



Politecnico
di Torino

Tecnologie e Servizi di Rete

Computer Engineering

Marco Lampis

28 novembre 2022

Indice

0	Informazioni	1
1	IPv4 Summary	3
1.1	Indirizzi speciali	3
1.2	Indirizzamento IP con classi	3
1.3	Indirizzamento IP senza classi (CIDR)	3
1.4	IP routing	4
1.5	IP addressing methodology	6
1.5.1	Esercizio 1	7
1.5.2	Esercizio 2	8
1.5.3	Esercizio 3	8
1.5.4	Esercizio 4	9
1.5.5	Esercizio 5	10
1.5.6	Esercizio 6	11
1.5.7	Esercizio 7	12
1.5.8	Esercizio 8	14

0 Informazioni

I seguenti appunti sono stati presi nell'anno accademico 2022-2023 durante il corso di Tecnologie e Servizi di Rete.

Il materiale non è ufficiale e non è revisionato da alcun docente, motivo per cui non mi assumo responsabilità per eventuali errori o imprecisioni.

Per qualsiasi suggerimento o correzione non esitate a contattarmi.

E' possibile riutilizzare il materiale con le seguenti limitazioni:

- Utilizzo non commerciale
- Citazione dell'autore
- Riferimento all'opera originale

E' per tanto possibile:

- Modificar parzialmente o interamente il contenuto

Questi appunti sono disponibili su GitHub al seguente link:

1 https://github.com/Guray00/polito_lectures/tree/main/Tecnologie%20e%20Servizi%20di%20Rete



Figura 1: Repository GitHub

1 IPv4 Summary

In questo capitolo viene fatto un ripasso generico su quanto visto nei corsi precedenti, con particolare riferimento a Reti Informatiche (o equivalenti).

In ogni sottorete tutti i dispositivi che ne fanno parte avranno lo stesso indirizzo ip.

1.1 Indirizzi speciali

- tutti i bit a 1: indirizzo di broadcast, non può essere assegnato
- 127 . x . x . x: indirizzo di loopback, è una classe di indirizzi e servono a identificare l'host stesso e per tale motivo vengono solitamente utilizzate a scopo di debug.

Spesso oggi giorno non è consentito l'invio di messaggi in broadcast per motivi di sicurezza.

1.2 Indirizzamento IP con classi

Le rappresentazioni possono essere classes (a classe) o classness (senza l'utilizzo di classi). In particolare esistono di tre tipologie:

- **A:** prevede i primi 8 bit per l'indirizzo di rete, i rimanenti sono per identificare i dispositivi. Il totale degli indirizzi è 2^7 per la rete e 2^{24} per i dispositivi. Si possono avere 128 networks.
- **B:** 2 bit per la classe, 14 bit per la rete e 16 bit per i dispositivi. Si possono avere 16384 networks.
- **C:** 3 bit per la classe, 21 bit per la rete e 8 bit per gli host.

Basta guardare il primo bit per capire se era una classe A o B o C.

Nota: I bit di riconoscimento servono per sapere quali bit individuano la rete e quali gli host.

1.3 Indirizzamento IP senza classi (CIDR)

Il sistema **Classless InterDomain Routing** permette di indirizzare la porzione più precisa di indirizzi tra rete e dispositivi. La porzione di rete è dunque di lunghezza arbitraria. Il formato con cui può essere

rappresentato un indirizzo è il seguente: `networkID` + `prefix length` oppure `netmask`.

Il prefix length, specificato con `/x`, è il numero di bit di network.

La netmask è identificata da una serie di bit posti a 1 che determinano quali bit identificano la rete, attraverso un and bit a bit.

Esempio:

1	200.23.16.0/23	# prefix length
2	200.23.16.0 255.255.255.254.0	# netmask

L'indirizzo viene espresso attraverso gruppi di 8 bit, rappresentanti in modo decimale puntato (4 gruppi in quanto 32 bit totali). Ogni raggruppamento avrà un valore da 0 a 255.

Non tutti i valori sono permessi possibili, il più piccolo è 252. Questo è dovuto al fatto che abbiamo l'indirizzo dell'intera sottorete e l'indirizzo del inter broadcast che non possono essere utilizzati nell'assegnazione.

Un modo per sapere se un indirizzo è scritto in modo corretto è prendere il prefix length `/x` e controllare che ci l'ultimo numero puntato sia multiplo di $2^{(32-x)}$.

Esempi:

1	130.192.1.4/30	=>	$4\%2^{(32-30)} = 4\%4 = 0$	si!
2	130.192.1.16/30	=>	$16\%2^{(32-30)} = 16\%4 = 0$	si!
3	130.192.1.16/29	=>	$16\%2^{(32-29)} = 16\%8 = 0$	si!
4				
5	130.192.1.1/30	=>	$1\%2^{(32-30)} = 1\%4 \neq 0$	no!
6	130.192.1.1/29	=>	$1\%2^{(32-29)} = 1\%8 \neq 0$	no!
7	130.192.1.1/28	=>	$1\%2^{(32-28)} = 1\%16 \neq 0$	no!

Per il ragionamento di sopra appare evidente che un indirizzo che termina con .1 non sarà mai un indirizzo corretto, in quanto ritornerà sempre un resto.

1.4 IP routing

Il routing degli host avviene attraverso la routing table, caratterizzata da due colonne che identificano:

- **destinazione** (indirizzi ip)
- **interfaccia** (eth0...)

Quando viene inviato un pacchetto, si cerca un match all'interno della tabella per identificare dove inviare un pacchetto IP. Se è presente più di un match, viene considerato quello con il prefisso più lungo.

nota: i router sono identificati solitamente con un cerchio con dentro una x.

Di seguito è mostrato un esempio di routing:

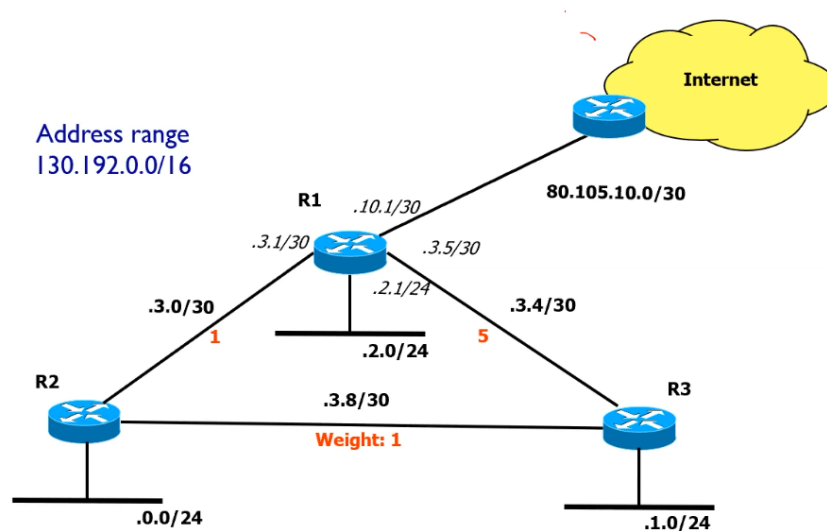


Figura 1.1: routing

Sono presenti in totale 7 sottoreti, di cui 3 reti locali e 4 reti punto punto. Tutta la sottorete ha come indirizzo quello raffigurato in alto a sinistra. Gli indirizzi di ciascuna di queste sono come segue:

Scriviamo la routing table del router identificando le reti direttamente connesse e raggiungibili. Prendiamo come riferimento **R1**:

Destination	Next	Type
130.192.3.0/30	130.192.3.1	direct
130.192.3.4/30	130.192.3.5	direct
130.192.2.0/24	130.192.2.1	direct
80.105.10.0/30	80.105.10.1	direct
80.105.10.0/30	80.105.10.1	direct
130.192.0.0/24	130.192.3.2	static
130.192.3.8/30	130.192.3.2	static
130.192.1.6/24	130.192.3.2	static

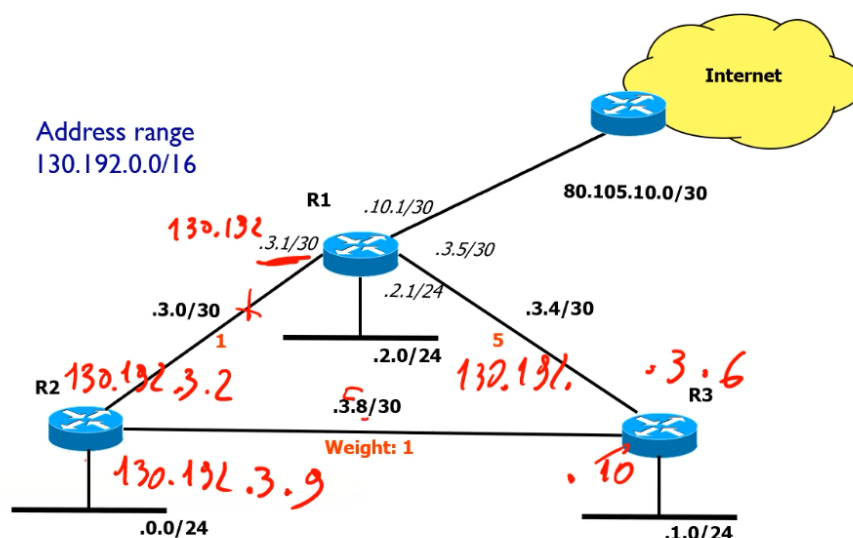


Figura 1.2: routing2

1.5 IP addressing methodology

La metodologia da adoperare è la seguente:

1. Localizzare le reti IP, in questo caso 3.
2. Individuare il numero di indirizzi richiesti, in questo caso nel router in alto a destra è sufficiente /30 perché ne sono richiesti 4 (2^2), /26 a sinistra (2^6) e /25 in basso a destra (2^7).
3. Quanti indirizzi posso allocare.
4. Il range di validità degli indirizzi, in questo caso /26, /25 e /30 dunque mi basterebbe o tutti e 3, o due /25 o infine un solo /24
5. netmask / prefix length
6. Address range
7. Host addresses

Nota: in basso a sinistra sono richiesti 43 indirizzi per 40 dispositivi. Ciò è dovuto al fatto che oltre ai 40 richiesti serve l'indirizzo di rete, l'indirizzo di broadcast e l'indirizzo del router.

Per riuscire a trovare le sottoreti, si prosegue in ordine dal maggiore (decimale minore):

```

1 # tutta la rete
2 10.0.0.0/24
3
4 # subnet2 (/25), 32-25 = 7 => 2^7 = 128 indirizzi
5 # range: 0-127
6 10.0.0.0/25

```

IP Addressing: methodology

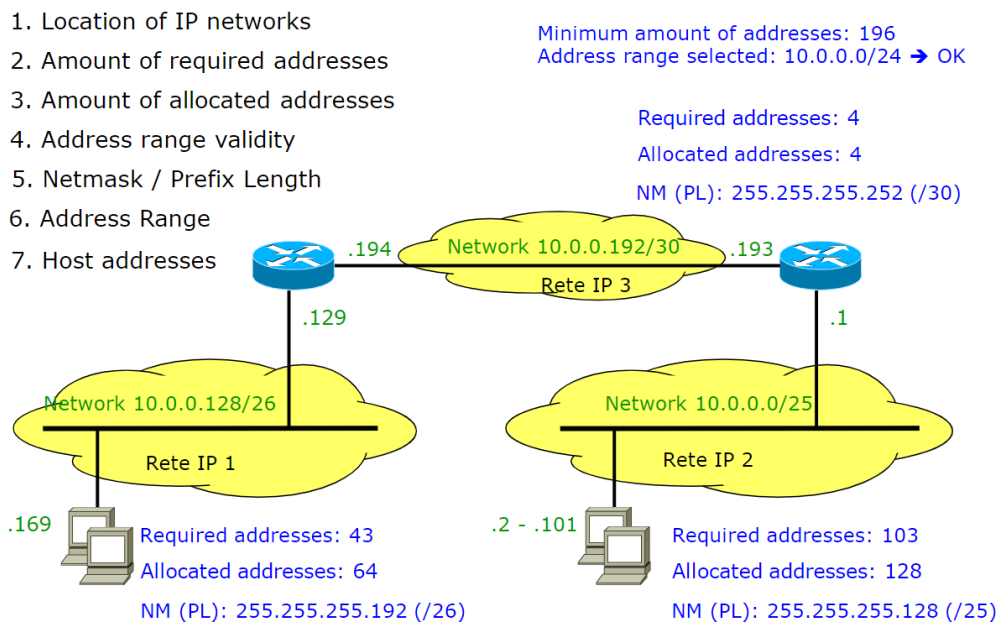


Figura 1.3: Rete di esempio

```

7 10.0.0.127 <- ultimo
8
9 # subnet3 (/26), 32-26 = 6 => 2^6 = 64 indirizzi
10 # range: 128-191
11 10.0.0.128/26
12 10.0.0.191 <- ultimo
13
14 #subnet4 (/30), punto punto
15 10.0.0.192/30

```

1.5.1 Esercizio 1

Numero di hosts	NetMask	Prefix Length	Available Addresses
2	255.255.255.252	(32-2) -> /30	$2^2 - 2 = 2$
27	255.255.255.224	(32-5) -> /27	$2^5 - 2 = 30$
5	255.255.255.248	(32-3) -> /29	$2^3 - 2 = 6$
100	255.255.255.128	(32-7) -> /25	$2^7 - 2 = 126$

Numero di hosts	NetMask	Prefix Length	Available Addresses
10	255.255.255.240	(32-4) -> /28	$2^4 - 2 = 14$
300	255.255.254.000	(32-9) -> /23	$2^9 - 2 = 510$
1010	255.255.252.000	(32-10) -> /22	$2^{10} - 2 = 1022$
55	255.255.255.192	(32-6) -> /26	$2^6 - 2 = 62$
167	255.255.255.000	(32-8) -> /24	$2^8 - 2 = 254$
1540	255.255.248.000	(32-11) -> /21	$2^{11} - 2 = 2046$

Nota: per calcolare la netmask, si esegue $256 - 2^{\text{bit}}$

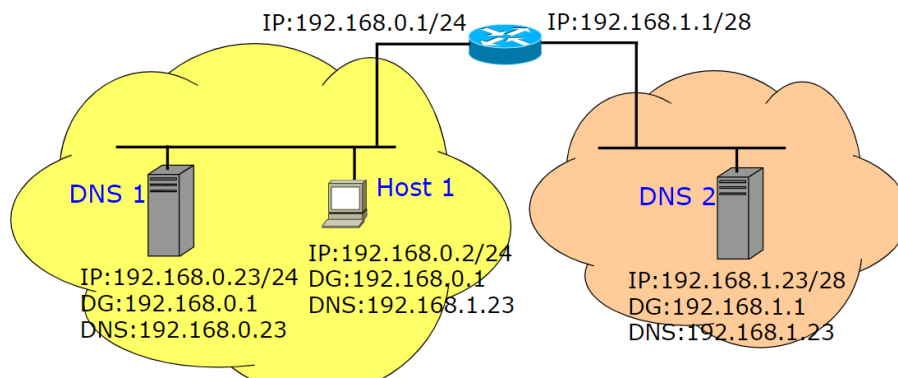
1.5.2 Esercizio 2

Verifica se i seguenti indirizzi sono validi o meno.

IP / Prefix Length pair	Valido?
192.168.5.0/24	Si, gli ultimi 8bit sono a 0
192.168.4.23/23	No
192.168.2.36/30	Si, $36 \bmod 2^{(32-30)} = 0$
192.168.2.36/29	No, $36 \bmod 2^{(32-29)} \neq 0$
192.168.2.32/28	Si, $32 \bmod 2^{(32-28)} = 0$
192.168.2.32/27	Si, $32 \bmod 2^{(32-27)} = 0$
192.168.3.0/23	No, $3 \bmod 2^{(1)} \neq 0$
192.168.2.0/31	No, /31 non ha senso
192.168.2.0/23	Si, $2 \bmod 2^{(1)} \neq 0$
192.168.16.0/21	Si, $16 \bmod 2^3 = 0$
192.168.12.0/21	No, $12 \bmod 2^3 \neq 0$

1.5.3 Esercizio 3

Trova l'errore di configurazione nella rete indicata di seguito e spiega il motivo per cui questa non funziona come dovrebbe.



1.5.4 Esercizio 4

Definisci un piano di indirizzamento IP per la rete in figura. Considera entrambi i tipi di indirizzamento: “tradizionale” (senza minimizzare) e una soluzione che minimizzi il numero di indirizzi IP utilizzati. Assumi di utilizzare il range 10.0.0.0/16.

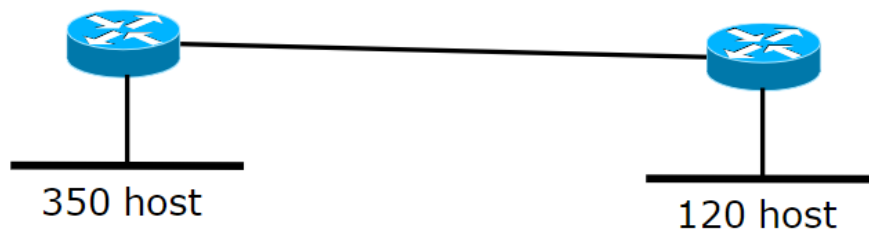


Figura 1.4: Rete

Partiamo evidenziando come il router a sinistra, al fine di servire 350 host, ha in realtà bisogno di 353 indirizzi: 350 host + 1 indirizzo di rete + 1 indirizzo di broadcast + 1 indirizzo del router, dunque /23. Stesso ragionamento è applicabile al router di destra, che ha bisogno di 123 indirizzi /25.

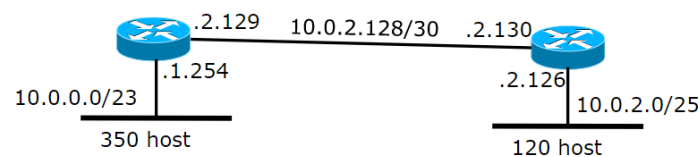
Troviamo così che 10.0.0.0/23 è la rete A (sinistra). Il suo indirizzo di broadcast sarà 10.0.1.255 in quanto adoperiamo 9 bit (quindi gli ultimi 8 bit a 1 e il primo bit del terzo gruppo a 1).

La sottorete C (destra) sarà identificata da 10.0.2.0/25 in quanto l'indirizzo immediatamente successivo. Il suo indirizzo di broadcast sarà 10.0.2.127.

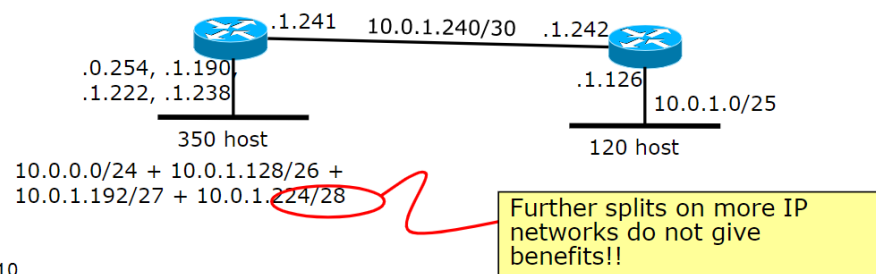
La sottorete B (centrale) sarà identificata da $10.0.2.128/30$, con $/30$ proveniente dal fatto che è una sottorete punto punto.

Questa soluzione comporta un grosso spreco, in quanto c'è un $/25$ che non viene utilizzato.

Solution1



Solution2



10

Figura 1.5: Soluzioni

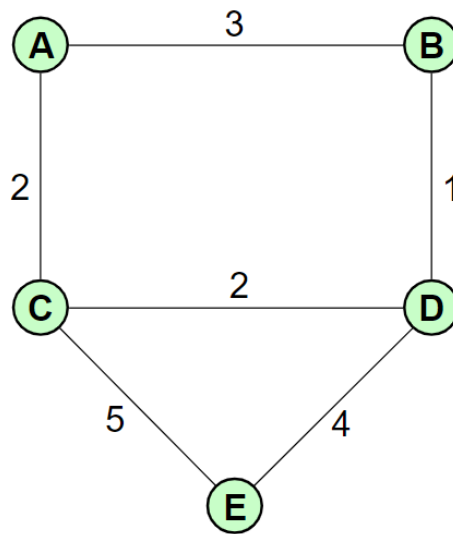
1.5.5 Esercizio 5

Definisci un albero di routing per tutti i nodi della rete mostrata di seguito.

L'**albero di instradamento** è quello che, a partire da un router della rete, stabilisce i percorsi minimi per raggiungere tutti i nodi.

Per calcolare l'albero di instradamento si prende un router come riferimento, ad esempio **A**.

dest	next
B	3 (ramo dx)
C	2 (ramo inf)
D	4 (sia dx che inf)
E	7 (ramo inf)

**Figura 1.6:** Rete esercizio 5

La stessa procedura dovrà essere poi eseguita per tutti i nodi rimanenti, minimizzando le distanze. A parità di distanza solitamente ci sono motivi differenti per cui si sceglie un percorso piuttosto che un altro (es router più nuovi).

Node A		Node B		Node C	
Destination	Next-hop	Destination	Next-hop	Destination	Next-hop
B	B	A	A	A	A
C	C	C	D	B	D
D	B/C	D	D	D	D
E	C	E	D	E	E

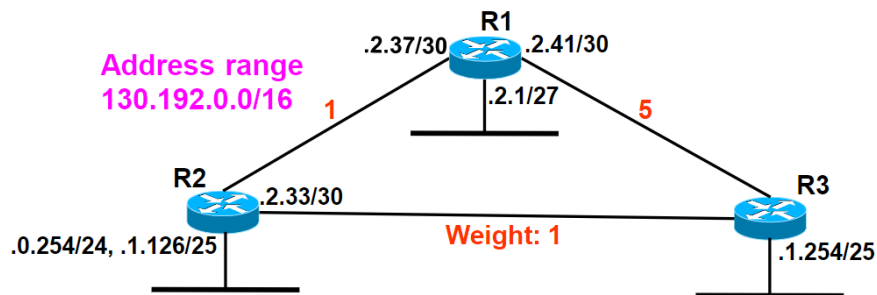
Node D		Node E	
Destination	Next-hop	Destination	Next-hop
A	B/C	A	C
B	B	B	D
C	C	C	C
E	E	D	D

Figura 1.7: Soluzione esercizio 5

1.5.6 Esercizio 6

Data la rete mostrata di seguito, definire la routing table di R1. La route aggregation deve essere massimizzata. Gli indirizzi ip mostrati in figura sono relativi all'interfaccia del router più vicino.

Cominciamo scrivendo la routing table di **R1**:

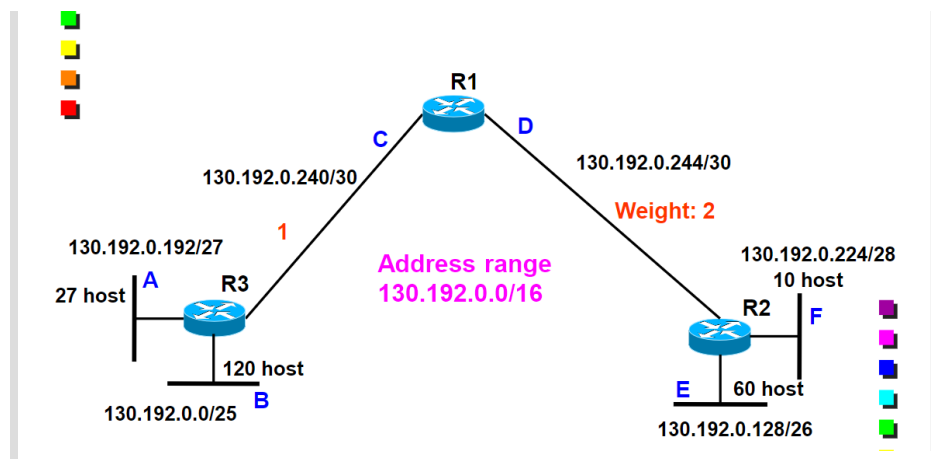
**Figura 1.8:** Esercizio 6

dest	next hop	Type
130.192.2.36/30 (A)	130.192.2.37	D
130.192.2.0/30 (B)	130.192.2.1	D
130.192.2.40/30 (C)	130.192.2.41	D
130.192.1.126/30 (D)	130.192.2.38	S
130.192.0.0/24 (E)	130.192.2.38	S
130.192.1.128/25 (F)	130.192.2.38	S
130.192.2.32/30 (G)	130.192.2.38	S

D ed **F** possono essere accorpati con 130.192.1.0/24, che a sua volta può essere aggregato con **E** ottenendo l'indirizzo 130.192.0.0/23 avendo il valore di broadcast pari a 130.192.1.255, per includere anche **G** è possibile usare 130.192.0.0/22. Dobbiamo però stare attenti a controllare come questi si rapportano con le entry statiche. In questo caso le include tutte, e non è un problema.

1.5.7 Esercizio 7

Realizzare un piano di indirizzamento che minimizza il numero di indirizzi necessari.



Troviamo la routing table di **R1**, analizzando ogni nodo a partire dai collegamenti diretti:

- Nella sottorete **A** sono presenti 27 host, per cui sono necessari 27+3 indirizzi e un prefix length di $(32 - 5) = 27$.
- Nella sottorete **B** sono invece necessari 120+3 indirizzi, per cui un prefix length di $(32 - 7) = 25$.
- Le sottorete C e D sono invece una sottoreti punto punto, per cui è necessario un prefix length di 30.
- La sottorete E ha bisogno di 60+3 indirizzi, per cui un prefix length di $(32 - 6) = 26$. Infine la sottorete F ha bisogno di 10+3 indirizzi, per cui un prefix length di $(32 - 4) = 28$.

Troviamo adesso quali sono gli indirizzi delle sottoreti, partendo da quella di dimensione maggiore (B, in quanto /25).

- **B**: 130.192.0.0/25, con indirizzo di broadcast 130.192.0.127 in quanto gli ultimi 7 bit sono a 1.
- **E**: 130.192.0.128/26 con indirizzo di broadcast 130.192.0.191
- **A**: 130.192.0.192/27, con indirizzo di broadcast 130.192.0.223
- **F**: 130.192.0.224/28, con indirizzo di broadcast 130.192.0.239
- **C**: 130.192.0.240/30, con indirizzo di broadcast 130.192.0.243
- **D**: 130.192.0.244/30, con indirizzo di broadcast 130.192.0.247

E' ora possibile calcolare gli indirizzi dei next hop, prendendo come riferimento il router più vicino:

dest	Gateway	Type
130.192.0.240/30 (C)	130.192.0.241	D
130.192.0.244/30 (D)	130.192.0.245	D
130.192.0.192/27 (A)	130.192.0.242	S

dest	Gateway	Type
130.192.0.0/25 (B)	130.192.0.242	S
130.192.0.128/26 (E)	130.192.0.246	S
130.192.0.224/28 (F)	130.192.0.246	S

Di queste entry bisogna valutare se è possibile fare qualche aggregazione. E' possibile farlo con **E** ed **F** in quanto: avendo /26 e 28, possono essere racchiusi in un /25 (quindi 2^7) con il medesimo indirizzo di **E** ($130.192.0.128/25$ è valido perché $128 \% 128 = 0$). La soluzione risulta comunque inefficiente perché non abbiamo ottenuto solo una entry.

1.5.8 Esercizio 8

Realizzare un piano di indirizzamento che minimizza il numero di indirizzi necessari. Utilizzare il risultato della routing table di R1.

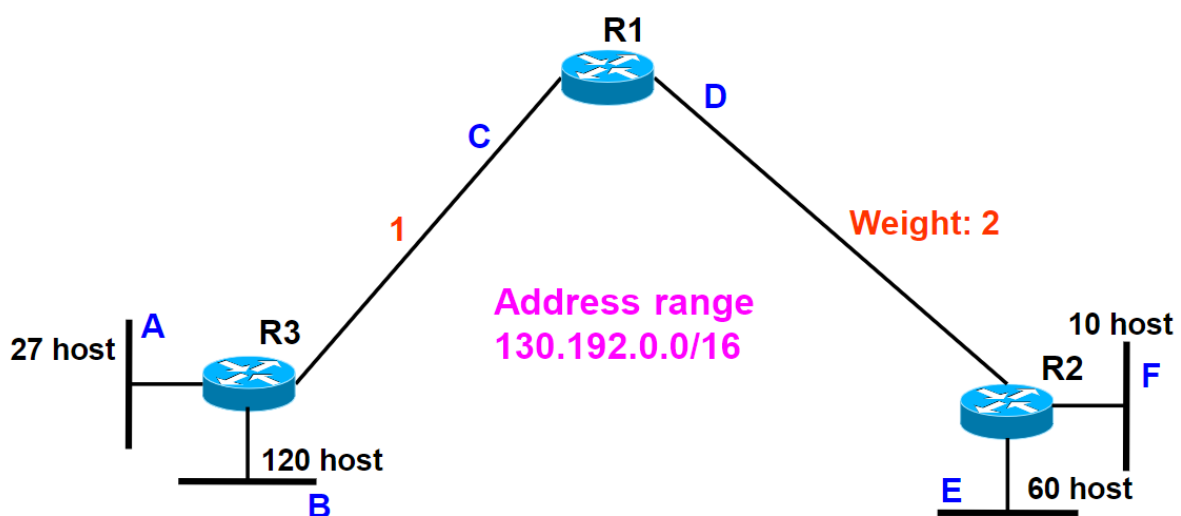


Figura 1.9: Esercizio 9