University of Porto
Faculdade de Engenharia

Protocols OSPF and BGP4

Lab work 4

Alena Tesařová
Rodrigo Caldas
Tiago Ribeiro

May 24, 2020

Contents

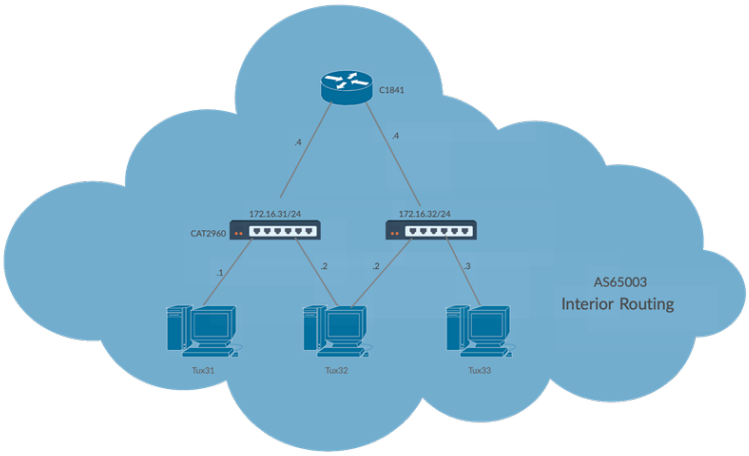
1 Introduction

The aim of this work is to understand clearly, using a case study, the concepts of Autonomous System (AS), Routing Interior and Exterior Routing Protocols respective examples of OSPF and BGP4.

2 Configuration

All created configuration can be found in Annexes.

3 Interior routing



1. *Present the routing tables of all systems running OSPF.*

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.31.4, eth1, 17:47:07
O   172.16.31.0/24 [110/10] is directly connected, eth1, 17:47:08
O>* 172.16.32.0/24 [110/20] via 172.16.31.4, eth1, 17:47:08
  *                         via 172.16.31.2, eth1, 17:47:08
```

(a) TUX 31

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.31.4, eth1, 5d15h29m
                      via 172.16.32.4, eth2, 5d15h29m
O   172.16.31.0/24 [110/10] is directly connected, eth1, 18:12:16
O   172.16.32.0/24 [110/10] is directly connected, eth2, 5d20h01m
```

(b) TUX 32

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.32.4, eth1, 5d17h45m
O>* 172.16.31.0/24 [110/20] via 172.16.32.2, eth1, 18:14:50
  *                         via 172.16.32.4, eth1, 18:14:50
O   172.16.32.0/24 [110/10] is directly connected, eth1, 5d20h04m
```

(c) TUX 33

Figure 1: All the computers running OSPF

```
tux-rtr3#show ip ospf route

            OSPF Router with ID (172.16.31.4) (Process ID 1)


                Base Topology (MTID 0)


    Area BACKBONE(0.0.0.0)

    Intra-area Route List
*   172.16.31.0/24, Intra, cost 10, area 0.0.0.0, Connected
      via 172.16.31.4, GigabitEthernet0/1.2
*   172.16.32.0/24, Intra, cost 10, area 0.0.0.0, Connected
      via 172.16.32.4, GigabitEthernet0/1.3
```

Figure 2: Cisco router running OSPF

2. *Simulate a fault in the circuit linking the GNUy2 to the network it shares with the GNUy1. Check the resulting changes in the routing tables of the multiple routers.*

Firstly we let the OSPF protocol run, exchange information about the routes and then all TUX [1-3] learn the path to each other, through TUX2 or through the Cisco router OSPF.

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.31.4, eth1, 17:47:07
O   172.16.31.0/24 [110/10] is directly connected, eth1, 17:47:08
O>* 172.16.32.0/24 [110/20] via 172.16.31.4, eth1, 17:47:08
  *                         via 172.16.31.2, eth1, 17:47:08
```

(a) TUX 31

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.31.4, eth1, 5d15h29m
                      via 172.16.32.4, eth2, 5d15h29m
O   172.16.31.0/24 [110/10] is directly connected, eth1, 18:12:16
O   172.16.32.0/24 [110/10] is directly connected, eth2, 5d20h01m
```

(b) TUX 32

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.32.4, eth1, 5d17h45m
O>* 172.16.31.0/24 [110/20] via 172.16.32.2, eth1, 18:14:50
  *                         via 172.16.32.4, eth1, 18:14:50
O   172.16.32.0/24 [110/10] is directly connected, eth1, 5d20h04m
```

(c) TUX 33

Figure 3: Routing tables before the fault in the circuit

Then as was asked, the connection between TUX32 and TUX31 was cut and what happened was that TUX31 is no longer able to use TUX32 to reach TUX33 and started using the Cisco router OSPF connection exclusively to reach both TUX32 and TUX33. Other changes were that TUX32 and TUX33 can only communicate now with TUX31 through the Cisco router OSPF.

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.31.4, eth1, 18:08:54
O   172.16.31.0/24 [110/10] is directly connected, eth1, 18:08:55
O>* 172.16.32.0/24 [110/20] via 172.16.31.4, eth1, 00:00:14
```

(a) TUX 31

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.32.4, eth2, 00:00:18
O>* 172.16.31.0/24 [110/20] via 172.16.32.4, eth2, 00:00:19
O   172.16.32.0/24 [110/10] is directly connected, eth2, 5d20h20m
```

(b) TUX 32

```
quagga-router# show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

O   0.0.0.0/0 [110/1] via 172.16.32.4, eth1, 5d18h01m
O>* 172.16.31.0/24 [110/20] via 172.16.32.4, eth1, 00:00:20
O   172.16.32.0/24 [110/10] is directly connected, eth1, 5d20h20m
```

(c) TUX 33

Figure 4: Routing tables after the fault in the circuit

3. *Present the results of the **traceroute** from the system TUX31 to TUX32.*

   Here we decided to present the two traceroute options:

   - from TUX31 to TUX32 directly attached to eth1 **(5a)**
   - from TUX31 to TUX32 via the Cisco router OSPF **(5b)**

```
tux31:~# traceroute 172.16.31.2
traceroute to 172.16.31.2 (172.16.31.2), 30 hops max, 60 byte packets
 1  172.16.31.2 (172.16.31.2)  0.155 ms  0.152 ms  0.147 ms
```

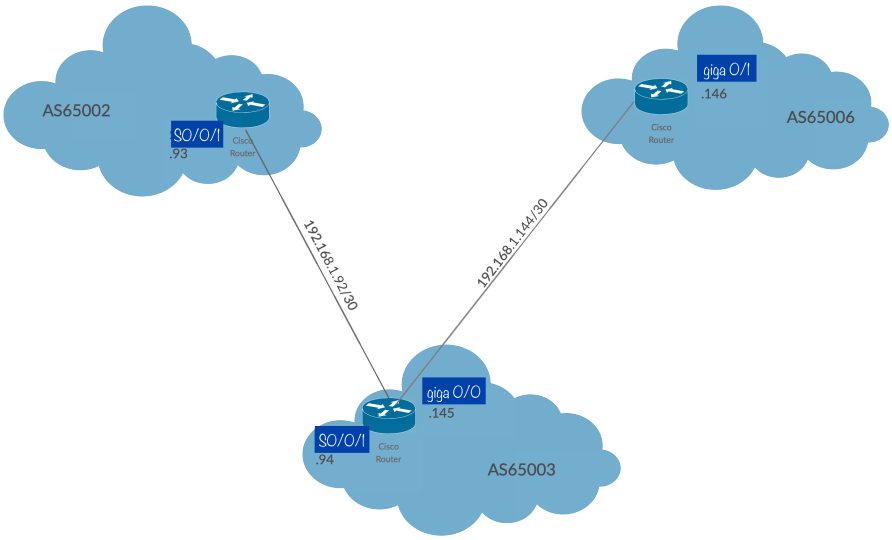
(a) via eth1

```
tux31:~# traceroute 172.16.32.2
traceroute to 172.16.32.2 (172.16.32.2), 30 hops max, 60 byte packets
 1  172.16.31.4 (172.16.31.4)  0.502 ms  0.545 ms  0.577 ms
 2  172.16.32.2 (172.16.32.2)  0.339 ms  0.350 ms  0.339 ms
```

(b) via Cisco router

Figure 5: Traceroute results from TUX31 to TUX32

4 Exterior routing



1. *Present the routing table of the ABR of each AS.*

```
tux-rtr3#show bgp
BGP table version is 80, local router ID is 192.168.0.238
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
r> 172.16.1.0/24    192.168.1.93             0             40 65002 ?
*> 172.16.11.0/24   192.168.1.93                           40 65002 65001 i
*> 172.16.12.0/24   192.168.1.93                           40 65002 65001 i
*> 172.16.21.0/24   192.168.1.93             0             40 65002 i
*> 172.16.22.0/24   192.168.1.93             0             40 65002 i
*> 172.16.31.0/24   0.0.0.0                  0         32768 i
*> 172.16.32.0/24   0.0.0.0                  0         32768 i
*> 172.16.41.0/24   192.168.1.93                           40 65002 65004 i
*> 172.16.42.0/24   192.168.1.93                           40 65002 65004 i
*> 172.16.51.0/24   192.168.1.93                           40 65002 65004 65005 i
*> 172.16.52.0/24   192.168.1.93                           40 65002 65004 65005 i
*> 192.168.1.48/30  192.168.1.93             0             40 65002 ?
r> 192.168.1.92/30  192.168.1.93             0             40 65002 ?
*> 192.168.1.96/30  192.168.1.93             0             40 65002 ?
```

Figure 6: BGP Peering

2. *Present the results of traceroute between the systems GNUy3 of an AS with the correspondents of the neighboring ASes and configure your BGP routing process to ensure that your AS's traffic with neighbors is done using the neighbor with the highest AS number.*

- **Connectivity results between AS65003 and AS65002**

```
root@tux33:~# ping 172.16.21.1
PING 172.16.21.1 (172.16.21.1) 56(84) bytes of data.
64 bytes from 172.16.21.1: icmp_seq=1 ttl=62 time=1.22 ms
64 bytes from 172.16.21.1: icmp_seq=2 ttl=62 time=1.19 ms
64 bytes from 172.16.21.1: icmp_seq=3 ttl=62 time=1.18 ms
64 bytes from 172.16.21.1: icmp_seq=5 ttl=62 time=1.22 ms
64 bytes from 172.16.21.1: icmp_seq=7 ttl=62 time=1.21 ms
```

(a) ping 172.16.21.1

```
root@tux33:~# traceroute 172.16.21.1
traceroute to 172.16.21.1 (172.16.21.1), 30 hops max, 60 byte packets
 1  172.16.32.4 (172.16.32.4)  0.660 ms  0.689 ms  0.729 ms
 2  192.168.1.93 (192.168.1.93)  0.946 ms  1.170 ms  1.448 ms
 3  172.16.21.1 (172.16.21.1)  1.902 ms * *
```

(b) traceroute 172.16.21.1

Figure 7: Results from tux33 to tux21

```
root@tux33:~# ping 172.16.22.3
PING 172.16.22.3 (172.16.22.3) 56(84) bytes of data.
64 bytes from 172.16.22.3: icmp_seq=2 ttl=62 time=1.25 ms
64 bytes from 172.16.22.3: icmp_seq=4 ttl=62 time=1.21 ms
64 bytes from 172.16.22.3: icmp_seq=6 ttl=62 time=1.22 ms
64 bytes from 172.16.22.3: icmp_seq=8 ttl=62 time=1.22 ms
64 bytes from 172.16.22.3: icmp_seq=9 ttl=62 time=1.20 ms
64 bytes from 172.16.22.3: icmp_seq=10 ttl=62 time=1.20 ms
64 bytes from 172.16.22.3: icmp_seq=11 ttl=62 time=1.21 ms
64 bytes from 172.16.22.3: icmp_seq=12 ttl=62 time=1.22 ms
```

(a) ping 172.16.22.3

```
root@tux33:~# traceroute 172.16.22.3
traceroute to 172.16.22.3 (172.16.22.3), 30 hops max, 60 byte packets
 1  172.16.32.4 (172.16.32.4)  0.638 ms  0.673 ms  0.715 ms
 2  192.168.1.93 (192.168.1.93)  0.930 ms  1.153 ms  1.443 ms
 3  172.16.22.3 (172.16.22.3)  1.906 ms * *
```

(b) traceroute 172.16.22.3

Figure 8: Results from tux33 to tux23

- **Connectivity results between AS65003 and AS65006:**

It was not possible to elaborate because the group responsible for bench 6 did not make the necessary BGP configurations.
In short, what was supposed to be observed would be that traffic with neighbors would necessarily pass through the neighbor with the highest AS in this case AS65006, as we can see that it was configured in the router's BGP (see annex) where AS65006 has a higher weight because more weight means that the algorithm will choose first.

A Annexes

A.1 Switch configuration

Listing 1: Creating 2 VLANS

```
tux-sw3#conf t
tux-sw3(config)#vlan 2
tux-sw3(config-vlan)#name 31 #VLAN - TUX31 AND TUX32
tux-sw3(config-vlan)#exit
tux-sw3(config)#vlan 3
tux-sw3(config-vlan)#name 32 #VLAN - TUX32 AND TUX33
tux-sw3(config-vlan)#exit
tux-sw3(config)#inter Fa0/2
tux-sw3(config-if)#swi
tux-sw3(config-if)#switchport acc vlan 2
tux-sw3(config-if)#exit
tux-sw3(config)#inter Fa0/4
tux-sw3(config-if)#switchport acc vlan 2
tux-sw3(config-if)#exit
tux-sw3(config)#inter Fa0/5
tux-sw3(config-if)#switchport acc vlan 3
tux-sw3(config-if)#exit
tux-sw3(config)#inter Fa0/7
tux-sw3(config-if)#switchport acc vlan 3
tux-sw3(config-if)#end
```

A.2 Configs of the various tuxys with OSPF "Quagga"

TUX 31

```
root@tux31:~# cat /etc/quagga/ospfd.conf
! Zebra configuration saved from vty
!   2020/05/14 17:54:56
hostname ospfd
password zebra
enable password zebra
service advanced-vty
!
interface dummy0
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 network 172.16.31.0/24 area 0.0.0.0
!
line vty
!
```

TUX 32

```
root@tux32:~# cat /etc/quagga/ospfd.conf
! Zebra configuration saved from vty
! 2020/05/14 18:13:52
hostname ospfd
password zebra
enable password zebra
service advanced-vty
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 network 172.16.31.0/24 area 0.0.0.0
 network 172.16.32.0/24 area 0.0.0.0
!
line vty
!
```

TUX 33

```
root@tux33:~# cat /etc/quagga/ospfd.conf
!
! Zebra configuration saved from vty
! 2020/05/14 18:03:25
!
hostname ospfd
password zebra
enable password zebra
service advanced-vty
!
!
!
interface eth0
!
interface eth1
!
interface eth2
!
interface lo
!
router ospf
 network 172.16.32.0/24 area 0.0.0.0
!
line vty
!
```

A.3 Configs Router Cisco (OSPF + BGP)

```
rtr(config)#router ospf 1
rtr(config-router)#router-id 172.16.31.4
rtr(config-router)#log-adjacency-changes
rtr(config-router)# no auto-cost
rtr(config-router)#network 172.16.31.0 0.0.0.255 area 0.0.0.0
rtr(config-router)#network 172.16.32.0 0.0.0.255 area 0.0.0.0
rtr(config-router)#default-information originate always

rtr(config)#router bgp 65003
rtr(config-router)#no synchronization
rtr(config-router)# neighbor 192.168.1.93 remote-as 65002
rtr(config-router)# neighbor 192.168.1.93 weight 40
rtr(config-router)# neighbor 192.168.1.146 remote-as 65006
rtr(config-router)# neighbor 192.168.1.146 weight 100
rtr(config-router)# address-family ipv4 unicast
```