

## STATIC-ROUTING

- 1- **lab.conf** -> indica la configurazione della rete con le interfacce e i collegamenti sulle varie LAN

```
r1[0]="A"
r1[1]="B"
r1[image]="kathara/base"

r2[0]="C"
r2[1]="B"
r2[image]="kathara/base"

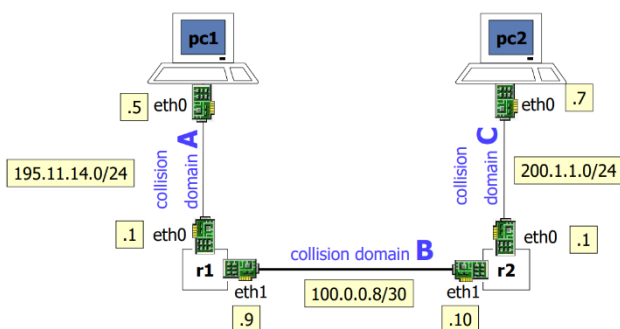
pc1[0]="A"
pc1[image]="kathara/base"

pc2[0]="C"
pc2[image]="kathara/base"
```

- 2- **r1.startup** -> indica l'aggiunta di indirizzi con cui può collaborare un certo router

```
ip address add 195.11.14.1/24 dev eth0
ip address add 100.0.0.9/30 dev eth1
#ip route add 200.1.1.0/24 via 100.0.0.10 dev eth1
```

- 3- Per testare la connettività uso il comando **ping** verso un indirizzo di una macchina. Se non possono comunicare mi darà network unreachable
- 4- Aggiunta rotta di default sulla macchina da cui voglio fare il ping -> **ip route add default via 195.11.14.1 dev eth0**, con questa posso raggiungere tutte le reti perché funge da gateway (tramite me raggiungi le altre lan)
- 5- **tcdump -t eth1** -> indica il comando per uno sniffer su eth1. Mi dice quali pacchetti passano e lo metto sul router da cui voglio vedere
- 6- quando non so raggiungere una macchina, aggiungo staticamente una rotta nella tabella di instradamento specificando il next hop : **ip route add 195.11.14.0/24 via 100.0.0.9 dev eth1**



## FRR-ROUTING

- 1- devo vedere quali demoni in esecuzione e decido quale far partire

```
root@r1:/# cat /etc/frr/daemons
zebra=yes
bgpd=no
ospfd=no
ospf6d=no
ripd=yes
ripngd=no
isisd=no
pimd=no
ldpd=no
nhd=no
nhrpd=no
eigrpd=no
babeld=no
sharpd=no
staticd=no
pbrd=no
bfd=no
fabricd=no
```

- 2- lab.conf -> definisco la configurazione della rete

```
r1[0]="A"
r1[1]="B"
r1[image]="kathara/frr"

r2[0]="B"
r2[1]="C"
r2[image]="kathara/frr"

r3[0]="C"
r3[1]="D"
r3[image]="kathara/frr"
```

- 3- r1.startup -> con il comando systemctl start frr, ci pensa lui a far partire il demone frr

```
ip address add 100.0.0.1/24 dev eth0
ip address add 150.0.0.1/30 dev eth1
systemctl start frr
```

- 4- vtysh -> comando per parlare con il demone ed entrare nell'interfaccia di configurazione di frr  
? -> mi mostra tutti i comandi che posso fare dentro vtysh

```
r1#
add          Add registration
clear        clear
configure    Configuration from vty interface
copy         Apply a configuration file
debug        Debugging functions
disable      Turn off privileged mode command
enable       Turn on privileged mode command
end          End current mode and change to enable mode
exit         Exit current mode and down to previous mode
find         Find CLI command matching a regular expression
graceful-restart Graceful Restart commands
list         Print command list
mgmt         Management Daemon (NGMTD) information
mtrace       Multicast trace route to multicast source
no           Negate a command or set its defaults
output       Direct vtysh output to file
ping         Send echo messages
quit         Exit current mode and down to previous mode
rpkd         Control rpkd specific settings
show         Show running system information
terminal     Set terminal line parameters
traceroute   Trace route to destination
watchfrr     Watchfrr Specific sub-command
write        Write running configuration to memory, network, or terminal
```

- 5- **show ip route** -> ci fa vedere le rotte del protocollo ip, tabella di instradamento

```
rl# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

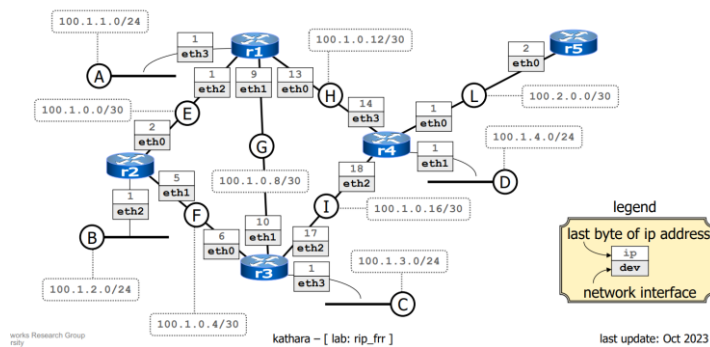
C>* 100.0.0.0/24 is directly connected, eth0, 00:00:14
C>* 150.0.0.0/30 is directly connected, eth1, 00:00:14
R>* 200.0.0.0/30 [120/2] via 150.0.0.2, eth1, weight 1, 00:00:13
R>* 220.0.0.0/24 [120/3] via 150.0.0.2, eth1, weight 1, 00:00:12
```

- 6- **show running-config** -> stampa la configurazione

```
rl# show running-config
Building configuration...

Current configuration:
!
frr version 9.0.1
frr defaults traditional
hostname rl
domainname localdomain
log file /var/log/frr/frr.log
no ipv6 forwarding
no service integrated-vtysh-config
!
router rip
 network 150.0.0.0/8
 redistribute connected
exit
!
end
```

## RIP



- 1- **lab.conf** -> definisco la rete

```
r1[0]="H"
r1[1]="G"
r1[2]="E"
r1[3]="A"
r1[image]="kathara/frr"

r2[0]="E"
r2[1]="F"
r2[2]="B"
r2[image]="kathara/frr"

r3[0]="F"
r3[1]="G"
r3[2]="I"
r3[3]="C"
r3[image]="kathara/frr"

r4[0]="L"
r4[1]="D"
r4[2]="I"
r4[3]="H"
r4[image]="kathara/frr"

r5[0]="L"
r5[image]="kathara/frr"
```

- 2- **frr.conf** -> configuration file rip

```
root@r1:/# cat /etc/frr/frr.conf
!
! FRRouting configuration file
!
!
!   RIP CONFIGURATION
!
router rip
 redistribute connected
 network 100.1.0.0/24
!
 log file /var/log/frr/frr.log
```

**redistribute connected** dice che devi parlare solo di ciò che è connesso a te

- 3- **ip route** -> vedo le lan direttamente connesse

```
root@r4:/# ip route
100.1.0.12/30 dev eth3 proto kernel scope link src 100.1.0.14
100.1.0.16/30 dev eth2 proto kernel scope link src 100.1.0.18
100.1.4.0/24 dev eth1 proto kernel scope link src 100.1.4.1
100.2.0.0/30 dev eth0 proto kernel scope link src 100.2.0.1
```

- 4- **systemctl start frr** -> avvio il demone manualmente che usa come protocollo rip (specificato in /etc/frr/daemons) su tutte le macchine

```
root@r4:/# ping 100.1.2.1
ping: connect: Network is unreachable
```

- 5- così facendo rendo le destinazioni remote tutte raggiungibili perché le ho aggiunte con rip nella tabella di instradamento e facendo di nuovo ip route la tabella di instradamento si aggiorna perché è basata su distance vector (gli arrivano le info dai vicini e aggiorno)
- 6- **tcpdump -ttni eth2 -v** -> vedo i pacchetti che passano su interfaccia eth2 di r4

```

root@r4:/# tcpdump -ttni eth2 -v
tcpdump: listening on eth2, link-type EN10MB (Ethernet), snapshot length 262144 bytes
72:7e:b9:ad:5c:fd > 01:00:5e:00:00:09, ethertype IPv4 (0x0800), length 66: (tos 0xc0, ttl 1, id 6145, offset 0, flags [D
F], proto UDP (17), length 52)
100.1.0.18.520 > 228.0.0.9.520:
RIPv2, Request, length: 24, routes: 1 or less
  AFI 0, 0.0.0.0/0, tag 0x0000, metric: 16, next-hop: self
f2:6c:6f:2c:c8:b5 > 72:7e:b9:ad:5c:fd, ethertype IPv4 (0x0800), length 166: (tos 0xc0, ttl 64, id 43351, offset 0, flags
[DF], proto UDP (17), length 152)
100.1.0.17.520 > 100.1.0.18.520:
RIPv2, Response, length: 124, routes: 6 or less
  AFI IPv4, 100.1.0.0/30, tag 0x0000, metric: 2, next-hop: self
  AFI IPv4, 100.1.0.4/30, tag 0x0000, metric: 1, next-hop: self
  AFI IPv4, 100.1.0.8/30, tag 0x0000, metric: 1, next-hop: self
  AFI IPv4, 100.1.1.0/24, tag 0x0000, metric: 2, next-hop: self
  AFI IPv4, 100.1.2.0/24, tag 0x0000, metric: 2, next-hop: self
  AFI IPv4, 100.1.3.0/24, tag 0x0000, metric: 1, next-hop: self

```

- 7- **vttysh** -> parlo col demone di rip
- 8- **show ip rip** -> vedo configurazione di rip e delle lan di ogni router singolo (sono tutte uguali perché le ho messe tutte raggiungibili)

```

r4# show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
  (n) - normal, (s) - static, (d) - default, (r) - redistribute,
  (i) - interface

Network        Next Hop      Metric From      Tag Time
R(n) 100.1.0.0/30 100.1.0.13    2 100.1.0.13    0 02:38
R(n) 100.1.0.4/30 100.1.0.17    2 100.1.0.17    0 02:41
R(n) 100.1.0.8/30 100.1.0.17    2 100.1.0.17    0 02:41
C(i) 100.1.0.12/30 0.0.0.0       1 self         0
C(i) 100.1.0.16/30 0.0.0.0       1 self         0
R(n) 100.1.1.0/24 100.1.0.13    2 100.1.0.13    0 02:38
R(n) 100.1.2.0/24 100.1.0.17    3 100.1.0.17    0 02:41
R(n) 100.1.3.0/24 100.1.0.17    2 100.1.0.17    0 02:41
C(r) 100.1.4.0/24 0.0.0.0       1 self         0
C(r) 100.2.0.0/30 0.0.0.0       1 self         0

```

- 9- aggiungo rotta statica su r5 perché è l'unico fuori dalla rete. La aggiungo per farlo comunicare con gli altri (tramite il router vicino).

```

root@r5:/# ip route add 100.1.0.0/16 via 100.2.0.1
root@r5:/# ping 100.1.2.1
PING 100.1.2.1 (100.1.2.1) 56(84) bytes of data.
64 bytes from 100.1.2.1: icmp_seq=1 ttl=62 time=0.598 ms
64 bytes from 100.1.2.1: icmp_seq=2 ttl=62 time=0.193 ms
64 bytes from 100.1.2.1: icmp_seq=3 ttl=62 time=0.227 ms
64 bytes from 100.1.2.1: icmp_seq=4 ttl=62 time=0.129 ms

```

- 10- configuro la route di default su r4 che è il router vicino a r5 in modo tale da propagare le info

```

root@r4:/# ip route add default via 100.2.0.2
root@r4:/# ip route
default via 100.2.0.2 dev eth0
100.1.0.0/30 nhid 12 via 100.1.0.13 dev eth3 proto rip metric 20
100.1.0.4/30 nhid 10 via 100.1.0.17 dev eth2 proto rip metric 20
100.1.0.8/30 nhid 10 via 100.1.0.17 dev eth2 proto rip metric 20
100.1.0.12/30 dev eth3 proto kernel scope link src 100.1.0.14
100.1.0.16/30 dev eth2 proto kernel scope link src 100.1.0.18
100.1.1.0/24 nhid 12 via 100.1.0.13 dev eth3 proto rip metric 20
100.1.2.0/24 nhid 10 via 100.1.0.17 dev eth2 proto rip metric 20
100.1.3.0/24 nhid 10 via 100.1.0.17 dev eth2 proto rip metric 20
100.1.4.0/24 dev eth1 proto kernel scope link src 100.1.4.1
100.2.0.0/30 dev eth0 proto kernel scope link src 100.2.0.1

```

11- propago la default route dentro rip

```
root@r4:/# vtysh

Hello, this is FRRouting (version 9.0.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

r4# configure
r4(config)# router rip
r4(config-router)# route 0.0.0.0/0
r4(config-router)# quit
r4(config)# quit
r4# exit
```

12- default route iniettata nella rete via rip (in r1 da interfaccia eth3 di r4, in r2 da interfaccia eth0 di r3)

```
root@r1:/# ip route
default nhid 14 via 100.1.0.14 dev eth0 proto rip metric 20
100.1.0.0/30 dev eth2 proto kernel scope link src 100.1.0.1
100.1.0.4/30 nhid 12 via 100.1.0.10 dev eth1 proto rip metric 20
100.1.0.8/30 dev eth1 proto kernel scope link src 100.1.0.9
100.1.0.12/30 dev eth0 proto kernel scope link src 100.1.0.13
100.1.0.16/30 nhid 12 via 100.1.0.10 dev eth1 proto rip metric 20
100.1.1.0/24 dev eth3 proto kernel scope link src 100.1.1.1
100.1.2.0/24 nhid 13 via 100.1.0.2 dev eth2 proto rip metric 20
100.1.3.0/24 nhid 12 via 100.1.0.10 dev eth1 proto rip metric 20
100.1.4.0/24 nhid 14 via 100.1.0.14 dev eth0 proto rip metric 20
100.2.0.0/30 nhid 14 via 100.1.0.14 dev eth0 proto rip metric 20

root@r2:/# ip route
default nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
100.1.0.0/30 dev eth0 proto kernel scope link src 100.1.0.2
100.1.0.4/30 dev eth1 proto kernel scope link src 100.1.0.5
100.1.0.8/30 nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
100.1.0.12/30 nhid 10 via 100.1.0.1 dev eth0 proto rip metric 20
100.1.0.16/30 nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
100.1.1.0/24 nhid 10 via 100.1.0.1 dev eth0 proto rip metric 20
100.1.2.0/24 dev eth2 proto kernel scope link src 100.1.2.1
100.1.3.0/24 nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
100.1.4.0/24 nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
100.2.0.0/30 nhid 8 via 100.1.0.6 dev eth1 proto rip metric 20
```

13- ping 193.2004.161.1 -> verso qualsiasi destinazione (perché in altro dominio)

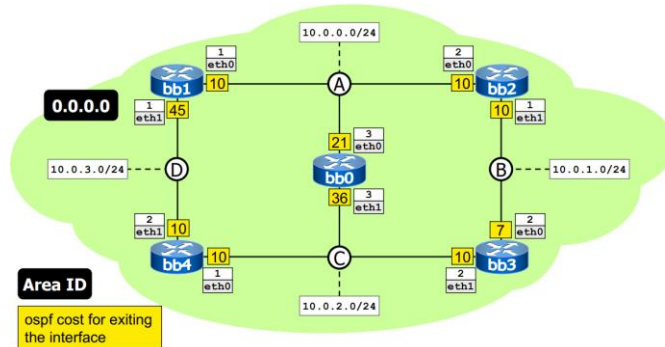
```
root@r1:/# ping 193.204.161.1
PING 193.204.161.1 (193.204.161.1) 56(84) bytes of data.
From 100.2.0.2 icmp_seq=1 Destination Net Unreachable
From 100.2.0.2 icmp_seq=2 Destination Net Unreachable
From 100.2.0.2 icmp_seq=3 Destination Net Unreachable
From 100.2.0.2 icmp_seq=4 Destination Net Unreachable
From 100.2.0.2 icmp_seq=5 Destination Net Unreachable
From 100.2.0.2 icmp_seq=6 Destination Net Unreachable
From 100.2.0.2 icmp_seq=7 Destination Net Unreachable
From 100.2.0.2 icmp_seq=8 Destination Net Unreachable
From 100.2.0.2 icmp_seq=9 Destination Net Unreachable
From 100.2.0.2 icmp_seq=10 Destination Net Unreachable
From 100.2.0.2 icmp_seq=11 Destination Net Unreachable
From 100.2.0.2 icmp_seq=12 Destination Net Unreachable
From 100.2.0.2 icmp_seq=13 Destination Net Unreachable
From 100.2.0.2 icmp_seq=14 Destination Net Unreachable
From 100.2.0.2 icmp_seq=15 Destination Net Unreachable
From 100.2.0.2 icmp_seq=16 Destination Net Unreachable
From 100.2.0.2 icmp_seq=17 Destination Net Unreachable
From 100.2.0.2 icmp_seq=18 Destination Net Unreachable
From 100.2.0.2 icmp_seq=19 Destination Net Unreachable
From 100.2.0.2 icmp_seq=20 Destination Net Unreachable
From 100.2.0.2 icmp_seq=21 Destination Net Unreachable
^C
--- 193.204.161.1 ping statistics ---
21 packets transmitted, 0 received, +21 errors, 100% packet loss, time 20827ms
```

14- se butto giù un'interfaccia faccio un'altra strada. Prima andavo diretto a r3 con un solo router, se taglio eth1 passo per eth0 dove trovo r4 e poi vado a r3.

**ip link set eth1 down**

```
root@r1:/# traceroute 100.1.0.10
traceroute to 100.1.0.10 (100.1.0.10), 30 hops max, 60 byte packets
 1 100.1.0.10 (100.1.0.10) 2.264 ms 2.113 ms 1.881 ms
root@r1:/# ip link set eth1 down
root@r1:/# traceroute 100.1.0.10
traceroute to 100.1.0.10 (100.1.0.10), 30 hops max, 60 byte packets
 1 100.1.0.14 (100.1.0.14) 2.331 ms 2.152 ms 2.057 ms
 2 100.1.0.10 (100.1.0.10) 1.911 ms 1.730 ms 1.588 ms
```

## OSPF SINGLE-AREA



- 1- **frr.config** -> per definire la configurazione di frr, lo faccio per ogni singolo router

```
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-sing
learea> cd bb0
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-sing
learea\bb0> cd .\etc\frr\
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-sing
learea\bb0\etc\frr> cat frr.conf

! FRRouting configuration file

!
! OSPF CONFIGURATION
!
! Default cost for exiting an interface is 10
interface eth0
ospf cost 21
interface eth1
ospf cost 36
!
router ospf
! Speak OSPF on all interfaces falling in the listed subnets
network 10.0.0.0/16 area 0.0.0.0
redistribute connected
!
log file /var/log/frr/frr.log
```

- 2- **lab.conf** -> configurazione della rete

```
bb0[0]=A
bb0[1]=C
bb0[image]="kathara/frr"

bb1[0]=A
bb1[1]=D
bb1[image]="kathara/frr"

bb2[0]=A
bb2[1]=B
bb2[image]="kathara/frr"

bb3[0]=B
bb3[1]=C
bb3[image]="kathara/frr"

bb4[0]=C
bb4[1]=D
bb4[image]="kathara/frr"
```

- 3- **bb0.startup** -> file di configurazione ip address

```
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-sing
learea> cat bb0.startup
ip address add 10.0.0.3/24 dev eth0
ip address add 10.0.2.3/24 dev eth1
systemctl start frr
```

- 4- **ip route** su bb1, dopo aver fatto **vttysh** (scelgo bb1 ma potrei scegliere qualsiasi router tanto con ospf è uguale per tutti)  
Vedo che ci sono varie rotte di cui due sono apprese direttamente o via ospf



```
bb1# sh ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, F - PBR,
       f - OpenFabric,
       > - selected route, * - FIB route, q - queued, r - rejected, b - backup
       t - trapped, o - offload failure

O 10.0.0.0/24 [110/10] is directly connected, eth0, weight 1, 00:01:11
C> 10.0.0.0/24 is directly connected, eth0, 00:01:12
O> 10.0.1.0/24 [110/20] via 10.0.0.2, eth0, weight 1, 00:00:16
O> 10.0.2.0/24 [110/30] via 10.0.0.2, eth0, weight 1, 00:00:16
O 10.0.3.0/24 [110/40] via 10.0.0.2, eth0, weight 1, 00:00:16
C> 10.0.3.0/24 is directly connected, eth1, 00:01:12
```

Il demone di zebra preferisce interfaccia direttamente connessa e non tramite ospf che magari fa un giro più lungo.

- 5- **sh ip ospf route** -> come il demone di ospf vede la rete, tutti in area 0 perché ho una sola area

```
bb1# sh ip ospf route
===== OSPF network routing table =====
N 10.0.0.0/24 [10] area: 0.0.0.0
   directly attached to eth0
N 10.0.1.0/24 [20] area: 0.0.0.0
   via 10.0.0.2, eth0
N 10.0.2.0/24 [30] area: 0.0.0.0
   via 10.0.0.2, eth0
N 10.0.3.0/24 [40] area: 0.0.0.0
   via 10.0.0.2, eth0

===== OSPF router routing table =====
R 10.0.1.1 [10] area: 0.0.0.0, ASBR
   via 10.0.0.2, eth0
R 10.0.2.2 [20] area: 0.0.0.0, ASBR
   via 10.0.0.2, eth0
R 10.0.2.3 [10] area: 0.0.0.0, ASBR
   via 10.0.0.3, eth0
R 10.0.3.2 [30] area: 0.0.0.0, ASBR
   via 10.0.0.2, eth0

===== OSPF external routing table =====
```

- 6- **sh ip ospf interface** -> mi dice le interfacce di quel router e mi dice che bb1 è il DR per la lan A perché eth0 is up

C'è anche eth1 in cui però è State Backup perché nell'elezione arriva dopo.

Il DR è l'identificatore della lan e viene usato il numero IP dell'interfaccia.

Posso farlo su ogni macchina e vedere i DR per identificare le lan.

```
bb1# sh ip ospf interface
eth0 is up
  ifindex 13, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.0.1/24, Broadcast 10.0.0.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 10.0.3.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.0.3.1 Interface Address 10.0.0.1/24
  Backup Designated Router (ID) 10.0.2.3, Interface Address 10.0.0.3
  Saved Network-LSA sequence number 0x80000002
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
  Hello due in 5.987s
  Neighbor Count is 2, Adjacent neighbor count is 2
  Graceful Restart hello delay: 10s
eth1 is up
  ifindex 23, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.3.1/24, Broadcast 10.0.3.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 10.0.3.1, Network Type BROADCAST, Cost: 45
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 10.0.3.2 Interface Address 10.0.3.2/24
  Backup Designated Router (ID) 10.0.3.1, Interface Address 10.0.3.1
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
  Hello due in 5.987s
  Neighbor Count is 1, Adjacent neighbor count is 1
  Graceful Restart hello delay: 10s
bb2# sh ip ospf interface
eth0 is up
  ifindex 17, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.0.2/24, Broadcast 10.0.0.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 10.0.1.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DROther, Priority 1
  Designated Router (ID) 10.0.3.1 Interface Address 10.0.0.1/24
  Backup Designated Router (ID) 10.0.2.3, Interface Address 10.0.0.3
  Multicast group memberships: OSPFAllRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
  Hello due in 4.031s
  Neighbor Count is 2, Adjacent neighbor count is 2
  Graceful Restart hello delay: 10s
eth1 is up
  ifindex 27, MTU 1500 bytes, BW 10000 Mbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.0.1.1/24, Broadcast 10.0.1.255, Area 0.0.0.0
  MTU mismatch detection: enabled
  Router ID 10.0.1.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 10.0.2.2 Interface Address 10.0.1.2/24
  Backup Designated Router (ID) 10.0.1.1, Interface Address 10.0.1.1
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
  Hello due in 4.031s
  Neighbor Count is 1, Adjacent neighbor count is 1
  Graceful Restart hello delay: 10s
```

bb1 è DR per  
10.0.0.1

bb2 è DR per  
10.0.0.2

- 7- **sh ip ospf database** -> mi fornisce il link state database che è uguale per tutte le macchine, se non ospf non funziona (mi serve per ricostruire la topologia della rete)
- OSPF Router with ID** -> come vengono identificati i router dentro ospf
- Router Link States** (Area 0.0.0.0) -> nomi dei router
- Net Link States** (Area 0.0.0.0) -> nomi delle lan con DR, le lan diventano nodi e non archi

```
bb1# sh ip ospf database

      OSPF Router with ID (10.0.3.1)

          Router Link States (Area 0.0.0.0)

Link ID      ADV Router   Age  Seq#       CkSum  Link count
10.0.1.1     10.0.1.1     1558 0x80000009 0xdd01 2
10.0.2.2     10.0.2.2     1560 0x80000009 0xd701 2
10.0.2.3     10.0.2.3     1538 0x8000000b 0xd7d5 2
10.0.3.1     10.0.3.1     1538 0x8000000a 0x248f 2
10.0.3.2     10.0.3.2     1577 0x80000009 0x3e92 2

          Net Link States (Area 0.0.0.0)

Link ID      ADV Router   Age  Seq#       CkSum
10.0.0.1     10.0.3.1     1528 0x80000003 0x67aa
10.0.1.2     10.0.2.2     1540 0x80000002 0x67bc
10.0.2.1     10.0.3.2     1537 0x80000003 0x709b
10.0.3.2     10.0.3.2     1567 0x80000002 0x69b4
```

- 8- **sh ip ospf database router** -> per creare gli archi e completare la rete

```
bb1# sh ip ospf database router

      OSPF Router with ID (10.0.3.1)

          Router Link States (Area 0.0.0.0)

LS age: 109
Options: 0x2 : *|---|---|E|
LS Flags: 0x6
Flags: 0x2 : ASBR
LS Type: router-LSA
Link State ID: 10.0.1.1
Advertising Router: 10.0.1.1
LS Seq Number: 8000000a
Checksum: 0xdb02
Length: 48

Number of Links: 2

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.0.1
(Link Data) Router Interface address: 10.0.0.2
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.1.2

LS age: 151
Options: 0x2 : *|---|---|E|
LS Flags: 0x6
Flags: 0x2 : ASBR
LS Type: router-LSA
Link State ID: 10.0.2.2
Advertising Router: 10.0.2.2
LS Seq Number: 8000000a
Checksum: 0xd502
Length: 48

Number of Links: 2

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.1.2
(Link Data) Router Interface address: 10.0.1.2
Number of TOS metrics: 0
TOS 0 Metric: 7

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.2.1
(Link Data) Router Interface address: 10.0.2.2
Number of TOS metrics: 0
TOS 0 Metric: 10
```

Archi per bb2

Archi per bb3

```

LS age: 139
Options: 0x2 : *|---|---|E|
LS Flags: 0x6
Flags: 0x2 : ASBR
LS Type: router-LSA
Link State ID: 10.0.2.3
Advertising Router: 10.0.2.3
LS Seq Number: 8000000c
Checksum: 0xd5d6
Length: 48

Number of Links: 2

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.0.1
(Link Data) Router Interface address: 10.0.0.3
Number of TOS metrics: 0
TOS 0 Metric: 21

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.0.2.1
(Link Data) Router Interface address: 10.0.2.3
Number of TOS metrics: 0
TOS 0 Metric: 36

```

Archi per bb0

MANCANO bb1 e bb4 MA è UGUALE

9- **sh ip ospf database network** -> mi dice per ogni lan quali sono i collegamenti che ha

```

bb1# sh ip ospf database network

    OSPF Router with ID (10.0.3.1)

        Net Link States (Area 0.0.0.0)

LS age: 396
Options: 0x2 : *|---|---|E|
LS Flags: 0x3
LS Type: network-LSA
Link State ID: 10.0.0.1 (address of Designated Router)
Advertising Router: 10.0.3.1
LS Seq Number: 80000004
Checksum: 0x65ab
Length: 36

Network Mask: /24
Attached Router: 10.0.3.1

Attached Router: 10.0.1.1

Attached Router: 10.0.2.3

```

SCENDENDO NELLA CONFIGURAZIONE TROVO LE ALTRE

10- **traceroute** da bb1 a 10.0.2.1 (bb4) -> “giro de peppe”, perché pago meno 45>30

```

bb1# traceroute 10.0.2.1
traceroute to 10.0.2.1 (10.0.2.1), 30 hops max, 60 byte packets
 1  10.0.0.2 (10.0.0.2)  1.718 ms  4.810 ms  4.770 ms
 2  10.0.1.2 (10.0.1.2)  5.120 ms  5.037 ms  4.972 ms
 3  10.0.2.1 (10.0.2.1)  4.925 ms  4.891 ms  4.872 ms

```

11- vado su bb3, spengo eth0 -> per arrivare a bb4 passo da bb0

**ip link set eth1 down**

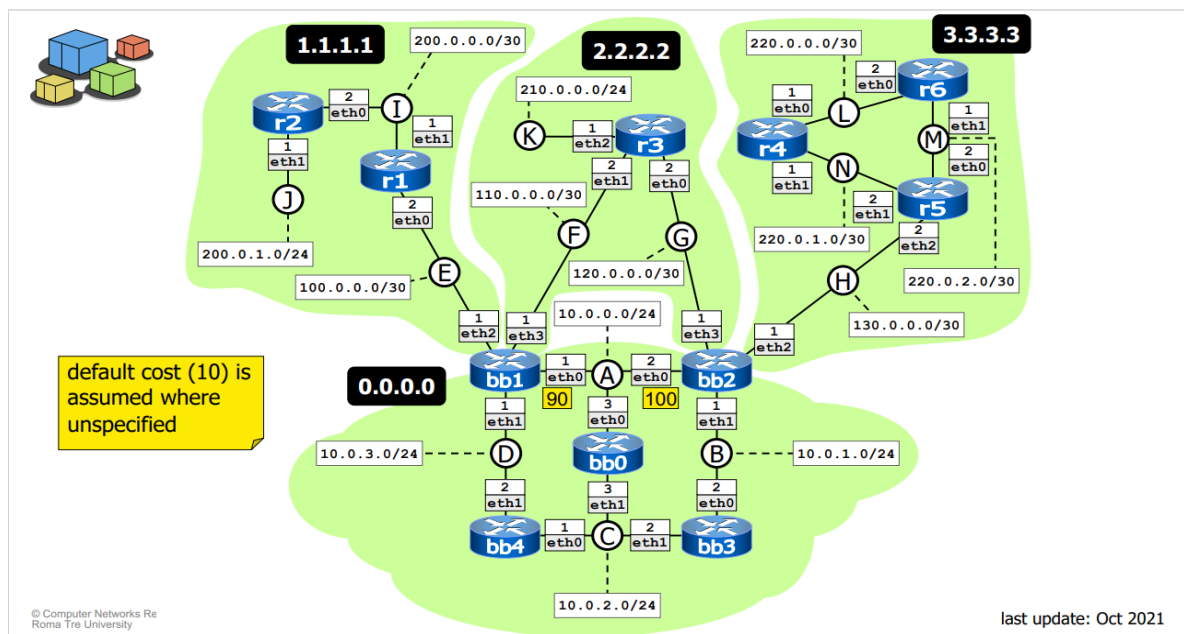
```

bb1# traceroute 10.0.2.1
traceroute to 10.0.2.1 (10.0.2.1), 30 hops max, 60 byte packets
 1  10.0.0.3 (10.0.0.3)  1.331 ms  1.141 ms  1.082 ms
 2  10.0.2.1 (10.0.2.1)  1.035 ms  0.949 ms  0.815 ms

```

## OSPF MULTI-AREA

L'area di backbone vede le sottoaree in maniera aggregata. Sa solo la loro esistenza e il prefisso aggregato. Ogni area si calcola il suo link state database e lo manda all'area 0. L'area 0 manda alle sottoaree la default.



- 1- `cd /etc/frr/frr.config` -> devo specificare i tipi di aree in base a dove appartengono i router

bb1, router di frontiera è collegato con area 0-1-2

```
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-mult
iarea\bb1\etc\frr> cat frr.conf
!
! FRRouting configuration file
!
!
! OSPF CONFIGURATION
!
interface eth0
ospf cost 90
!
router ospf
! Speak OSPF on all interfaces falling in the listed subnets
network 10.0.0.0/16 area 0.0.0.0
network 100.0.0.0/30 area 1.1.1.1
network 110.0.0.0/30 area 2.2.2.2
area 1.1.1.1 stub
area 2.2.2.2 stub
redistribute connected
!
!
log file /var/log/frr/frr.log
```

bb2, collegato con area 0-2-3

```
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\intradomain-routing\frr\ospf\kathara-lab_ospf\kathara-lab_ospf_frr-mult
iarea\bb2\etc\frr> cat frr.conf
!
! FRRouting configuration file
!
!
! OSPF CONFIGURATION
!
interface eth0
ospf cost 100
!
router ospf
! Speak OSPF on all interfaces falling in the listed subnets
network 10.0.0.0/16 area 0.0.0.0
network 120.0.0.0/30 area 2.2.2.2
network 130.0.0.0/30 area 3.3.3.3
area 2.2.2.2 stub
area 3.3.3.3 stub
redistribute connected
!
!
log file /var/log/frr/frr.log
```

- 2- `lab.conf` -> configurazione rete

```
bb0[0]=A
bb0[1]=C
bb0[image]="kathara/frr"

bb1[0]=A
bb1[1]=D
bb1[2]=E
bb1[3]=F
bb1[image]="kathara/frr"

bb2[0]=A
bb2[1]=B
bb2[2]=H
bb2[3]=G
bb2[image]="kathara/frr"

bb3[0]=B
bb3[1]=C
bb3[image]="kathara/frr"

bb4[0]=C
bb4[1]=D
bb4[image]="kathara/frr"
```

E' TUTTO UGUALE A SINGLE AREA IL RESTO

- 3- **show ip ospf neighbor** -> per vedere i vicini

```
r2# show ip ospf neighbor
Neighbor ID    Pri State           Up Time         Dead Time Address      Interface      RXmtL Rqs
200.0.0.1      1 ExStart/Backup    0.320s         32.141s 200.0.0.1     eth0:200.0.0.2 0
```

- 4- **show ip ospf database router** -> per definire sempre gli archi

```
OSPF Router with ID (200.0.1.1)

Router Link States (Area 1.1.1.1 [Stub])

LS age: 131
Options: 0x0 : *|-|-|-|-|-
LS Flags: 0x6
Flags: 0x1 : ABR
LS Type: router-LSA
Link State ID: 110.0.0.1
Advertising Router: 110.0.0.1
LS Seq Number: 80000006
Checksum: 0x2b70
Length: 36

Number of Links: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 100.0.0.2
(Link Data) Router Interface address: 100.0.0.1
Number of TOS metrics: 0
TOS 0 Metric: 10
```

- 5- **show ip ospf database network** -> per Vedere ogni lan che collegamenti ha  
L'importante è che nel database ci siano sia i router propri di quella lan che i router di frontiera.

```
OSPF Router with ID (200.0.1.1)

Net Link States (Area 1.1.1.1 [Stub])

LS age: 286
Options: 0x0 : *|-|-|-|-|-
LS Flags: 0x6
LS Type: network-LSA
Link State ID: 100.0.0.2 (address of Designated Router)
Advertising Router: 200.0.0.1
LS Seq Number: 80000001
Checksum: 0x09ec
Length: 32

Network Mask: /30
Attached Router: 110.0.0.1
Attached Router: 200.0.0.1
```

Dalla macchina r2 che appartiene a 1.1.1.1, avrò r1, r2 e bb1 come router collegati da cui vedo gli archi. Questo schema è lo stesso che avrei se digitassi il comando su r1 o su un'altra macchina della rete (seguono altre due macchine)

```

OSPF Router with ID (210.0.0.1)

      Router Link States (Area 2.2.2.2 [Stub])

LS age: 762
Options: 0x0 : *|---|---|---|
LS Flags: 0x6
Flags: 0x1 : ABR
LS Type: router-LSA
Link State ID: 110.0.0.1
Advertising Router: 110.0.0.1
LS Seq Number: 80000005
Checksum: 0xe1a6
Length: 36

Number of Links: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 110.0.0.2
(Link Data) Router Interface address: 110.0.0.1
Number of TOS metrics: 0
TOS 0 Metric: 10

```

```

Link State ID: 210.0.0.1
Advertising Router: 210.0.0.1
LS Seq Number: 8000000a
Checksum: 0xd2ec
Length: 60

Number of Links: 3

Link connected to: Stub Network
(Link ID) Net: 210.0.0.0
(Link Data) Network Mask: 255.255.255.0
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: a Transit Network
(Link ID) Designated Router address: 110.0.0.2
(Link Data) Router Interface address: 110.0.0.2
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: a Transit Network
(Link ID) Designated Router address: 120.0.0.2
(Link Data) Router Interface address: 120.0.0.2
Number of TOS metrics: 0
TOS 0 Metric: 10

```

Dalla macchina r3 che appartiene a 2.2.2.2, avrò r3,bb2 e bb1 come router collegati da cui vedo gli archi. Questo schema è lo stesso che avrei se digitassi il comando su bb1 o su un'altra macchina della rete (seguono altre due macchine)

SONO COLLEGATE SEMPRE CON L'INTERFACCIA DOPO IL NODO

## DNS

Il Domain Name System (DNS) è un sistema fondamentale per il funzionamento di Internet. Serve a tradurre gli indirizzi IP (Internet Protocol) numerici in nomi di dominio leggibili e viceversa. Ecco una spiegazione di cosa è il DNS e come funziona:

### **\*\*Cosa è il DNS:\*\***

Il DNS è un sistema di distribuzione gerarchica e distribuita per la risoluzione di nomi di dominio. Questi nomi di dominio sono stringhe leggibili dagli esseri umani (come "www.google.com") che vengono utilizzate per identificare risorse su Internet, come siti web, server di posta elettronica e altro. Il DNS traduce questi nomi di dominio in indirizzi IP (come "192.168.1.1") che i computer possono utilizzare per localizzare le risorse online.

### **\*\*Come funziona il DNS:\*\***

Il DNS opera in modi simili a una rubrica telefonica in cui puoi cercare il nome di una persona e trovare il suo numero di telefono associato. Ecco come funziona in dettaglio:

1. **\*\*Richiesta DNS\*\***: Quando un utente digita un nome di dominio nella barra del browser o avvia una richiesta su Internet, il suo dispositivo avvia una richiesta DNS per tradurre quel nome di dominio in un indirizzo IP.
2. **\*\*Cache locale\*\***: Il dispositivo verifica prima la sua cache locale DNS per vedere se ha già memorizzato l'associazione nome di dominio / indirizzo IP richiesta. Questo aiuta a ridurre il carico sui server DNS.
3. **\*\*Server DNS locale\*\***: Se il nome di dominio non è presente nella cache locale, il dispositivo invia una richiesta al server DNS locale o al server DNS del provider Internet.
4. **\*\*Server DNS radice\*\***: Se il server DNS locale non ha l'informazione richiesta nella sua cache, inizia la ricerca inviando una richiesta al server DNS radice. Questi server rappresentano il livello più alto della gerarchia DNS.
5. **\*\*Server DNS di dominio di primo livello (TLD)\*\***: Il server DNS radice instrada quindi la richiesta al server DNS del dominio di primo livello (TLD) appropriato. I TLD includono estensioni come .com, .org, .net e così via.
6. **\*\*Server DNS autoritario\*\***: Il server TLD reindirizza la richiesta al server DNS autoritario del dominio specifico (come ad esempio il server DNS di Google per il dominio google.com).
7. **\*\*Risoluzione finale\*\***: Il server DNS autoritario risponde con l'indirizzo IP associato al nome di dominio richiesto.
8. **\*\*Cache locale e risposta al dispositivo\*\***: Il server DNS locale memorizza l'associazione tra il nome di dominio e l'indirizzo IP nella sua cache locale e invia la risposta al dispositivo dell'utente.
9. **\*\*Accesso alla risorsa richiesta\*\***: Ora che il dispositivo ha l'indirizzo IP, può contattare il server corretto per accedere alla risorsa richiesta su Internet.



Nuove macchine definite come *dnsroot*, *dnsorg* che sono autorità rispettivamente su root e org. *localuni* e *localstart* sono resolver locali usati dalle macchine pc, a cui si affidano per scoprire i nomi di interesse.

- 1- **resolv.conf** -> specifica del default name server, resolver locale

```
PS C:\Users\39339\desktop\Kathara-Labs\main-labs\application-level\dns\kathara-lab_dns\pc1\etc> cat resolv.conf
nameserver 192.168.0.110
search uniroma3.it
```

- 2- **named.conf** -> mi dice dove trovare le info riguardo al root ns e mi dice anche che siamo autorità per una certa zona, configurazione del demone named

```
root@dnsuni:/# cat /etc/bind/named.conf
include "/etc/bind/named.conf.options";

zone "." {
    type hint;
    file "/etc/bind/db.root";
};

zone "uniroma3.it" {
    type master;
    file "/etc/bind/db.it.uniroma3";
};
```

- 3- **named.conf.options** -> serve per definire la cache del dns dentro la quale vedo se ha già memorizzato l'associazione nome di dominio / indirizzo IP richiesta

```
root@dnsuni:/# cat /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";
};
```

- 4- **db.root** -> per vedere la configurazione di un resource record

```
root@dnsuni:/# cat /etc/bind/db.root
.                IN      NS      ROOT-SERVER.
ROOT-SERVER.     IN      A      192.168.0.5
```

il dominio "." ha come nameserver una macchina di nome ROOT-SERVER. Il secondo record associa un indirizzo IP al nome della macchina (glue record).

- 5- **db.it.uniroma3** -> mi dice per quanto tempo un record rimane in cache

```
root@dnsuni:/# cat /etc/bind/db.it.uniroma3
$TTL      60000
@         IN      SOA      dnsuni.uniroma3.it.  root.dnsuni.uniroma3.it. (
                                2006031201 ; serial
                                28 ; refresh
                                14 ; retry
                                3600000 ; expire
                                0 ; negative cache ttl
                                )
@         IN      NS      dnsuni.uniroma3.it.
dnsuni.uniroma3.it.  IN      A      192.168.0.11
pc1.uniroma3.it.    IN      A      192.168.0.111
localuni.uniroma3.it. IN      A      192.168.0.110
```

Il file si riferisce a uniroma3.it

Il primo RR indica che il server DNS autoritario per il dominio "uniroma3.it" è "dnsuni.uniroma3.it". Il dominio "uniroma3.it" è responsabile per la gestione dei record DNS associati a questo dominio e per la risoluzione dei nomi all'interno di questa zona. Dopo che ho trovato il ns per uniroma3.it posso trovare l'indirizzo ip corrispondente così che quando qualcuno tenta di accedere verrà indirizzato lì.

- 6- **db.it** -> mi dice le stesse info di sopra ma per il dominio it.

```
root@dnsit:/# cat /etc/bind/db.it
$TTL 60000
@      IN      SOA     dnsit.it.  root.dnsit.it. (
        2006031201 ; serial
        28800 ; refresh
        14400 ; retry
        3600000 ; expire
        0 ; negative cache ttl
)
@      IN      NS      dnsit.it.
dnsit.it.  IN      A      192.168.0.1
uniroma3.it.  IN      NS      dnsuni.uniroma3.it.
dnsuni.uniroma3.it.  IN      A      192.168.0.11
```

il nameserver di it delega il controllo di uniroma3.it al nameserver autorità su quel dominio, aggiunge il gluerecord con indirizzo IP.

- 7- vado su pc1 e faccio **ping pc2.startup.net**

```
root@pc1:/# ping pc2.startup.net
PING pc2.startup.net (192.168.0.222) 56(84) bytes of data.
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=1 ttl=64 time=2.25 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=2 ttl=64 time=0.176 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=3 ttl=64 time=0.144 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=4 ttl=64 time=0.146 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=5 ttl=64 time=0.170 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=6 ttl=64 time=0.192 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=7 ttl=64 time=0.175 ms
64 bytes from 192.168.0.222 (192.168.0.222): icmp_seq=8 ttl=64 time=0.130 ms
```

- 8- **dig pc2.startup.net** -> query al dns

```
root@pc1:/# dig +trace pc2.startup.net

; <<>> DiG 9.18.19-1~deb12u1-Debian <<>> +trace pc2.startup.net
;; global options: +cmd
.                59916      IN      NS      ROOT-SERVER.
;; Received 80 bytes from 192.168.0.110#53(192.168.0.110) in 0 ms

net.             60000      IN      NS      dnsnet.net.
;; Received 112 bytes from 192.168.0.5#53(ROOT-SERVER) in 10 ms

startup.net.     60000      IN      NS      dnsstart.startup.net.
;; Received 111 bytes from 192.168.0.2#53(dnsnet.net) in 10 ms

pc2.startup.net. 60000      IN      A      192.168.0.222
startup.net.     60000      IN      NS      dnsstart.startup.net.
;; Received 127 bytes from 192.168.0.22#53(dnsstart.startup.net) in 10 ms
```

Faccio prima una richiesta a root per sapere qual è il ns di default, mi dice che è net e quindi trovo il ns che gestisce net. Stessa cosa per startup fino a che non traduco pc2.startup.net in IP così che e chi accede viene reindirizzato lì.

## WEB SERVER

- 1- vado su server e faccio partire il demone apache2 per la creazione di pagine web con html

```
root@server:/# systemctl start apache2
root@server:/# ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.3  0.0   4604   3676 pts/0    Ss+  09:54   0:00 bash
root        45  0.0  0.0   6560   4600 ?        Ss   09:54   0:00 /usr/sbin/apache2 -k start
www-data   46  0.0  0.1 1998920 12732 ?        Sl   09:54   0:00 /usr/sbin/apache2 -k start
www-data   47  0.0  0.1 1998920 12712 ?        Sl   09:54   0:00 /usr/sbin/apache2 -k start
root       120  0.3  0.0   4604   3708 pts/1    Ss   09:54   0:00 /bin/bash
root       136 100  0.0   8532   4312 pts/1    R+   09:55   0:00 ps aux
root@server:/#
```

- 2- in `/var/www/html/index.html` -> ho la pagina che mostro e posso modificare
- 3- vado su client e faccio `links http://10.0.0.1`, per visualizzare la pagina web

