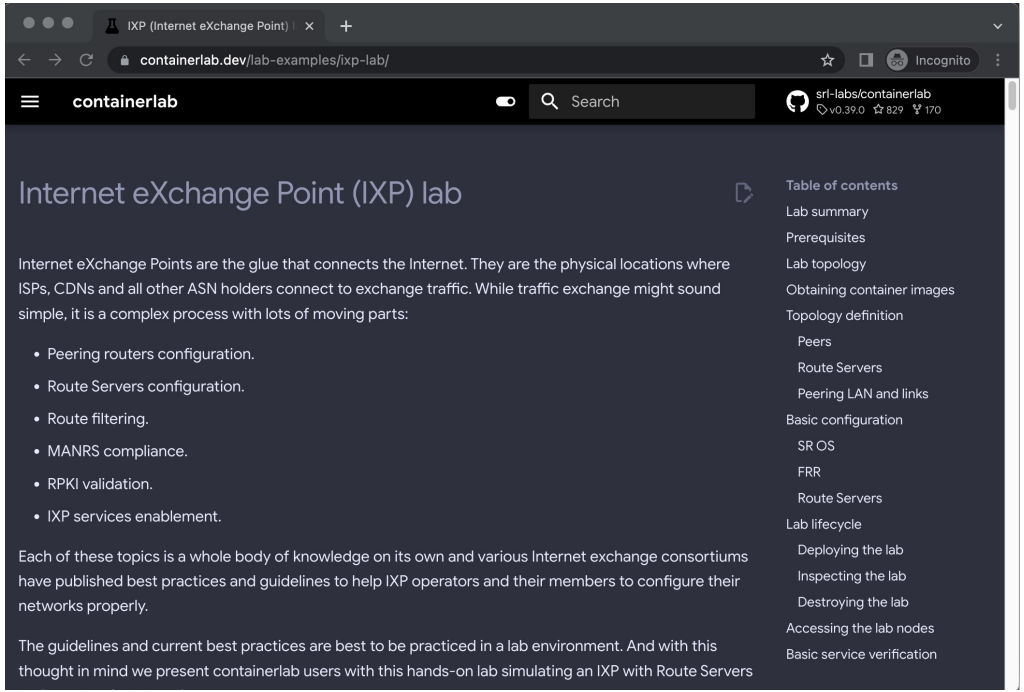
All IXP use cases

- ACL filtering
- MAC filtering
- BUM filtering
- RPKI validation
- MANRS conformance
- Route servers config (ARouteServer)
- Looking glass integration
- Vendor try-out



Lab

A to Z explanation



IXP-Lab

Supported systems

srl
vr-sros

crpd
vr-vmx
vr-vqfx

ceos
vr-veos

vr-xrv9k
vr-csr
vr-n9kv

sonic-vs
frr

cvx

vr-pan

vr-ftosv

keysight_ixia-c

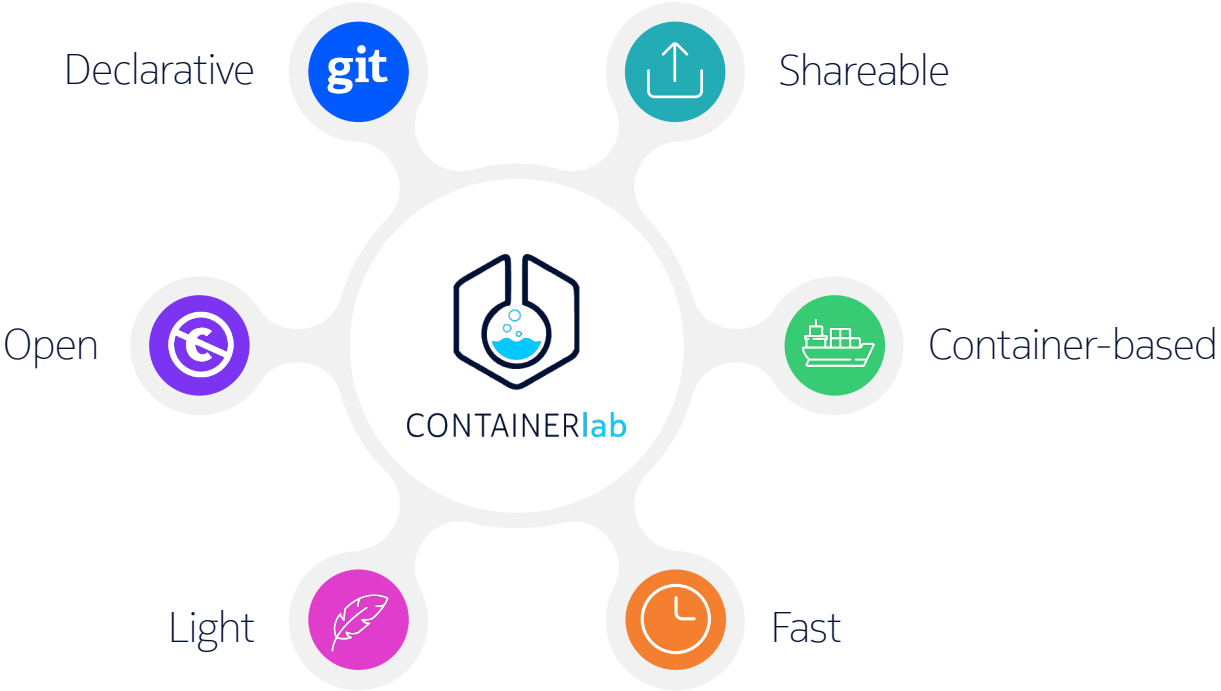
vr-ros

ipinfusion_ocnos

checkpoint_cloudgard

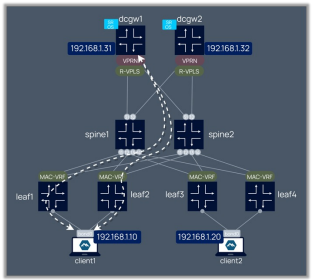
Containerlab

A different look at labs



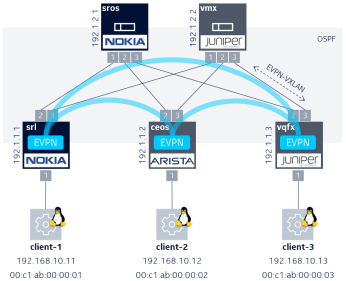
Labs We've Built For The Community

Nokia EVPN Interop



srl-labs/nokia-evpn-lab

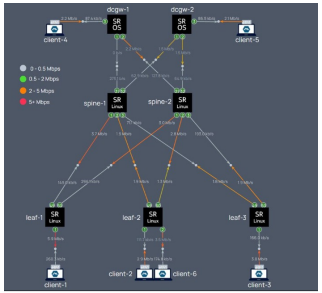
Multivendor EVPN



srl-labs/multivendor-evpn-lab

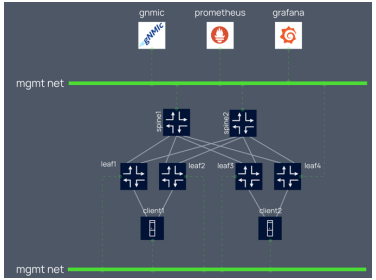
CONTAINERlab

SR Linux & SROS Telemetry



srl-labs/srl-sros-telemetry-lab

SR Linux Oper-Group



srl-labs/opergroup-lab

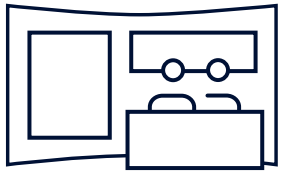
Containerlab

Get in touch

https://containerlab.dev

Discord server

srl-labs/containerlab

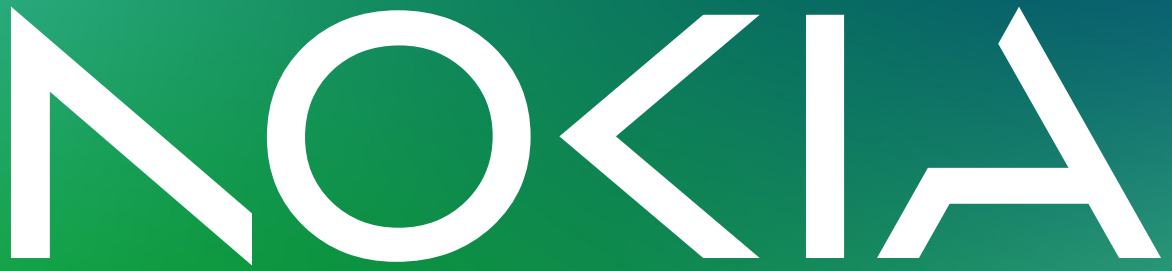


I am here today!

Invitation

Webinar

https://www.lacnic.net/7007/1/lacnic/


NOKIA