HY-335

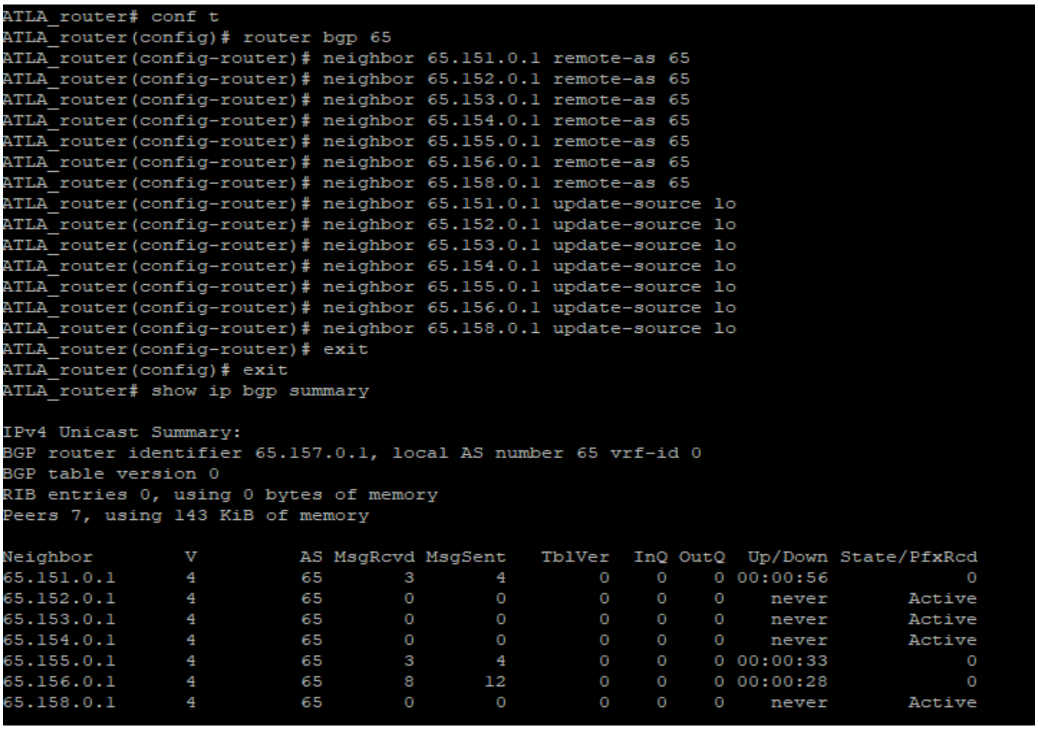
Project Phase-B

Lymperis Lymperidis csd4813

Giorgos Mathioudakis csd4674

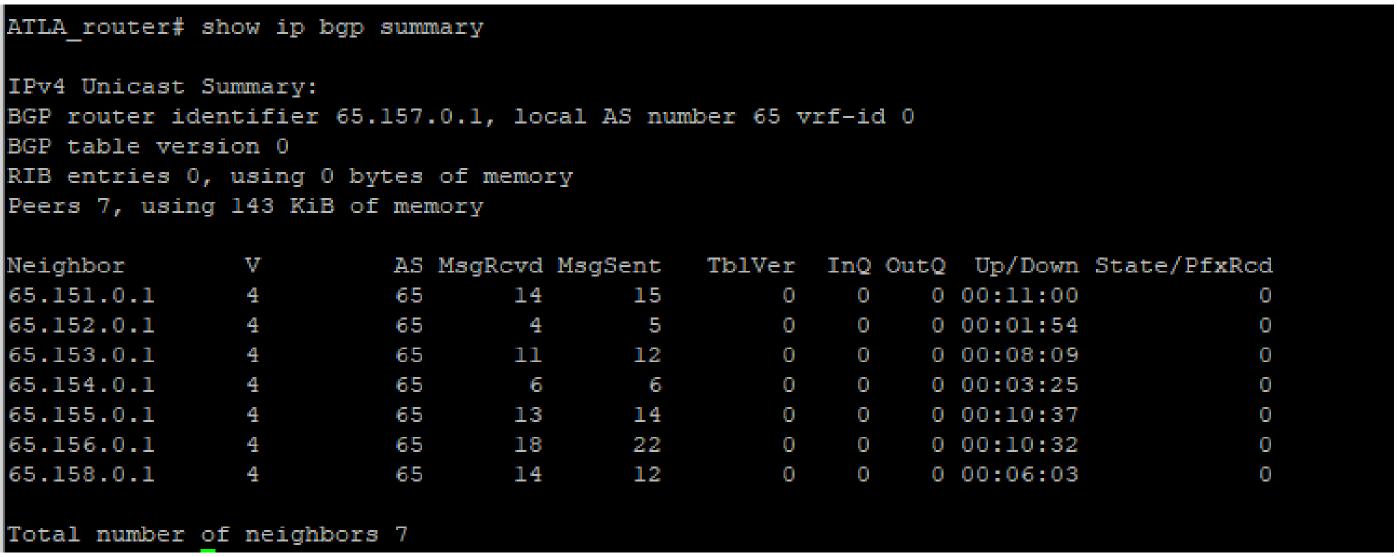
Question 2.1

In the first part of the project we must configure internal BGP sessions(iBGP) between all pair of routers. We need to go every router and establish connection with every other router configuring the BGP with all the neighbours (in ATLA we had already some other routers configured).



```
ATLA_router# conf t
ATLA_router(config)# router bgp 65
ATLA_router(config-router)# neighbor 65.151.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.152.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.153.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.154.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.155.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.156.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.158.0.1 remote-as 65
ATLA_router(config-router)# neighbor 65.151.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.152.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.153.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.154.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.155.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.156.0.1 update-source lo
ATLA_router(config-router)# neighbor 65.158.0.1 update-source lo
ATLA_router(config-router)# exit
ATLA_router(config)# exit
ATLA_router# show ip bgp summary

IPv4 Unicast Summary:
BGP router identifier 65.157.0.1, local AS number 65 vrf-id 0
BGP table version 0
RIB entries 0, using 0 bytes of memory
Peers 7, using 143 KiB of memory

Neighbor        V         AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down State/PfxRcd
65.151.0.1      4         65       3       4        0    0    0 00:00:56            0
65.152.0.1      4         65       0       0        0    0    0    never       Active
65.153.0.1      4         65       0       0        0    0    0    never       Active
65.154.0.1      4         65       0       0        0    0    0    never       Active
65.155.0.1      4         65       3       4        0    0    0 00:00:33            0
65.156.0.1      4         65       8      12        0    0    0 00:00:28            0
65.158.0.1      4         65       0       0        0    0    0    never       Active
```

This is the output of the show ip bgp summary in ATLA



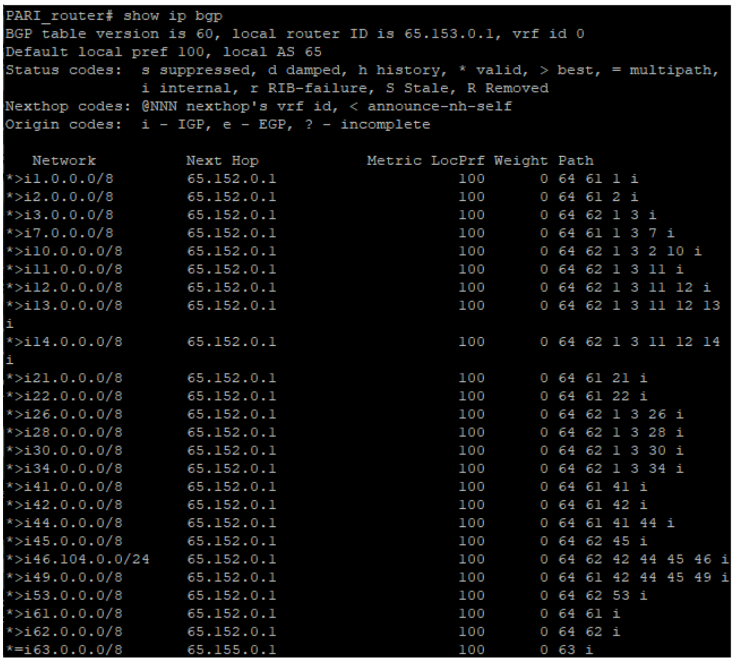
```
ATLA_router# show ip bgp summary

IPv4 Unicast Summary:
BGP router identifier 65.157.0.1, local AS number 65 vrf-id 0
BGP table version 0
RIB entries 0, using 0 bytes of memory
Peers 7, using 143 KiB of memory

Neighbor        V         AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down State/PfxRcd
65.151.0.1      4         65      14      15        0    0    0 00:11:00            0
65.152.0.1      4         65       4       5        0    0    0 00:01:54            0
65.153.0.1      4         65      11      12        0    0    0 00:08:09            0
65.154.0.1      4         65       6       6        0    0    0 00:03:25            0
65.155.0.1      4         65      13      14        0    0    0 00:10:37            0
65.156.0.1      4         65      18      22        0    0    0 00:10:32            0
65.158.0.1      4         65      14      12        0    0    0 00:06:03            0

Total number of neighbors 7
```

The update-source command in BGP configuration is crucial for specifying which interface or address should be used as the source IP address for BGP sessions. The main reason we use the loopback interface is the fact that this interface is always up and running and gives stability and reliability.

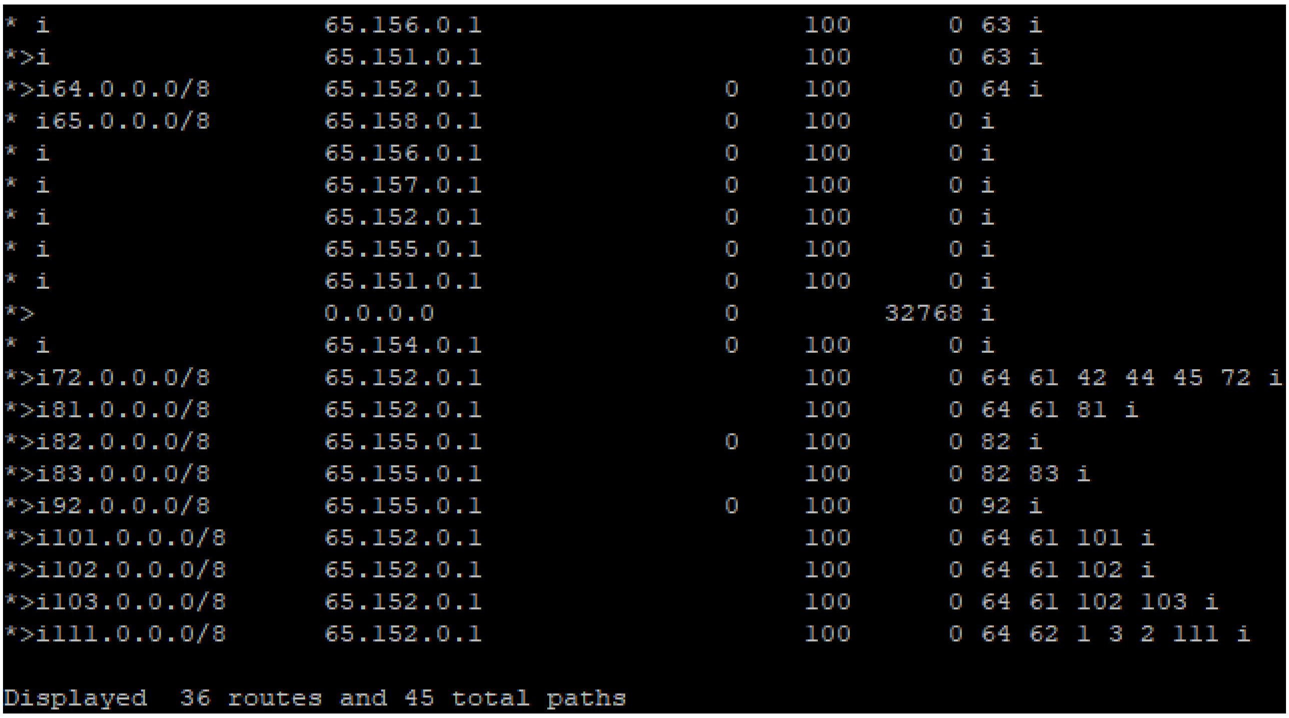
Question 2.2

In this question we need to configure the next-hop-self so that the neighbouring routers can reach an external AS via a router inside the current AS with an external interface and advertise our prefix to every other AS.

Next-hop-self: Basically, this command is introduced to every router because every router has an external interface that is connected to an outside AS. This command is needed because if a neighbouring router (For Example R3) wants to send packages to an outside AS that is connected through a router (For Example R1) with an external interface (For Example 10.0.0.2), it does not know how to reach that IP(No OSPF configured). So next-hop-self tells to every router inside an AS that if you want to reach that IP you can go through a specific router (This example is R1).



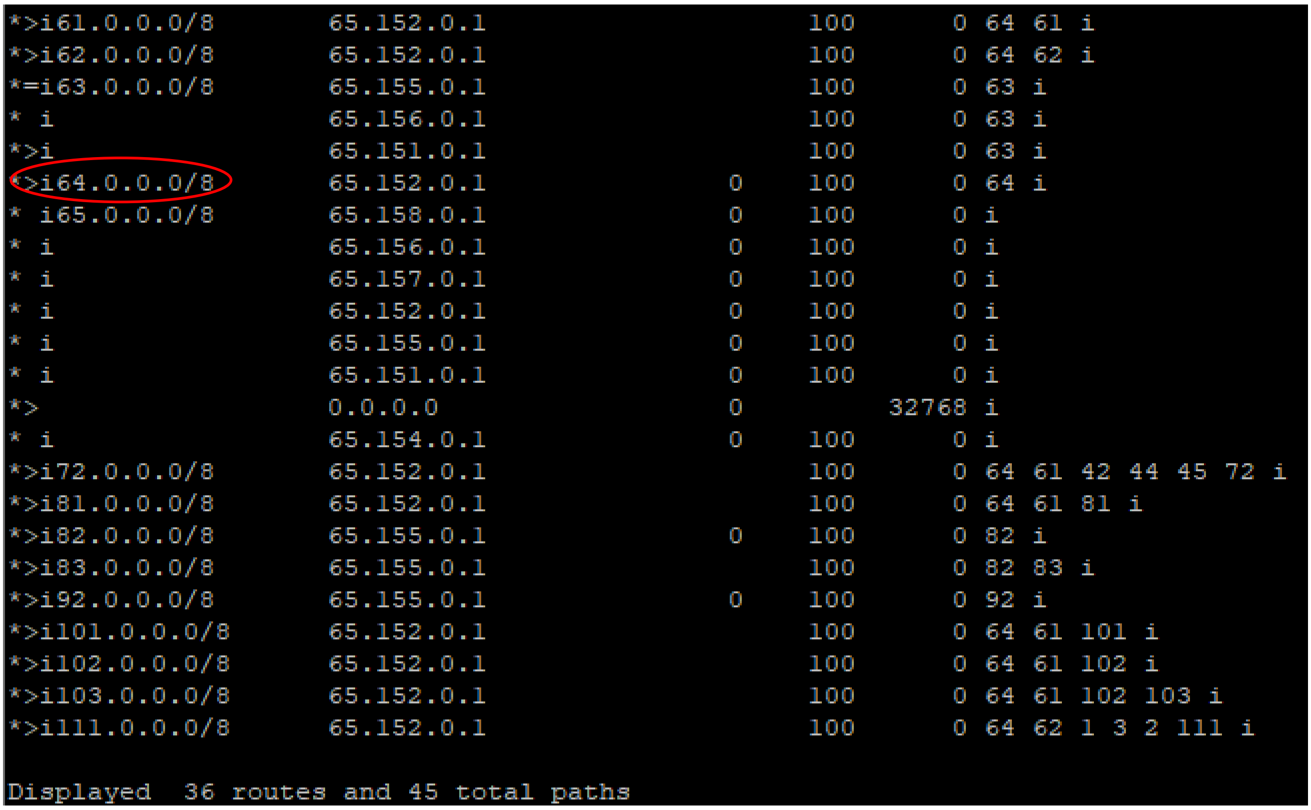
```
PARI_router# show ip bgp
BGP table version is 60, local router ID is 65.153.0.1, vrf id 0
Default local pref 100, local AS 65
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i1.0.0.0/8        65.152.0.1                    100      0 64 61 1 i
*>i2.0.0.0/8        65.152.0.1                    100      0 64 61 2 i
*>i3.0.0.0/8        65.152.0.1                    100      0 64 62 1 3 i
*>i7.0.0.0/8        65.152.0.1                    100      0 64 61 1 3 7 i
*>i10.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 2 10 i
*>i11.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 11 i
*>i12.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 11 12 i
*>i13.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 11 12 13
i
*>i14.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 11 12 14
i
*>i21.0.0.0/8       65.152.0.1                    100      0 64 61 21 i
*>i22.0.0.0/8       65.152.0.1                    100      0 64 61 22 i
*>i26.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 26 i
*>i28.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 28 i
*>i30.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 30 i
*>i34.0.0.0/8       65.152.0.1                    100      0 64 62 1 3 34 i
*>i41.0.0.0/8       65.152.0.1                    100      0 64 61 41 i
*>i42.0.0.0/8       65.152.0.1                    100      0 64 61 42 i
*>i44.0.0.0/8       65.152.0.1                    100      0 64 61 41 44 i
*>i45.0.0.0/8       65.152.0.1                    100      0 64 62 45 i
*>i46.104.0.0/24    65.152.0.1                    100      0 64 62 42 44 45 46 i
*>i49.0.0.0/8       65.152.0.1                    100      0 64 61 42 44 45 49 i
*>i53.0.0.0/8       65.152.0.1                    100      0 64 62 53 i
*>i61.0.0.0/8       65.152.0.1                    100      0 64 61 i
*>i62.0.0.0/8       65.152.0.1                    100      0 64 62 i
*=i63.0.0.0/8       65.155.0.1                    100      0 63 i
```



```
* i                 65.156.0.1                    100      0 63 i
*>i                 65.151.0.1                    100      0 63 i
*>i64.0.0.0/8       65.152.0.1               0    100      0 64 i
* i65.0.0.0/8       65.158.0.1               0    100      0 i
* i                 65.156.0.1               0    100      0 i
* i                 65.157.0.1               0    100      0 i
* i                 65.152.0.1               0    100      0 i
* i                 65.155.0.1               0    100      0 i
* i                 65.151.0.1               0    100      0 i
*>                  0.0.0.0                  0         32768 i
* i                 65.154.0.1               0    100      0 i
*>i72.0.0.0/8       65.152.0.1                    100      0 64 61 42 44 45 72 i
*>i81.0.0.0/8       65.152.0.1                    100      0 64 61 81 i
*>i82.0.0.0/8       65.155.0.1               0    100      0 82 i
*>i83.0.0.0/8       65.155.0.1                    100      0 82 83 i
*>i92.0.0.0/8       65.155.0.1               0    100      0 92 i
*>i101.0.0.0/8      65.152.0.1                    100      0 64 61 101 i
*>i102.0.0.0/8      65.152.0.1                    100      0 64 61 102 i
*>i103.0.0.0/8      65.152.0.1                    100      0 64 61 102 103 i
*>i111.0.0.0/8      65.152.0.1                    100      0 64 62 1 3 2 111 i

Displayed  36 routes and 45 total paths
```

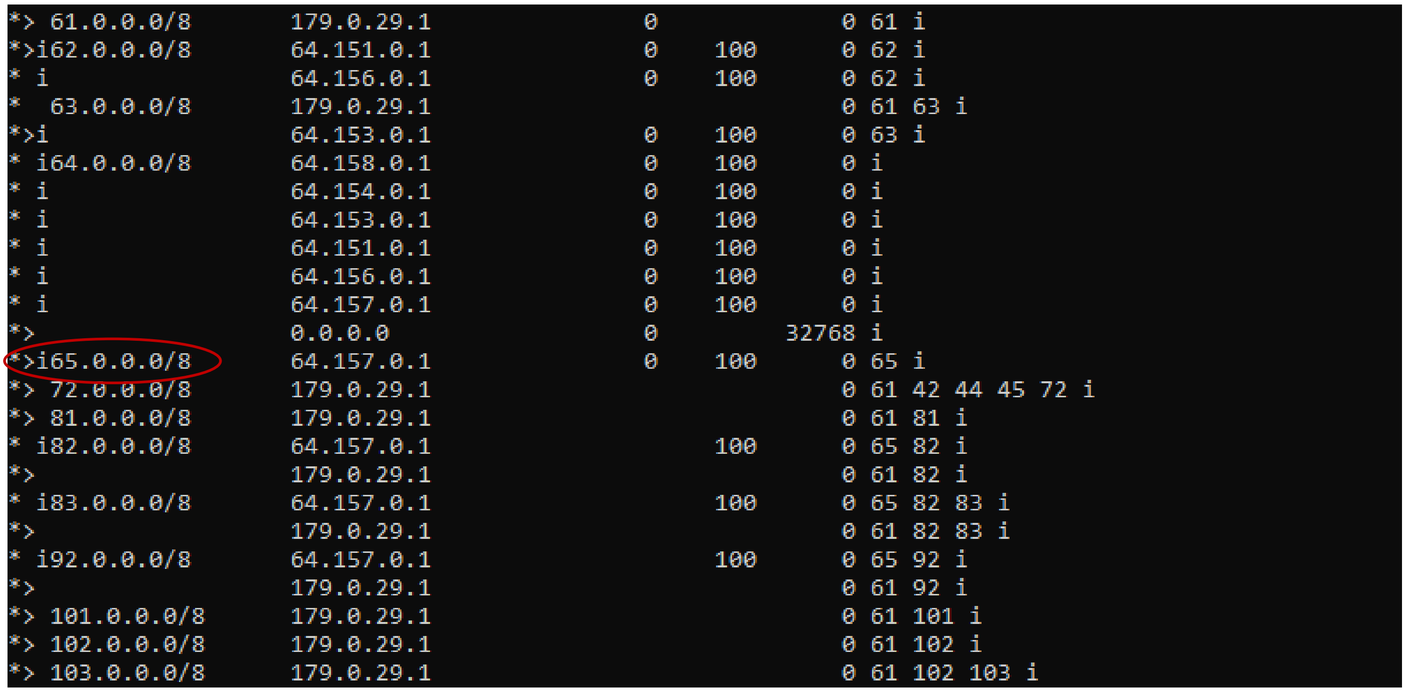
So after doing the command self-hop-next to all the routers in my AS lets see if our neighbours have taken our advertised IP.

This is a ***show ip bgp*** command from the PARI router in AS 65.



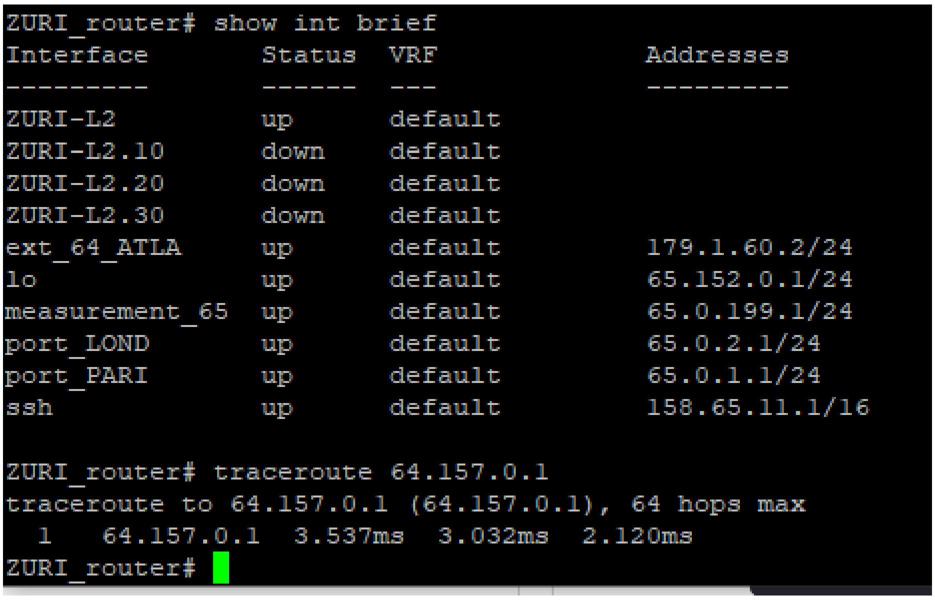
```
*>i61.0.0.0/8       65.152.0.1                    100      0 64 61 i
*>i62.0.0.0/8       65.152.0.1                    100      0 64 62 i
*=i63.0.0.0/8       65.155.0.1                    100      0 63 i
* i                 65.156.0.1                    100      0 63 i
*>i                 65.151.0.1                    100      0 63 i
*>i64.0.0.0/8       65.152.0.1               0    100      0 64 i
* i65.0.0.0/8       65.158.0.1               0    100      0 i
* i                 65.156.0.1               0    100      0 i
* i                 65.157.0.1               0    100      0 i
* i                 65.152.0.1               0    100      0 i
* i                 65.155.0.1               0    100      0 i
* i                 65.151.0.1               0    100      0 i
*>                  0.0.0.0                  0         32768 i
* i                 65.154.0.1               0    100      0 i
*>i72.0.0.0/8       65.152.0.1                    100      0 64 61 42 44 45 72 i
*>i81.0.0.0/8       65.152.0.1                    100      0 64 61 81 i
*>i82.0.0.0/8       65.155.0.1               0    100      0 82 i
*>i83.0.0.0/8       65.155.0.1                    100      0 82 83 i
*>i92.0.0.0/8       65.155.0.1               0    100      0 92 i
*>i101.0.0.0/8      65.152.0.1                    100      0 64 61 101 i
*>i102.0.0.0/8      65.152.0.1                    100      0 64 61 102 i
*>i103.0.0.0/8      65.152.0.1                    100      0 64 61 102 103 i
*>i111.0.0.0/8      65.152.0.1                    100      0 64 62 1 3 2 111 i

Displayed  36 routes and 45 total paths
```

Now we run from the looking glass from 64 – ZURI

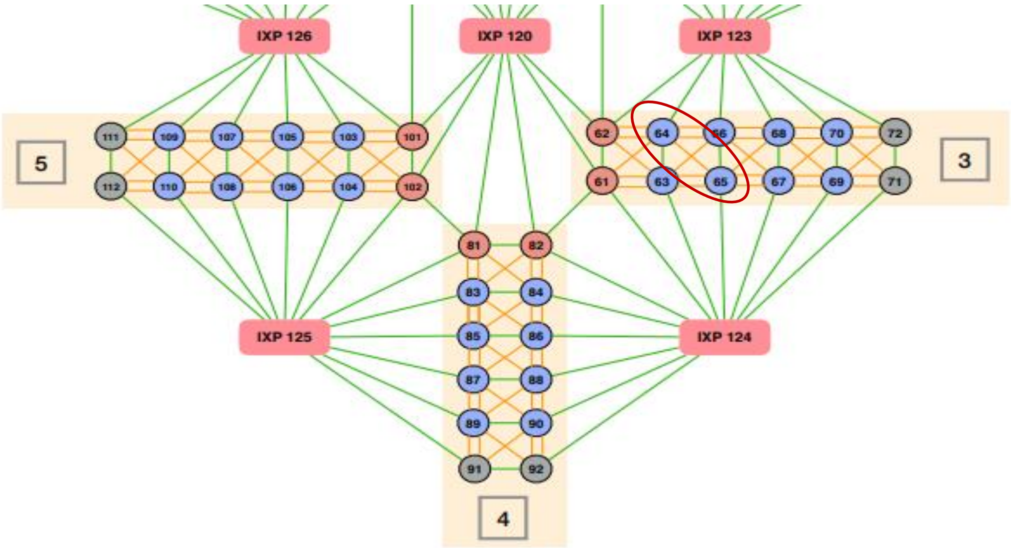


```
*> 61.0.0.0/8       179.0.29.1               0              0 61 i
*>i62.0.0.0/8       64.151.0.1               0    100       0 62 i
* i                 64.156.0.1               0    100       0 62 i
*  63.0.0.0/8       179.0.29.1                              0 61 63 i
*>i                 64.153.0.1               0    100       0 63 i
* i64.0.0.0/8       64.158.0.1               0    100       0 i
* i                 64.154.0.1               0    100       0 i
* i                 64.153.0.1               0    100       0 i
* i                 64.151.0.1               0    100       0 i
* i                 64.156.0.1               0    100       0 i
* i                 64.157.0.1               0    100       0 i
*>                  0.0.0.0                  0          32768 i
*>i65.0.0.0/8       64.157.0.1               0    100       0 65 i
*> 72.0.0.0/8       179.0.29.1                              0 61 42 44 45 72 i
*> 81.0.0.0/8       179.0.29.1                              0 61 81 i
* i82.0.0.0/8       64.157.0.1                    100       0 65 82 i
*>                  179.0.29.1                              0 61 82 i
* i83.0.0.0/8       64.157.0.1                    100       0 65 82 83 i
*>                  179.0.29.1                              0 61 82 83 i
* i92.0.0.0/8       64.157.0.1                    100       0 65 92 i
*>                  179.0.29.1                              0 61 92 i
*> 101.0.0.0/8      179.0.29.1                              0 61 101 i
*> 102.0.0.0/8      179.0.29.1                              0 61 102 i
*> 103.0.0.0/8      179.0.29.1                              0 61 102 103 i
```

As you can see our AS (65) is broadcasted to our neighbours.



```
ZURI_router# show int brief
Interface         Status  VRF             Addresses
---------         ------  ---             ---------
ZURI-L2           up      default
ZURI-L2.10        down    default
ZURI-L2.20        down    default
ZURI-L2.30        down    default
ext_64_ATLA       up      default         179.1.60.2/24
lo                up      default         65.152.0.1/24
measurement_65    up      default         65.0.199.1/24
port_LOND         up      default         65.0.2.1/24
port_PARI         up      default         65.0.1.1/24
ssh               up      default         158.65.11.1/16

ZURI_router# traceroute 64.157.0.1
traceroute to 64.157.0.1 (64.157.0.1), 64 hops max
  1   64.157.0.1  3.537ms  3.032ms  2.120ms
ZURI_router#
```



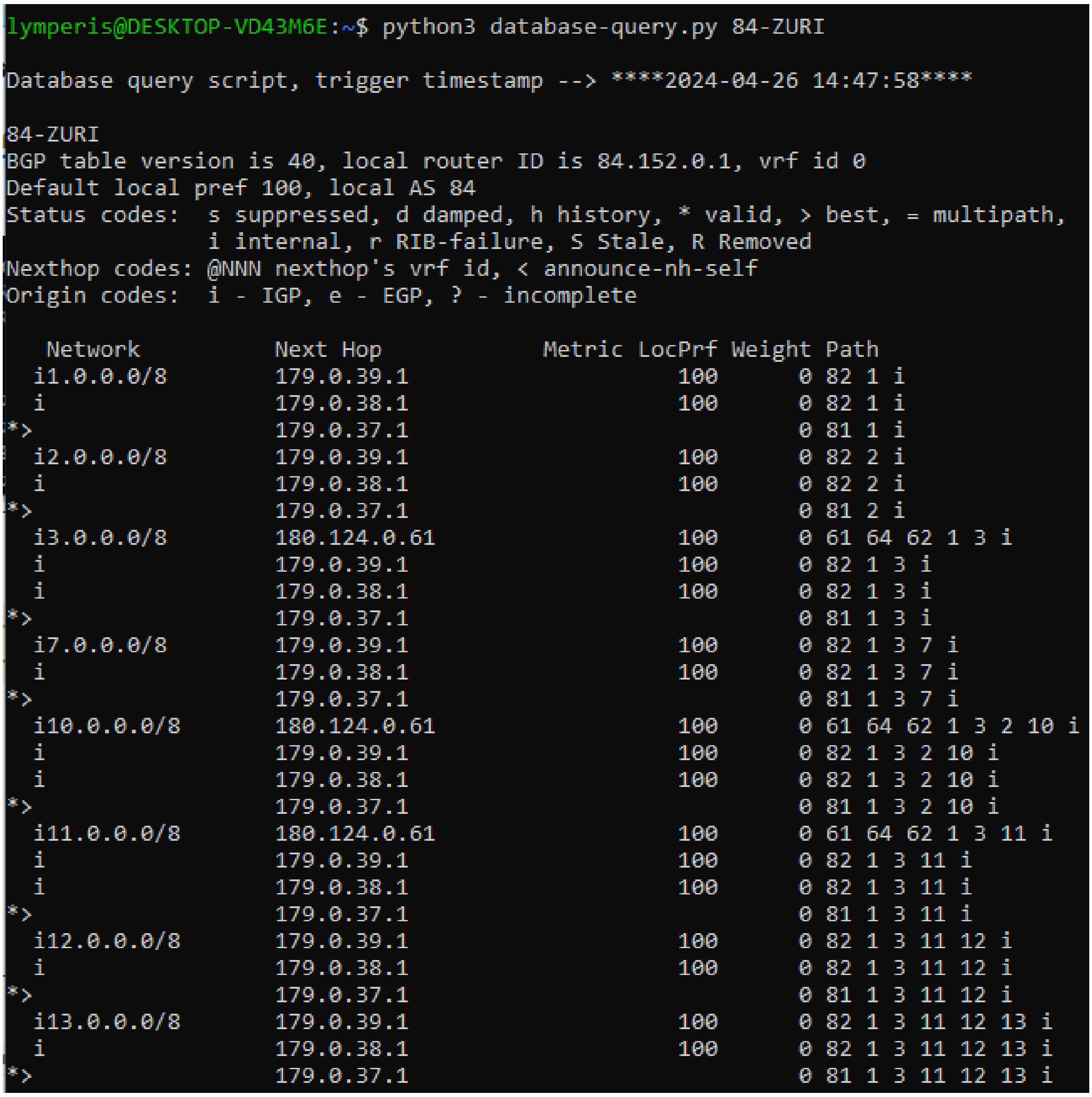
We ran a traceroute from ZURI router to the 64 AS ATLA router as you can see the hop is only one as it should be. (Our AS is 65)

Question 2.3

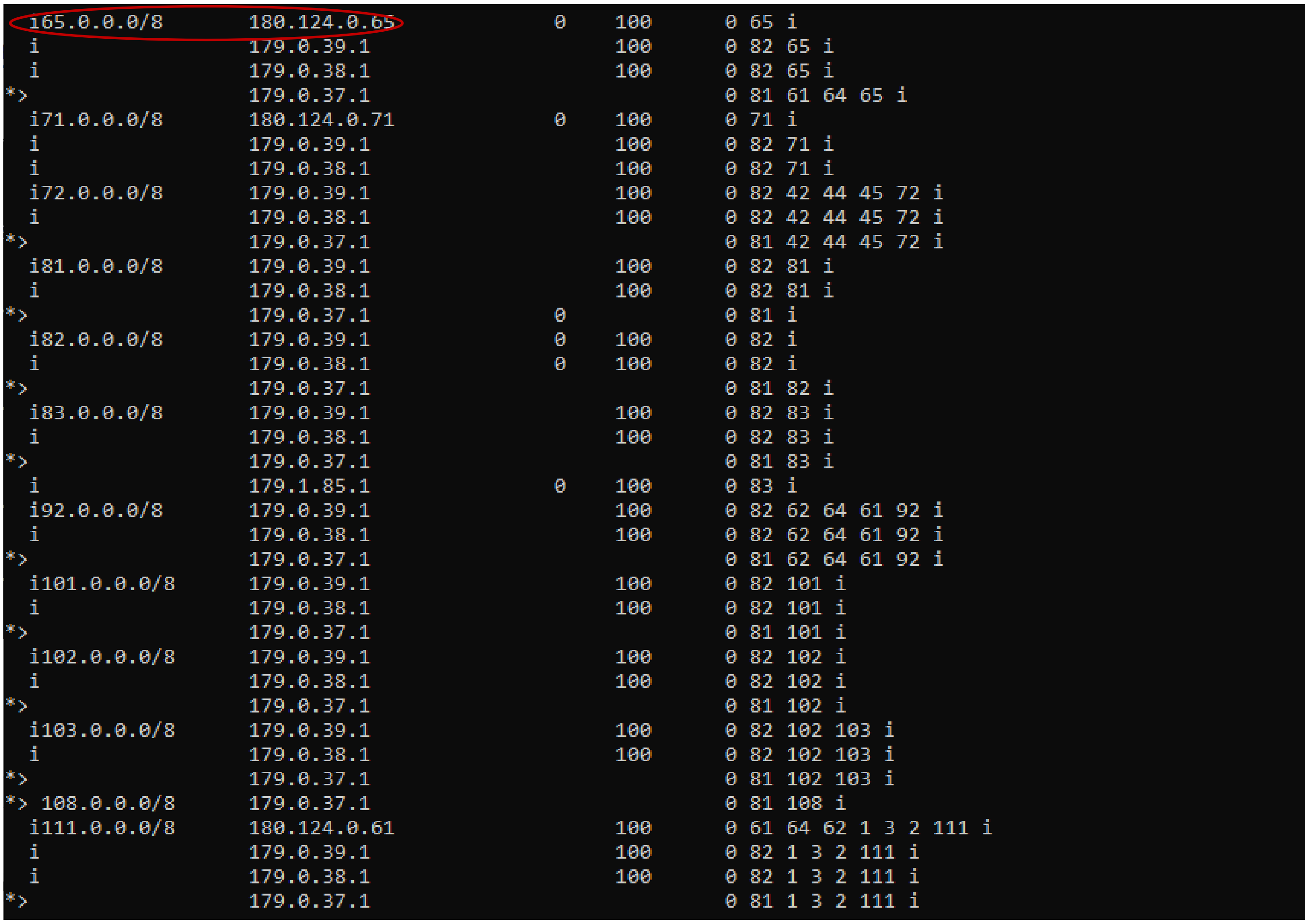
For our AS to communicate with the IXP and can handle traffic from one AS to another AS with one hop we need to do the following commands:

```
NEWY-router# conf t
NEWY-router# ip prefix-list OWN_PREFIX seq 5 permit 65.0.0.0/8 (Create a prefix-list named OWN_PREFIX that stores our subnet)
NEWY-router# route-map IXP_OUT permit 10 (Create a route-map named IXP_OUT used for filtering)
NEWY-router# match ip address prefix-list OWN_PREFIX
NEWY-router# 124:71 124:69 124:67 124:63 124:61 124:82 124:84 124:86 124:88 124:90 124:92
(Take every IP that matches the prefix-list OWN_PREFIX establish connection  with this community)
NEWY-router# exit
NEWY-router# router bgp 65
NEWY-router# neighbour 180.124.0.124 route-map IXP_OUT out (Apply the route-map filter to the external traffic from the IXP)
```

We used looking glass from AS-84 and as you can see the 65 prefix is announced.

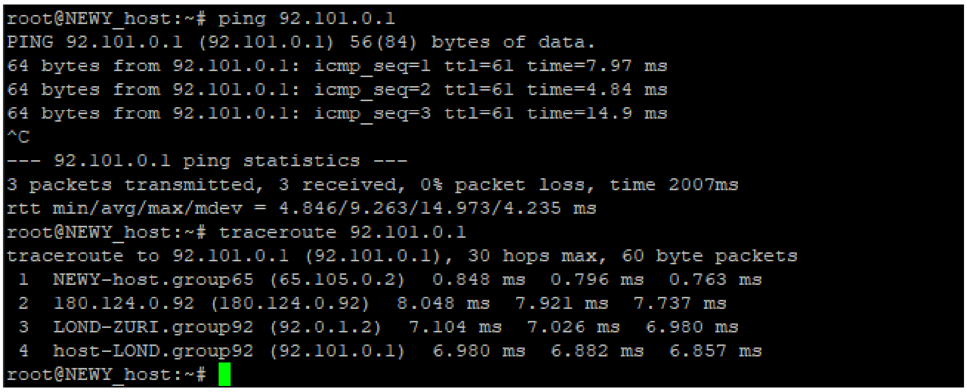


```
lymperis@DESKTOP-VD43M6E:~$ python3 database-query.py 84-ZURI

Database query script, trigger timestamp --> ****2024-04-26 14:47:58****

84-ZURI
BGP table version is 40, local router ID is 84.152.0.1, vrf id 0
Default local pref 100, local AS 84
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
  i1.0.0.0/8        179.0.39.1                    100      0 82 1 i
  i                 179.0.38.1                    100      0 82 1 i
*>                  179.0.37.1                             0 81 1 i
  i2.0.0.0/8        179.0.39.1                    100      0 82 2 i
  i                 179.0.38.1                    100      0 82 2 i
*>                  179.0.37.1                             0 81 2 i
  i3.0.0.0/8        180.124.0.61                  100      0 61 64 62 1 3 i
  i                 179.0.39.1                    100      0 82 1 3 i
  i                 179.0.38.1                    100      0 82 1 3 i
*>                  179.0.37.1                             0 81 1 3 i
  i7.0.0.0/8        179.0.39.1                    100      0 82 1 3 7 i
  i                 179.0.38.1                    100      0 82 1 3 7 i
*>                  179.0.37.1                             0 81 1 3 7 i
  i10.0.0.0/8       180.124.0.61                  100      0 61 64 62 1 3 2 10 i
  i                 179.0.39.1                    100      0 82 1 3 2 10 i
  i                 179.0.38.1                    100      0 82 1 3 2 10 i
*>                  179.0.37.1                             0 81 1 3 2 10 i
  i11.0.0.0/8       180.124.0.61                  100      0 61 64 62 1 3 11 i
  i                 179.0.39.1                    100      0 82 1 3 11 i
  i                 179.0.38.1                    100      0 82 1 3 11 i
*>                  179.0.37.1                             0 81 1 3 11 i
  i12.0.0.0/8       179.0.39.1                    100      0 82 1 3 11 12 i
  i                 179.0.38.1                    100      0 82 1 3 11 12 i
*>                  179.0.37.1                             0 81 1 3 11 12 i
  i13.0.0.0/8       179.0.39.1                    100      0 82 1 3 11 12 13 i
  i                 179.0.38.1                    100      0 82 1 3 11 12 13 i
*>                  179.0.37.1                             0 81 1 3 11 12 13 i
```

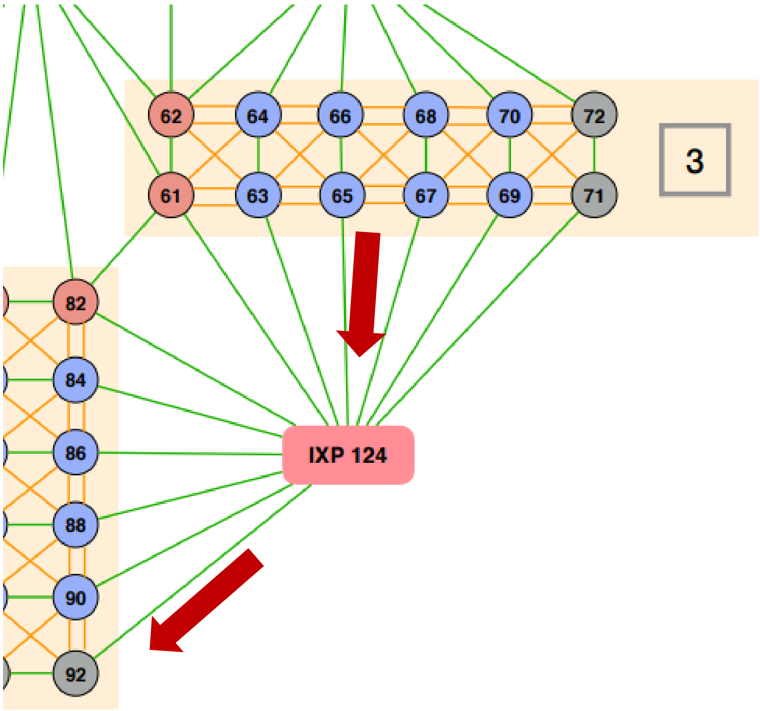


```
  i165.0.0.0/8      180.124.0.65             0    100      0 65 i
  i                 179.0.39.1                    100      0 82 65 i
  i                 179.0.38.1                    100      0 82 65 i
*>                  179.0.37.1                             0 81 61 64 65 i
  i71.0.0.0/8       180.124.0.71             0    100      0 71 i
  i                 179.0.39.1                    100      0 82 71 i
  i                 179.0.38.1                    100      0 82 71 i
  i72.0.0.0/8       179.0.39.1                    100      0 82 42 44 45 72 i
  i                 179.0.38.1                    100      0 82 42 44 45 72 i
*>                  179.0.37.1                             0 81 42 44 45 72 i
  i81.0.0.0/8       179.0.39.1                    100      0 82 81 i
  i                 179.0.38.1                    100      0 82 81 i
*>                  179.0.37.1               0             0 81 i
  i82.0.0.0/8       179.0.39.1               0    100      0 82 i
  i                 179.0.38.1               0    100      0 82 i
*>                  179.0.37.1                             0 81 82 i
  i83.0.0.0/8       179.0.39.1                    100      0 82 83 i
  i                 179.0.38.1                    100      0 82 83 i
*>                  179.0.37.1                             0 81 83 i
  i                 179.1.85.1               0    100      0 83 i
  i92.0.0.0/8       179.0.39.1                    100      0 82 62 64 61 92 i
  i                 179.0.38.1                    100      0 82 62 64 61 92 i
*>                  179.0.37.1                             0 81 62 64 61 92 i
  i101.0.0.0/8      179.0.39.1                    100      0 82 101 i
  i                 179.0.38.1                    100      0 82 101 i
*>                  179.0.37.1                             0 81 101 i
  i102.0.0.0/8      179.0.39.1                    100      0 82 102 i
  i                 179.0.38.1                    100      0 82 102 i
*>                  179.0.37.1                             0 81 102 i
  i103.0.0.0/8      179.0.39.1                    100      0 82 102 103 i
  i                 179.0.38.1                    100      0 82 102 103 i
*>                  179.0.37.1                             0 81 102 103 i
*> 108.0.0.0/8      179.0.37.1                             0 81 108 i
  i111.0.0.0/8      180.124.0.61                  100      0 61 64 62 1 3 2 111 i
  i                 179.0.39.1                    100      0 82 1 3 2 111 i
  i                 179.0.38.1                    100      0 82 1 3 2 111 i
*>                  179.0.37.1                             0 81 1 3 2 111 i
```

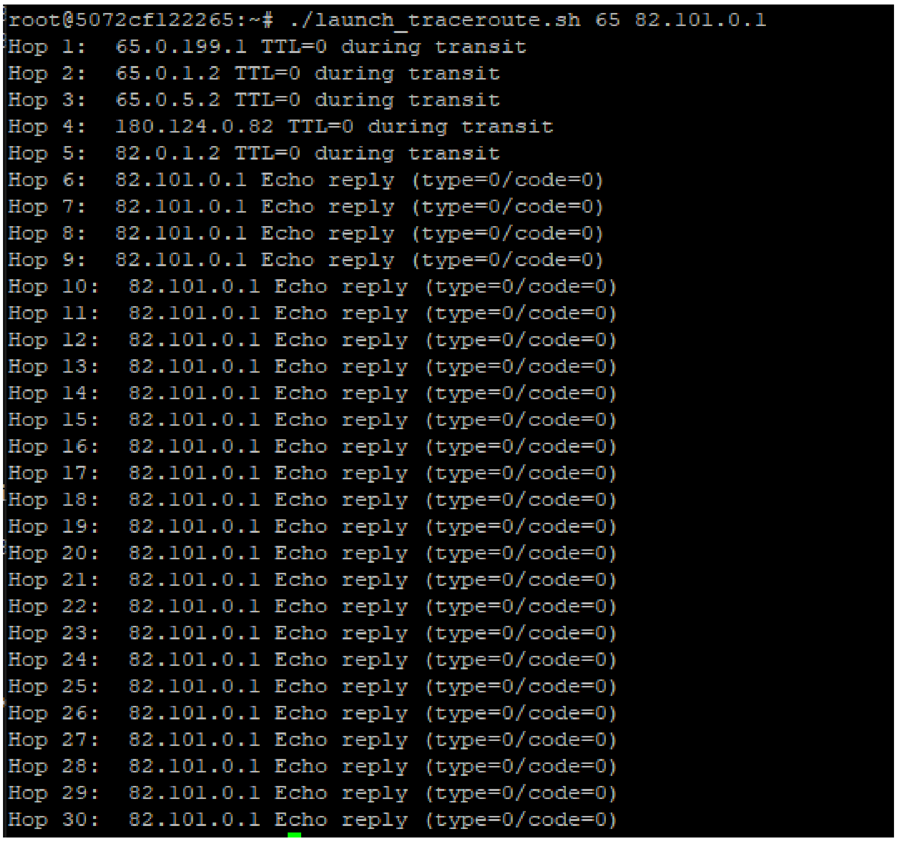
That is a ping from my NEWY host to the AS 92 through the IXP.



```
root@NEWY_host:~# ping 92.101.0.1
PING 92.101.0.1 (92.101.0.1) 56(84) bytes of data.
64 bytes from 92.101.0.1: icmp_seq=1 ttl=61 time=7.97 ms
64 bytes from 92.101.0.1: icmp_seq=2 ttl=61 time=4.84 ms
64 bytes from 92.101.0.1: icmp_seq=3 ttl=61 time=14.9 ms
^C
--- 92.101.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 4.846/9.263/14.973/4.235 ms
root@NEWY_host:~# traceroute 92.101.0.1
traceroute to 92.101.0.1 (92.101.0.1), 30 hops max, 60 byte packets
 1  NEWY-host.group65 (65.105.0.2)  0.848 ms  0.796 ms  0.763 ms
 2  180.124.0.92 (180.124.0.92)  8.048 ms  7.921 ms  7.737 ms
 3  LOND-ZURI.group92 (92.0.1.2)  7.104 ms  7.026 ms  6.980 ms
 4  host-LOND.group92 (92.101.0.1)  6.980 ms  6.882 ms  6.857 ms
root@NEWY_host:~#
```



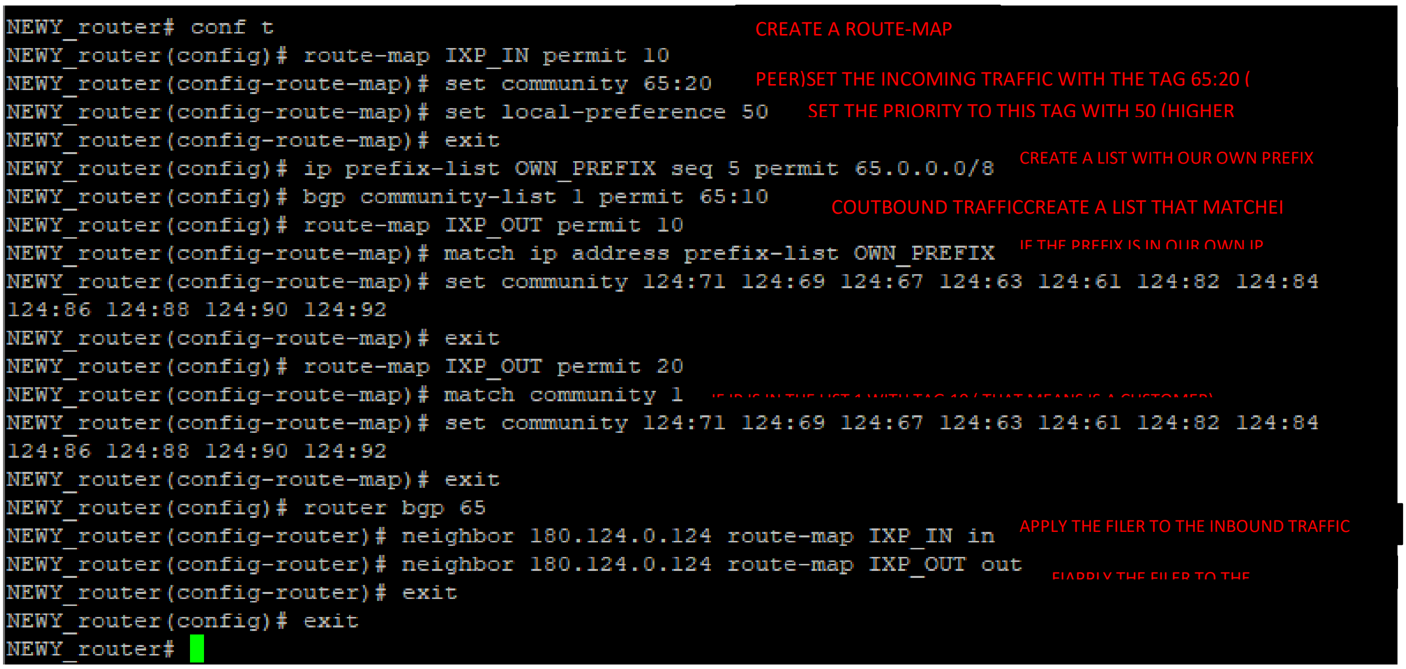
Now we run a NEWY traceroute from AS 65 to AS 85 to through the IXP from the looking glass:



```
root@5072cf122265:~# ./launch_traceroute.sh 65 82.101.0.1
Hop 1:  65.0.199.1 TTL=0 during transit
Hop 2:  65.0.1.2 TTL=0 during transit
Hop 3:  65.0.5.2 TTL=0 during transit
Hop 4:  180.124.0.82 TTL=0 during transit
Hop 5:  82.0.1.2 TTL=0 during transit
Hop 6:  82.101.0.1 Echo reply (type=0/code=0)
Hop 7:  82.101.0.1 Echo reply (type=0/code=0)
Hop 8:  82.101.0.1 Echo reply (type=0/code=0)
Hop 9:  82.101.0.1 Echo reply (type=0/code=0)
Hop 10:  82.101.0.1 Echo reply (type=0/code=0)
Hop 11:  82.101.0.1 Echo reply (type=0/code=0)
Hop 12:  82.101.0.1 Echo reply (type=0/code=0)
Hop 13:  82.101.0.1 Echo reply (type=0/code=0)
Hop 14:  82.101.0.1 Echo reply (type=0/code=0)
Hop 15:  82.101.0.1 Echo reply (type=0/code=0)
Hop 16:  82.101.0.1 Echo reply (type=0/code=0)
Hop 17:  82.101.0.1 Echo reply (type=0/code=0)
Hop 18:  82.101.0.1 Echo reply (type=0/code=0)
Hop 19:  82.101.0.1 Echo reply (type=0/code=0)
Hop 20:  82.101.0.1 Echo reply (type=0/code=0)
Hop 21:  82.101.0.1 Echo reply (type=0/code=0)
Hop 22:  82.101.0.1 Echo reply (type=0/code=0)
Hop 23:  82.101.0.1 Echo reply (type=0/code=0)
Hop 24:  82.101.0.1 Echo reply (type=0/code=0)
Hop 25:  82.101.0.1 Echo reply (type=0/code=0)
Hop 26:  82.101.0.1 Echo reply (type=0/code=0)
Hop 27:  82.101.0.1 Echo reply (type=0/code=0)
Hop 28:  82.101.0.1 Echo reply (type=0/code=0)
Hop 29:  82.101.0.1 Echo reply (type=0/code=0)
Hop 30:  82.101.0.1 Echo reply (type=0/code=0)
```

Question 2.4

The community tags we have been instructed to use (following the Gao_Raxford method) are 65:10 for the Customers 65:20 for the Peers and 65:30 for the Providers. Im using local-preference 50 for Peers, 100 for customers and 20 for Providers



```
NEWY_router# conf t
NEWY_router(config)# route-map IXP_IN permit 10
NEWY_router(config-route-map)# set community 65:20
NEWY_router(config-route-map)# set local-preference 50
NEWY_router(config-route-map)# exit
NEWY_router(config)# ip prefix-list OWN_PREFIX seq 5 permit 65.0.0.0/8
NEWY_router(config)# bgp community-list 1 permit 65:10
NEWY_router(config)# route-map IXP_OUT permit 10
NEWY_router(config-route-map)# match ip address prefix-list OWN_PREFIX
NEWY_router(config-route-map)# set community 124:71 124:69 124:67 124:63 124:61 124:82 124:84
124:86 124:88 124:90 124:92
NEWY_router(config-route-map)# exit
NEWY_router(config)# route-map IXP_OUT permit 20
NEWY_router(config-route-map)# match community 1
NEWY_router(config-route-map)# set community 124:71 124:69 124:67 124:63 124:61 124:82 124:84
124:86 124:88 124:90 124:92
NEWY_router(config-route-map)# exit
NEWY_router(config)# router bgp 65
NEWY_router(config-router)# neighbor 180.124.0.124 route-map IXP_IN in
NEWY_router(config-router)# neighbor 180.124.0.124 route-map IXP_OUT out
NEWY_router(config-router)# exit
NEWY_router(config)# exit
NEWY_router#
```

CREATE A ROUTE-MAP

PEER)SET THE INCOMING TRAFFIC WITH THE TAG 65:20 (

SET THE PRIORITY TO THIS TAG WITH 50 (HIGHER

CREATE A LIST WITH OUR OWN PREFIX

COUTBOUND TRAFFICCREATE A LIST THAT MATCHEI

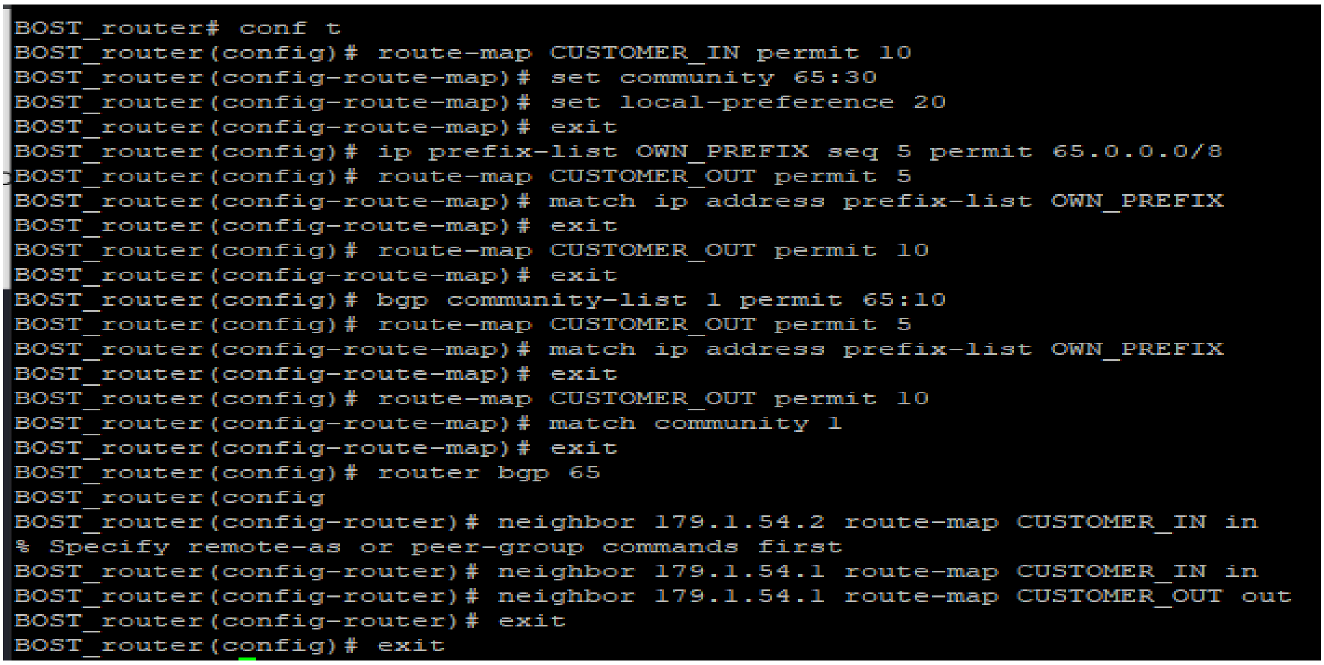
IF THE PREFIX IS IN OUR OWN IP

IF IP IS IN THE LIST 1 WITH TAG 10 (THAT MEANS IS A CUSTOMER)

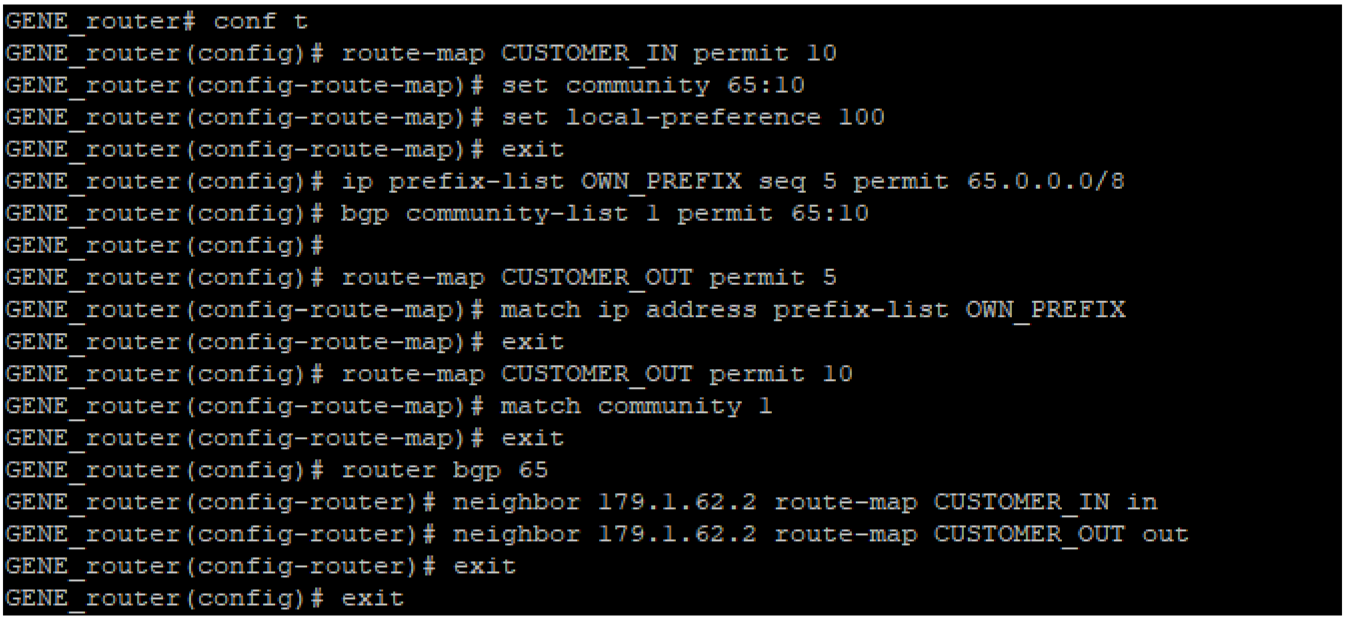
APPLY THE FILER TO THE INBOUND TRAFFIC

FIAPPLY THE FILER TO THE

Example for a customer and a provider:



```
BOST_router# conf t
BOST_router(config)# route-map CUSTOMER_IN permit 10
BOST_router(config-route-map)# set community 65:30
BOST_router(config-route-map)# set local-preference 20
BOST_router(config-route-map)# exit
BOST_router(config)# ip prefix-list OWN_PREFIX seq 5 permit 65.0.0.0/8
BOST_router(config)# route-map CUSTOMER_OUT permit 5
BOST_router(config-route-map)# match ip address prefix-list OWN_PREFIX
BOST_router(config-route-map)# exit
BOST_router(config)# route-map CUSTOMER_OUT permit 10
BOST_router(config-route-map)# exit
BOST_router(config)# bgp community-list 1 permit 65:10
BOST_router(config)# route-map CUSTOMER_OUT permit 5
BOST_router(config-route-map)# match ip address prefix-list OWN_PREFIX
BOST_router(config-route-map)# exit
BOST_router(config)# route-map CUSTOMER_OUT permit 10
BOST_router(config-route-map)# match community 1
BOST_router(config-route-map)# exit
BOST_router(config)# router bgp 65
BOST_router(config
BOST_router(config-router)# neighbor 179.1.54.2 route-map CUSTOMER_IN in
% Specify remote-as or peer-group commands first
BOST_router(config-router)# neighbor 179.1.54.1 route-map CUSTOMER_IN in
BOST_router(config-router)# neighbor 179.1.54.1 route-map CUSTOMER_OUT out
BOST_router(config-router)# exit
BOST_router(config)# exit
```



```
GENE_router# conf t
GENE_router(config)# route-map CUSTOMER_IN permit 10
GENE_router(config-route-map)# set community 65:10
GENE_router(config-route-map)# set local-preference 100
GENE_router(config-route-map)# exit
GENE_router(config)# ip prefix-list OWN_PREFIX seq 5 permit 65.0.0.0/8
GENE_router(config)# bgp community-list 1 permit 65:10
GENE_router(config)#
GENE_router(config)# route-map CUSTOMER_OUT permit 5
GENE_router(config-route-map)# match ip address prefix-list OWN_PREFIX
GENE_router(config-route-map)# exit
GENE_router(config)# route-map CUSTOMER_OUT permit 10
GENE_router(config-route-map)# match community 1
GENE_router(config-route-map)# exit
GENE_router(config)# router bgp 65
GENE_router(config-router)# neighbor 179.1.62.2 route-map CUSTOMER_IN in
GENE_router(config-router)# neighbor 179.1.62.2 route-map CUSTOMER_OUT out
GENE_router(config-router)# exit
GENE_router(config)# exit
```

Now after configuring all the routers correctly if we run a traceroute from one of our Customers, the customer will go through our AS (we are the provider) and will give the traffic through our provider as well.

Expected output : We can see the traceroute from our Peer go through our AS and we are going to send the traffic not to our provider but to an IXP(or another peer).

Question 2.5

In this question we must make a python script which will monitor the latency and the number of intermediate routers (hops) among a path. (Can not run without the docker (problem with ssh)).