

2) Instalar e configurar a plataforma de roteamento

Instalação FRRouting (fork do Quagga)

Versão Debian/Ubuntu: <https://deb.frrouting.org/>

Adicionar repositório, instalar dependências, habilitar daemons necessários (zebra, ospfd, ripd):

#adicionar chave GPG

```
curl -s https://deb.frrouting.org/frr/keys.gpg | sudo tee /usr/share/keyrings/frrouting.gpg > /dev/null
```

#adicionar repositório e verificar

```
echo "deb [signed-by=/usr/share/keyrings/frrouting.gpg] https://deb.frrouting.org/frr jammy frr-stable" | sudo tee /etc/apt/sources.list.d/frr.list
```

```
cat /etc/apt/sources.list.d/frr.list
```

#output deve ser deb [signed-by=/usr/share/keyrings/frrouting.gpg] https://deb.frrouting.org/frr jammy frr-stable , caso o sistema também tenha suporte a 32 bits pode haver uma notificação ao atualizar repositórios porque o FRR só tem 64, nesse caso dá pra resolver adicionando arch=amd64 logo depois de frrouting.gpg

#dependências libjson-c-dev e pkg-config

```
sudo apt install -y libjson-c-dev pkg-config
```

#instalar FRR

```
sudo apt install -y frr frr-pythontools
```

#no Mint, cuidado para a instalação não ser deb https://deb.frrouting.org/frr **virginia** frr-stable,: o

#Mint usa nomes de versões diferentes das do Ubuntu e o repositório FRR espera nomes Ubuntu.

#Como o Mint Virginia é baseado no Ubuntu 22.04, o nome apropriado é **jammy**. Usar o nome

#errado significa não conseguir encontrar pacotes FRR já que o nome do diretório no repositório

#reflete o nome da versão do distro.

#é possível instalar usando chaves GPG ou o método mais antigo, apt-key add para adicionar ao

#system keyring. Usar um ou o outro pra não haver conflitos. Verificar em

```
#/etc/apt/sources.list.d/frr.list
```

```
#caso haja conflitos e seja necessário reinstalar:
```

```
sudo apt purge frr frr-pythontools
```

```
#cuidado para não usar sudo apt autoremove !!
```

```
sudo rm -rf /etc/frr
```

```
sudo rm /etc/apt/sources.list.d/frr.list
```

```
sudo rm /usr/share/keyrings/frrouting.gpg
```

```
sudo apt update
```

#verificar instalação

```
dpkg -l | grep frr
```

```
ls /usr/bin/vtysh
```

```
ls /lib/systemd/system/frr.service
```

```
#setar daemons: zebra=yes ospfd=yes ripd=yes
```

```
sudo vim /etc/frr/daemons
```

#zebra é usado por default, é responsável por gerenciar a tabela de roteamento do kernel e coordenar outros daemons. Então zebra=yes seria redundante.

#arquivo de configuração (genérico p/ não precisar um para cada daemon) e permissões.

#640 equivale a u=rw,g=r,o=

```
sudo touch /etc/frr/frr.conf
```

```
sudo chown frr:frr /etc/frr/frr.conf
```

```
sudo chmod 640 /etc/frr/frr.conf
```

#permissões do vtysh, o arquivo deve ser acessível ao grupo frrvty com permissões ug=rw,o=

```
sudo chown frr:frrvty /etc/frr/frr.conf
```

```
sudo chmod 660 /etc/frr/frr.conf
```

#verificar permissões com sudo ls -l /etc/frr/frr.conf

#começar frr em si

```
sudo systemctl enable frr
```

```
sudo systemctl start frr
```

```
sudo systemctl status frr
```

#testar vtysh

```
sudo vtysh
```

```
show version
```

#output correto abaixo, com isso a instalação estará pronta

```
rodrigo@rodrigo-Legion:~$ sudo vtysh
Hello, this is FRRouting (version 10.3).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

rodrigo-Legion# show version
FRRouting 10.3 (rodrigo-Legion) on Linux(5.15.0-139-generic).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
configured with:
  '--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info' '--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d' '--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

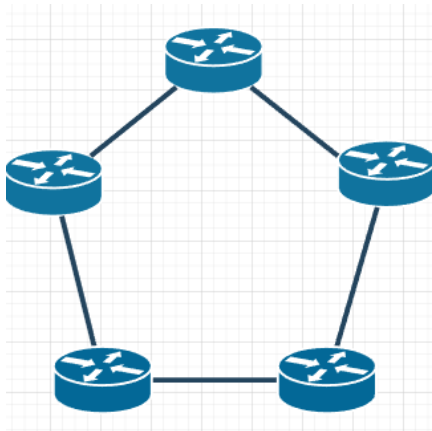
3) Definir e configurar a topologia física

Linux network namespaces permitem isolar recursos por processo, como containers no host, cada um com um conjunto diferente de recursos. Cada namespace tem suas próprias interfaces, tabelas de roteamento, firewall, etc. Nesse contexto, cada namespace representará um roteador.

A topologia tem 5 roteadores que formam um anel, cada nó se conectando a outros dois,

R1 ↔ R2 ↔ R3 ↔ R4 ↔ R5 ↔ R1.

Esboço:



Endereços IP (links entre roteadores):

- R1–R2: 10.0.1.0/24 (R1: 10.0.1.1, R2: 10.0.1.2)
- R2–R3: 10.0.2.0/24 (R2: 10.0.2.1, R3: 10.0.2.2)
- R3–R4: 10.0.3.0/24 (R3: 10.0.3.1, R4: 10.0.3.2)
- R4–R5: 10.0.4.0/24 (R4: 10.0.4.1, R5: 10.0.4.2)
- R5–R1: 10.0.5.0/24 (R5: 10.0.5.1, R1: 10.0.5.2)

Instâncias FRR: Cada roteador (namespace) executa sua própria instância FRR com zebra, ospfd e ripd para lidar com protocolos de roteamento.

Criação de namespaces:

```
sudo ip netns add r1
```

```
sudo ip netns add r2
```

```
sudo ip netns add r3
```

```
sudo ip netns add r4
```

```
sudo ip netns add r5
```

Verificar com:

```
ip netns list
```

Interfaces virtuais: veth (virtual Ethernet) para conectar roteadores

R1–R2:

```
sudo ip link add veth-r1-r2 type veth peer name veth-r2-r1
```

```
sudo ip link set veth-r1-r2 netns r1
```

```
sudo ip link set veth-r2-r1 netns r2
```

R2-R3:

```
sudo ip link add veth-r2-r3 type veth peer name veth-r3-r2
```

```
sudo ip link set veth-r2-r3 netns r2
```

```
sudo ip link set veth-r3-r2 netns r3
```

R3-R4:

```
sudo ip link add veth-r3-r4 type veth peer name veth-r4-r3
```

```
sudo ip link set veth-r3-r4 netns r3
```

```
sudo ip link set veth-r4-r3 netns r4
```

R4-R5:

```
sudo ip link add veth-r4-r5 type veth peer name veth-r5-r4
```

```
sudo ip link set veth-r4-r5 netns r4
```

```
sudo ip link set veth-r5-r4 netns r5
```

R5-R1:

```
sudo ip link add veth-r5-r1 type veth peer name veth-r1-r5
```

```
sudo ip link set veth-r5-r1 netns r5
```

```
sudo ip link set veth-r1-r5 netns r1
```

Configurar interfaces em cada namespace: associar endereços IP e levantar interfaces.

R1:

```
sudo ip netns exec r1 ip addr add 10.0.1.1/24 dev veth-r1-r2
```

```
sudo ip netns exec r1 ip addr add 10.0.2.1/24 dev veth-r1-r3
```

```
sudo ip netns exec r1 ip link set veth-r1-r2 up
```

```
sudo ip netns exec r1 ip link set veth-r1-r3 up
```

```
sudo ip netns exec r1 ip link set lo up
```

R2:

```
sudo ip netns exec r2 ip addr add 10.0.1.2/24 dev veth-r2-r1
```

```
sudo ip netns exec r2 ip addr add 10.0.5.2/24 dev veth-r2-r5
```

```
sudo ip netns exec r2 ip link set veth-r2-r1 up
```

```
sudo ip netns exec r2 ip link set veth-r2-r5 up
```

```
sudo ip netns exec r2 ip link set lo up
```

R3:

```
sudo ip netns exec r3 ip addr add 10.0.2.2/24 dev veth-r3-r1
```

```
sudo ip netns exec r3 ip addr add 10.0.3.1/24 dev veth-r3-r4
```

```
sudo ip netns exec r3 ip link set veth-r3-r1 up
```

```
sudo ip netns exec r3 ip link set veth-r3-r4 up
```

```
sudo ip netns exec r3 ip link set lo up
```

R4:

```
sudo ip netns exec r4 ip addr add 10.0.3.2/24 dev veth-r4-r3
```

```
sudo ip netns exec r4 ip addr add 10.0.4.1/24 dev veth-r4-r5
```

```
sudo ip netns exec r4 ip link set veth-r4-r3 up
```

```
sudo ip netns exec r4 ip link set veth-r4-r5 up
```

```
sudo ip netns exec r4 ip link set lo up
```

R5:

```
sudo ip netns exec r5 ip addr add 10.0.4.2/24 dev veth-r5-r4
```

```
sudo ip netns exec r5 ip addr add 10.0.5.1/24 dev veth-r5-r2
```

```
sudo ip netns exec r5 ip link set veth-r5-r4 up
```

```
sudo ip netns exec r5 ip link set veth-r5-r2 up
```

```
sudo ip netns exec r5 ip link set lo up
```

Obs.: no caso de configuração incorreta, deletar o par veth e reconfigurar.

Ex.: `sudo ip netns exec r2 ip link delete veth-r2-r3`

Interfaces LAN para testar alcance end-to-end, representam sub-redes locais a cada roteador.

R1 LAN:

```
sudo ip netns exec r1 ip link add dummy-r1 type dummy
sudo ip netns exec r1 ip addr add 192.168.1.1/24 dev dummy-r1
sudo ip netns exec r1 ip link set dummy-r1 up
```

R2 LAN:

```
sudo ip netns exec r2 ip link add dummy-r2 type dummy
sudo ip netns exec r2 ip addr add 192.168.2.1/24 dev dummy-r2
sudo ip netns exec r2 ip link set dummy-r2 up
```

R3 LAN:

```
sudo ip netns exec r3 ip link add dummy-r3 type dummy
sudo ip netns exec r3 ip addr add 192.168.3.1/24 dev dummy-r3
sudo ip netns exec r3 ip link set dummy-r3 up
```

R4 LAN:

```
sudo ip netns exec r4 ip link add dummy-r4 type dummy
sudo ip netns exec r4 ip addr add 192.168.4.1/24 dev dummy-r4
sudo ip netns exec r4 ip link set dummy-r4 up
```

R5 LAN:

```
sudo ip netns exec r5 ip link add dummy-r5 type dummy
sudo ip netns exec r5 ip addr add 192.168.5.1/24 dev dummy-r5
sudo ip netns exec r5 ip link set dummy-r5 up
```

Habilitar IP forwarding em cada namespace:

```
sudo ip netns exec r1 sysctl -w net.ipv4.ip_forward=1
sudo ip netns exec r2 sysctl -w net.ipv4.ip_forward=1
sudo ip netns exec r3 sysctl -w net.ipv4.ip_forward=1
sudo ip netns exec r4 sysctl -w net.ipv4.ip_forward=1
sudo ip netns exec r5 sysctl -w net.ipv4.ip_forward=1
```

Tornar persistente:

```
sudo ip netns exec r1 sh -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'
sudo ip netns exec r2 sh -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'
sudo ip netns exec r3 sh -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'
sudo ip netns exec r4 sh -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'
sudo ip netns exec r5 sh -c 'echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf'
```

Rodar FRR em cada namespace: cada roteador precisa de sua própria instância e configuração

```
sudo mkdir -p /etc/frr/r1 /etc/frr/r2 /etc/frr/r3 /etc/frr/r4 /etc/frr/r5
```

Copiar config padrão para cada roteador:

```
sudo cp /etc/frr/frr.conf /etc/frr/r1/frr.conf
sudo cp /etc/frr/frr.conf /etc/frr/r2/frr.conf
sudo cp /etc/frr/frr.conf /etc/frr/r3/frr.conf
sudo cp /etc/frr/frr.conf /etc/frr/r4/frr.conf
sudo cp /etc/frr/frr.conf /etc/frr/r5/frr.conf
sudo cp /etc/frr/daemons /etc/frr/r1/daemons
sudo cp /etc/frr/daemons /etc/frr/r2/daemons
sudo cp /etc/frr/daemons /etc/frr/r3/daemons
sudo cp /etc/frr/daemons /etc/frr/r4/daemons
sudo cp /etc/frr/daemons /etc/frr/r5/daemons
```

Permissões para os diretórios de cada roteador/namespace:

```
sudo chown -R frr:frrvty /etc/frr/r{1,2,3,4,5}/frr.conf
sudo chmod 660 /etc/frr/r{1,2,3,4,5}/frr.conf
sudo chown -R frr:frr /etc/frr/r{1,2,3,4,5}/daemons
sudo chmod 640 /etc/frr/r{1,2,3,4,5}/daemons
```

Start FRR em cada namespace:

Exemplo de output:

```
* Starting watchfrr with command: ' /usr/lib/frr/watchfrr -N r1 -d -F traditional zebra mgmtd ripd
ospfd staticd'
```

```
* Started watchfrr
```

R1:

```
sudo ip netns exec r1 /usr/lib/frr/frinit.sh start r1
```

R2:

```
sudo ip netns exec r2 /usr/lib/frr/frinit.sh start r2
```

R3:

```
sudo ip netns exec r3 /usr/lib/frr/frinit.sh start r3
```

R4:

```
sudo ip netns exec r4 /usr/lib/frr/frinit.sh start r4
```

R5:

```
sudo ip netns exec r5 /usr/lib/frr/frinit.sh start r5
```

Verificando conectividade entre vizinhos:

A partir de R1:

```
sudo ip netns exec r1 ping 10.0.1.2 # R1 a R2
```

```
sudo ip netns exec r1 ping 10.0.5.1 # R1 a R5
```

A partir de R2:

```
sudo ip netns exec r2 ping 10.0.1.1 # R2 a R1
```

```
sudo ip netns exec r2 ping 10.0.2.2 # R2 a R3
```

A partir de R3:

```
sudo ip netns exec r3 ping 10.0.2.1 # R3 a R2
```

```
sudo ip netns exec r3 ping 10.0.3.2 # R3 a R4
```

A partir de R4:

```
sudo ip netns exec r4 ping 10.0.3.1 # R4 a R3
```

```
sudo ip netns exec r4 ping 10.0.4.2 # R4 a R5
```

A partir de R5:

```
sudo ip netns exec r5 ping 10.0.4.1 # R5 a R4
```

```
sudo ip netns exec r5 ping 10.0.5.2 # R5 a R1
```

Possíveis problemas

Interfaces down (exemplo com r1):

Verificar status com `sudo ip netns exec r1 ip link`, cada namespace deve ter 2 links veth

Caso down, `sudo ip netns exec r1 ip link set veth-r1-r2 up`

FRR não começa:

```
sudo ip netns exec r1 cat /var/log/frr/frr.log
```

Permissões:

```
sudo ls -l /etc/frr/r1/frr.conf
```

Se necessário:

```
sudo chown frr:frrvty /etc/frr/r1/frr.conf
```

```
sudo chmod 660 /etc/frr/r1/frr.conf
```

Verificar FRR em cada namespace:

```
sudo ip netns exec r1 vtysh -c "show version"
```

Repetir para r2,3,4 e 5. Deve mostrar versão do FRR, exemplo:

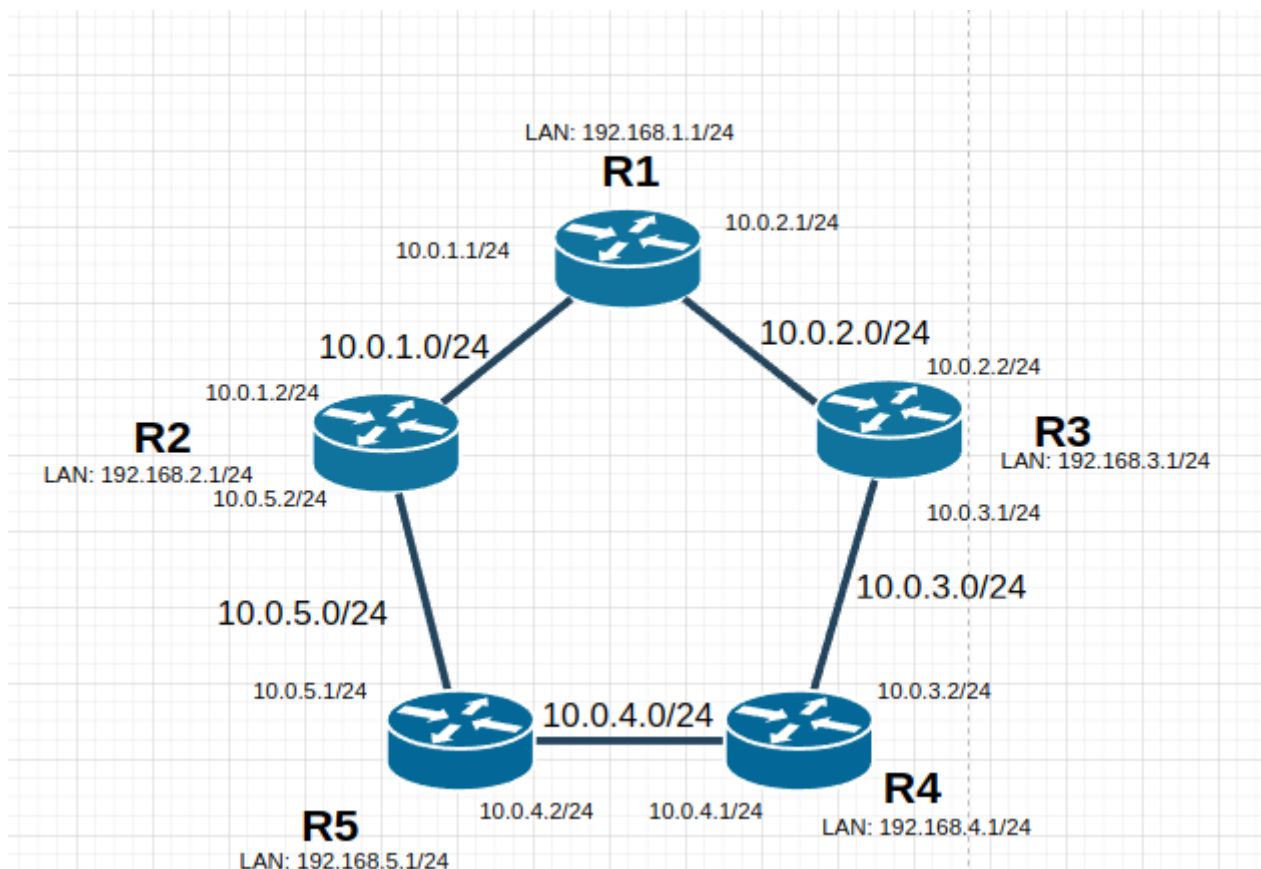
Visualização da topologia final

```
FRRouting 10.3 (rodrigo-L
Copyright 1996-2005 Kunih
configured with:
  '--build=x86_64-linux
  '--mandir=${prefix}/share
  '--localstatedir=/var
  '--libdir=${prefix}/lib/x
```

Interfaces dummy (como dummy-r1) representam LANs, não conexão a outro roteador.

Endereços IP

Router	Interface	Endereços IP	Conectado a
R1	veth-r1-r2	10.0.1.1/24	R2 (10.0.1.2)
R1	veth-r1-r3	10.0.2.1/24	R3 (10.0.2.2)
R1	dummy-r1	192.168.1.1/24	LAN
R2	veth-r2-r1	10.0.1.2/24	R1 (10.0.1.1)
R2	veth-r2-r5	10.0.5.2/24	R5 (10.0.5.1)
R2	dummy-r2	192.168.2.1/24	LAN
R3	veth-r3-r1	10.0.2.2/24	R1 (10.0.2.1)
R3	veth-r3-r4	10.0.3.1/24	R4 (10.0.3.2)
R3	dummy-r3	192.168.3.1/24	LAN
R4	veth-r4-r3	10.0.3.2/24	R3 (10.0.3.1)
R4	veth-r4-r5	10.0.4.1/24	R5 (10.0.4.2)
R4	dummy-r4	192.168.4.1/24	LAN
R5	veth-r5-r4	10.0.4.2/24	R4 (10.0.4.1)
R5	veth-r5-r2	10.0.5.1/24	R2 (10.0.5.2)
R5	dummy-r5	192.168.5.1/24	LAN



Interfaces e seus endereços IP podem ser verificados com `sudo ip netns exec r1 ip addr (r2, r3, etc.)`

4)Configurar os dois protocolos de roteamento – valor 2.0.

Escolhi OSPF e RIP para contrastar link-state e vetor de distância.

```
sudo ip netns exec r1 vtysh
```

```
configure terminal
```

```
router ospf
```

```
network 10.0.1.0/24 area 0
```

```
network 10.0.2.0/24 area 0
```

```
network 192.168.1.0/24 area 0
```

```
exit
```

```
write
```

Repetir para R2, R3, R4 e R5 com suas redes.

Caso o daemon OSPF (ospfd) não esteja habilitado na instância FRR do namespace em questão, o terminal config avisará “ospfd not running”. O daemon para IPv6 é diferente, abaixo de ospfd.

Caso OSPF não funcione apesar de estar habilitado, reiniciar o FRR no namespace (no exemplo o R1):

```
sudo ip netns exec r1 /usr/lib/frr/frinit.sh restart r1
```

Depois verificar os processos do namespace (ps é nível de kernel e listará todos os processos, não só do r1):

```
sudo ip netns exec r1 ps aux | grep ospfd
```

watchfrr -N r1 monitora daemons do FRR e ospfd -N r1 é o daemon do OSPF).

Se o processo ospfd -N r1 (r2, r3, etc.) estiver ativo, agora ospfd deverá funcionar no vtysh.

Caso ainda não funcione, pode ser que o vtysh não consegue comunicar-se com o processo ospfd.

Possíveis motivos:

- ospfd não cria ou binds ao seu socket de controle (geralmente /var/run/frr/r1/ospfd.sock), ou o vtysh não consegue acessá-lo devido a permissões.

- /etc/frr/r1/frr.conf tem algum erro

- frr user/group não tem acesso aos arquivos socket ou diretórios de config

- a instância FRR específica a Rx pode não estar isolada, fazendo ospfd tentar bind no socket/port errado.

Verificar sockets com `sudo ip netns exec r1 ls -l /var/run/frr/r1/`, devem existir: `ospfd.sock`, `zserv.api`, `zebra.sock`. Caso socket não exista e reiniciar novamente não resolva, verificar logs do FRR: `sudo ip netns exec r1 cat /var/log/frr/frr.log`, se o log deveria estar aqui mas não está, editar `/etc/frr/r1/frr.conf` para incluir:

```
hostname r1
```

```
log file /var/log/frr/frr.log
```

```
service integrated-vtysh-config
```

Permissões:

```
sudo chown frr:frrvty /etc/frr/r1/frr.conf
```

```
sudo chmod 660 /etc/frr/r1/frr.conf
```

Reiniciar FRR: `sudo ip netns exec r1 /usr/lib/frr/frinit.sh restart r1`

Testar ospfd diretamente e não pelo vtysh (porta padrão 2604): `sudo ip netns exec r1 telnet 127.0.0.1 2604`

Caso diga “Vty password is not set”, editar `/etc/frr/r1/frr.conf` adicionando:

```
!
```

```
line vty
```

```
    no login
```

Verificar permissões para arquivos frr e sockets:

```
sudo ls -l /etc/frr/r1/
```

```
sudo ls -ld /var/run/frr/r1/
```

mgmtd pode impedir a criação de sockets (como `ospfd.sock` e `zebra.sock`) já que tenta minimizar a utilização de sockets. Desabilitá-lo explicitamente em `/etc/frr/r1/daemons` com `mgmtd=no` e remover ou comentar `mgmtd_options=" -A 127.0.0.1"`

Caso ainda haja processos mgmtd, parar todos os processos FRR (globais e dos namespaces):

```
for i in {1..5}; do
```

```
    sudo ip netns exec r$i /usr/lib/frr/frinit.sh stop r$i
```

```
done
```

```
sudo /usr/lib/frr/frinit.sh stop
```

Matar processos remanescentes:

```
sudo pkill -f "frr"
```

```
sudo ip netns exec r1 pkill -f "frr"
```

```
sudo ip netns exec r2 pkill -f "frr"
```

```
sudo ip netns exec r3 pkill -f "frr"
```

```
sudo ip netns exec r4 pkill -f "frr"
```

```
sudo ip netns exec r5 pkill -f "frr"
```

Verificar que não resta nenhum processo FRR :

```
sudo ps aux | grep frr
```

```
sudo ip netns exec r1 ps aux | grep frr
```

Se houver algum, matar:

```
sudo kill -9 $(sudo ps aux | grep frr | grep -v grep | awk '{print $2}')
```

```
for i in {1..5}; do
```

```
    sudo ip netns exec r$i kill -9 $(sudo ip netns exec r$i ps aux | grep frr | grep -v grep | awk '{print $2}')
```

```
done
```

Se ainda houver processos pendentes, limpar sockets:

```
for i in {1..5}; do
```

```
    sudo ip netns exec r$i rm -rf /var/run/frr/r$i/*
```

```
done
```

```
sudo rm -rf /var/run/frr/*
```

Remover configuração global do FRR:

```
sudo mv /etc/frr/daemons /etc/frr/daemons.bak
```

```
sudo touch /etc/frr/daemons
```

```
sudo chown frr:frr /etc/frr/daemons
```

```
sudo chmod 640 /etc/frr/daemons
```

Reinstalar:

```
sudo apt remove --purge frr
```

```
sudo apt install frr
```

Verificar frr.conf e daemons (line vty no login/mgmt=no):

```
sudo cat /etc/frr/r1/frr.conf
```

```
sudo cat /etc/frr/r1/daemons
```

Se frinit.sh continuar usando watchfrr para criar sockets mgmtd manualmente (que é a possível causa de não criar sockets ospf e zebra) pode haver um problema com a versão do FRR.

Executar watchfrr manualmente, sem intermédio do frinit.sh:

```
for i in {1..5}; do  
    sudo ip netns exec r$i /usr/lib/frr/watchfrr -N r$i -d -F traditional zebra ripd ospfd staticd  
done
```

Analisar log:

```
sudo ip netns exec r1 cat /var/log/frr/frr.log
```

Alguns exemplos de erros:

Can't create pid lock file /var/run/frr/r1/watchfrr.pid (No such file or directory), exiting

ls: cannot access '/var/run/frr/r1/': No such file or directory

BE-client: msg_conn_send_msg: can't send message on closed connection

NB_OP_CHANGE: oper_walk_done: ERROR: Error sending notification message for path: /frr-interface:lib/interface[name="dummy-rX"]/state

STATIC: [MRN6F-AYZC4] Terminating on signal

OSPF: [W9T04-QWK6B] Terminating on signal

../zebra/zebra_ptm.c:1285 failed to find process pid registration

Esse erro se refere Port Tracking Module do Zebra, que não consegue registrar ou comunicar adequadamente, pode ser por falta de suporte do kernel ou problemas de configuração.

Tentar desabilitar o mgmtd nos 5 namespaces:

```
for i in {1..5}; do  
    sudo ip netns exec r$i sh -c 'echo "no service mgmtd" >> /etc/frr/r$i/frr.conf'  
done
```

Verificar se mgmtd ainda tem processos:

```
for i in {1..5}; do  
    echo "Checking mgmtd in r$i..."  
    sudo ip netns exec r$i pidof mgmtd || echo "No mgmtd running"  
done
```

Já que os logs não ajudaram, verificando versão do FRR, tentando descobrir se tem alguma incompatibilidade com o SO que não permite desabilitar o mgmt dado que anteriormente não foi possível matar os processos dele, que continuavam sendo criados mesmo após mudar os arquivos config com instruções explícitas de não usar mgmt (no service mgmtd e mgmtd=no):

```
for i in {1..5}; do
```

```
    echo "FRR version in r$i..."
```

```
    sudo ip netns exec r$i /usr/lib/frr/zebra --version
```

```
done
```

```
FRR version in r1...
```

```
zebra version 10.3
```

```
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
configured with:
```

```
'--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info'
'--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${
prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-
gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d'
'--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-
ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-
configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

```
FRR version in r2...
```

```
zebra version 10.3
```

```
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
configured with:
```

```
'--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info'
'--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${
prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-
gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d'
'--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-
ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-
configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

```
FRR version in r3...
```

```
zebra version 10.3
```

```
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
configured with:
```

```
'--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info'
'--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${
prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-
gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d'
'--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-
ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-
configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

```
FRR version in r4...
```

```
zebra version 10.3
```

```
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
configured with:
```

```
'--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info'
'--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${
prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-
gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d'
```

```
'--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

FRR version in r5...

zebra version 10.3

Copyright 1996-2005 Kunihiro Ishiguro, et al.

configured with:

```
'--build=x86_64-linux-gnu' '--prefix=/usr' '--includedir=${prefix}/include' '--mandir=${prefix}/share/man' '--infodir=${prefix}/share/info' '--sysconfdir=/etc' '--localstatedir=/var' '--disable-option-checking' '--disable-silent-rules' '--libdir=${prefix}/lib/x86_64-linux-gnu' '--libexecdir=${prefix}/lib/x86_64-linux-gnu' '--disable-maintainer-mode' '--sbindir=/usr/lib/frr' '--with-vtysh-pager=/usr/bin/pager' '--libdir=/usr/lib/x86_64-linux-gnu/frr' '--with-moduledir=/usr/lib/x86_64-linux-gnu/frr/modules' '--disable-dependency-tracking' '--enable-rpki' '--disable-scripting' '--enable-pim6d' '--disable-grpc' '--with-libpam' '--enable-doc' '--enable-doc-html' '--enable-snmp' '--enable-fpm' '--disable-protobuf' '--disable-zeromq' '--enable-ospfapi' '--enable-bgp-vnc' '--enable-multipath=256' '--enable-pcre2posix' '--enable-user=frr' '--enable-group=frr' '--enable-vty-group=frrvty' '--enable-configfile-mask=0640' '--enable-logfile-mask=0640' 'build_alias=x86_64-linux-gnu' 'PYTHON=python3'
```

uname -a

cat /etc/os-release

Linux rodrigo-Legion 5.15.0-139-generic #149-Ubuntu SMP Fri Apr 11 22:06:13 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux

NAME="Linux Mint"

VERSION="21.3 (Virginia)"

ID=linuxmint

ID_LIKE="ubuntu debian"

PRETTY_NAME="Linux Mint 21.3"

VERSION_ID="21.3"

HOME_URL="https://www.linuxmint.com/"

SUPPORT_URL="https://forums.linuxmint.com/"

BUG_REPORT_URL="http://linuxmint-troubleshooting-guide.readthedocs.io/en/latest/"

PRIVACY_POLICY_URL="https://www.linuxmint.com/"

VERSION_CODENAME=virginia

UBUNTU_CODENAME=jammy

```
for i in {1..5}; do
```

```
    echo "Adding 'no service mgmtd' to r$i..."
```

```
    sudo ip netns exec r$i sh -c "echo 'no service mgmtd' >> /etc/frr/r$i/frr.conf"
```

```
done
```

Adding 'no service mgmtd' to r1...

Adding 'no service mgmtd' to r2...

Adding 'no service mgmtd' to r3...

Adding 'no service mgmtd' to r4...

Adding 'no service mgmtd' to r5...

```
for i in {1..5}; do
```

```
    echo "Checking mgmtd setting in r$i..."
```

```
sudo ip netns exec r$i grep "no service mgmtd" /etc/frr/r$i/frr.conf
```

done

Checking mgmtd setting in r1...

no service mgmtd

Checking mgmtd setting in r2...

no service mgmtd

Checking mgmtd setting in r3...

no service mgmtd

Checking mgmtd setting in r4...

no service mgmtd

Checking mgmtd setting in r5...

no service mgmtd

```
for i in {1..5}; do
```

```
    echo "Checking daemons file in r$i..."
```

```
    sudo ip netns exec r$i grep mgmtd /etc/frr/r$i/daemons
```

```
    sudo ip netns exec r$i sed -i 's/mgmtd=yes/mgmtd=no' /etc/frr/r$i/daemons
```

```
    sudo ip netns exec r$i grep mgmtd /etc/frr/r$i/daemons
```

```
done
```

Checking daemons file in r1...

mgmtd=no

mgmtd=no

Checking daemons file in r2...

mgmtd=no

mgmtd=no

Checking daemons file in r3...

mgmtd=no

mgmtd=no

Checking daemons file in r4...

mgmtd=no

mgmtd=no

Checking daemons file in r5...

mgmtd=no

mgmtd=no

```
for i in {1..5}; do
```

```
    echo "Stopping all FRR processes in r$i..."
```

```
    sudo ip netns exec r$i pkill -9 -f "/usr/lib/frr/" || true
```

```
    sudo ip netns exec r$i rm -f /var/run/frr/r$i/*.pid
```

```
    echo "Starting watchfrr in r$i..."
```

```
    sudo ip netns exec r$i /usr/lib/frr/watchfrr -N r$i -d -F traditional zebra ripd ospfd staticd
```

done

Stopping all FRR processes in r1...

Killed

Starting watchfrr in r1...

Stopping all FRR processes in r2...

Killed

Starting watchfrr in r2...

Stopping all FRR processes in r3...

Killed

Starting watchfrr in r3...

Stopping all FRR processes in r4...

Killed

Starting watchfrr in r4...

Stopping all FRR processes in r5...

Killed

Starting watchfrr in r5...

for i in {1..5}; do

echo "Checking mgmtd in r\$i..."

sudo ip netns exec r\$i pidof mgmtd || echo "No mgmtd running"

done

Checking mgmtd in r1...

102236

Checking mgmtd in r2...

102236

Checking mgmtd in r3...

102236

Checking mgmtd in r4...

102236

Checking mgmtd in r5...

102236

Verificar se namespaces estão isolados:

for i in {1..5}; do

echo "Checking namespace isolation in r\$i..."

sudo ip netns exec r\$i cat /proc/self/status | grep NSpid

done

NSpid parecem diferentes, sugerindo estar isolados, mas deveria haver um segundo valor, como NSpid: 103018 1, e não somente 103018.

.

```
sudo pidof mgmtd
```

PID = 102236, então mgmtd está ativo no namespace do host

Tentando matar mgmtd:

```
for i in {1..5}; do
```

```
    echo "Killing mgmtd in r$i..."
```

```
    sudo ip netns exec r$i pkill -9 mgmtd || true
```

```
done
```

Verificar se ainda está em pé:

```
for i in {1..5}; do
```

```
    echo "Checking mgmtd in r$i..."
```

```
    sudo ip netns exec r$i pidof mgmtd || echo "No mgmtd running"
```

```
done
```

Verificar se há novos erros no log:

```
for i in {1..5}; do
```

```
    echo "Checking logs in r$i..."
```

```
    sudo ip netns exec r$i tail -n 20 /var/log/frr/r$i/frr.log
```

```
done
```

Se mgmtd não pode ser desabilitado, configurar o Zebra pra desabilitar suas notificações:

```
for i in {1..5}; do
```

```
    echo "Disabling mgmtd notifications in r$i..."
```

```
    sudo ip netns exec r$i sh -c "echo 'no mgmtd notifications' >> /etc/frr/r$i/zebra.conf"
```

```
done
```

Depois disso, reiniciar FRR novamente:

```
for i in {1..5}; do
    echo "Stopping all FRR processes in r$i..."
    sudo ip netns exec r$i pkill -9 -f "/usr/lib/frr/" || true
    sudo ip netns exec r$i rm -f /var/run/frr/r$i/*.pid
    echo "Starting watchfrr in r$i..."
    sudo ip netns exec r$i /usr/lib/frr/watchfrr -N r$i -d -F traditional zebra ripd ospfd staticd
done
```

E verificar logs para ver se há novos erros.

Verificar funcionalidade do OSPF:

```
for i in {1..5}; do
    echo "Checking OSPF neighbors in r$i..."
    sudo ip netns exec r$i vtysh -N r$i -c "show ip ospf neighbor"
done
```

Verificar tabela de roteamento:

```
for i in {1..5}; do
    echo "Checking routing table in r$i..."
    sudo ip netns exec r$i ip route
done
```

Verificando processos, mgmtd ainda em pé:

```
sudo ip netns exec r1 sh -c "ps -C watchfrr,zebra,mgmtd,ripd,ospfd,staticd -o pid,comm"
```

PID COMMAND

102859 watchfrr

102871 zebra

102876 mgmtd

102878 ripd

102881 ospfd

102884 staticd

Os daemons ospfd, ripd e zebra estão em pé e frr.conf não tem erros de sintaxe. O problema parece ainda ser a comunicação dos daemons com o vtysh mas o motivo ainda é um mistério. As interfaces veth de cada namespace estão corretamente conectadas.

5) Avaliar o desempenho dos protocolos de roteamento com diferentes métricas, tais como: tamanho da tabela de roteamento, quantidade de pacotes de roteamento enviados na rede, taxa de transmissão utilizada pelo protocolo de roteamento, delay, etc. – valor 2.0.

Como o sistema não funcionou não há como analisar a performance. Porém, o plano era:

Capturar pacotes com tcpdump:

```
sudo ip netns exec r1 tcpdump -i veth-r1-r2 -w ospf-r1.pcap
```

Analisar no wireshark; usar iperf3 para analisar bandwidth:

```
sudo ip netns exec r1 iperf3 -s
sudo ip netns exec r3 iperf3 -c 192.168.1.1
```

Ping para analisar delay:

```
sudo ip netns exec r1 ping 192.168.3.1 -i 0.1
```

Contar entradas na tabela de roteamento:

```
sudo ip netns exec r1 vtysh -c "show ip route" | wc -l
```

Contar pacotes do dump:

```
tcpdump -r ospf-r1.pcap | wc -l
```

Criar gráficos em Python com matplotlib. Ex.:

```
import matplotlib.pyplot as plt
times = [0.1, 0.2, 0.3, ...] # output do ping
plt.plot(times)
plt.savefig("convergencia.png")
```

<https://docs.frrouting.org>

Info sobre pacotes como o NetworkManager da wiki arch, man pages, etc.:

<https://wiki.archlinux.org/title/NetworkManager>

<https://man7.org/linux/man-pages/man1/unshare.1.html>

<https://forum.openwrt.org>

<https://forum.netgate.com>

<https://github.com/FRRouting/frr/>

<https://vyos.dev/T5439>

<https://packages.ubuntu.com/jammy/frr>

<https://lists.frrouting.org/>

https://www.kernel.org/doc/html/latest/admin-guide/namespaces/pid_namespaces.html

<https://iperf.fr/>