
S1E5-6-7 - Static vs Dynamic routing

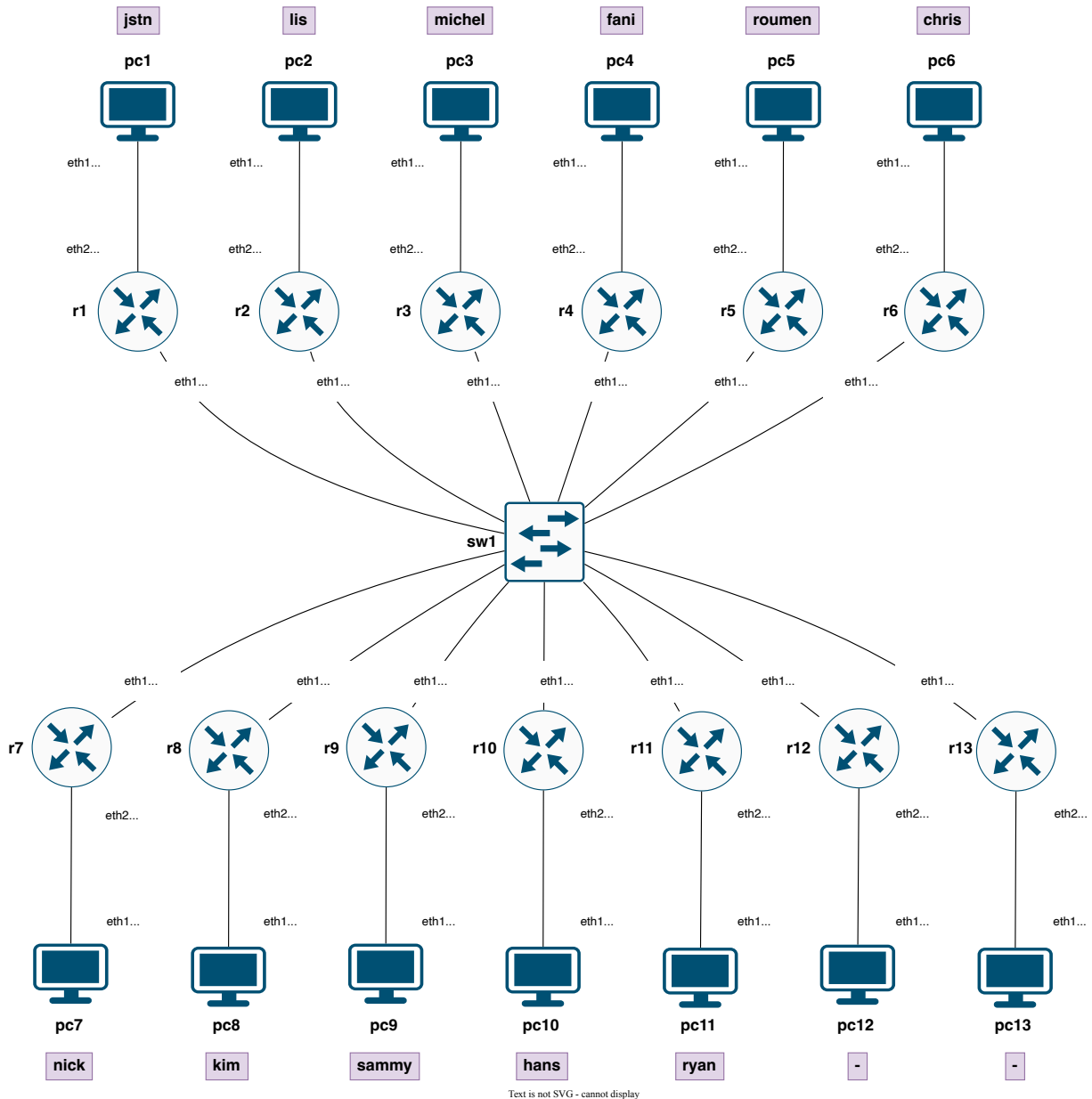
Nicholas Morrison nick@nanocat.net



Topology

Topology: five-routers-five-pcs

(You may notice that since creating this topology, there are now more than five routers and more than five pcs. Please do your best to ignore this naming discrepancy.)



Catch-up cheat-sheet (1/4)

Connect to the lab server:

```
local$ ssh-keygen -R netlab.nanocat.net
```

```
local$ ssh lab@netlab.nanocat.net
```

Password: (see discord)

Connect to your router:

```
lab@netlab$ list-routers          <- list all lab routers
lab@netlab$ list-pcs              <- list all lab PCs
lab@netlab$ docker exec -it clab-clintro-r7 Cli <- connect to your router
```

Catch-up cheat-sheet (2/4)

Configure your router interfaces:

```
r7>enable                          <- enable mode
r7#configure                        <- configuration mode
r7(config)#interface ethernet 1
r7(config-if-eth1)#description Link to Switch
r7(config-if-eth1)#no switchport
r7(config-if-eth1)#ip address 10.0.0.7/24
r7(config-if-eth1)#interface ethernet 2
r7(config-if-eth2)#description Link to PC
r7(config-if-eth2)#no switchport
r7(config-if-eth2)#ip address 192.168.7.1/24
r7(config-if-eth2)#exit
r7(config)#
```

Catch-up cheat-sheet (3/4)

Enable routing, and configure your static routes:

```
r7(config)#ip routing
r7(config)#ip route 192.168.1.0/24 10.0.0.1
r7(config)#ip route 192.168.2.0/24 10.0.0.2
r7(config)#ip route 192.168.3.0/24 10.0.0.3
r7(config)#ip route 192.168.4.0/24 10.0.0.4
r7(config)#ip route 192.168.5.0/24 10.0.0.5
r7(config)#ip route 192.168.6.0/24 10.0.0.6
r7(config)#ip route 192.168.7.0/24 10.0.0.7
r7(config)#ip route 192.168.8.0/24 10.0.0.8
r7(config)#ip route 192.168.9.0/24 10.0.0.9
r7(config)#ip route 192.168.10.0/24 10.0.0.10
r7(config)#ip route 192.168.11.0/24 10.0.0.11
r7(config)#ip route 192.168.12.0/24 10.0.0.12
r7(config)#ip route 192.168.13.0/24 10.0.0.13
r7(config)#end
r7#wr                               <- save your configuration
```

Catch-up cheat-sheet (4/4)

Verify your configuration:

```
r7#ping 192.168.7.2      <- can you ping your PC?
r7#ping 10.0.0.5        <- can you ping your neighbours?
r7#ping 10.0.0.8        <- try a bunch
r7#ping 10.0.0.2        <- to make sure
r7#ping 192.168.5.2     <- can you ping your neighbours' PC?
r7#ping 192.168.9.2     <- try some more
r7#ping 192.168.1.2     <- then you're sure
r7#show run             <- take a look at your configuration
r7#show run | include route <- filter for specific text
r7#show ip route        <- look at your routing table
```

Static Routes

“For packets with a destination in *this* network, send them to *this* router.”

```
!
ip route 192.168.7.0/24 10.0.0.7
           ^ destination   ^ router
!
```

- Receive a packet (dst: 192.168.7.2)
- Is there a matching entry in the routing table? (show ip route or show ip route 192.168.7.2)
- Yes! 192.168.7.0/24 via 10.0.0.7
- Forward that packet to 10.0.0.7

Dynamic Routing

- Programming routes manually (configure, ip route x.x.x.x/xx x.x.x.x) takes time and is error-prone.
- Static routes must be manually updated if the network topology changes.
- A **routing protocol** can take the pain away (and potentially introduce a brand new kind of pain.)
- We will configure RIPv2 - the Routing Information Protocol

About RIP

- RIP is the Routing Information Protocol
- RFC 1058, <https://datatracker.ietf.org/doc/html/rfc1058> from 1988.
- **Distance Vector**: counts “hops” to decide which path is best
- Aside from **Distance Vector** protocols, we also have **Link State Vector** and **Path Vector** protocols.

Configuring RIP

Remove all of your static routes. You can use a text editor if you want to.

```
!  
no ip route 192.168.1.0/24 10.0.0.1  
no ip route 192.168.2.0/24 10.0.0.2  
no ip route 192.168.3.0/24 10.0.0.3  
! etc  
!
```

Add the following configuration:

```
!  
router rip  
  no shutdown          <- enable RIP (it's shutdown by default)  
  network 10.0.0.0/24  <- run RIP on any interface in this range  
!
```

Verify your RIP configuration

Look at the config:

```
rX#show run | section router rip
```

Check your routing table. Do you see any RIP routes?

```
rX#show ip route  
...  
rX#show ip route rip  
...
```

Check the RIP process itself:

```
rX#show ip rip database  
...  
rX#show ip rip neighbors  
...
```

Next session: a presentation about Routing Protocols