

CONFIGURAZIONE DELLA RETE

Si è scelto di utilizzare un peso pari a undici sulle interfacce ethernet 1 del router 5 e ethernet 3 del router 6, per forzare il protocollo OSPF a scegliere come percorso quello indicato dai requisiti di progetto.

File .startup

```

r5.startup
1 ifconfig eth0 10.1.0.17 netmask 255.255.255.252
2 ifconfig eth1 10.1.0.13 netmask 255.255.255.252
3 ifconfig eth2 10.1.0.10 netmask 255.255.255.252
4 ifconfig eth3 10.1.0.6 netmask 255.255.255.252
5
6 /etc/init.d/zebra start
7
8 route add default gw 10.1.0.18
9 route add -net 10.1.0.24 netmask 255.255.255.248 gw 10.1.0.14
10 route add -net 10.1.0.32 netmask 255.255.255.252 gw 10.1.0.14
11 route add -net 10.1.1.128 netmask 255.255.255.128 gw 10.1.0.14

r6.startup
1 ifconfig eth0 10.1.0.21 netmask 255.255.255.252
2 ifconfig eth1 10.1.0.30 netmask 255.255.255.252
3 ifconfig eth2 10.1.0.26 netmask 255.255.255.252
4 ifconfig eth3 10.1.0.14 netmask 255.255.255.252
5
6 /etc/init.d/zebra start
7
8 route add default gw 10.1.0.22
9 route add -net 10.1.0.0 netmask 255.255.255.240 gw 10.1.0.13
10 route add -net 10.1.1.0 netmask 255.255.255.128 gw 10.1.0.13

r7.startup
1 ifconfig eth0 10.2.0.1 netmask 255.255.255.252
2 ifconfig eth1 10.1.0.37 netmask 255.255.255.252
3 ifconfig eth2 10.1.0.18 netmask 255.255.255.252
4 ifconfig eth3 10.1.0.65 netmask 255.255.255.192
5
6 /etc/init.d/zebra start
7
8 route add default gw 10.2.0.2

r8.startup
1 ifconfig eth0 10.2.0.5 netmask 255.255.255.252
2 ifconfig eth1 10.1.0.129 netmask 255.255.255.192
3 ifconfig eth2 10.1.0.22 netmask 255.255.255.252
4 ifconfig eth3 10.1.0.38 netmask 255.255.255.252
5
6 /etc/init.d/zebra start
7
8 route add default gw 10.2.0.6
  
```

File .conf e daemons del router 5

```

daemons
r5 > etc > quagga > daemons
1 # This file tells the quagga package
2 # which daemons to start.
3 # Entries are in the format: <daemon>=(yes|no|priority)
4 # where 'yes' is equivalent to infinitely low priority, and
5 # lower numbers mean higher priority. Read
6 # /usr/doc/quagga/README.Debian.gz for details.
7 # Daemons are: bgpd zebra ospfd ospf6d ripd ripngd
8 zebra=yes
9 bgpd=no
10 ospfd=yes
11 ospf6d=no
12 ripd=yes
13 ripngd=no

zebra.conf
r5 > etc > quagga > zebra.conf
1 ! -- zebra --
2 !
3 ! zebra configuration file
4 !
5 hostname r5
6 password zebra
7 enable password zebra
8 !
9 ! Static default route sample.
10 !
11 !ip route 0.0.0.0/0 203.181.89.241
12 !
13 log file /var/log/quagga/zebra.log
14

ripd.conf
r5 > etc > quagga > ripd.conf
1 !
2 hostname ripd
3 password zebra
4 enable password zebra
5 !
6 router rip
7 redistribute connected
8 network 10.1.0.0/28
9 route 0.0.0.0/0
10 route 10.1.0.32/30
11 route 10.1.0.24/29
12 route 10.1.1.128/25
13 !
14 log file /var/log/quagga/ripd.log

ospfd.conf
r5 > etc > quagga > ospfd.conf
1 !
2 hostname ospfd
3 password zebra
4 enable password zebra
5 !
6 router ospf
7 ! Speak OSPF on all interfaces falling in 10.0.0.0/16
8 network 10.1.0.0/27 area 0.0.0.0
9 redistribute connected
10 redistribute kernel
11 interface eth1
12 ip ospf cost 11
13 !
14 log file /var/log/zebra/ospfd.log
  
```

File .conf e daemons del router 7

```

daemons
r7 > etc > quagga > daemons
1 # This file tells the quagga package
2 # which daemons to start.
3 # Entries are in the format: <daemon>=(yes|no|priority)
4 # where 'yes' is equivalent to infinitely low priority, and
5 # lower numbers mean higher priority. Read
6 # /usr/doc/quagga/README.Debian.gz for details.
7 # Daemons are: bgpd zebra ospfd ospf6d ripd ripngd
8 zebra=yes
9 bgpd=no
10 ospfd=yes
11 ospf6d=no
12 ripd=no
13 ripngd=no

ospfd.conf
r7 > etc > quagga > ospfd.conf
1 !
2 hostname ospfd
3 password zebra
4 enable password zebra
5 !
6 router ospf
7 ! Speak OSPF on all interfaces falling in 10.0.0.0/16
8 network 10.1.0.0/26 area 0.0.0.0
9 redistribute connected
10 redistribute kernel
11 default-information originate
12 !
13 log file /var/log/zebra/ospfd.log
  
```

DESIGNATED ROUTER DELLA RETE 10.1.0.36/30

Come si evince dall'immagine mostrata, l'Interface Address del Designated Router è 10.1.0.38, appartenente all'interfaccia ethernet 3 del router r8 e ciò viene confermato dalla presenza di State DR.

```
eth3 is up
  ifindex 151, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 10.1.0.38/30, Broadcast 10.1.0.39, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 10.2.0.5, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 10.2.0.5, Interface Address 10.1.0.38
  Backup Designated Router (ID) 10.2.0.1, Interface Address 10.1.0.37
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
  Hello due in 2.156s
  Neighbor Count is 1, Adjacent neighbor count is 1
```

RICOSTRUZIONE DELLA TOPOLOGIA DELLA ZONA OSPF

Tramite il comando `SHOW IP OSPF DATABASE ROUTER` è stato possibile estrapolare le informazioni dei router appartenenti alla zona OSPF, come mostrato dalle seguenti immagini. Partendo dal router r5, sono individuate le reti a cui esso è collegato. Per ultimo, verrà analizzato il router r8, poiché non direttamente connesso al router di partenza. Prendendo in esame l'immagine del router r5 (individuabile dal Link State ID), si nota come esso è direttamente connesso a due reti appartenenti alla zona OSPF (individuabili dal Link ID del designated router). Le restanti due, sono viste come stub network. Analizzando successivamente l'immagine del router r7, si nota come anch'esso sia direttamente connesso a due reti appartenenti alla zona OSPF, una delle quali lo collega al router r5. Infatti, è possibile individuare la rete a cui due router sono connessi quando i rispettivi designated router address coincidono. Analogamente a quanto sopracitato, si analizzano i router r6 e r8. Una volta terminata la fase di analisi del database router, sarà possibile "ricostruire" parzialmente una mappa della zona OSPF. Per concludere la ricostruzione di tale mappa, sarà necessario invocare il comando `SHOW IP OSPF DATABASE NETWORK`. Questo comando rende possibile l'assegnazione degli indirizzi delle LAN, i quali vengono identificati grazie alla netmask riportata da esso.

Database Router

```
LS age: 670
Options: 0x2 : *|-----|E|*
LS Flags: 0x3
Flags: 0x2 : ASBR
LS Type: router-LSA
Link State ID: 10.1.0.17
Advertising Router: 10.1.0.17
LS Seq Number: 8000000a
Checksum: 0xad85
Length: 72
Number of Links: 4

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.0.18
(Link Data) Router Interface address: 10.1.0.17
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.0.14
(Link Data) Router Interface address: 10.1.0.13
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: Stub Network
(Link ID) Net: 10.1.0.8
(Link Data) Network Mask: 255.255.255.252
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: Stub Network
(Link ID) Net: 10.1.0.4
(Link Data) Network Mask: 255.255.255.252
Number of TOS metrics: 0
TOS 0 Metric: 10
```

```
LS age: 667
Options: 0x2 : *|-----|E|*
LS Flags: 0x6
Flags: 0x2 : ASBR
LS Type: router-LSA
Link State ID: 10.2.0.1
Advertising Router: 10.2.0.1
LS Seq Number: 80000007
Checksum: 0x5a19
Length: 48
Number of Links: 2

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.0.38
(Link Data) Router Interface address: 10.1.0.37
Number of TOS metrics: 0
TOS 0 Metric: 10

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.0.18
(Link Data) Router Interface address: 10.1.0.18
Number of TOS metrics: 0
TOS 0 Metric: 10
```

Database Network

```

LS age: 18
Options: 0x2 : *|---|---|E|*
LS Flags: 0x6
LS Type: network-LSA
Link State ID: 10.1.0.14 (address of Designated Router)
Advertising Router: 10.1.0.30
LS Seq Number: 80000001
Checksum: 0x567f
Length: 32
Network Mask: /30
    Attached Router: 10.1.0.17
    Attached Router: 10.1.0.30

```

ANALISI DEI TEMPI DI RIPRISTINO DEL SERVIZIO

Come suggerito, si è scelto di creare due host h1 e h2 (rispettivamente collegati alle LAN A e D) per effettuare le rispettive analisi dei tempi di ripristino.

CASO 1: GUASTO AL COLLEGAMENTO ROUTER R1 – ROUTER R5

Per l'analisi di questo caso, si è scelto di effettuare il ping dall'host h1 al router r7 e la cattura sull'interfaccia ethernet 3 di quest'ultimo. I pacchetti inviati dall'host h1 passano inizialmente per i router r1 e r5. Come mostrato dalla cattura (/SHARED/LINKFAULT_R1_R5.PCAP), al verificarsi del guasto sul collegamento tra i router r1 e r5, il protocollo RIP impiega circa venticinque secondi ad individuare un nuovo percorso.

CASO 2: GUASTO AL ROUTER R7

Per l'analisi del secondo caso, si è scelto di utilizzare due host h1 e h2 per effettuare il ping tra essi e la cattura sull'interfaccia ethernet 0 di h2. Questa scelta è dovuta ai requisiti di progetto, poiché dalla zona RIP1, si raggiunge la LAN D passando per i router r5, r7 e r8. Come mostrato dalla cattura (/SHARED/R7FAULT.PCAP), al verificarsi del guasto del router r7 (forzato da terminale tramite il comando `DOCKER STOP ID_CONTAINER`), il protocollo OSPF impiega circa trentaquattro secondi ad individuare un nuovo percorso. Il calcolo del nuovo percorso è verificato dalle seguenti immagini.

Traceroute precedente e successiva al guasto del router 7

```

root@h1:/# traceroute 10.1.0.130
traceroute to 10.1.0.130 (10.1.0.130), 64 hops max
 1  10.1.1.1  0.004ms  0.004ms  0.004ms
 2  10.1.0.6  0.005ms  0.004ms  0.005ms
 3  10.1.0.18 0.008ms  0.003ms  0.004ms
 4  10.1.0.22 0.011ms  0.004ms  0.005ms
 5  10.1.0.130 0.004ms  0.004ms  0.004ms

```

```

root@h1:/# traceroute 10.1.0.130
traceroute to 10.1.0.130 (10.1.0.130), 64 hops max
 1  10.1.1.1  0.005ms  0.003ms  0.002ms
 2  10.1.0.6  0.022ms  0.003ms  0.003ms
 3  10.1.0.14 0.004ms  0.004ms  0.004ms
 4  10.1.0.22 0.004ms  0.002ms  0.003ms
 5  10.1.0.130 0.008ms  0.004ms  0.004ms

```