iu-ne-lab-04-Iskander_Nafikov

- **Name**: Iskander Nafikov
- **E-mail**: i.nafikov@innopolis.university
- **Username**: iskanred
- **Hostname**: `lenovo`

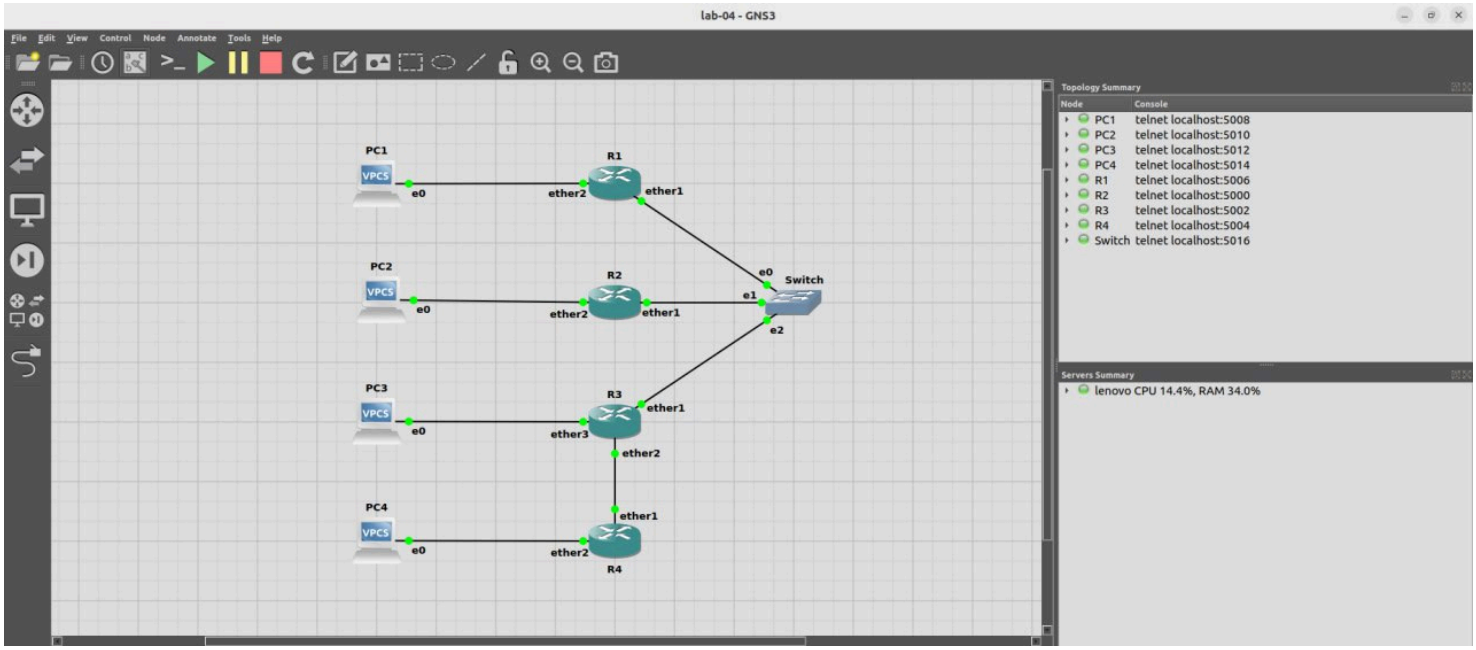
Task 1 - Prepare your network topology

> **1.** `In the GNS3 project, select and install a virtual routing solution that you would like to use: Mikrotik (recommended), Pfsense, vyos and so on.`

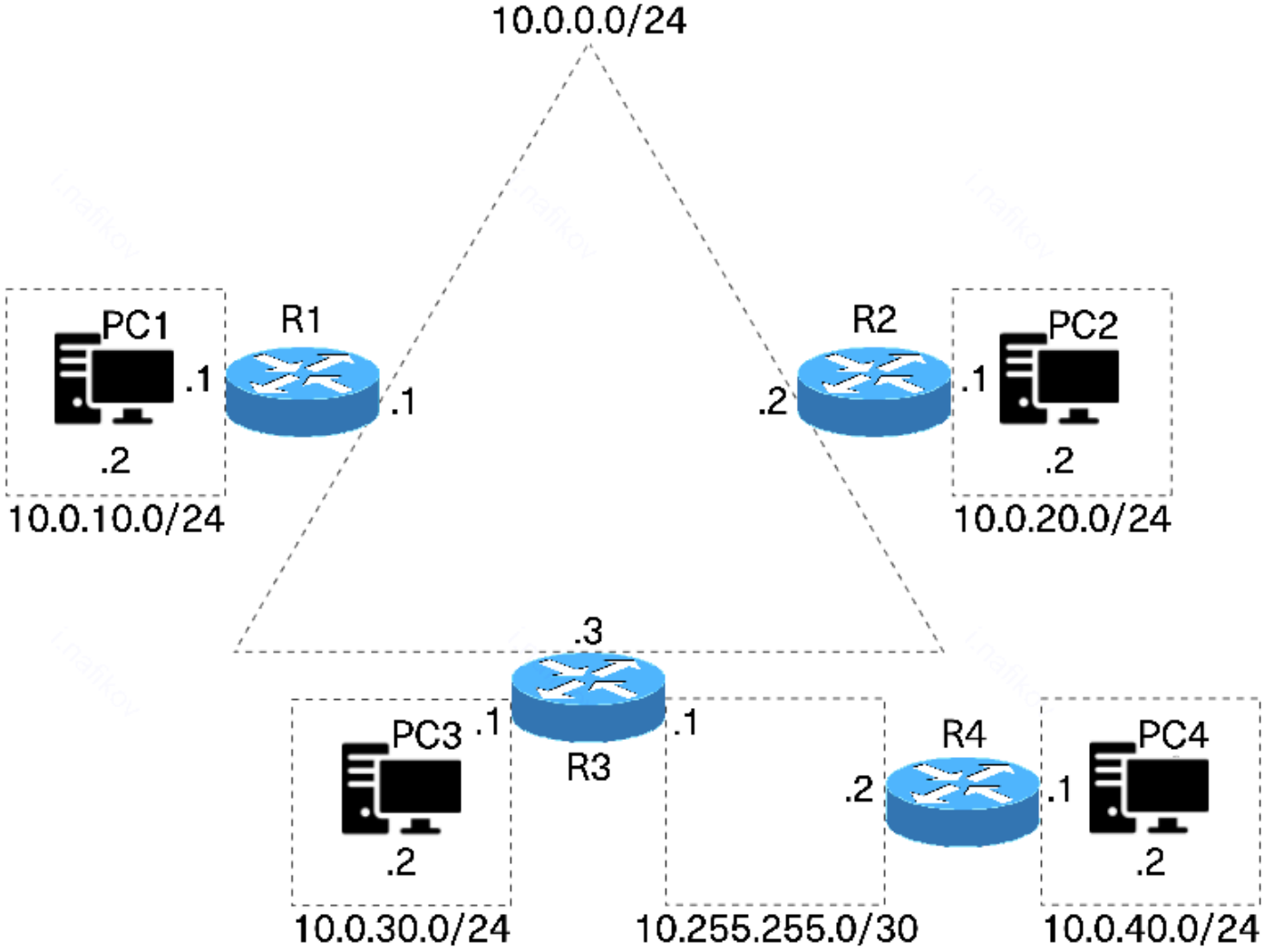
I decided to use Mikrotik as recommended - `Mikrotik CHR 7.16`

> **2.** `Prepare a simple network consisting of at least 3 routers, each one of them has a different subnet, and they should be able to reach each other (for example by a switch/router in the middle or a bus topology). Do not write a static routes between different networks.`

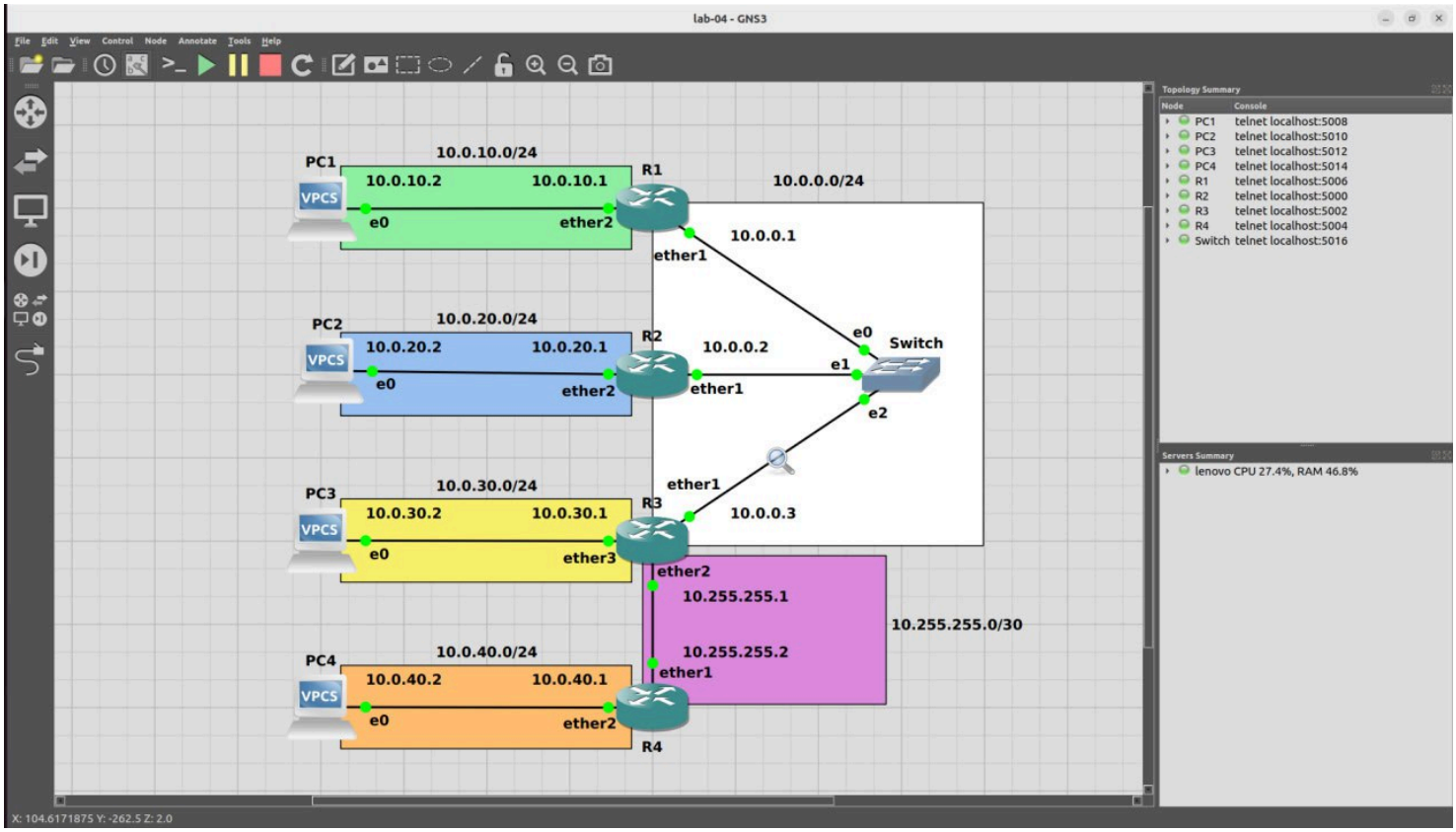
- I prepared my topology as it was suggested:



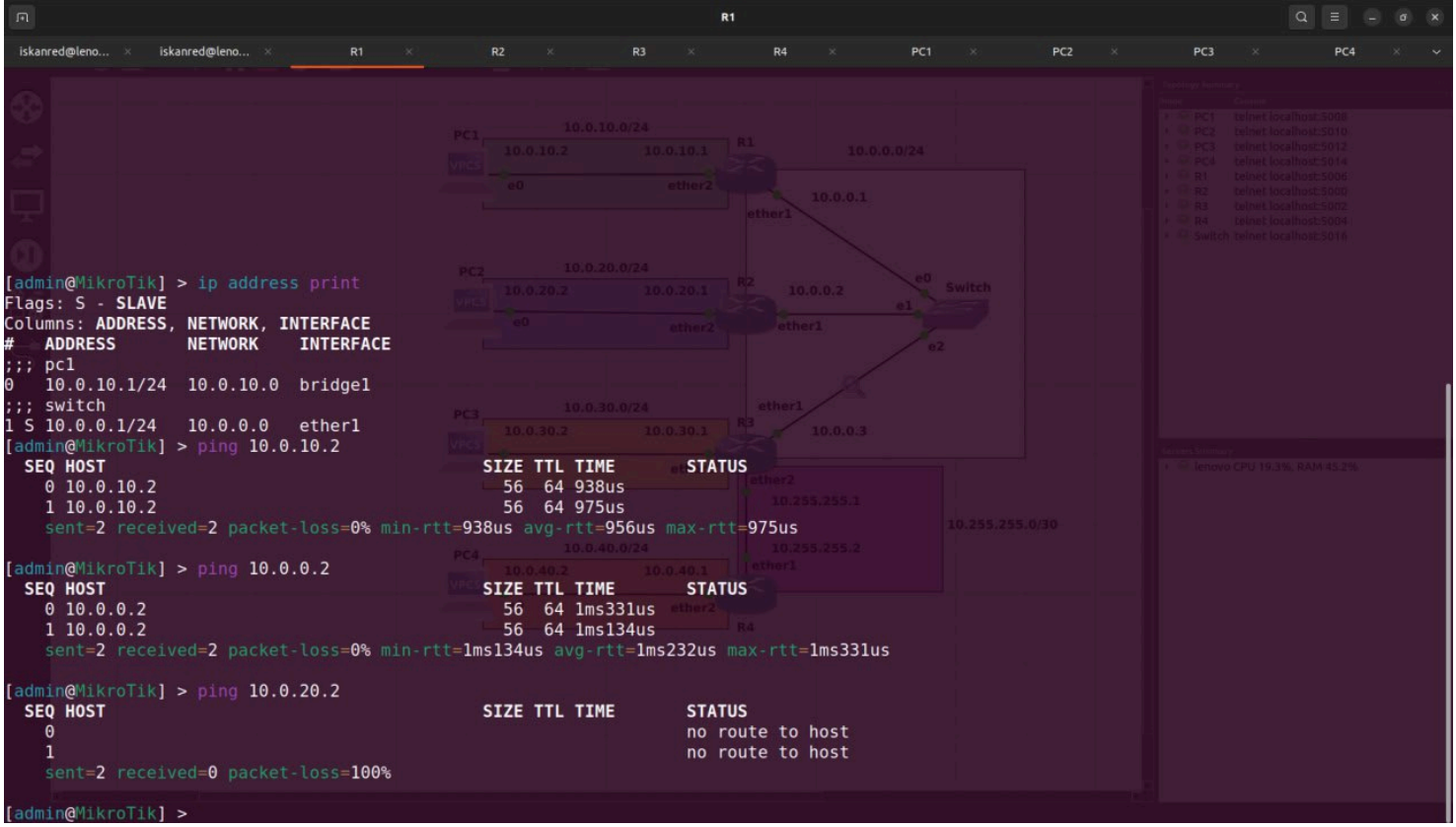
- However, to prepare it more accurately I was recommended to draw a network scheme before starting the lab. So, I decided to draw the scheme



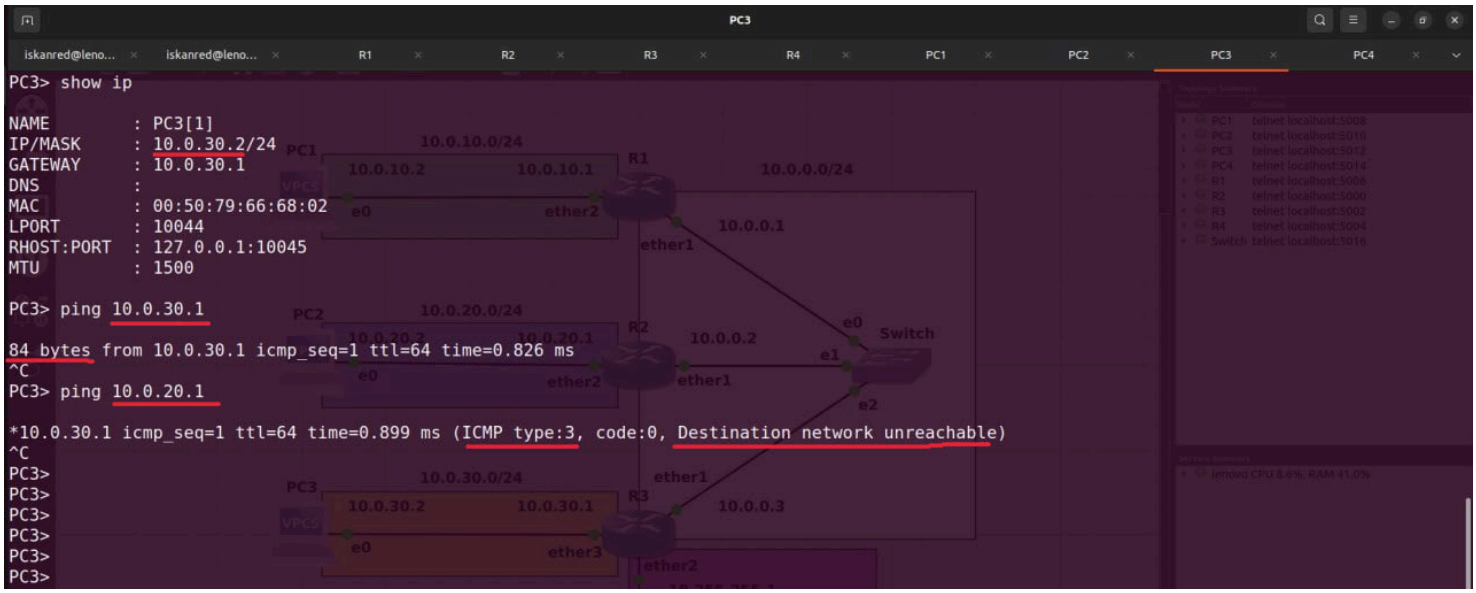
- Finally, I configured the whole topology as following:



- So now routers can reach each other and the PC node inside their own LAN, but cannot reach others' routers LANs.



- The similar was applied to PC nodes: we can reach the router, but no other routers or PCs



So far, I configured my topology without static or dynamic routing with only having direct connections between the routers.

Task 2 - OSPF Learning & Configuring

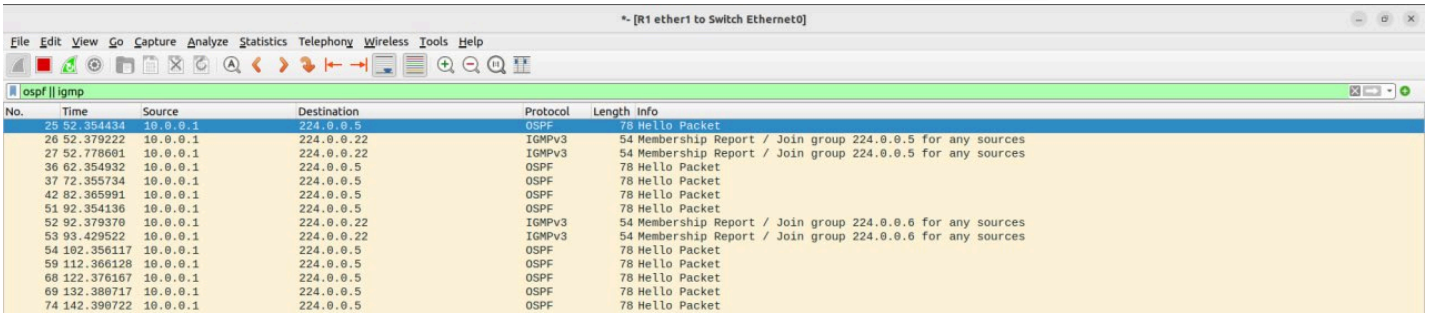
> **1.** `Deploy OSPF in your chosen network topology.`

Afterwards, I stared deploying OSPF in my topology. Actually, since I had started using IPv4 I used OSPFv2.

1. Below is the example of configuring OSPF for the router `R1` -- `ether1` interface
   - First, I added **OSPF-instance** (`ospf-instance-1`) with the **router-id =** `1.1.1.1` , **OSPF-area** (`ospf-area-1` - `0.0.0.0`) and OSPF-interface (`ether1` - `10.0.0.1/24`).

```
[admin@MikroTik] > routing/ospf
[admin@MikroTik] /routing/ospf> instance/print
Flags: X - disabled, I - inactive
[admin@MikroTik] /routing/ospf> instance add
comment      domain-id          mpls-te-address     originate-default     redistribute      use-dn
copy-from    domain-tag         mpls-te-area        out-filter-chain      router-id         version
disabled     in-filter-chain    name                out-filter-select     routing-table     vrf
[admin@MikroTik] /routing/ospf> instance add router-id=1.1.1.1
[admin@MikroTik] /routing/ospf> instance print
Flags: X - disabled, I - inactive
 0   name="ospf-instance-1" version=2 vrf=main router-id=1.1.1.1
[admin@MikroTik] /routing/ospf> area print
Flags: X - disabled, I - inactive, D - dynamic; T - transit-capable
[admin@MikroTik] /routing/ospf> area add instance=ospf-instance-1
[admin@MikroTik] /routing/ospf> area print
Flags: X - disabled, I - inactive, D - dynamic; T - transit-capable
 0    name="ospf-area-1" instance=ospf-instance-1 area-id=0.0.0.0 type=default
[admin@MikroTik] /routing/ospf> interface-template/add in
instance-id     interfaces
[admin@MikroTik] /routing/ospf> interface-template/add interfaces=ether
ether1     ether2     ether3
[admin@MikroTik] /routing/ospf> interface-template/add interfaces=ether1
area: ospf-area-1
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=waiting network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=waiting network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
[admin@MikroTik] /routing/ospf>
```

- What is noticeable above is that router's state changed from `waiting` to `dr` after 40s (== `dead-interval`) which means that the router was selected as Designated Router inside the OSPF network where there is only one the same router 😄
- Also, with the help of Wireshark we can see that after adding the interface (`ether1`) to the OSPF the router started sending OSPF `Hello` packets broadcasting it (Destination == `224.0.0.5`) over this interface.



2. Okay, then my task was to deploy OSFP to the router `R2` -- `ether1` interface
   - I repeated the same actions for the `R2` setting its router-id = `2.2.2.2`

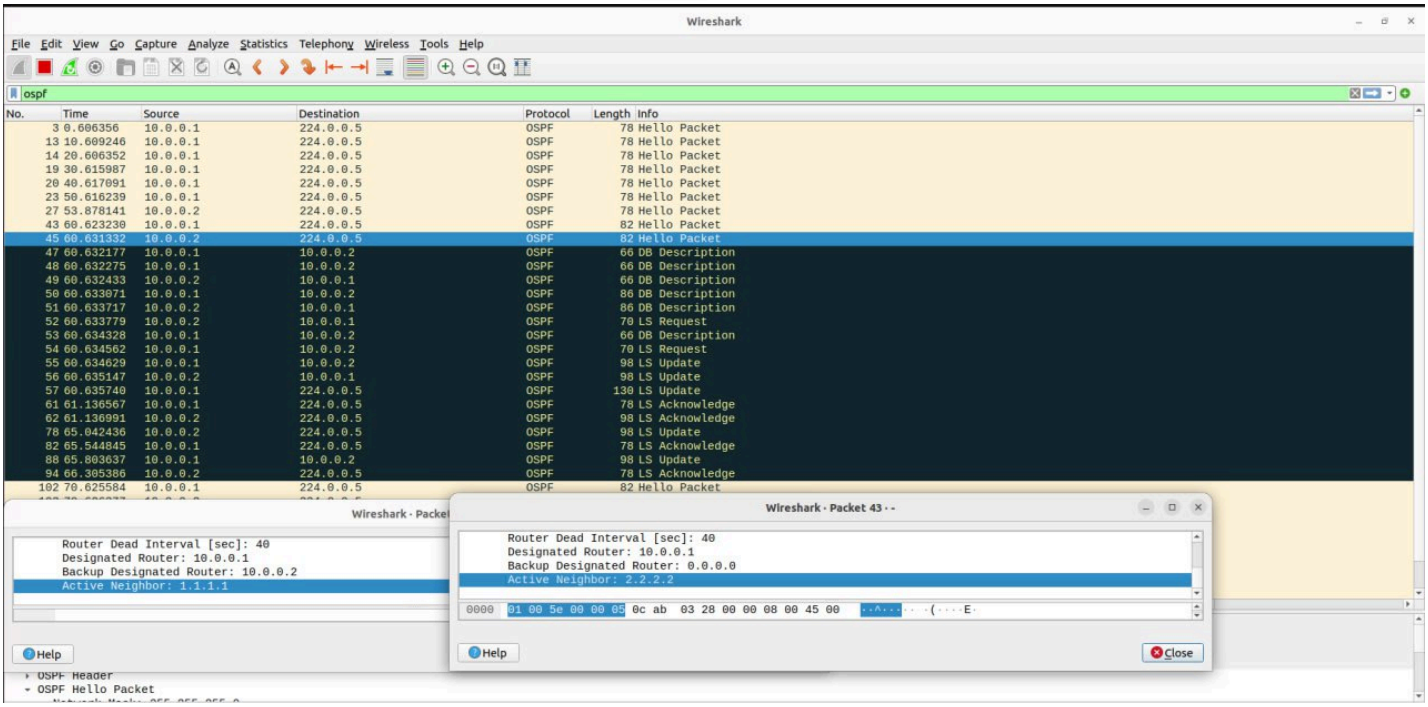
```
[admin@MikroTik] > routing ospf
[admin@MikroTik] /routing/ospf> instance add rout
router-id     routing-table
[admin@MikroTik] /routing/ospf> instance add router-id=2.2.2.2
[admin@MikroTik] /routing/ospf> area add instance=ospf-instance-1
[admin@MikroTik] /routing/ospf> interface-template add area=ospf-area-1 interfaces=ether1
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.2%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
```

   - And we immediately see that `R2` got updates from `R1` becoming `bdr` while `R1` kept being the Designated Router

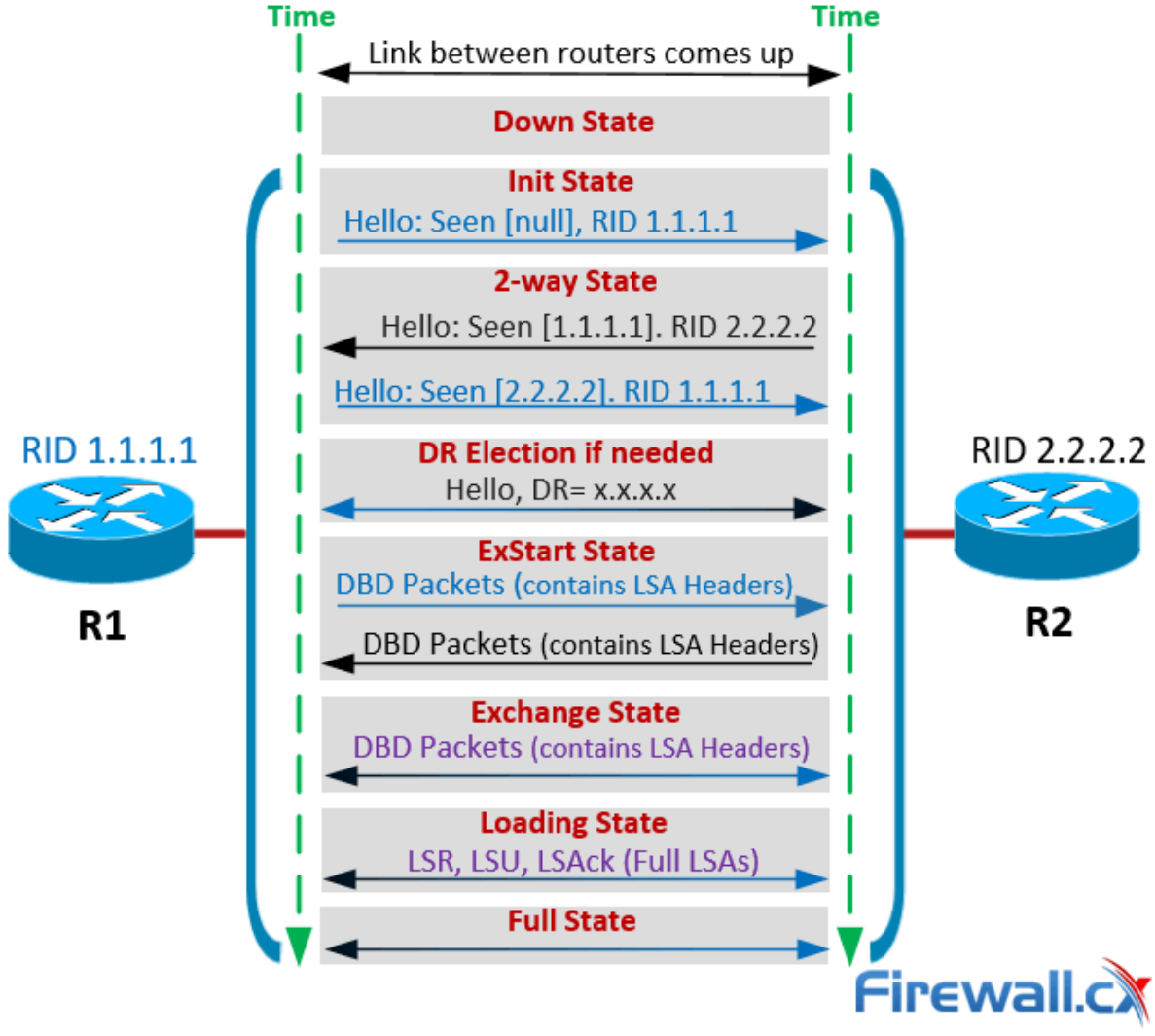
```
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.2%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
```

   - Ant it's easy to notice from the perspective of Wireshark. Both routers sent information about their neighbours in `Hello` packets and started exchanging their LSA summary in `DB Description` packets. Then, they requested

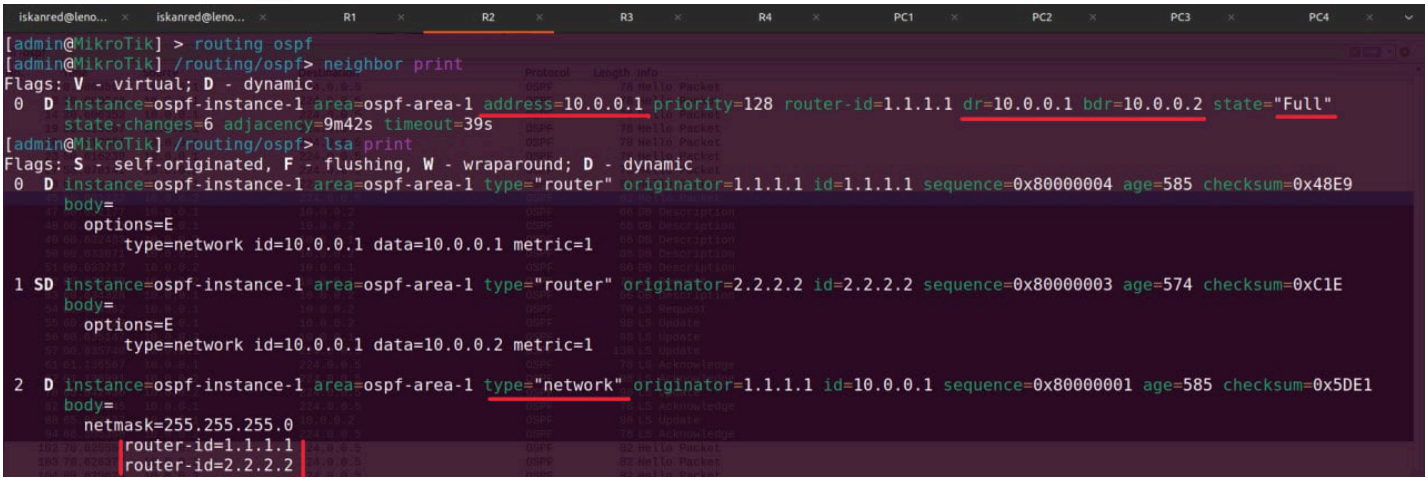
absent LSAs, acknowledged it, and moved to the `FULL` state finally.



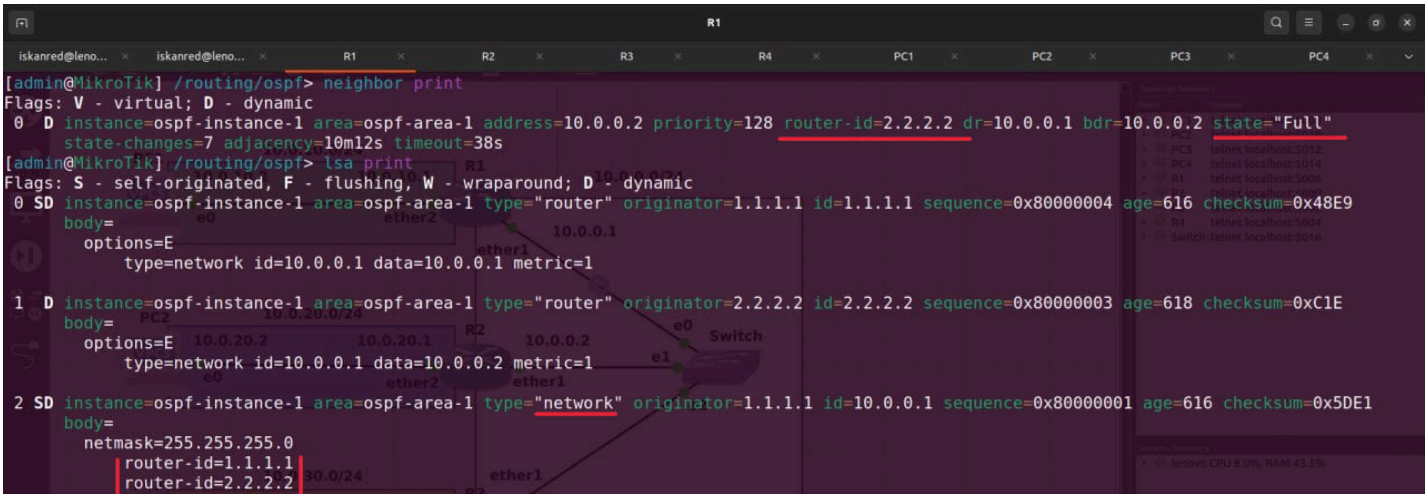
- This process can be represented in such a [state machine]



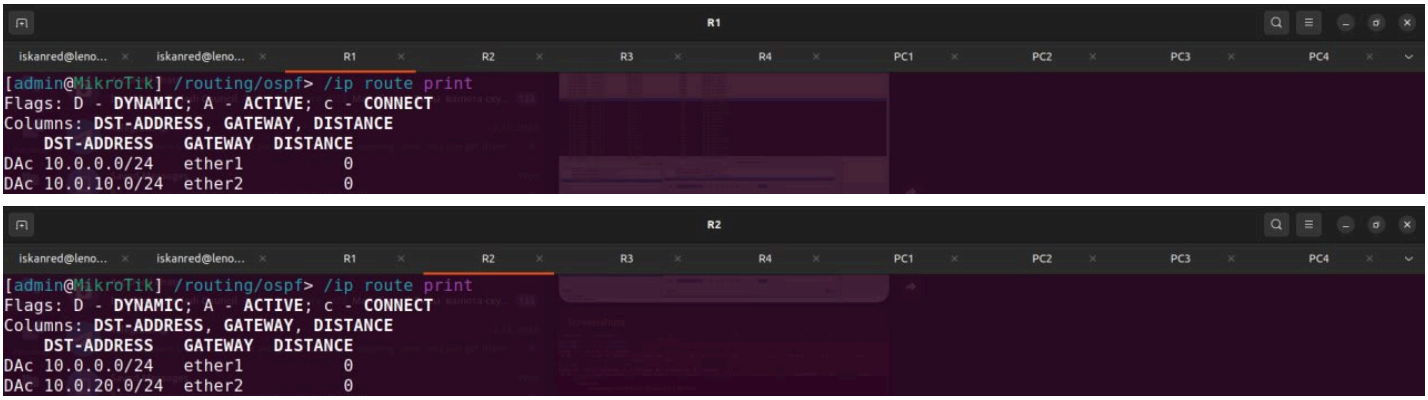
- Now we can check `R2` node's neighbour table and LSDB



- We see that it was updated and we actually see that our neighbour is `R1` with the `Full` state. What is more, we can see an LSA with `type 2 Network` which means that in this particular subnet there are only 2 routers right now. Let's check neighbour table and LSDB of the router `R1`



- The same is applied here. However if we check the routing tables on both of the routers, we see the same direct connected networks.



- To advertise other subnets, we need to deploy OSPF to all the routers' interfaces

3. Let's deploy OSPF to `R1` -- `ether2` and `R2` -- `ether2`
   - First, let's just add interfaces to the OSPF domain

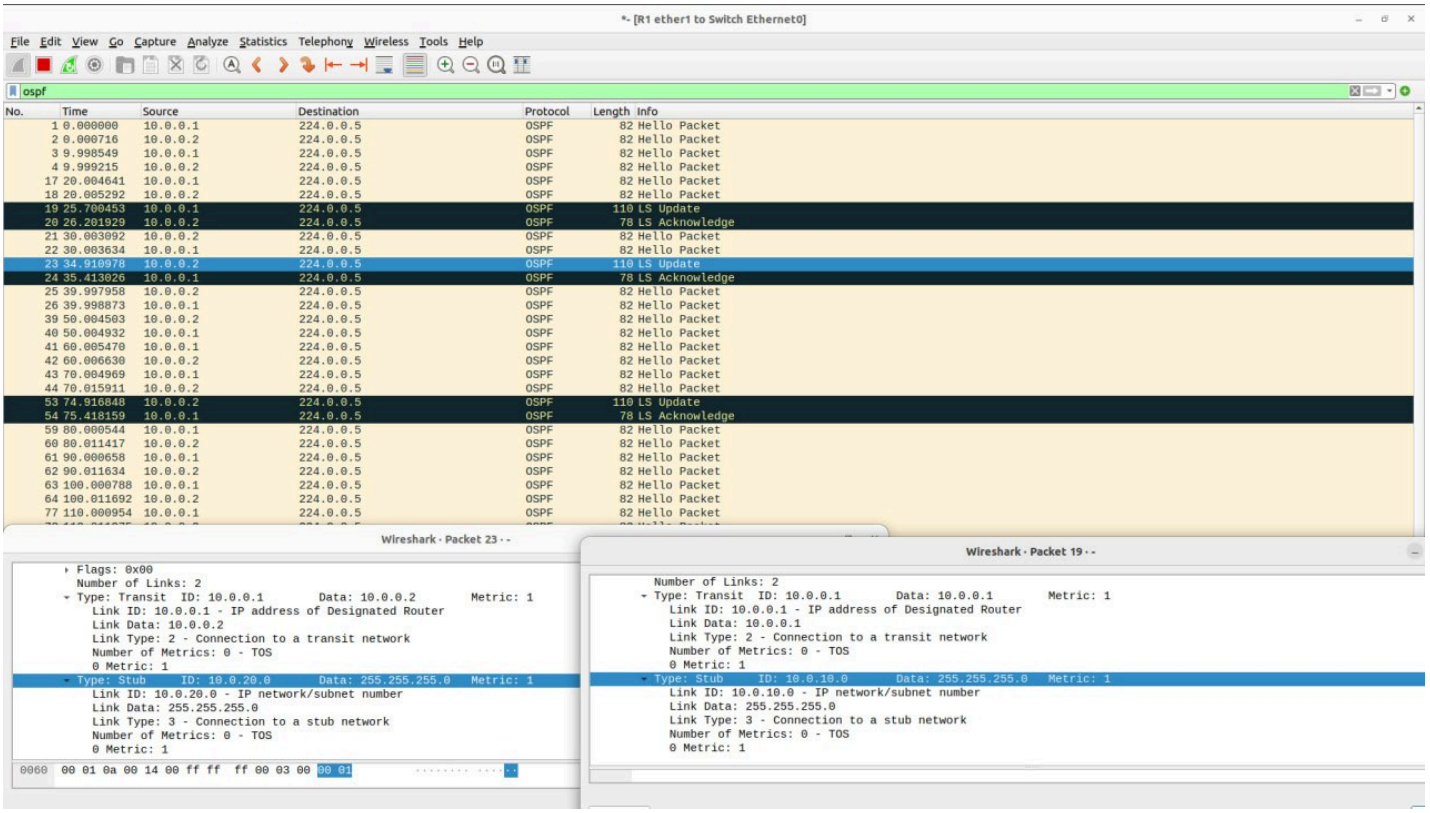
- R1

```
[admin@MikroTik] /routing/ospf> interface-template add
area            comment         disabled        networks        priority        use-bfd
auth            copy-from       hello-interval  passive         retransmit-interval  vlink-neighbor-id
auth-id         cost            instance-id     place-before    transmit-delay  vlink-transit-area
auth-key        dead-interval   interfaces      prefix-list     type
[admin@MikroTik] /routing/ospf> interface-template add interfaces=ether2
area: ospf-area-1
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=dr network-type=broadcast bdr=10.0.0.2 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.10.1%ether2 area=ospf-area-1 state=waiting network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
```

- R2

```
[admin@MikroTik] /routing/ospf> interface-template add interfaces=ether2 area=ospf-area-1
[admin@MikroTik] /routing/ospf>
[admin@MikroTik] /routing/ospf>
[admin@MikroTik] /routing/ospf>
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.2%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.20.1%ether2 area=ospf-area-1 state=waiting network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
```

- Afterwards, we notice that routers sent updates to their common subnet and advertised their own subnets with the help of `LS Update/Ack`

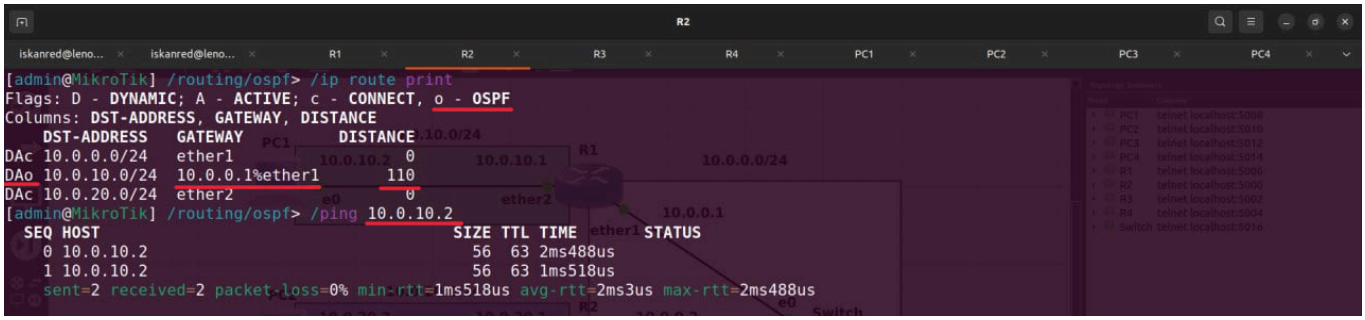


- Finally, we can see that the routing tables were updated and we can ping `PC2` from `R1` and `PC1` from `R2`

- R1

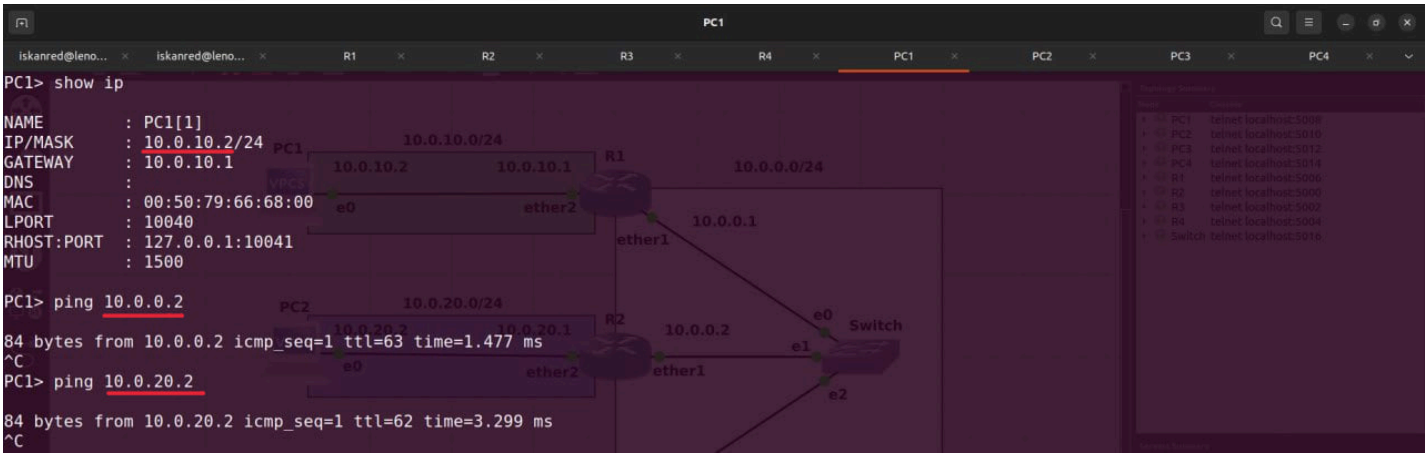
```
[admin@MikroTik] /routing/ospf> /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, o - OSPF
Columns: DST-ADDRESS, GATEWAY, DISTANCE
    DST-ADDRESS    GATEWAY     DISTANCE
DAc 10.0.0.0/24    ether1             0
DAc 10.0.10.0/24   ether2             0
DAo 10.0.20.0/24   10.0.0.2%ether1   110
[admin@MikroTik] /routing/ospf> /ping 10.0.20.2
  SEQ HOST                                     SIZE TTL TIME   STATUS
    0 10.0.20.2                                  56  63 2ms491us
    1 10.0.20.2                                  56  63 1ms819us
    sent=2 received=2 packet-loss=0% min-rtt=1ms819us avg-rtt=2ms155us max-rtt=2ms491us
```

- `R2`

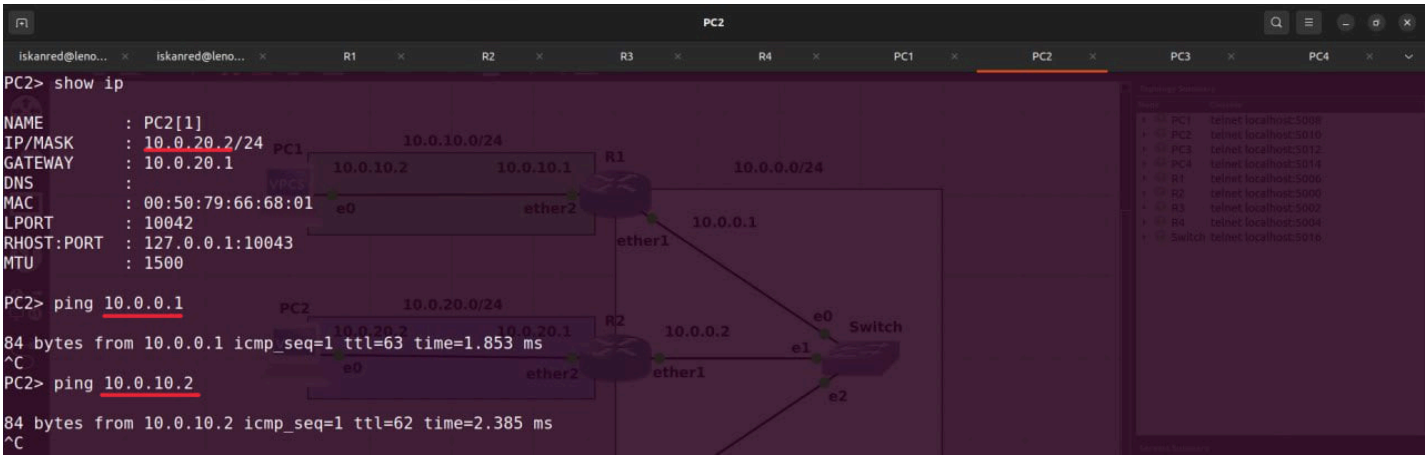


  - We could notice that Distance $= 110$ is exactly the OSPF's administrative distance

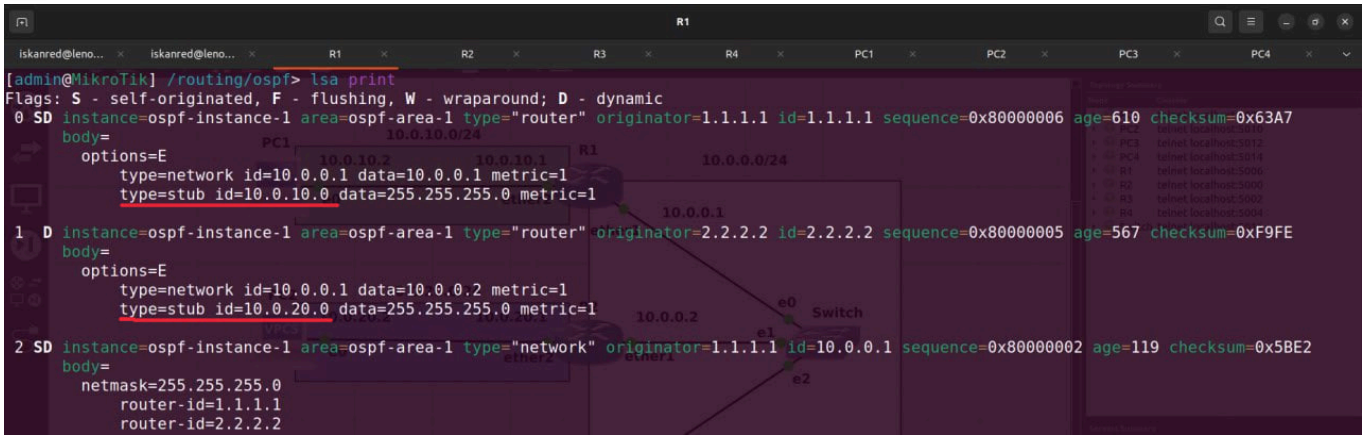
- And of course `PC1` can reach `R2` and `PC2`



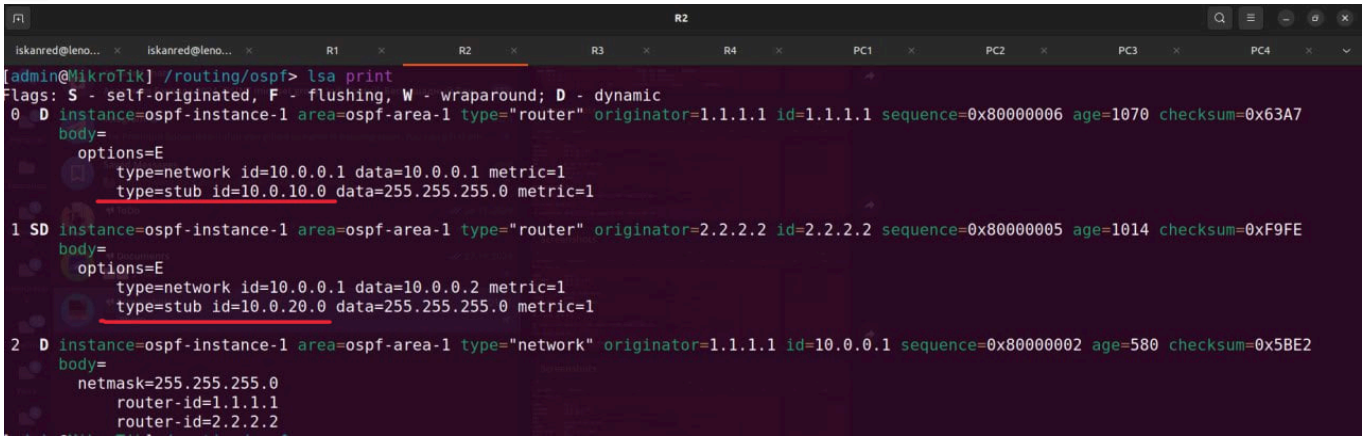
- Meanwhile `PC2` can reach `R1` and `PC1`



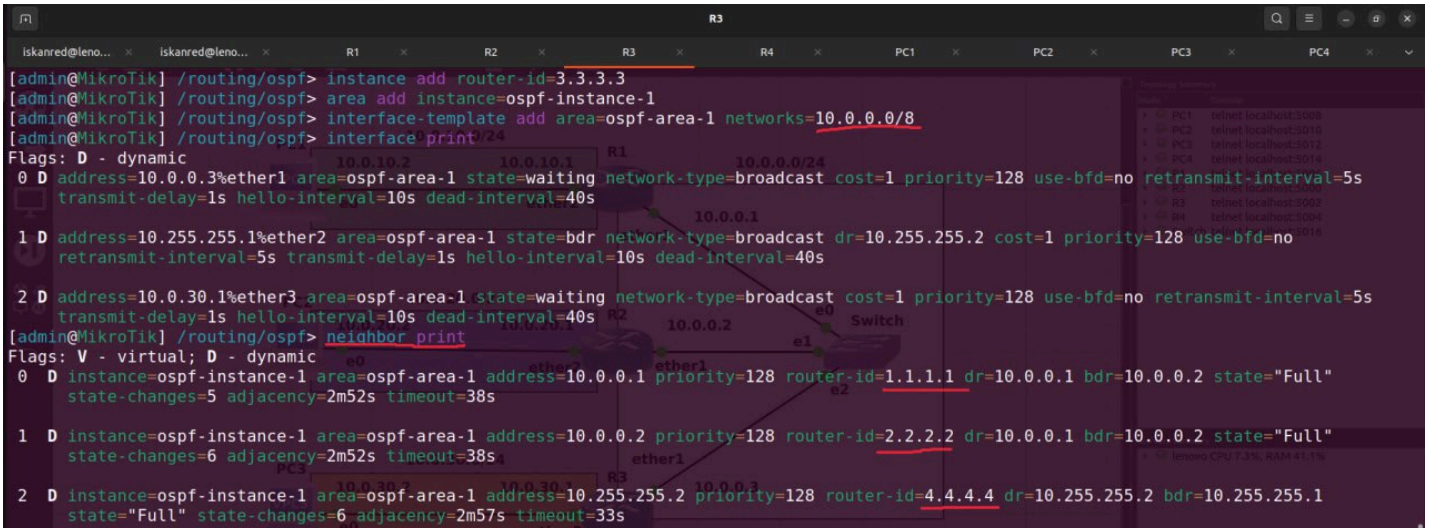
- `R1` 's and `R2` 's LSDBs were updated and synchronised
  - `R1` LSDB



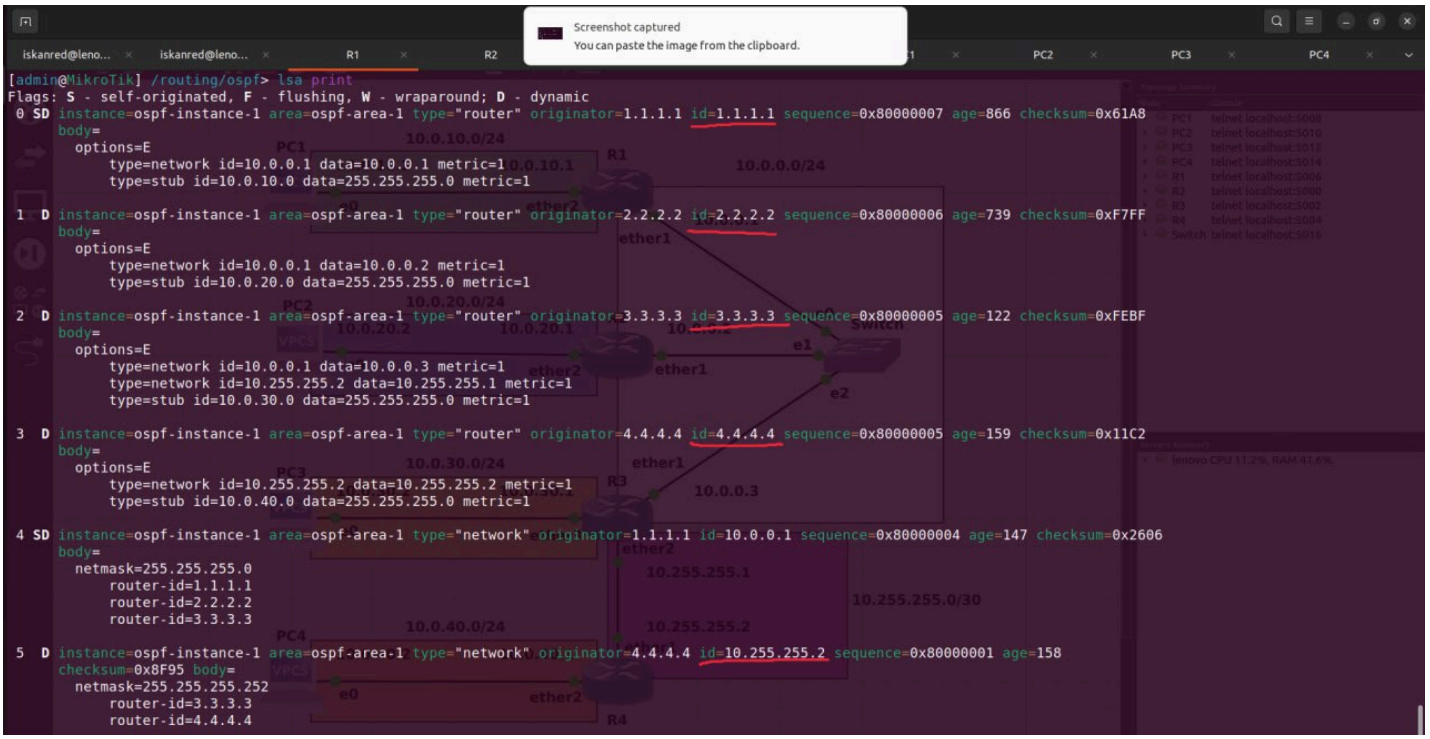
- `R2` LSDB



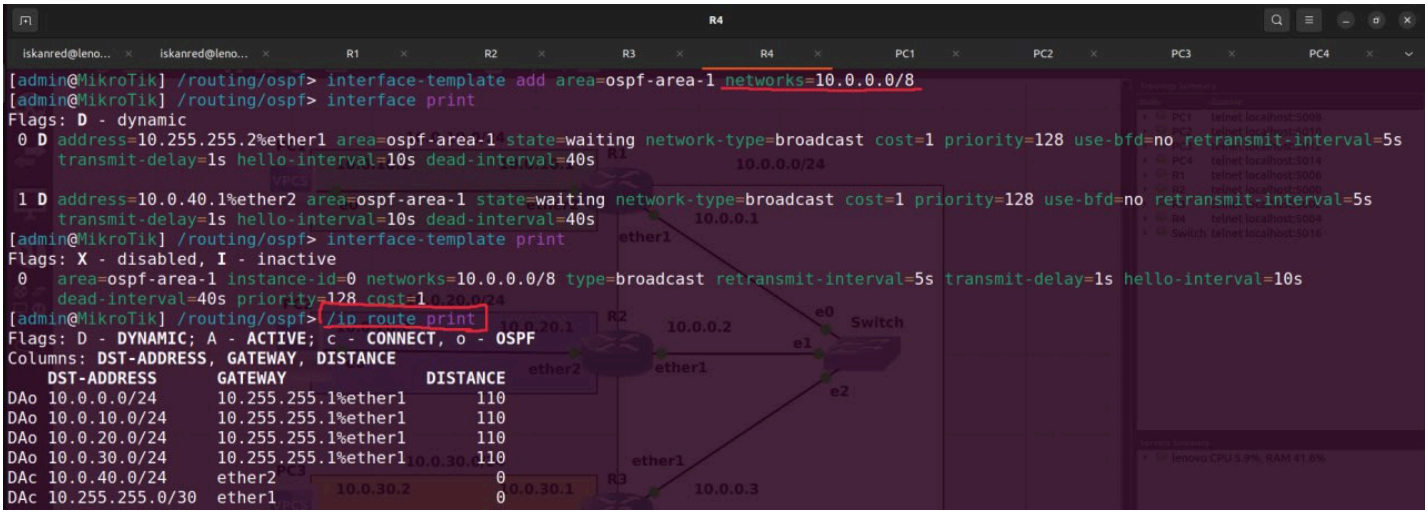
4. Finally, we can deploy OSPF in the same way for all the other routers and its interfaces
   - After doing it, we can see for example `R3` router's neighbour table. (Notice also that I added 3 interfaces of `R3` using 1 template `10.0.0.0/8`).



   - LSDB



- Or `R4`'s routing table



> **2.** `Which interface you will select as the OSPF router ID and why?`

- **Loopback interface**: following best practices, the best choice is to configure loopback Interface on a router. The highest loopback interface's IP address is selected as Router ID by default. his is because loopback interfaces are **always up** and do **not depend on physical interfaces**, providing constant **availability** and **stability**.
- **Highest non-loopback interface IP**: If a loopback interface is not configured, OSPF will automatically select the highest IP address on any of the **active** interfaces as the Router ID.
- **Explicit configuration**: In addition, it is always possible to configure Router ID manually. It may helps to avoid relying on the automatic selection and to prevent collisions.

In my case, I configured it manually and set it in the following way:

1. `R1 - 1.1.1.1`
2. `R2 - 2.2.2.2`
3. `R3 - 3.3.3.3`
4. `R4 - 4.4.4.4`

> **3.** `What is the difference between advertising all the networks VS manual advertising (per interface or per subnet)? Which one is better?`

⚠️ **Note**: In this exercise and further router `R5` exists in my topology because I started doing it after I completed Task 3 where I explained why and how I had added it

Generally, manual advertising is better. In addition, one can manually configure advertising all the networks (`0.0.0.0/0`). However, I believe that anyway administrators should be as specific as possible configuring the wildcard which will be applied to all the interfaces on which IP address matches the mask.

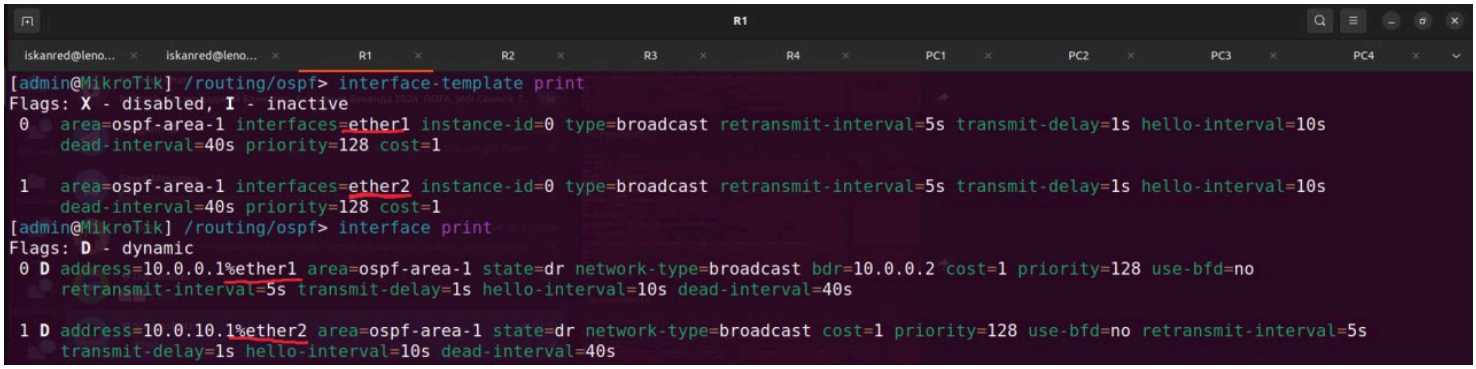
For example, other networks can be connected a router but these networks may use not OSPF protocol (RIP, BGP, etc.) or this network may lay in a different OSPF domain - in other words may lay in a different Autonomy. In such a case a redistribution of routes can be configured, while the router should become **ASBR**.

What is more, in Mikrotik example specifically, we can only configure a cost and other parameters with the help pf `interface-template` which can include many interfaces.

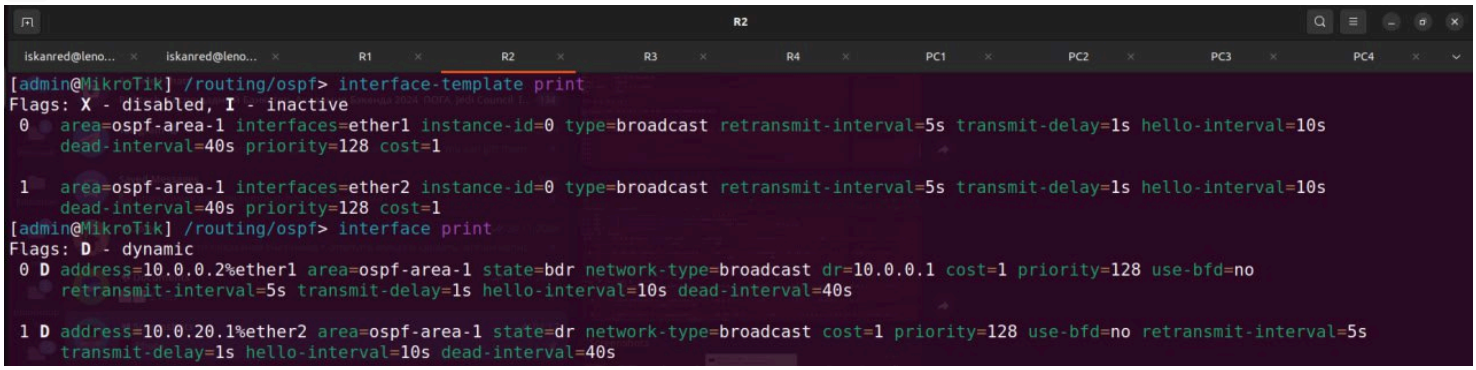
However, if an administrator is sure that there will no other autonomies, than it may be more convenient to configure advertising all the networks.

For instance, I configured `R3` and `R4` routers using the mask (`10.0.0.0/8`) while `R1` and `R2` was configured by interfaces `ether1` and `ether2` separately:

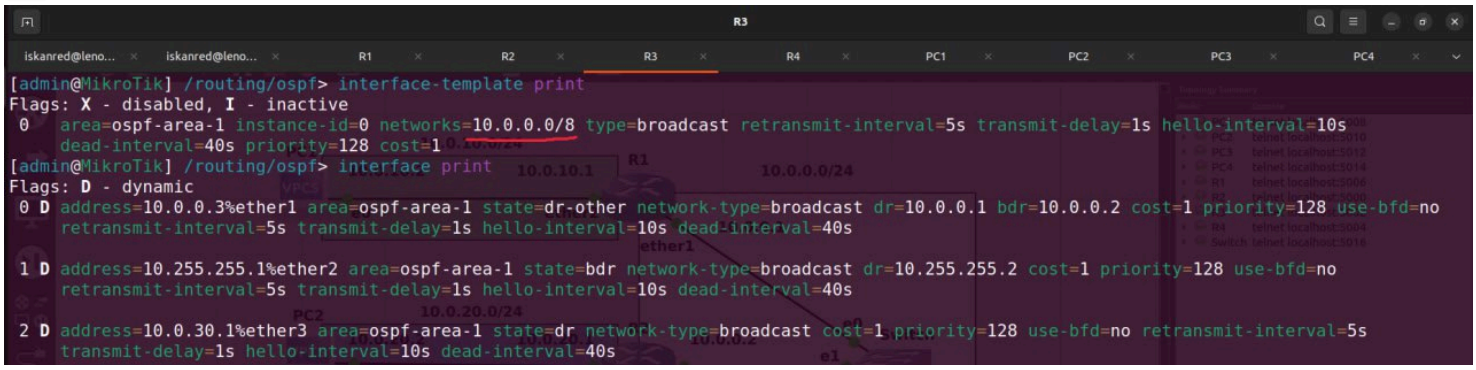
- R1



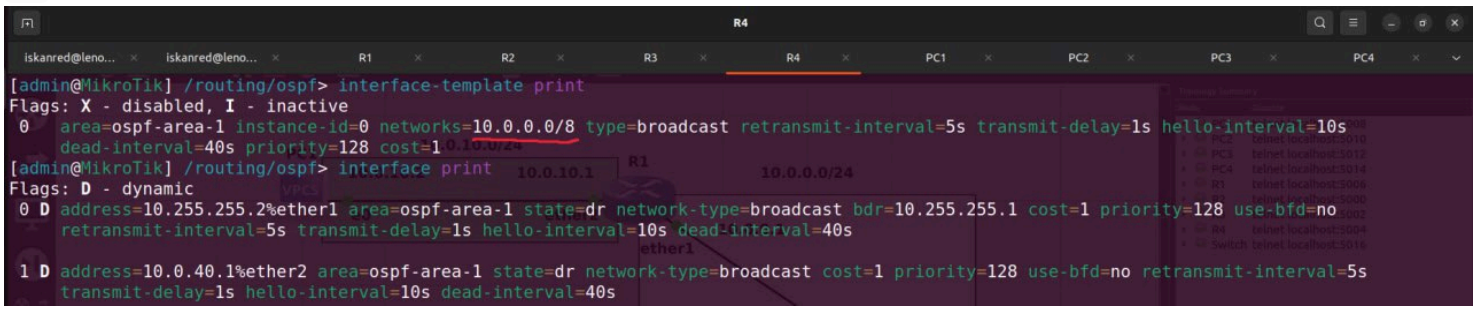
- R2



- R3



- R4



> **4.** `If you have a static route in a router, how can you let your OSPF neighbors know about it? Approve and show it on practice`

- Let's change the link between `R3` and `R4` to be not an OSPF

- From the perspective of `R3`

```
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.3%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.5 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.255.255.1%ether2 area=ospf-area-1 state=bdr network-type=broadcast dr=10.255.255.2 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 2 D address=10.0.30.1%ether3 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 instance-id=0 networks=10.0.0.0/8 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface-template set 0 networks=10.0.0.0/24
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.3%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.5 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- With not forgetting to return `10.0.30.0/24` network to keep connectivity to the `PC3`

```
[admin@MikroTik] /routing/ospf> interface-template add networks=10.0.30.0/24 area=ospf-area-1
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.3%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.5 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.30.1%ether3 area=ospf-area-1 state=waiting network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

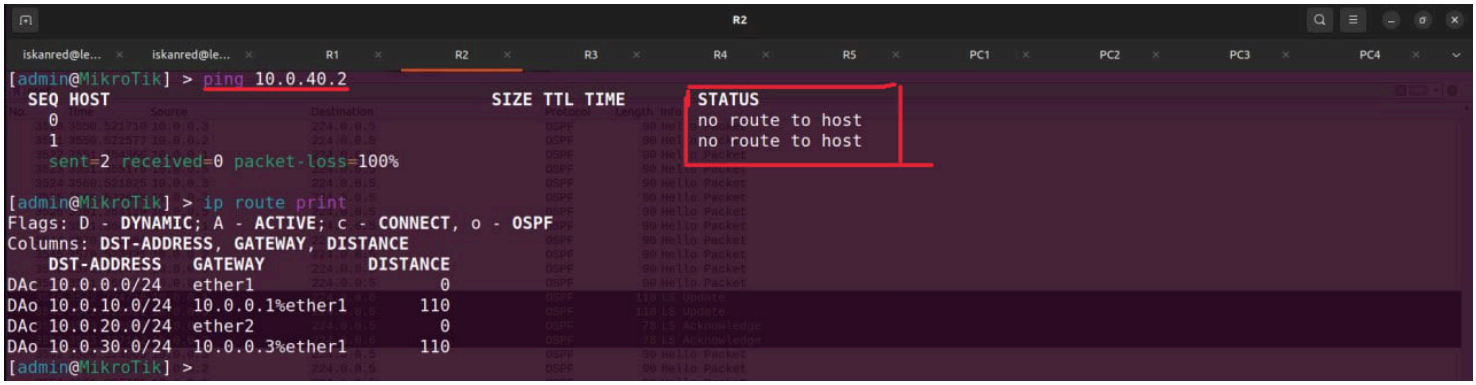
- And the same from the perspective of `R4`

```
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.255.255.2%ether1 area=ospf-area-1 state=dr network-type=broadcast bdr=10.255.255.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.40.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 instance-id=0 networks=10.0.0.0/8 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface-template set 0 network=10.0.40.0/24
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.40.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- Then, let's configure the static route from the `R3` to `10.0.40.0/24` through the `R4`

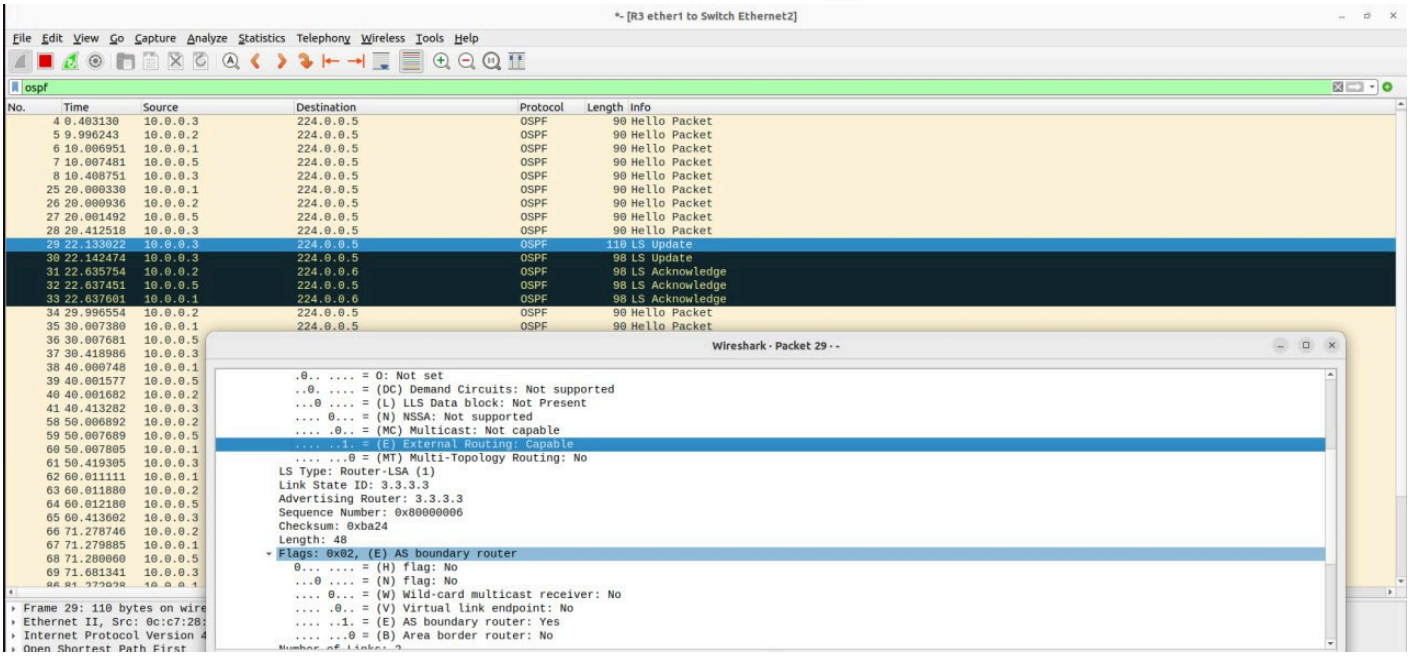
```
[admin@MikroTik] /routing/ospf> /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, o - OSPF
Columns: DST-ADDRESS, GATEWAY, DISTANCE
    DST-ADDRESS       GATEWAY          DISTANCE
DAc 10.0.0.0/24       ether1                0
DAo 10.0.10.0/24      10.0.0.1%ether1     110
DAo 10.0.20.0/24      10.0.0.2%ether1     110
DAc 10.0.30.0/24      ether3                0
DAc 10.255.255.0/30   ether2                0
[admin@MikroTik] /routing/ospf> /ping 10.0.40.2
  SEQ HOST                                     SIZE TTL TIME       STATUS
    0                                                              no route to host
    1                                                              no route to host
    sent=2 received=0 packet-loss=100%

[admin@MikroTik] /routing/ospf> /ip route add dst-address=10.0.40.0/24 gateway=10.255.255.2
[admin@MikroTik] /routing/ospf> /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC, o - OSPF
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#     DST-ADDRESS       GATEWAY          DISTANCE
  DAc 10.0.0.0/24       ether1                0
  DAo 10.0.10.0/24      10.0.0.1%ether1     110
  DAo 10.0.20.0/24      10.0.0.2%ether1     110
  DAc 10.0.30.0/24      ether3                0
0  As 10.0.40.0/24      10.255.255.2          1
  DAc 10.255.255.0/30   ether2                0
[admin@MikroTik] /routing/ospf> /ping 10.0.40.2
  SEQ HOST                                     SIZE TTL TIME       STATUS
    0 10.0.40.2                                  56  63 1ms690us
    1 10.0.40.2                                  56  63 1ms642us
    sent=2 received=2 packet-loss=0% min-rtt=1ms642us avg-rtt=1ms666us max-rtt=1ms690us

[admin@MikroTik] /routing/ospf>
```

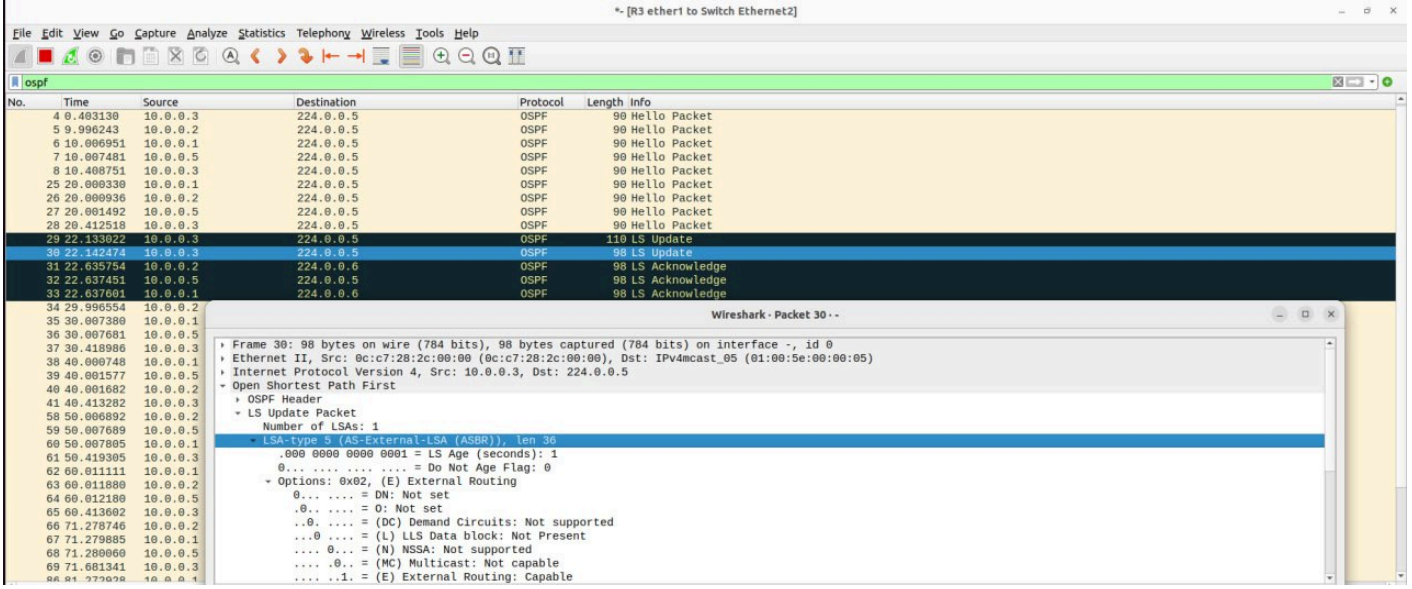
- However, we still cannot reach `10.0.40.0/24` from the other routers such as `R2` because they have no suitable route in their routing tables



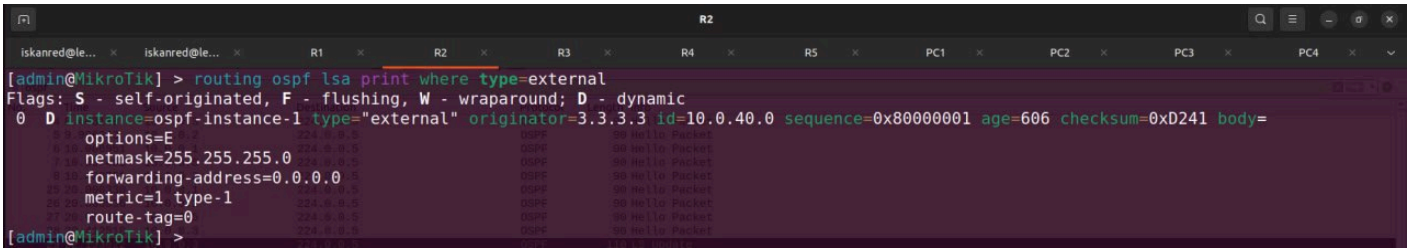
- So now, let's make `R3` an ASBR.
  - We can just add `redistribute` flag with the value `static` to our OSPF instance on the router `R3` to make it an ASBR for this particular OSPF domain redistributing its static routes as `type 5: External` LSAs throughout the whole domain

  - We can immediately see this LSDB update requests initiated by `R3`

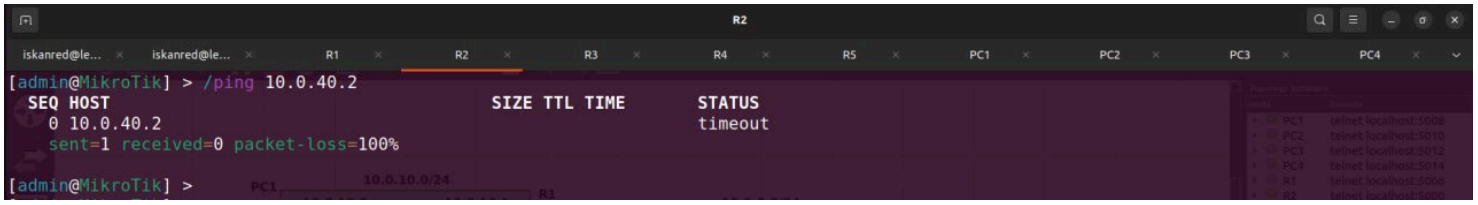




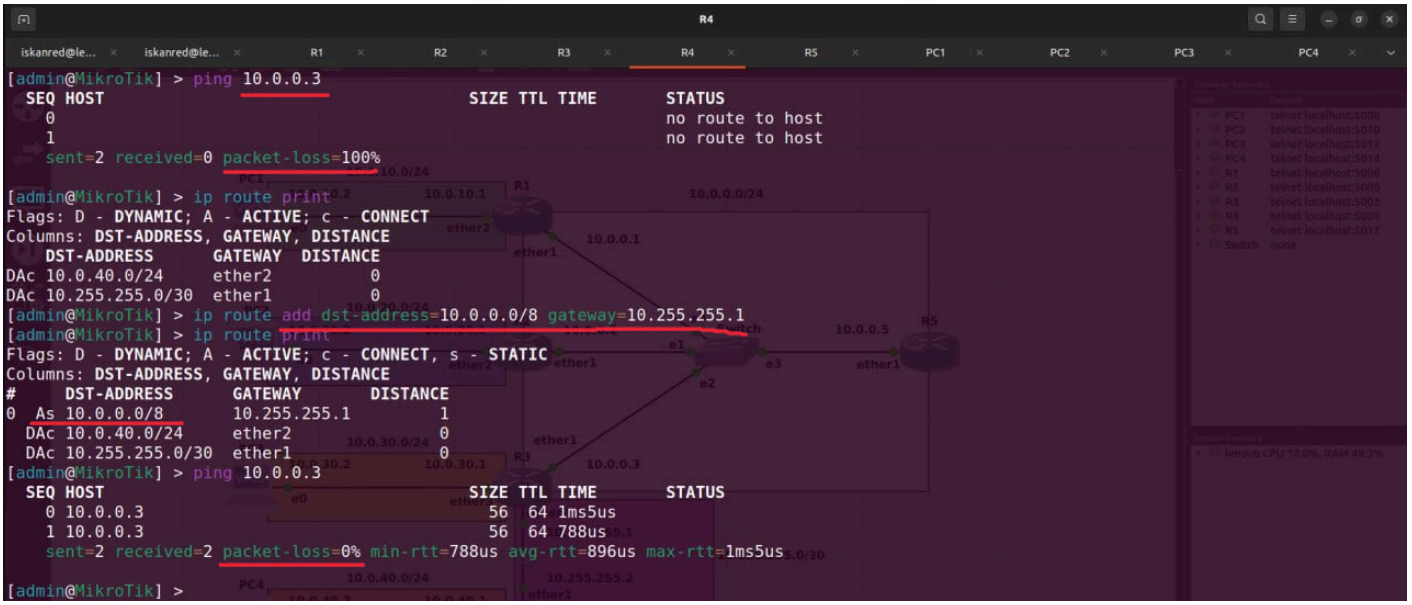
- As we see these requests contain exactly the information that `R3` became an ASBR for this OSPF domain and this particular area, and new external Autonomous System, which is is marked as `LSA-type 5`. Therefore, now we can actually make sure that this LSA exists in the LSDB



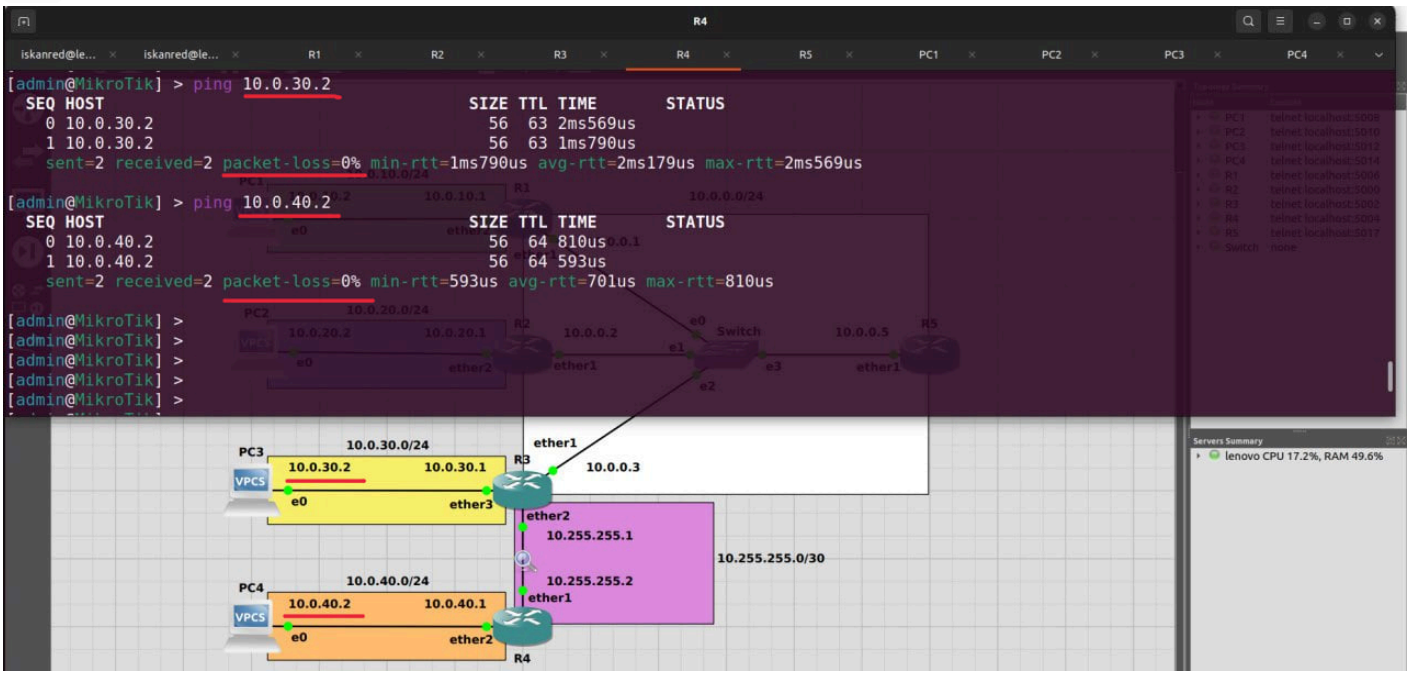
- Although now we can reach the `10.0.40.0/24` from our OSPF domain it was not enough and we got only timeouts



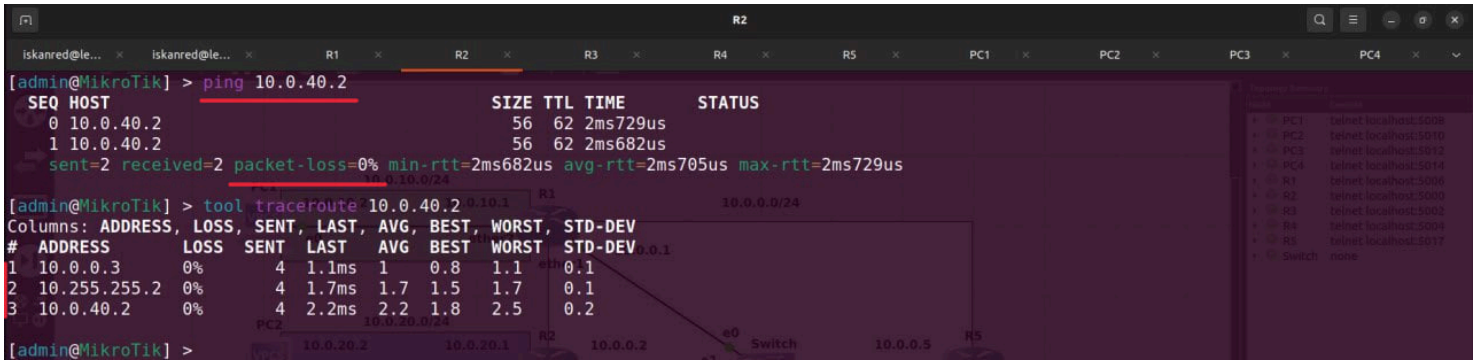
- They occurred because packets now can be easily routed outside the OSPF domain, but `R4` knows nothing about how to route them back.
  - So, let's also add a static route from `R4` to our OSPF domain through `R3`



- Notice that I have added `10.0.0.0/8` which is enough to ensure the coverage of the current addresses in the whole topology. A key point here is route precedence rules (packets go through the link which is specified as the most specific over all the routing table entries) keep the connectivity with `10.0.40.0/24` - and, thus, with `PC4`



- Finally, me made it possible to reach another Autonomous System that uses static routing from our OSPF domain



> **5.** `Enable OSPF with authentication between the neighbors and verify it`

I decided to enable OSPF authentication based on the simple plain-text password transmission inside the OSPF `Hello` packets' header

- First, I enabled the authentication on `R1`

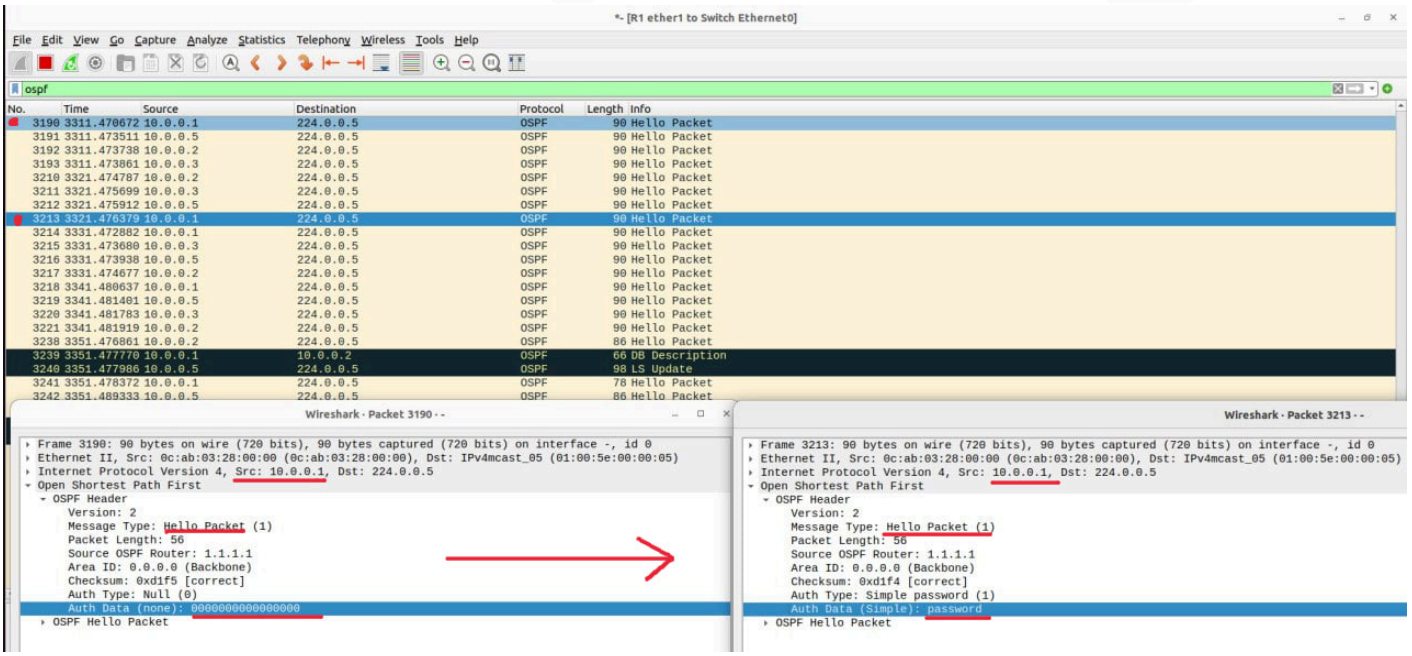
- We can see there are 3 `R1`'s neighbours. Also, other routers' subnets are reachable

```
[admin@MikroTik] /routing/ospf> /ip route print where ospf
Flags: D - DYNAMIC; A - ACTIVE; o - OSPF
Columns: DST-ADDRESS, GATEWAY, DISTANCE
    DST-ADDRESS    GATEWAY          DISTANCE
DAo 10.0.20.0/24   10.0.0.2%ether1       110
DAo 10.0.30.0/24   10.0.0.3%ether1       110
DAo 10.0.40.0/24   10.0.0.3%ether1       110
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0   D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.2 priority=128 router-id=2.2.2.2 dr=10.0.0.5 bdr=10.0.0.3 state="TwoWay"
     state-changes=4 timeout=39s

 1   D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.3 priority=128 router-id=3.3.3.3 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
     state-changes=7 adjacency=42m42s timeout=39s

 2   D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.5 priority=128 router-id=5.5.5.5 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
     state-changes=7 adjacency=42m42s timeout=39s
[admin@MikroTik] /routing/ospf> /ping 10.0.30.2
  SEQ HOST                                     SIZE TTL TIME       STATUS
    0 10.0.30.2                                  56  63 3ms2us
    1 10.0.30.2                                  56  63 2ms34us
    sent=2 received=2 packet-loss=0% min-rtt=2ms34us avg-rtt=2ms518us max-rtt=3ms2us

[admin@MikroTik] /routing/ospf>
```

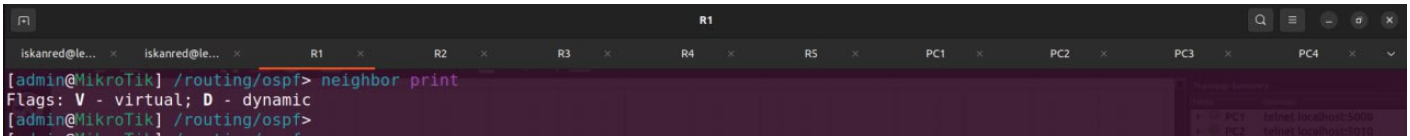
- I set up the password `password` for the interface `ether1` which goes to the subnet `10.0.0.0/24` where other routers are located also. The interesting part here is that it became `DR` inside its own authenticated multicast link.

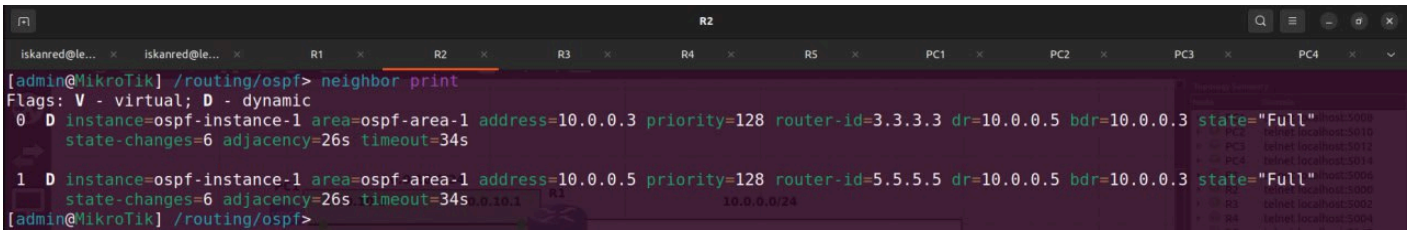
```
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 interfaces=ether1 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1

 1   area=ospf-area-1 interfaces=ether2 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface-template set 0 auth=simple auth-id=1 auth-key=password
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 interfaces=ether1 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1 auth=simple auth-key="password" auth-id=1

 1   area=ospf-area-1 interfaces=ether2 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=dr-other network-type=broadcast dr=10.0.0.5 bdr=10.0.0.3 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.10.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.10.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- And we can see the updates in Wireshark. `R1` had sent its auth data inside the OSPF `Hello` packet's header

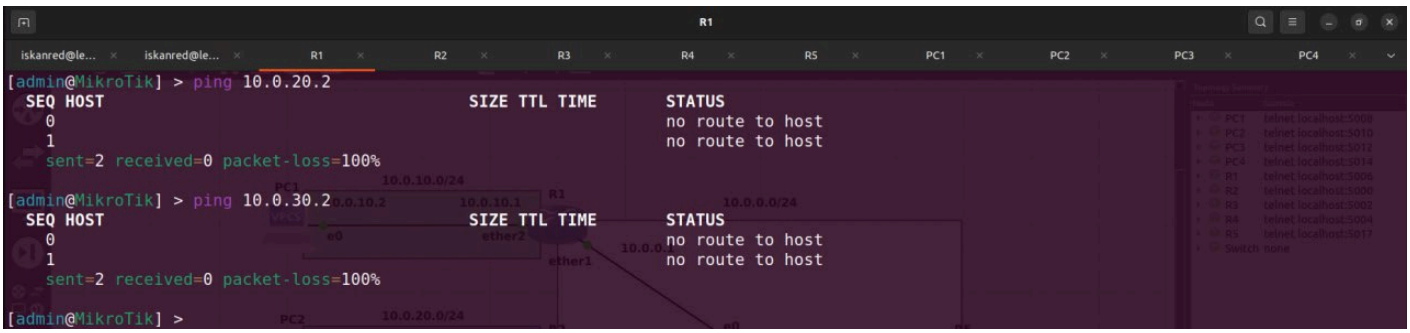


- Next, we can see that router `R1` "lost" all its neighbours, while other routers "lost" him as a neighbour.

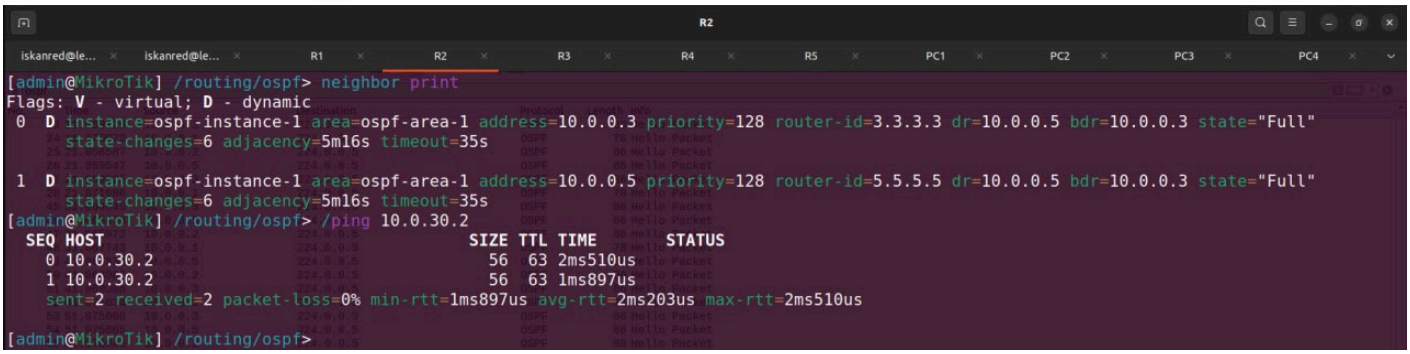




- What is more, since there is no router who can become a neighbour to `R1`, it lost the reachability to other routers and vice a versa



- Okay, now let's configure authentication with the same password on `R2`
  - We can see there are 2 `R1`'s neighbours. Also, these routers' subnets are reachable

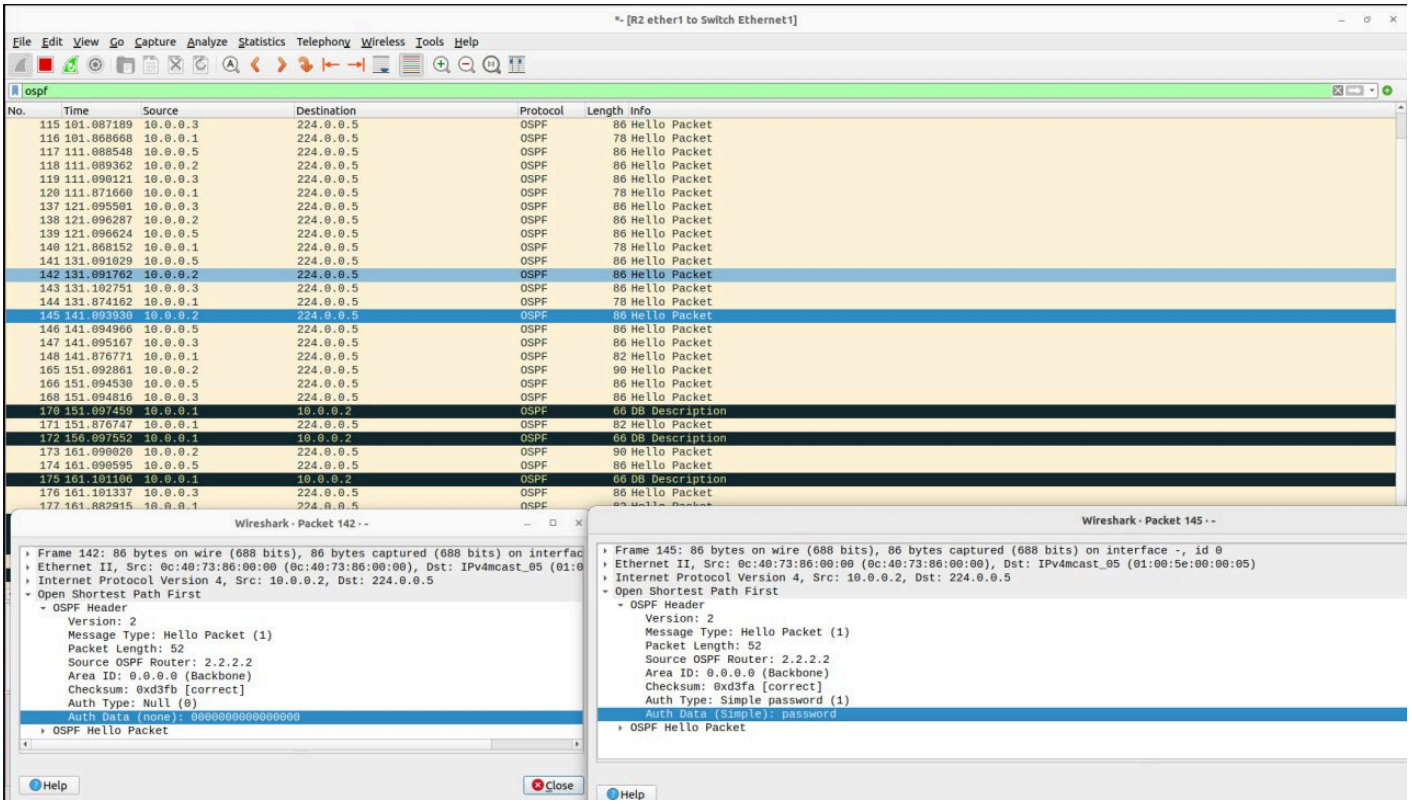


  - I set up the password `password` for the interface `ether1` which goes to the subnet `10.0.0.0/24` where other routers are located also. The interesting part here is that it became `BDR` inside its authenticated

multicast link, while `DR` remained the same - `R1`.

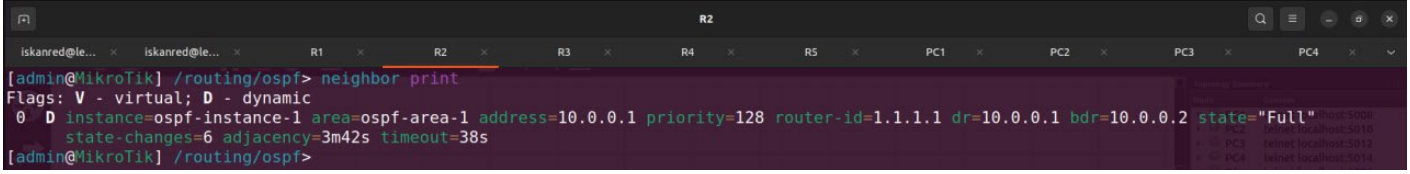
```
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 interfaces=ether1 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1

 1   area=ospf-area-1 interfaces=ether2 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface-template set 0 auth=simple auth-id=1 auth-key=password
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 interfaces=ether1 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1 auth=simple auth-key="password" auth-id=1

 1   area=ospf-area-1 interfaces=ether2 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.1 priority=128 router-id=1.1.1.1 dr=10.0.0.1 bdr=0.0.0.0 state="TwoWay"
      state-changes=2 timeout=36s

 1  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.3 priority=128 router-id=3.3.3.3 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=6m55s timeout=6s

 2  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.5 priority=128 router-id=5.5.5.5 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=6m55s timeout=6s
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.2%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.20.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- We can also see the updates in Wireshark.

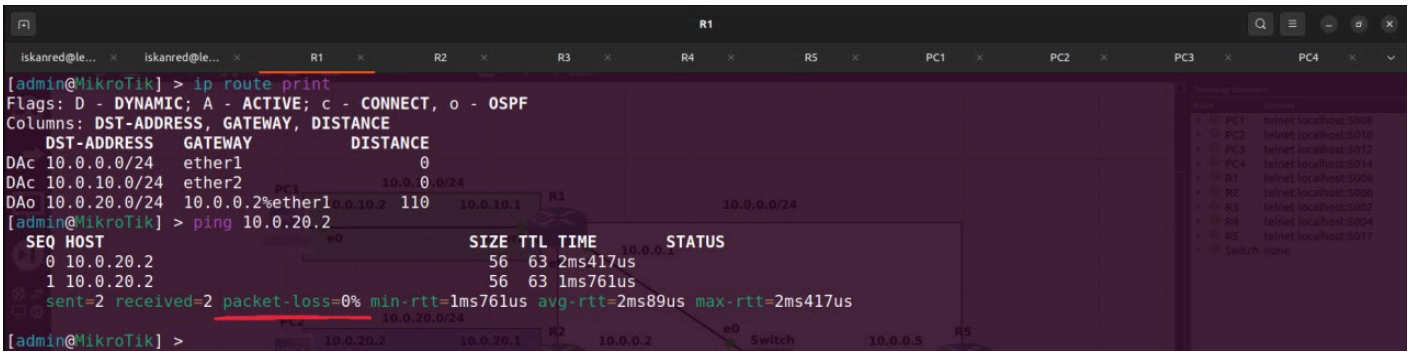


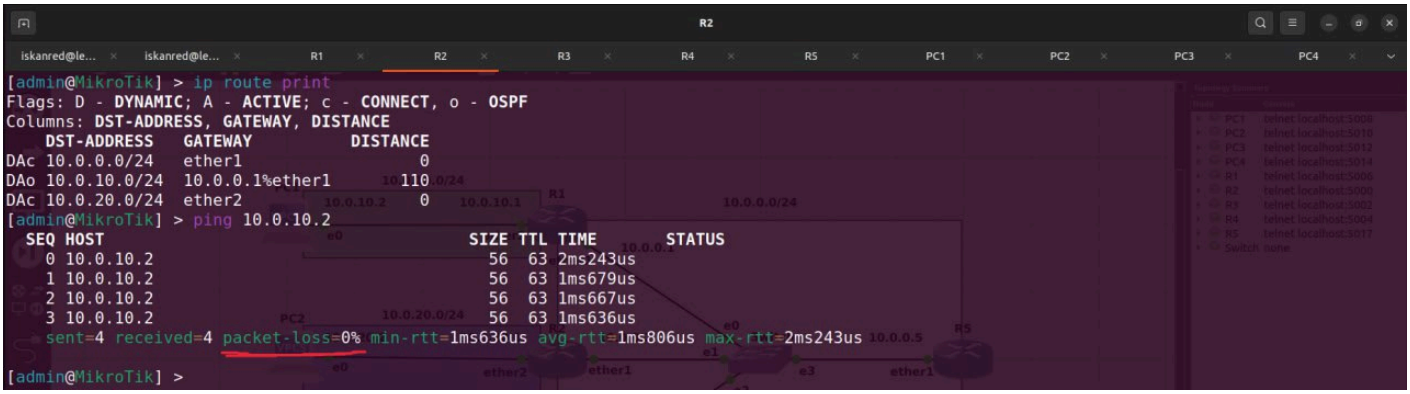
- Now, we can see that `R1` and `R2` became neighbours, while other routers "lost" them both.

```
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.2 priority=128 router-id=2.2.2.2 dr=10.0.0.1 bdr=10.0.0.2 state="Full"
      state-changes=6 adjacency=3m38s timeout=32s
[admin@MikroTik] /routing/ospf>
```



- And of course now we can reach both the routers from each other

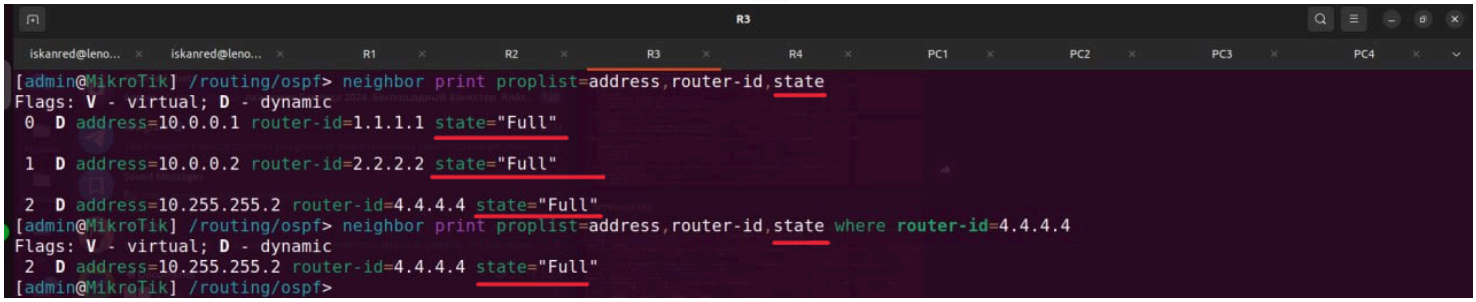




Task 3 - OSPF Verification

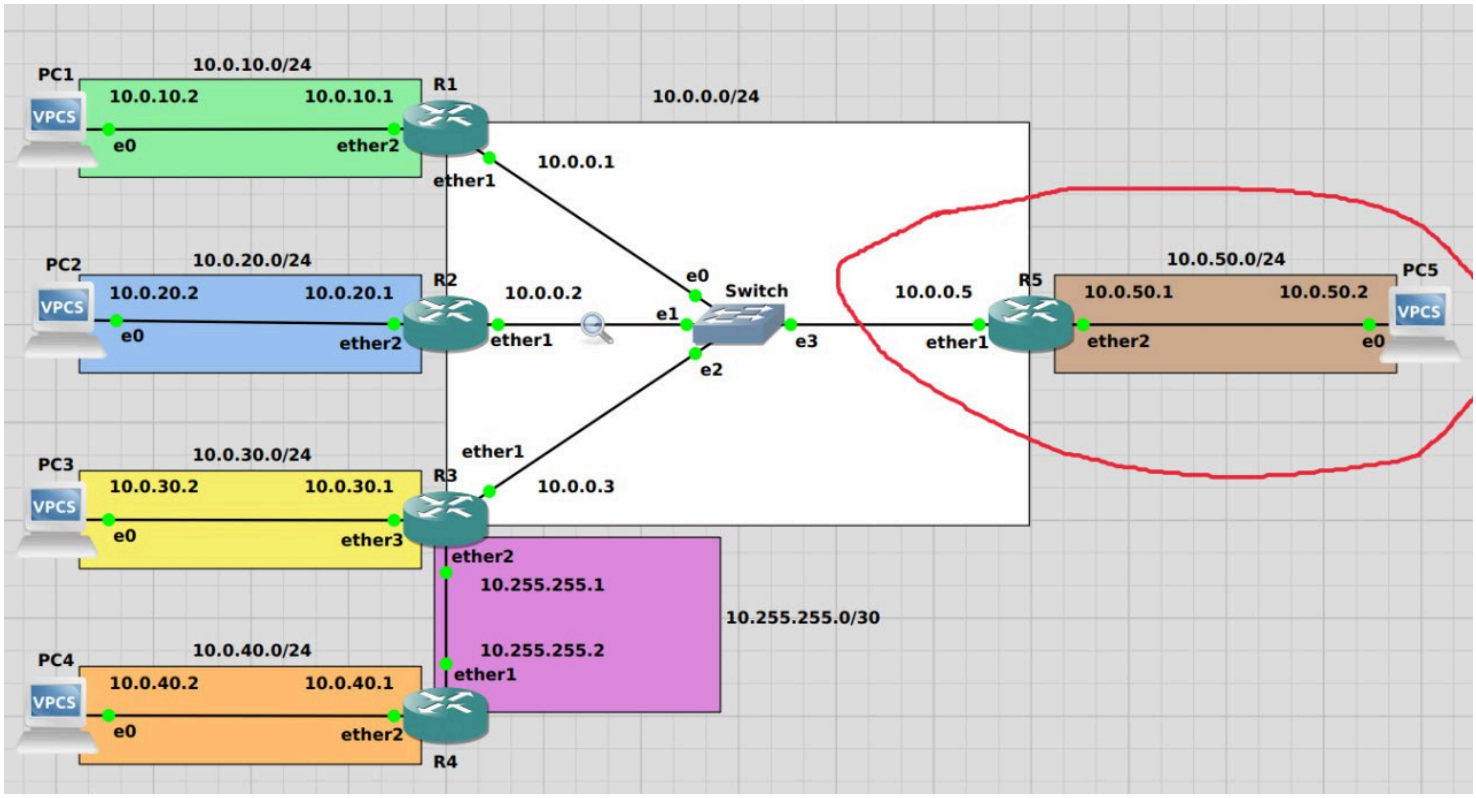
> **1.** `How can you check if you have a full adjacency with your router neighbor?`

- I can check it looking at neighbour's state. Let's see at the `R3` example:



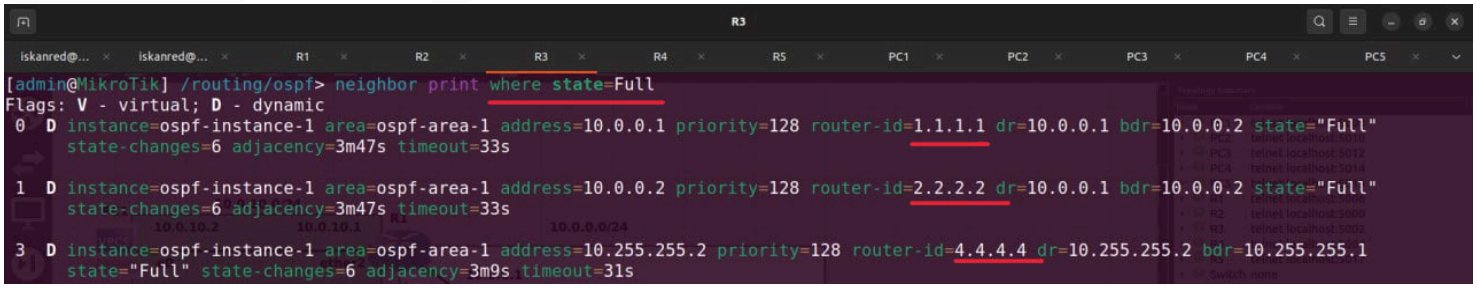
- I mentioned the state machine and possible state values above in the **Task 2**

- To make it more demonstrative, I added another one router `R5` (router-id= `5.5.5.5`) and PC `PC5`



- Then, I deployed OSPF to both `R5` 's interfaces

- After all we can see that `R3` has the same neighbours in `FULL` state: `R1` , `R2` in `10.0.0.0/24` , and `R4` in `10.255.255.0/30` . But where is `R5` ?

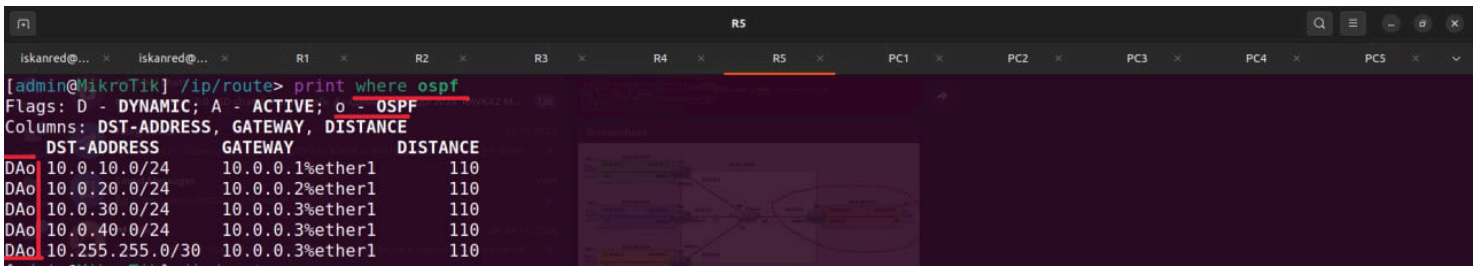


- And `R5` has indeed `TwoOnly` state from the `R3` 's perspective.

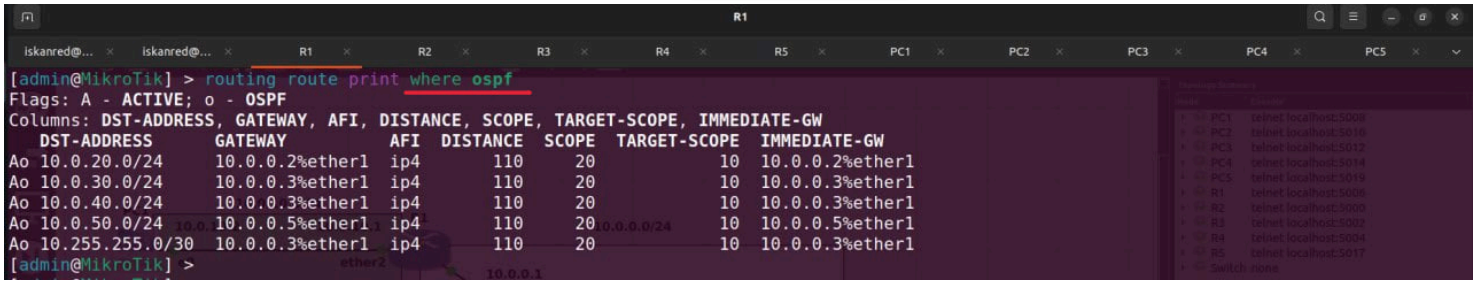
- This happened because `R5` and `R3` do not need to synchronise LSDB between each other. For this purpose the both have DR and BDR routers, `R1` and `R2` correspondingly

> **2.** `How can you check in the routing table which networks did you receive from your neighbors?`

- Since we receive all the updates for LSDB from our neighbours (and to be precise from DR or BDR) we can only check that entries of the routing table were generated by OSPF:

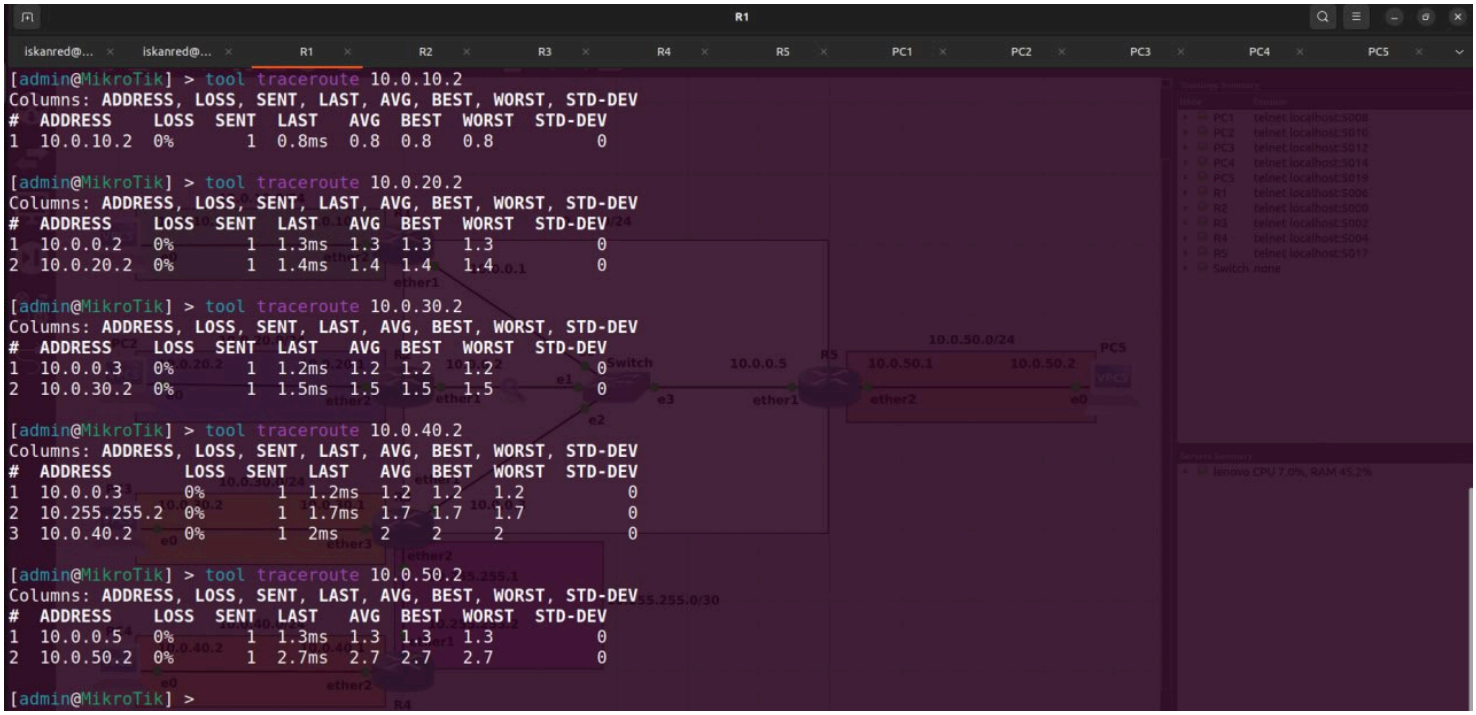


- Another way which is actually more detailed is to use `routing/route/` instead



> **3.** `Use traceroute to verify that you have a full OSPF network.`

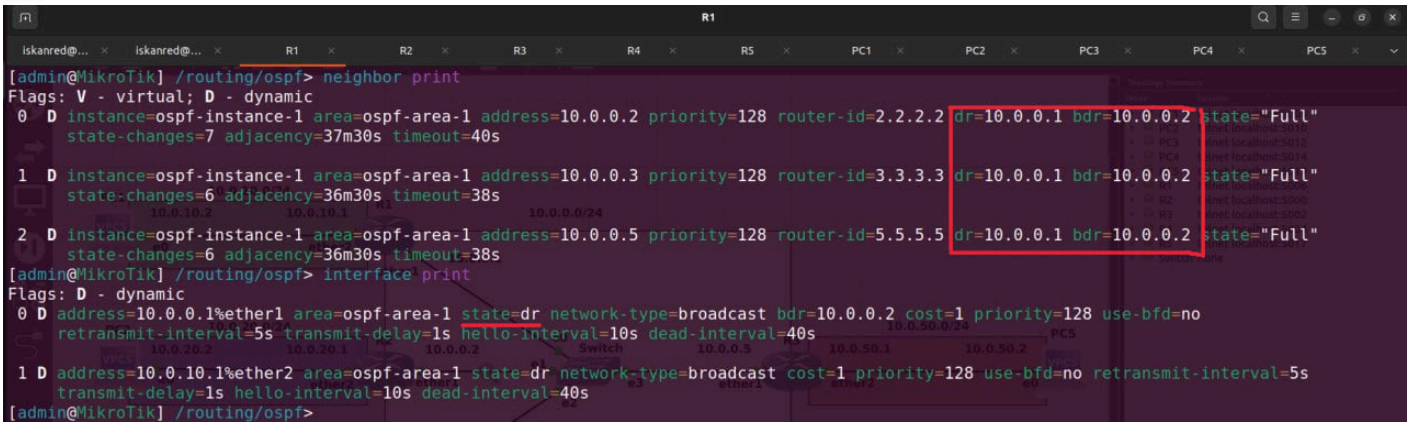
- I didn't get what does it mean to "have a full OSPF network", so I just performed `traceroute` from `R1` to every subnet



> **4.** `Which router is selected as DR and which one is BDR ?`

It really depends on the subnet we are considering, or to be more precise, **Multicast Link**.

- For example, in the `10.0.0.0/24` `R1` was selected as DR and `R2` as BDR
  - `R1`



- `R2`

```
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.1 priority=128 router-id=1.1.1.1 dr=10.0.0.1 bdr=10.0.0.2 state="Full"
      state-changes=6 adjacency=37m48s timeout=32s

 1  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.3 priority=128 router-id=3.3.3.3 dr=10.0.0.1 bdr=10.0.0.2 state="Full"
      state-changes=6 adjacency=36m48s timeout=31s

 2  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.5 priority=128 router-id=5.5.5.5 dr=10.0.0.1 bdr=10.0.0.2 state="Full"
      state-changes=6 adjacency=36m48s timeout=31s
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.2%ether1 area=ospf-area-1 state=bdr network-type=broadcast dr=10.0.0.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.20.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- However, inside the `10.255.255.0/30` `R4` was selected as DR and `R3` as BDR

```
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.255.255.1 priority=128 router-id=3.3.3.3 dr=10.255.255.2 bdr=10.255.255.1
      state="Full" state-changes=6 adjacency=46m51s timeout=30s
[admin@MikroTik] /routing/ospf> interface print where address=10.255.255.2%ether1
Flags: D - dynamic
 0 D address=10.255.255.2%ether1 area=ospf-area-1 state=dr network-type=broadcast bdr=10.255.255.1 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- Actually, there always an election process between routers in the same Multicast Link.
  1. Routers check if there are already a DR or BDR elected in their network. If yes, then they just accept it.
  2. If no, routers elect one of them with the **highest priority** as a DR.
  3. If all the priorities are identical, then to break the tie the **highest router-id** is used.
  4. The same election process is applied for BDR after DR is elected.
- So, that's why in the `10.0.0.0/24` `R1` and `R2` was DR and BDR correspondingly -- they were enabled first one by one, therefore, other routers just accepted that during exchanging `Hello` packets.
- However, if I restart my topology, I will get `R5` (router-id= `5.5.5.5`) as DR and `R3` as BDR (router-id= `3.3.3.3`) because all the routers have the same priority $= 128$ by default, but the highest router-ids belong to `R5` and `R4`
  - `R1` now in state `dr-other` in the `10.0.0.0/24` and among other things it is proven that is has `TwoWay` adjacency state with `R2` since thee both are not DR or BDR

```
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.2 priority=128 router-id=2.2.2.2 dr=10.0.0.5 bdr=10.0.0.3 state="TwoWay"
      state-changes=2 timeout=38s

 1  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.3 priority=128 router-id=3.3.3.3 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=4m51s timeout=39s

 2  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.5 priority=128 router-id=5.5.5.5 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=4m46s timeout=39s
[admin@MikroTik] /routing/ospf> interface print where address="10.0.0.1%ether1"
Flags: D - dynamic
 0 D address=10.0.0.1%ether1 area=ospf-area-1 state=dr-other network-type=broadcast dr=10.0.0.5 bdr=10.0.0.3 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

  - `R5` now is DR and it has `Full` adjacency with each router in the `10.0.0.0/24`

```
[admin@MikroTik] /routing/ospf> neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.1 priority=128 router-id=1.1.1.1 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=3m10s timeout=33s

 1  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.2 priority=128 router-id=2.2.2.2 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=3m15s timeout=34s

 2  D instance=ospf-instance-1 area=ospf-area-1 address=10.0.0.3 priority=128 router-id=3.3.3.3 dr=10.0.0.5 bdr=10.0.0.3 state="Full"
      state-changes=6 adjacency=3m15s timeout=35s
[admin@MikroTik] /routing/ospf> interface print where address="10.0.0.5%ether1"
Flags: D - dynamic
 0 D address=10.0.0.5%ether1 area=ospf-area-1 state=dr network-type=broadcast bdr=10.0.0.3 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf>
```

- And in Wireshark we can see the actual election process.

*- [Switch Ethernet3 to R5 ether1]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ospf

No.	Time	Source	Destination	Protocol	Length	Info
44	10.403292	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
46	10.604686	10.0.0.2	224.0.0.5	OSPF	90	Hello Packet
47	10.886164	10.0.0.3	224.0.0.5	OSPF	90	Hello Packet
52	11.728308	10.0.0.5	224.0.0.5	OSPF	90	Hello Packet
66	20.411106	10.0.0.1	224.0.0.5	OSPF	90	Hello Packet
67	20.602456	10.0.0.2	224.0.0.5	OSPF	90	Hello Packet
68	20.883783	10.0.0.3	224.0.0.5	OSPF	90	Hello Packet
69	21.735353	10.0.0.5	224.0.0.5	OSPF	90	Hello Packet
78	30.421037	10.0.0.1	224.0.0.5	OSPF	90	Hello Packet
83	30.601994	10.0.0.2	224.0.0.5	OSPF	90	Hello Packet
88	30.883335	10.0.0.3	224.0.0.5	OSPF	90	Hello Packet
93	31.745328	10.0.0.5	224.0.0.5	OSPF	90	Hello Packet
105	40.431291	10.0.0.1	224.0.0.5	OSPF	90	Hello Packet
107	40.603050	10.0.0.2	224.0.0.5	OSPF	90	Hello Packet
109	40.604522	10.0.0.2	10.0.0.5	OSPF	66	DB Description
111	40.885139	10.0.0.3	224.0.0.5	OSPF	90	Hello Packet
113	40.886211	10.0.0.3	10.0.0.5	OSPF	66	DB Description
115	41.728832	10.0.0.5	10.0.0.2	OSPF	66	DB Description
116	41.728958	10.0.0.5	10.0.0.3	OSPF	66	DB Description
117	41.729202	10.0.0.5	224.0.0.5	OSPF	90	Hello Packet
118	41.729949	10.0.0.2	10.0.0.5	OSPF	86	DB Description
120	41.730094	10.0.0.3	10.0.0.5	OSPF	86	DB Description
123	41.731071	10.0.0.5	10.0.0.1	OSPF	66	DB Description
125	41.731394	10.0.0.3	224.0.0.5	OSPF	90	Hello Packet
126	41.731523	10.0.0.5	10.0.0.2	OSPF	86	DB Description
127	41.731901	10.0.0.5	10.0.0.2	OSPF	70	LS Request
128	41.732225	10.0.0.5	10.0.0.3	OSPF	86	DB Description
129	41.732306	10.0.0.5	10.0.0.3	OSPF	70	LS Request
130	41.733000	10.0.0.2	10.0.0.5	OSPF	66	DB Description
131	41.733095	10.0.0.2	10.0.0.5	OSPF	70	LS Request
132	41.733193	10.0.0.2	10.0.0.5	OSPF	110	LS Update
133	41.733244	10.0.0.3	10.0.0.5	OSPF	66	DB Description
134	41.733836	10.0.0.3	10.0.0.5	OSPF	70	LS Request
135	41.734042	10.0.0.3	10.0.0.5	OSPF	122	LS Update
136	41.734103	10.0.0.1	10.0.0.5	OSPF	66	DB Description
137	41.734499	10.0.0.5	10.0.0.2	OSPF	110	LS Update
138	41.735235	10.0.0.5	224.0.0.5	OSPF	142	LS Update
139	41.735519	10.0.0.5	10.0.0.3	OSPF	110	LS Update
140	41.735945	10.0.0.2	10.0.0.5	OSPF	78	LS Acknowledge
145	42.229085	10.0.0.2	224.0.0.6	OSPF	118	LS Acknowledge
146	42.241825	10.0.0.5	224.0.0.5	OSPF	78	LS Acknowledge
147	42.242074	10.0.0.1	224.0.0.6	OSPF	138	LS Acknowledge
148	42.242187	10.0.0.3	224.0.0.5	OSPF	118	LS Acknowledge

  - What is interesting here is that `R1` and `R2` send `LS Ack` packets to `224.0.0.6` that is the Multicast IP address for OSPF group containing DR and BDR routers only.
  - At the same time, both `R3` and `R3` send `LS Ack` to `224.0.0.5` that is Multicast IP address for OSPF group containing all the routers in a current Multicast Link.

> **5.** `Check what is the cost for each network that has been received by OSPF in the routing table.`

- Let's check OSPF path cost which is called precisely `ospf.metric` on `R1`

R1

```
A H ether2                                link
[admin@MikroTik] > routing route print proplist=dst-address,gateway,afi,ospf.metric,immediate-gw where ospf
Flags: A - ACTIVE; o - OSPF
Columns: DST-ADDRESS, GATEWAY, AFI, OSPF.METRIC, IMMEDIATE-GW
   DST-ADDRESS      GATEWAY           AFI  OSPF.METRIC  IMMEDIATE-GW
Ao 10.0.20.0/24     10.0.0.2%ether1   ip4            2  10.0.0.2%ether1
Ao 10.0.30.0/24     10.0.0.3%ether1   ip4            2  10.0.0.3%ether1
Ao 10.0.40.0/24     10.0.0.3%ether1   ip4            3  10.0.0.3%ether1
Ao 10.0.50.0/24     10.0.0.5%ether1   ip4            2  10.0.0.5%ether1
Ao 10.255.255.0/30  10.0.0.3%ether1   ip4            2  10.0.0.3%ether1
[admin@MikroTik] >
```

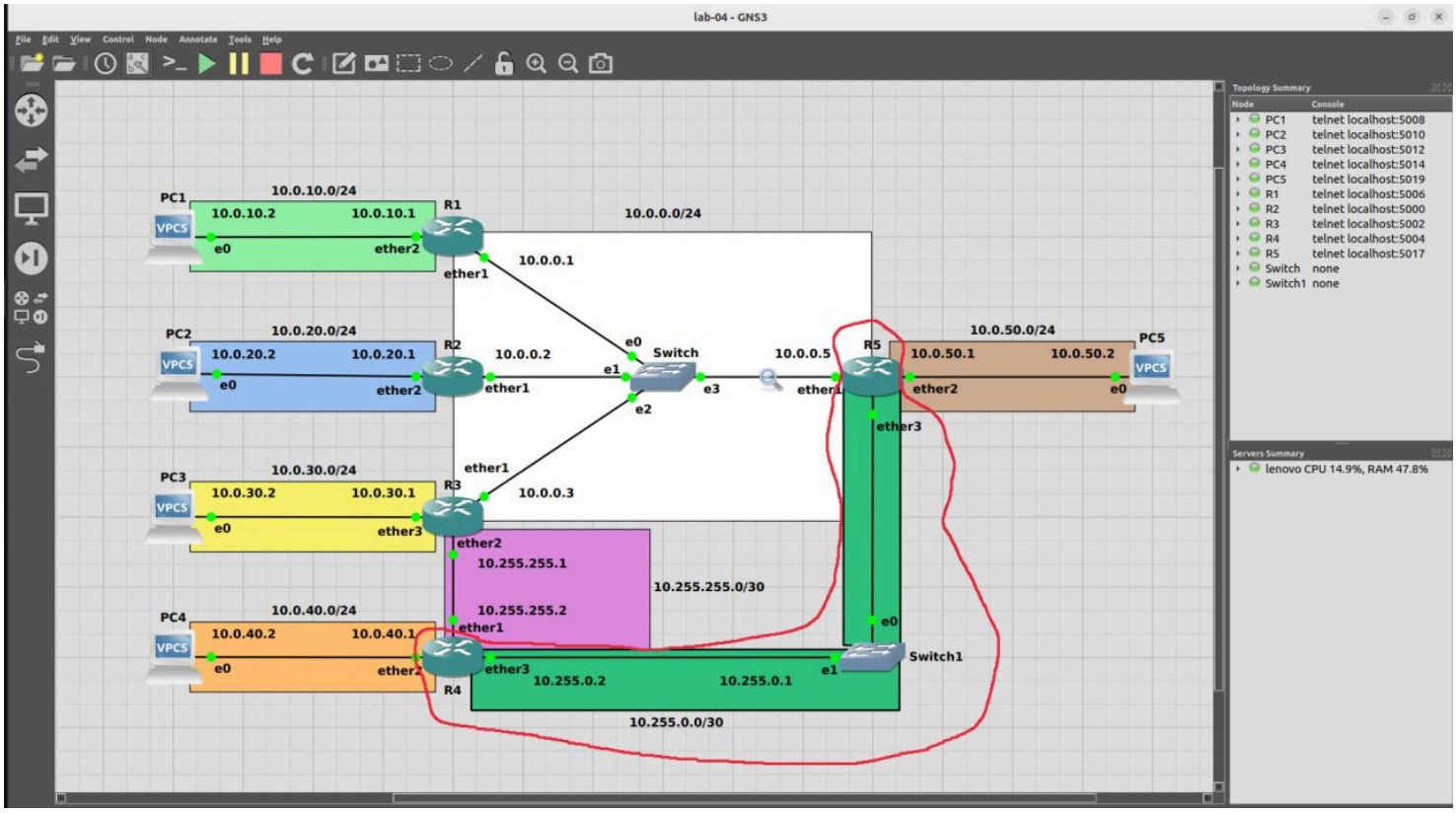
- By default, each link has a link with an OSPF cost $= 1$ in Mirkotik devices. Therefore, the cost from `R1` to `R4`'s subnet is 3 while others have only 4. In such a case cost are identical to the hops count. Since `R4` is separated from other routers by `R3`, the cost of path to it is 3 which is really `R1` -> `R3` -> `R4`
- Now, let's check this metric from `R5` to `R4`

R5

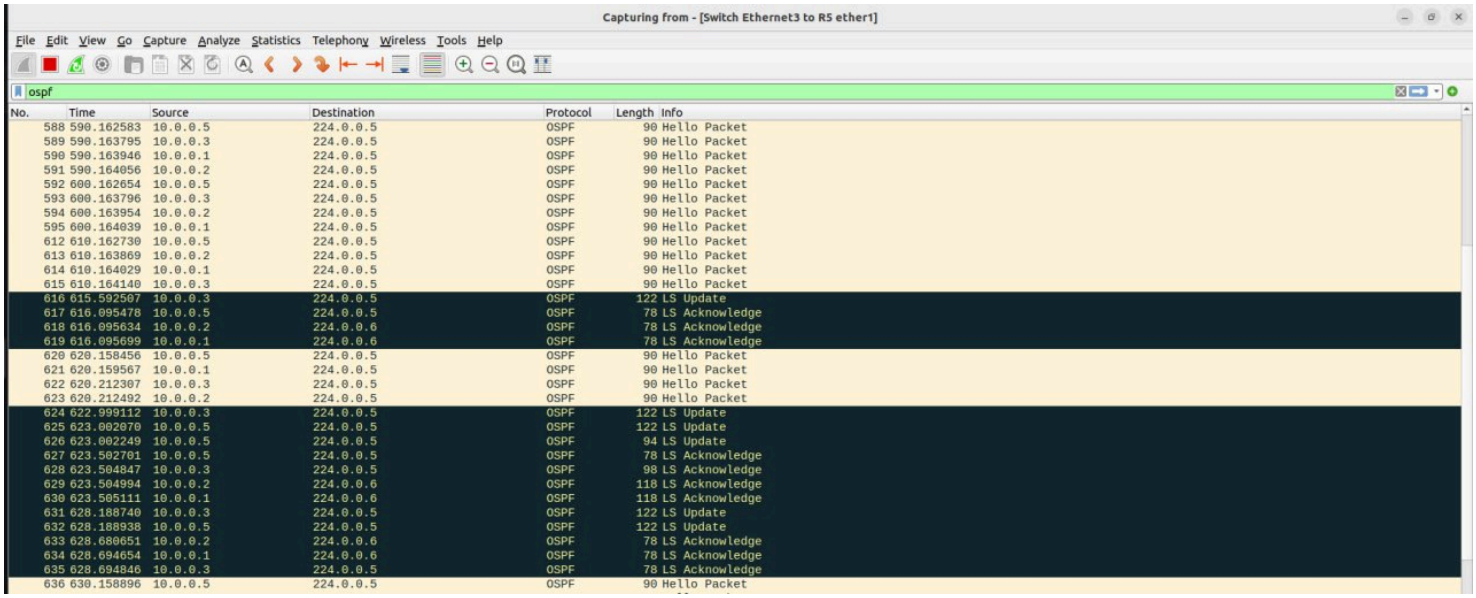
```
[admin@MikroTik] > routing route print proplist=dst-address,gateway,afi,distance,ospf.metric,immediate-gw where ospf and dst-address=10.0.40.
0/24
Flags: A - ACTIVE; o - OSPF
Columns: DST-ADDRESS, GATEWAY, AFI, DISTANCE, OSPF.METRIC, IMMEDIATE-GW
   DST-ADDRESS    GATEWAY           AFI  DISTANCE  OSPF.METRIC  IMMEDIATE-GW
Ao 10.0.40.0/24   10.0.0.3%ether1   ip4       110            3  10.0.0.3%ether1
[admin@MikroTik] >
```

- We can see that the cost to `R4`'s subnet `10.0.40.0/24` is exactly the same, which equals to 3

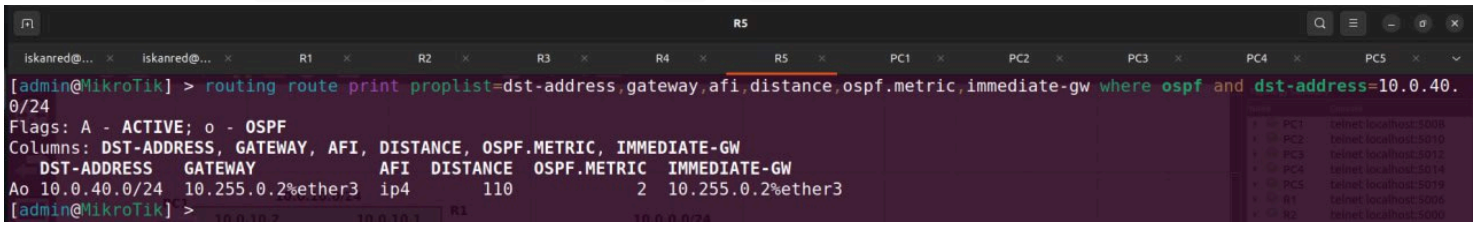
- Now let's add another change to my topology:



- We can see LSDB updates in Wireshark



- And let's check the `ospf.metric` for the path from `R5` to `R4`

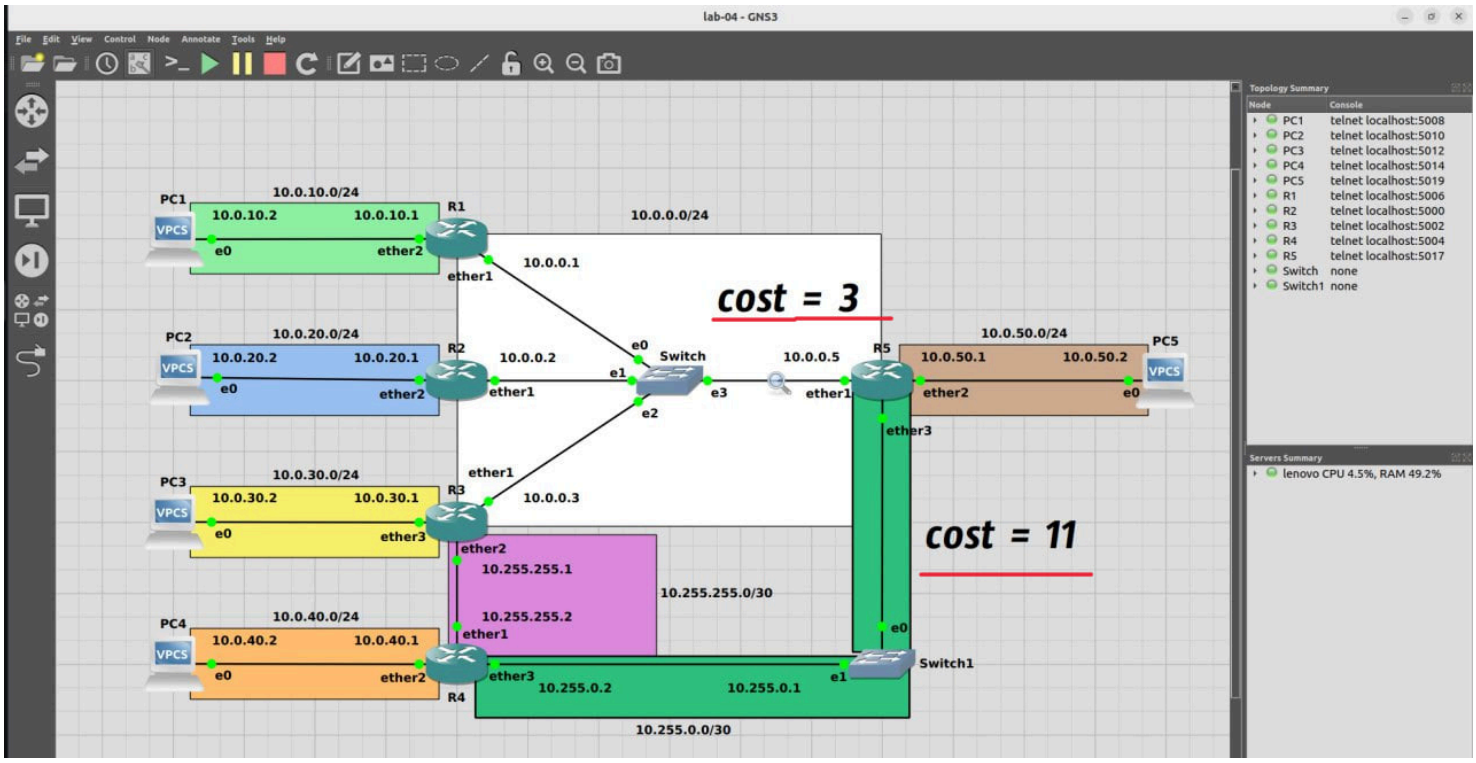


- Now we see that it equals to 2 because hops count now is 2: `R5` -> `R4`

- But what if change cost to 10 instead of 1 for the interface on `R5` for the link `R5` -- `R4`

```
[admin@MikroTik] /routing/ospf> interface-template print
Flags: X - disabled, I - inactive
 0   area=ospf-area-1 interfaces=ether1 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1

 1   area=ospf-area-1 interfaces=ether2 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1

 2   area=ospf-area-1 interfaces=ether3 instance-id=0 type=broadcast retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@MikroTik] /routing/ospf> interface-template set numbers=2 cost=10
[admin@MikroTik] /routing/ospf> interface print
Flags: D - dynamic
 0 D address=10.0.0.5%ether1 area=ospf-area-1 state=dr network-type=broadcast bdr=10.0.0.3 cost=1 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s

 1 D address=10.0.50.1%ether2 area=ospf-area-1 state=dr network-type=broadcast cost=1 priority=128 use-bfd=no retransmit-interval=5s
     transmit-delay=1s hello-interval=10s dead-interval=40s

 2 D address=10.255.0.1%ether3 area=ospf-area-1 state=dr network-type=broadcast bdr=10.255.0.2 cost=10 priority=128 use-bfd=no
     retransmit-interval=5s transmit-delay=1s hello-interval=10s dead-interval=40s
[admin@MikroTik] /routing/ospf> ../../routing route print proplist=dst-address,gateway,afi,distance,ospf.metric,immediate-gw where ospf and d
st-address=10.0.40.0/24
Flags: A - ACTIVE; o - OSPF
Columns: DST-ADDRESS, GATEWAY, AFI, DISTANCE, OSPF.METRIC, IMMEDIATE-GW
   DST-ADDRESS    GATEWAY         AFI  DISTANCE  OSPF.METRIC  IMMEDIATE-GW
Ao 10.0.40.0/24   10.0.0.3%ether1 ip4       110            3  10.0.0.3%ether1
[admin@MikroTik] /routing/ospf>
```

- We see that now `ospf.metric` equals to 3 for the path between `R5` and `R5`. This is because OSPF helped `R5` to help more efficient path with the less cost



- The path from `R5` to `R4` that lays through the `Switch 1` costs 11 in total. However, the most efficient path lays through the `Swtich` and `R3` which costs 3 in total. OSPF protocol uses Dijkstra's algorithm for the purpose of selecting the route.

References

- https://help.mikrotik.com/docs/spaces/ROS/pages/9863229/OSPF#OSPF-PropertyReference
- https://www.practicalnetworking.net/stand-alone/ospf-training-course-free-m1/

- https://www.firewall.cx/networking/routing-protocols/ospf-adjacency-neighbor-states-forming-process.html