

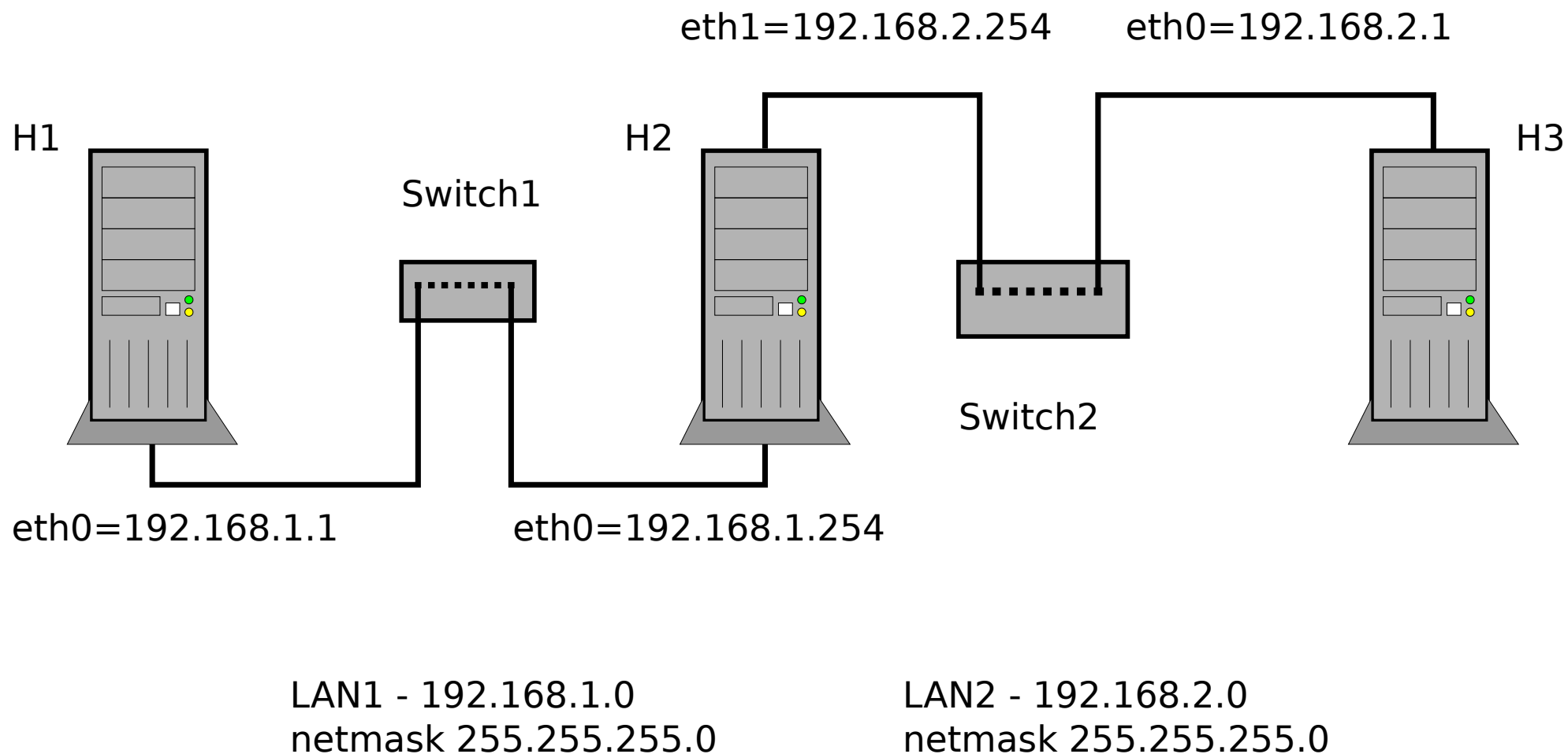
**PARTE D**

**ROUTING**

# **Modulo 1: Routing su Linux**

# Scenario

- Consideriamo la rete in figura



# Scenario

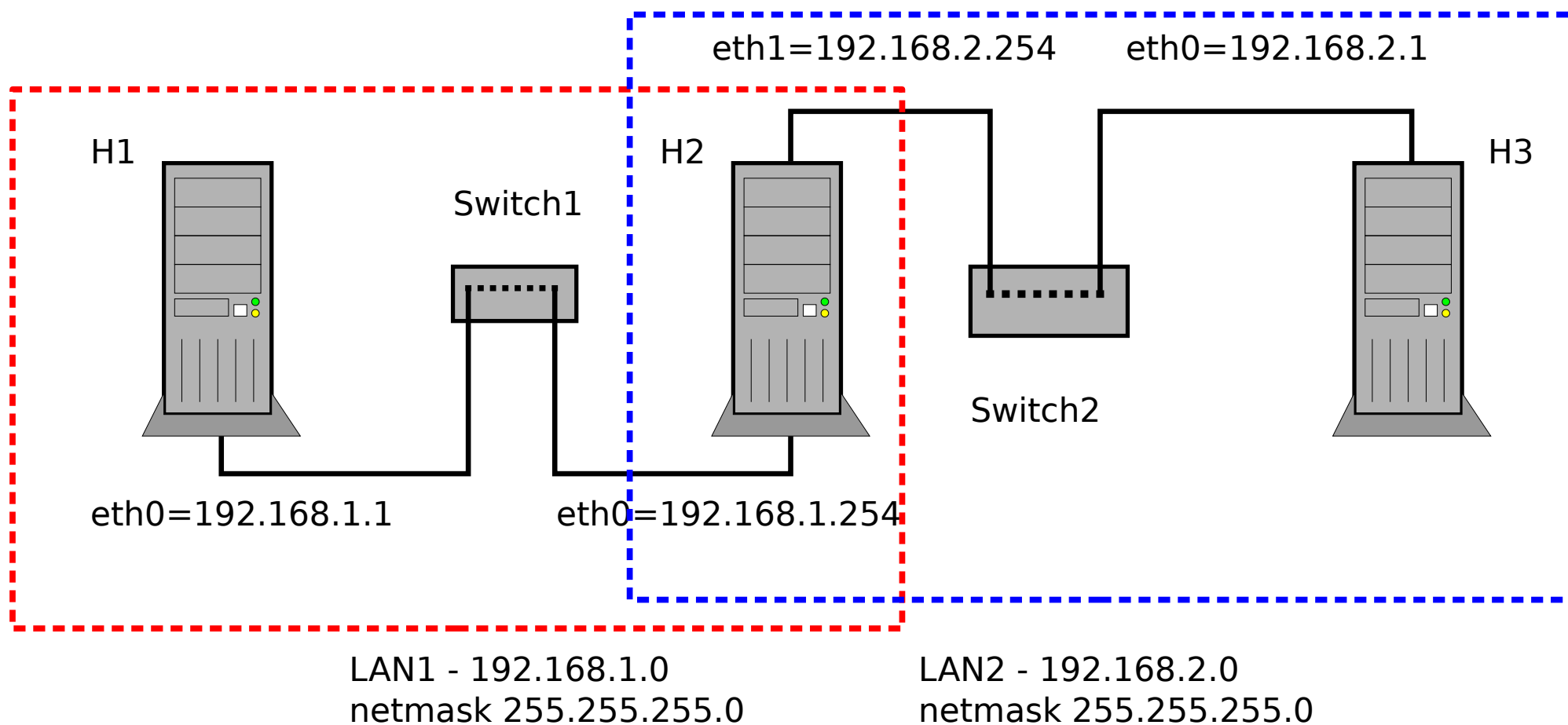
- **Tre nodi**
  - H1 :
    - eth0 = 192.168.1.1 netmask 255.255.255.0
  - H2 :
    - eth0 = 192.168.1.254 netmask 255.255.255.0
    - eth1 = 192.168.2.254 netmask 255.255.255.0
  - H3 :
    - eth0 = 192.168.2.1 netmask 255.255.255.0
- **Obiettivo: far comunicare tutti i nodi**

# Scenario

- **Usando una configurazione bridge, potremmo fare comunicare facilmente tutti i nodi, MA non rispetteremmo la configurazione richiesta**
- **Studio delle due sottoreti:**
  - 192.168.1.0/24
    - HostMin: 192.168.1.1 ; HostMax: 192.168.1.254
  - 192.168.2.0/24
    - HostMin: 192.168.2.1 ; HostMax: 192.168.2.254
- **Ogni nodo di una sottorete può comunicare a livello 2 solo con nodi appartenenti alla stessa sottorete. Per ogni altro nodo, è necessario ricorrere al routing dei pacchetti a livello IP.**

# Scenario

*A livello 2 possiamo collegare H1 con H2 e H2 con H3,  
ma non H1 e H3*



# IP Forwarding e Routing

- **Allo scopo di fare comunicare delle sottoreti, si individuano degli host che svolgono il ruolo di router, che inoltrano pacchetti da una sottorete a un'altra**
  - l'inoltro dei pacchetti viene determinato in base all'indirizzo IP destinazione (livello 3 dello stack TCP/IP)
  - il router riceve un pacchetto con un indirizzo IP destinazione non suo e, invece di scartarlo (azione comune di un host), lo inoltra secondo **regole di routing**
  - ogni host deve conoscere quali sono i router a cui può inviare i pacchetti nel caso in cui i destinatari non facciano parte della sua sottorete, e li identifica con il termine di **gateway**

# *IP Forwarding e Routing*

- **Problemi da risolvere in configurazioni che comprendono multiple sottoreti:**
  - ogni router deve sapere come raggiungere ogni rete, sia quelle alle quali è connesso, sia le altre
  - ogni host deve sapere a quali router (gateway) della propria sottorete deve inviare i pacchetti in base alla destinazione
- **Ai nostri scopi, la configurazione (statica) degli host e dei gateway deve comprendere:**
  - la definizione di **tabelle di routing** su tutti gli host
  - l'abilitazione dell'**ip forwarding** sugli host identificati come gateway



# IP Forwarding su Linux

- **La funzionalità di accettare pacchetti IP con destinazioni differenti e inoltrarli verso un destinatario viene chiamata ip forwarding, accessibile tramite il comando:**

```
sysctl net.ipv4.ip_forward
```

- **Se il parametro è 0 (default), l'ip forwarding è disabilitato. Per abilitare la funzionalità temporaneamente:**

```
sysctl -w net.ipv4.ip_forward=1
```

- **Altrimenti, per abilitarla permanentemente, impostare a 1 il campo net.ipv4.ip\_forward nel file /etc/sysctl.conf. In questo caso, riavviare networking o usare:**

```
sysctl -p /etc/sysctl.conf
```

per rendere effettiva la modifica

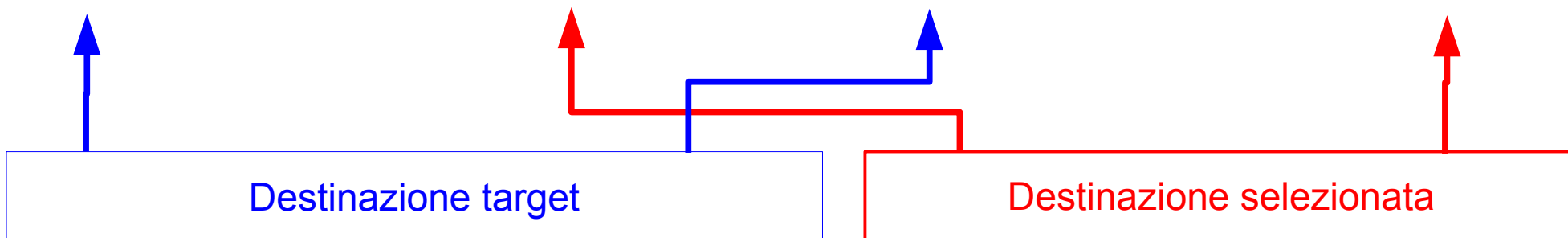
# Tabella di routing su Linux

Ogni host deve avere una tabella che raccoglie le regole di routing. Nei sistemi Linux, questa tabella è consultabile (e modificabile) tramite il comando **route** o **ip route show**:

```
# route -n # se non siamo root /sbin/route
```

*Kernel IP routing table*

<i>Destination</i>	<i>Gateway</i>	<i>Genmask</i>	<i>Flags</i>	<i>Iface</i>
192.168.1.0	*	255.255.255.0	U	eth1
155.185.48.128	*	255.255.255.192	U	eth0
192.168.2.5	192.168.1.254	255.255.255.255	UGH	eth1
0.0.0.0	155.185.54.190	0 0.0.0.0	UG	eth0



# Tabella di routing su Linux

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Iface
192.168.1.0	*	255.255.255.0	U	eth1
155.185.48.128	*	255.255.255.192	U	eth0
192.168.2.5	192.168.1.254	255.255.255.255	UGH	eth1
0.0.0.0	155.185.54.190	0 0.0.0.0	UG	eth0

## NB:

- la destinazione può essere una **subnet**, o un **host**
- è impostabile un **default gateway** per tutte le destinazioni target che non soddisfano nessuna delle destinazioni presenti nella tabella di routing (**NB**: ricordare che le destinazioni nella tabella sono definite da NetID + Netmask, e non solo da “Destination”)

# *Tabella di routing su Linux*

Output corrispondente con comando **ip route show**

```
default via 155.185.54.190 dev eth1 \
    proto static metric 1024
192.168.2.5 via 192.168.1.254 dev eth0 \
    proto static metric 1024
192.168.1.0/24 dev eth0 proto kernel \
    scope link src 192.168.1.35
155.185.48.128/26 dev eth1 proto kernel \
    scope link src 155.185.48.147
```

# Aggiungere regole di routing

**Routing verso un host:**

```
# route add -host <target> gw <gwaddr>
```

**Routing verso una subnet:**

```
# route add -net <target> gw <gwaddr>
```

**Impostazione del default gateway:**

```
# route add default gw <gwaddr>
```

**Esempi:**

```
# route add -host 192.161.4.1 gw 192.161.1.254
```

```
# route add -net 155.185.48.128 netmask \  
255.255.255.128 gw 192.161.1.254
```

```
# route add default gw 192.168.2.254
```

# Tabelle di routing permanenti

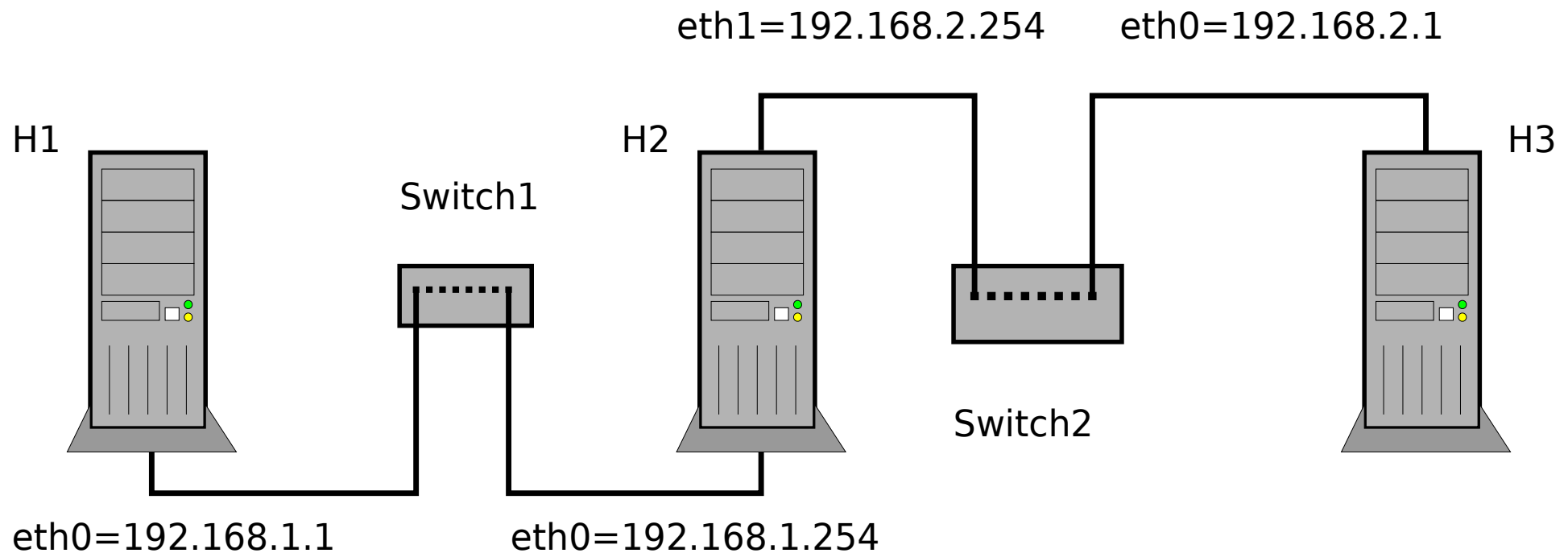
- Si agisce sui blocchi delle interfacce in ***/etc/network/interfaces***
- Il default gateway può essere impostato tramite comando ***gateway***.
- Altre regole di routing possono essere impostate tramite direttiva ***post-up***, che esegue un comando in seguito all'attivazione dell'interfaccia.

*Esempio:*

```
iface eth0 inet static
    address 192.168.1.1
    netmask 255.255.255.0
    gateway 192.168.1.254
    post-up route add -net 192.168.2.0 \
netmask 255.255.255.0 gw 192.168.1.253
```

# Risoluzione dello scenario

- Configurare correttamente la rete mostrate in precedenza



LAN1 - 192.168.1.0  
netmask 255.255.255.0

LAN2 - 192.168.2.0  
netmask 255.255.255.0

# Risoluzione dello scenario

- 2 subnet
  - LAN1: 192.168.1.0/24
  - LAN2: 192.168.2.0/24
- H2 fa parte di entrambe le subnet, non ha problemi a raggiungere gli altri due nodi
- H2 deve svolgere funzionalità di routing per permettere la comunicazione fra le due subnet
- *Soluzione*, considerando le interfacce di rete già configurate:
  - Abilitare l'**IP forward** su H2
  - Configurare la **tabella di routing** di H1 (H3) per raggiungere l'host H3 (H1) tramite H2. Possibile via:
    - routing verso l'host; routing verso la subnet, routing tramite default gateway



# Risoluzione dello scenario

## IP Forward:

```
H2 # sysctl -w net.ipv4.ip_forward=1
```

## Tabella di routing, alternative:

### *Routing basato su Host*

- H1 # route add 192.168.2.1 gw 192.168.1.254
- H3 # route add 192.168.1.1 gw 192.168.2.254

### *Routing basato su Subnet*

- H1 # route add -net 192.168.2.0 netmask \ 255.255.255.0 gw 192.168.1.254
- H3 # route add -net 192.168.1.0 netmask \ 255.255.255.0 gw 192.168.2.254

### *Default gateway*

- H1 # route add default gw 192.168.1.254
- H3 # route add default gw 192.168.2.254

## **Modulo 2: Subnetting**

# Subnetting

- La rete considerata (192.168.1.0, netmask 255.255.255.0) impiega subnet di **classe C** (CIDR /24). Identificare il range degli indirizzi disponibili per gli host è immediato:  
**192.168.1.1 - 192.168.1.254**
- In generale, con classi di indirizzi A, B, C (corrispondenti a /8, /16, /24) è molto facile calcolare il range degli indirizzi disponibili
- Un range di indirizzi di una delle suddette classi può essere organizzato in diverse sottoreti con notazioni potenzialmente classless. Definire i range di indirizzi può non essere scontato.
- Ad esempio, l'indirizzo di broadcast può non finire con 255, e il NetID può non finire con 0. Il numero di subnet disponibili in un range di classe A,B,C è una potenza di 2.

# Subnetting

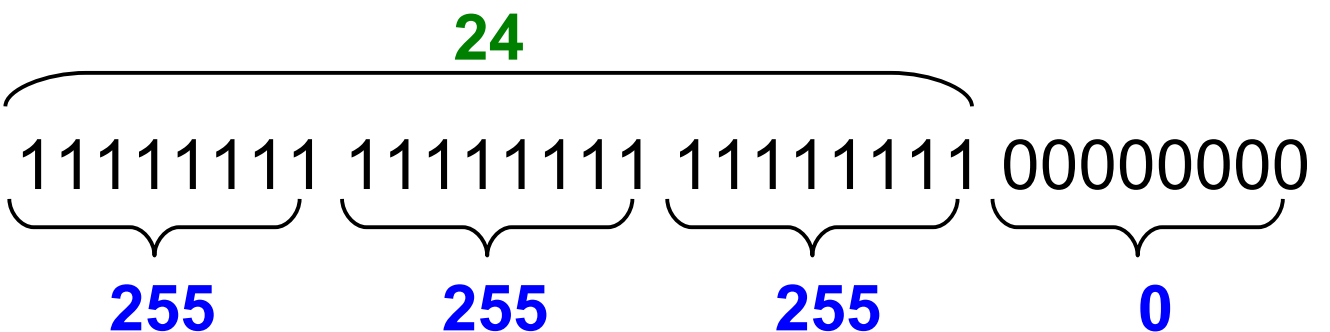
- Nella tabella di routing precedente avevamo

**NetID = 192.168.1.0, netmask = 255.255.255.0**

*ovvero,*

**Network = 192.168.1.0 / 24**

- Il numero **24** rappresenta la quantità di 1 nella rappresentazione binaria della netmask.

Netmask (32bit) => 

# Subnetting

Capiterà spesso di configurare subnet in notazioni CIDR non multiple di 8. Consideriamo l'indirizzo IP **155.185.48.147/26** dalla tabella di routing precedente. Identificare correttamente la subnet di cui fa parte richiede la risoluzione, in ordine, delle seguenti domande:

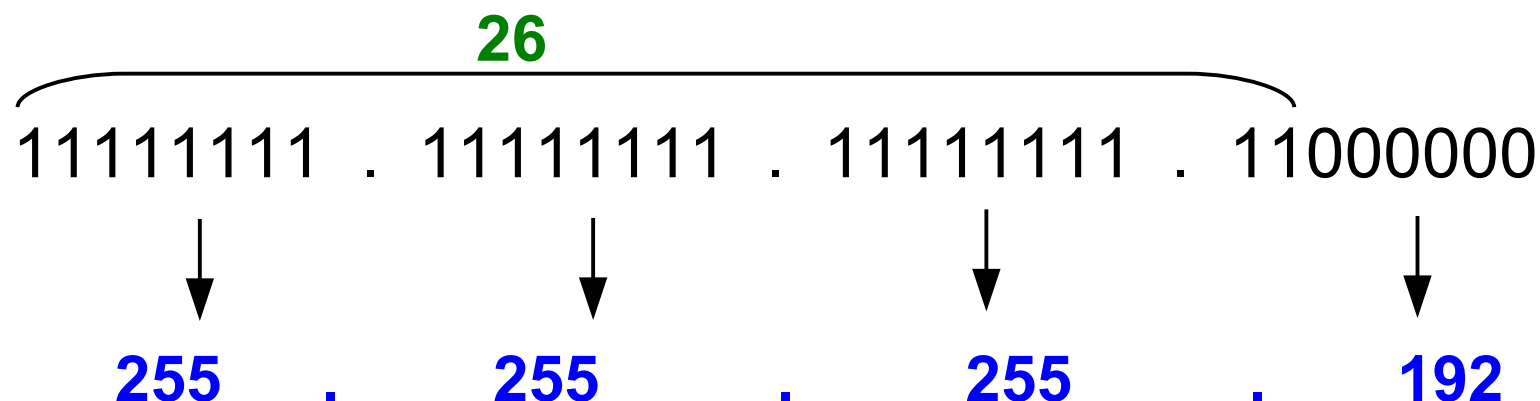
1. Qual è la netmask della rete a cui appartiene?
2. Qual è il NetID a cui appartiene questo indirizzo?
3. Qual è l'indirizzo di Broadcast della subnet?
4. Quali sono il primo e l'ultimo indirizzo disponibili in questa rete?

# Subnetting

- 1. Dato l'indirizzo IP **155.185.48.147/26** qual è la netmask della rete a cui appartiene?

Si ottiene dalla sola notazione CIDR, ovvero da **26**.

Si considera un numero binario a 32 bit con tanti 1 quanti indicati dal CIDR partendo dal MSB, introducendo 0 a destra come padding.



# Subnetting

- 2. Qual è il NetID a cui appartiene l'indirizzo IP **155.185.48.147** con netmask **255.255.255.192** ?

AND bitwise fra l'IP e la netmask, ovvero:

•	155	.	185	.	48	.	147
	&		&		&		&
	255	.	255	.	255	.	192
	=		=		=		=
	<b>155</b>	.	<b>185</b>	.	<b>48</b>	.	<b>128</b>

↑  
Naturalmente, l'unico valore che cambia è quello in corrispondenza di un blocco della netmask diverso da 255

# Subnetting

Considerando il NetID 155.185.48.128 e netmask 255.255.255.192:

3. Qual è l'indirizzo di **Broadcast** della subnet?

- Si ottiene tramite l'operazione OR bitwise fra il NetID e il complemento ad 1 della netmask, ovvero:

$$(155.185.48.128 \mid 0.0.0.127) \Rightarrow 155.185.48.191$$

4. Quali sono il primo e l'ultimo indirizzo disponibili in questa rete?

- Rispettivamente NetID + 1, e Broadcast - 1, ovvero:

$$\text{HostMin} \Rightarrow 155.185.48.129; \text{HostMax} \Rightarrow 155.185.48.190$$



# Subnetting

- A fini organizzativi o funzionali, potrebbe essere richiesto di dimensionare opportunamente diverse subnet all'interno di un range di indirizzi a nostra disposizione.
- Numero di host: in base alla scelta della netmask, la subnet supporta un diverso numero di host. Rappresentando con **N** la notazione CIDR della netmask:
- $\max n^{\circ} \text{ host} = 2^{(32-N)} - 2$
- Numero di subnet: dato un range di IP fisso, subnet più grandi comportano meno subnet. Se conosciamo il numero di IP (e.g. classe A,B,C), il numero di subnet necessarie (e.g. una per ogni unità operativa dell'azienda) e i requisiti in termini di quantità di host, possiamo dimensionare immediatamente le netmask corrispondenti.

# Subnetting: Esercizi

- **Calcolare tutti i parametri delle subnet per gli indirizzi IP:  
a) 192.168.5.157/20; b) 192.168.5.157/28 ; c) 155.185.48.200/26**
- **Calcolare il CIDR, HostMin, HostMax e il numero di host, dati  
NetID = 192.168.40.0 e netmask = 255.255.252.0**
- **Trovare tutti i parametri della subnet sapendo che uno degli  
host ha indirizzo IP 10.51.45.3/14**
- **Organizzare in quattro subnet la rete 155.185.48.0/24,  
sapendo che una delle reti includerà un massimo 120 host,  
mentre le altre sono di dimensioni inferiori. Si richiede di  
sfruttare completamente il range di indirizzi, e non  
sovradimensionare inutilmente le subnet (suggerimento =>  
 $1/8 + 1/8 + 1/4 + 1/2 = 1$ ).**

# Soluzioni esercizi

1. Calcolare tutti i parametri delle subnet per gli indirizzi IP:  
a) *192.168.5.157/20*

Netmask:    11111111.11111111.11110000.00000000  
              255   .   255       .   240       .       0

Indirizzo IP:    192       .   168       .   5       .       157  
                  11000000.10101000.00000101.10011101

Indirizzo IP: **11000000.10101000.0000** **0101.10011101** : 192.168.5.157  
NetId:        **11000000.10101000.0000** **0000.00000000** : 192.168.0.0  
Broadcast: **11000000.10101000.0000** **1111.11111111** : 192.168.15.255  
HostMin:     **11000000.10101000.0000** **0000.00000001** : 192.168.0.1  
HostMax:     **11000000.10101000.0000** **1111.11111110** : 192.168.15.254

# Soluzioni esercizi

- 1. Calcolare tutti i parametri delle subnet per gli indirizzi IP:  
b) 192.168.5.157/28

Netmask:    11111111.11111111.11111111.11110000  
              255   . 255   . 255   . 240

Indirizzo IP:    192   . 168   . 5   . 157  
                  11000000.10101000.00000101.10011101

Indirizzo IP: 11000000.10101000.00000101.1001 1101 : 192.168.5.157  
NetId:        11000000.10101000.00000101.1001 0000 : 192.168.5.144  
Broadcast: 11000000.10101000.00000101.1001 1111 : 192.168.5.159  
HostMin:     11000000.10101000.00000101.1001 0001 : 192.168.5.145  
HostMax:     11000000.10101000.00000101.1001 1110 : 192.168.5.158

# Soluzioni esercizi

1. Calcolare tutti i parametri delle subnet per gli indirizzi IP:  
c) 155.185.48.200/26

Netmask:    11111111.11111111.11111111.11000000  
              255   .   255       .   255       .   192

Indirizzo IP:    155       .   185       .   48       .   200  
                  10011011.10111001.00110000.11001000

Indirizzo IP: 10011011.10111001.00110000.11 001000 : 155.185.48.200  
NetId:        10011011.10111001.00110000.11 000000 : 155.185.48.192  
Broadcast: 10011011.10111001.00110000.11 111111 : 155.185.48.255  
HostMin:     10011011.10111001.00110000.11 000001 : 155.185.48.193  
HostMax:     10011011.10111001.00110000.11 111110 : 155.148.48.254

# Soluzioni esercizi

**2. Calcolare il CIDR, HostMin, HostMax e il numero di host, dati  
NetID = 192.168.40.0 e netmask = 255.255.252.0**

**255.255.252.0:  $252=128+64+32+16+8+4=11111100b$**

**CIDR :  $8 + 8 + 6 + 0 = 22$**

**HostMin: 192.168.40.1**

**HostMax: 192.168.43.254**

**N°Host:  $2^{(32-22)}-2 = 1022$**

# Soluzioni esercizi

3. Trovare tutti i parametri della subnet sapendo che uno degli host ha indirizzo IP 10.51.45.3/14

Netmask: 11111111.11111100.00000000.00000000  
          255 . 252 . 0 . 0

Indirizzo IP: 10 . 51 . 45 . 3  
              00001010.00110011.00101101.00000011

Indirizzo IP: 00001010.001100 11.00101101.00000011 : 10.51.45.3

NetId: 00001010.001100 00.00000000.00000000 : 10.48.0.0

Broadcast: 00001010.001100 11.11111111.11111111 : 10.51.255.255

HostMin: 00001010.001100 00.00000000.00000001 : 10.48.0.1

HostMax: 00001010.001100 11.11111111.11111110 : 10.51.255.254

# Soluzioni esercizi

**4. Organizzare in quattro subnet la rete 155.185.48.0/24, sapendo che la rete più grande includerà un massimo di 120 host. Si richiede di sfruttare completamente il range di indirizzi e di distribuire in modo equo (per quanto possibile) gli indirizzi.**

**120  $\rightarrow$   $\min\{2^N\} = 128 \rightarrow N = \log_2(128) = 7$   
 $\Rightarrow$  netmask della rete più grande =  $32-7 = 25$**

**Questa subnet occupa 1/2 della rete totale, ma ne servono 4 in tutto. È impossibile avere 3 subnet uguali, per cui due subnet occuperanno 1/8 (netmask 27) e una 1/4 (netmask 26) degli indirizzi disponibili, ovvero:**

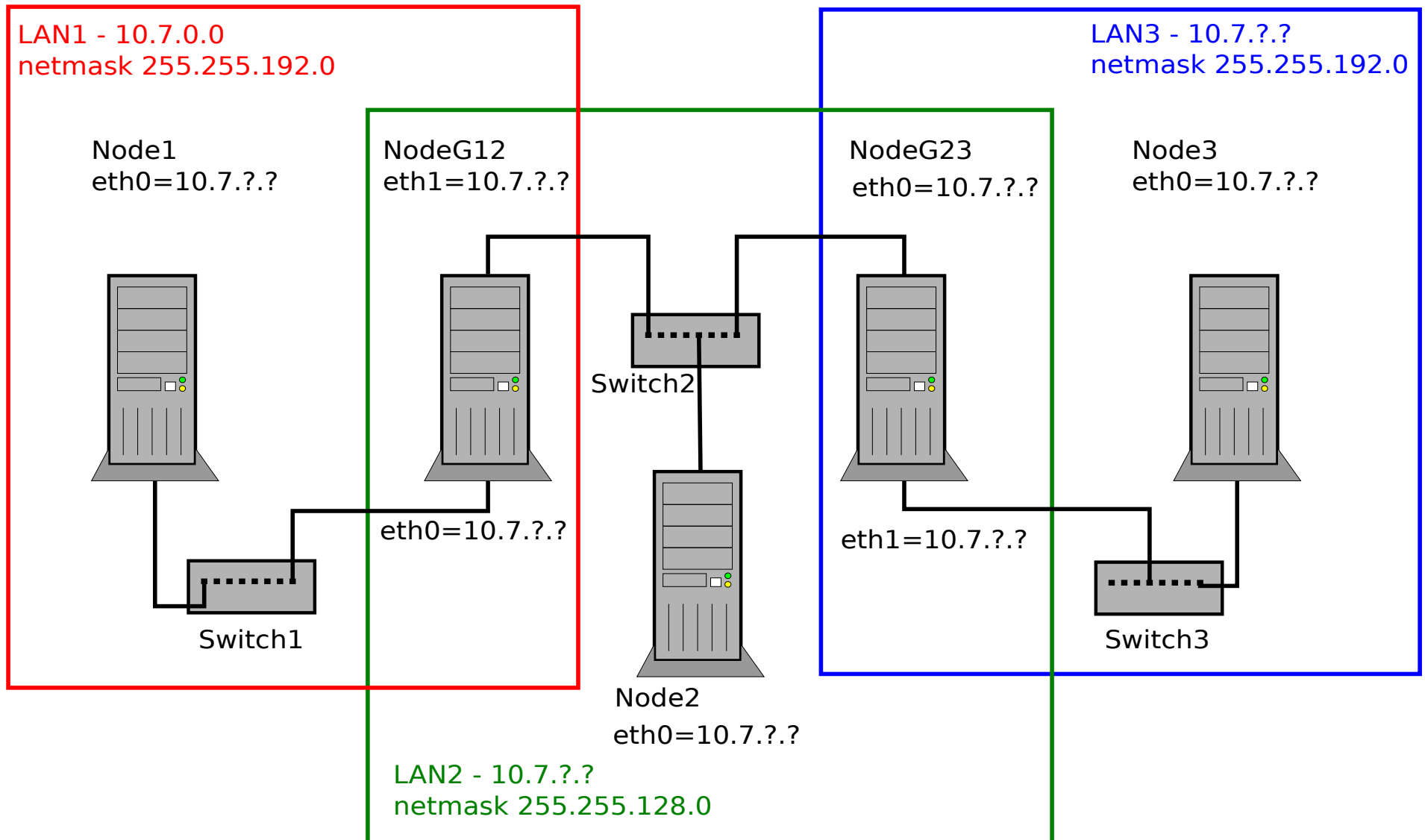
**NM 1 = 255.255.255.128  $\rightarrow$  Subnet 1 : 155.185.48.0/25  
NM 2 = 255.255.255.192  $\rightarrow$  Subnet 2 : 155.185.48.128/26  
NM 3 = 255.255.255.224  $\rightarrow$  Subnet 3 : 155.185.48.192/27  
NM 3 = 255.255.255.224  $\rightarrow$  Subnet 4 : 155.185.48.224/27**



**Modulo 3:  
Altri esercizi di  
routing**

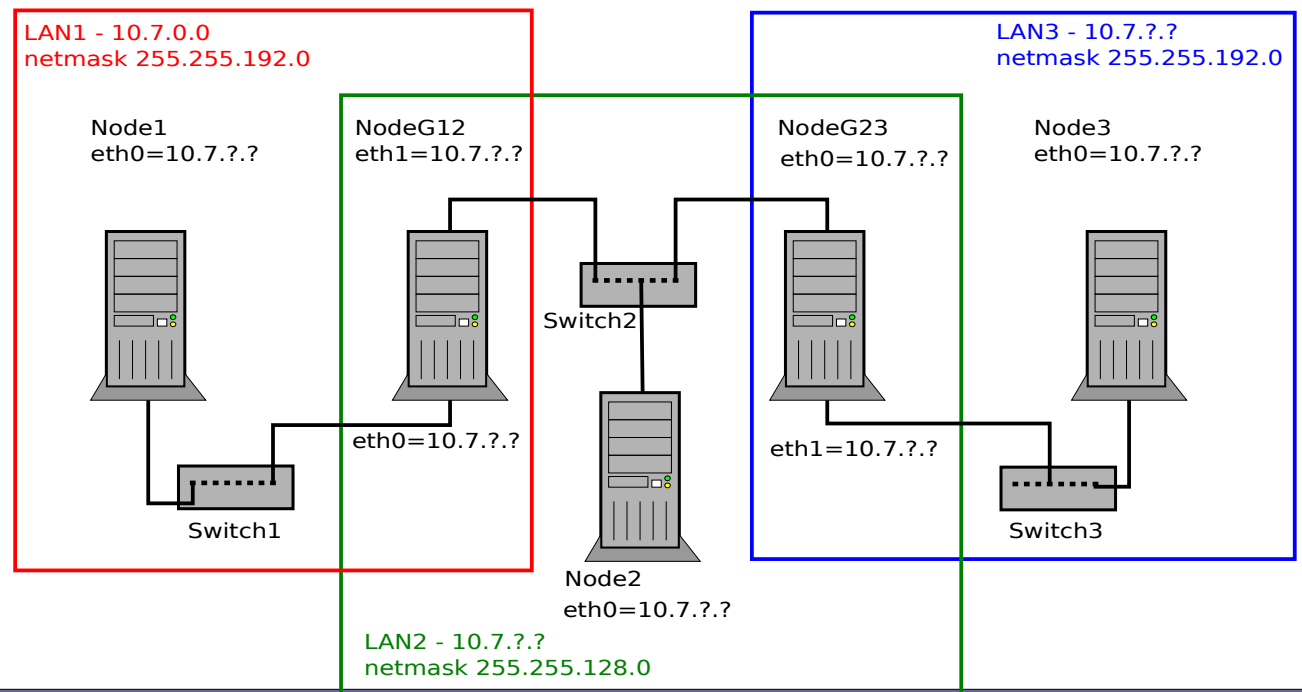
# Esercizio

- Configurare e virtualizzare la rete mostrata in figura



# Esercizio

- Configurare e virtualizzare la rete mostrata in figura:
  - Node1, Node2 e Node3 usino i primi indirizzi IP disponibili nelle rispettive sottoreti
  - NodeG12 occupi i penultimi indirizzi IP delle sottoreti di cui fa parte
  - NodeG23 occupi gli ultimi indirizzi IP delle sottoreti di cui fa parte



# Soluzione esercitazione: analisi

- Considerazioni:

128 corrisponde a un byte con il MSB a 1, mentre 192 (128+64) corrisponde a un byte con i due MSB a 1

LAN 2 ha una netmask formata da 17 bit a 1, mentre LAN1 e LAN 3 hanno netmask con 18 bit a 1

LAN1, LAN3: 255.255.192.0 → /18

LAN2: 255.255.128.0 → /17

LAN2 occupa la metà degli indirizzi, mentre LAN1 e LAN3 un quarto ciascuna

**Nella scelta dei NetId dobbiamo essere sicuri di non creare conflitti fra le subnet!**

# Soluzione esercitazione

- **LAN1 è già data dalla configurazione:**
  - LAN1: 10.7.0.0/18; Min: 10.7.0.1; Max: 10.7.63.254
- **Alternative nella scelta di LAN2 ( /17 → 1 bit 1 nel terzo byte):**
  - 10.7.**0**.0 → **0**0000000 NO, conflitto LAN1
  - 10.7.**128**.0 → **1**0000000 OK

# Soluzione esercitazione

- **Input della configurazione:**

- LAN1: 10.7.0.0/18; Min: 10.7.0.1; Max: 10.7.63.254

- **Alternative nella scelta di LAN3 ( /18 → 2 bit 1 nel terzo byte):**

- 10.7.**0**.0 → **00**000000 NO, conflitto LAN1
- 10.7.**64**.0 → **01**000000 OK
- 10.7.**128**.0 → **10**000000 NO, conflitto LAN2
- 10.7.**192**.0 → **11**000000 NO, conflitto LAN2

# Soluzione esercitazione

		<b>NetId</b>	<b>HostId</b>
<b>LAN1:</b>	<b>10.7.0.0/18</b>	<b>00001010.00000111.00</b>	<b>000000.00000000</b>
• <b>HostMin:</b>	<b>10.7.0.1</b>	<b>00001010.00000111.00</b>	<b>000000.00000001</b>
• <b>HostMax:</b>	<b>10.7.63.254</b>	<b>00001010.00000111.00</b>	<b>111111.11111110</b>
<b>LAN2:</b>	<b>10.7.128.0/17</b>	<b>00001010.00000111.1</b>	<b>000000.00000000</b>
• <b>HostMin:</b>	<b>10.7.128.1</b>	<b>00001010.00000111.1</b>	<b>000000.00000001</b>
• <b>HostMax:</b>	<b>10.7.255.254</b>	<b>00001010.00000111.1</b>	<b>1111111.11111110</b>
<b>LAN3:</b>	<b>10.7.64.0/18</b>	<b>00001010.00000111.01</b>	<b>000000.00000000</b>
• <b>HostMin:</b>	<b>10.7.64.1</b>	<b>00001010.00000111.01</b>	<b>000000.00000001</b>
• <b>HostMax:</b>	<b>10.7.127.254</b>	<b>00001010.00000111.01</b>	<b>111111.11111110</b>

# Soluzione esercitazione

*“Node1, Node2 e Node3 usino i primi indirizzi IP disponibili nelle rispettive sottoreti”*

- Node1, primo host della subnet 10.7.0.0/18:  
eth0: 10.7.0.1 /18
- Node2, primo host della subnet 10.7.128.0/17:  
eth0: 10.7.128.1 /17
- Node3, primo host della subnet 10.7.64.0/18:  
eth0: 10.7.64.1 /18



# Soluzione esercitazione

*“NodeG12 occupi i penultimi indirizzi IP delle sottoreti di cui fa parte”*

- NodeG12, penultimi indirizzi delle subnet 10.7.0.0/18 e 10.7.128.0/17:

eth0: 10.7.63.253 /18

eth1: 10.7.255.253 /17

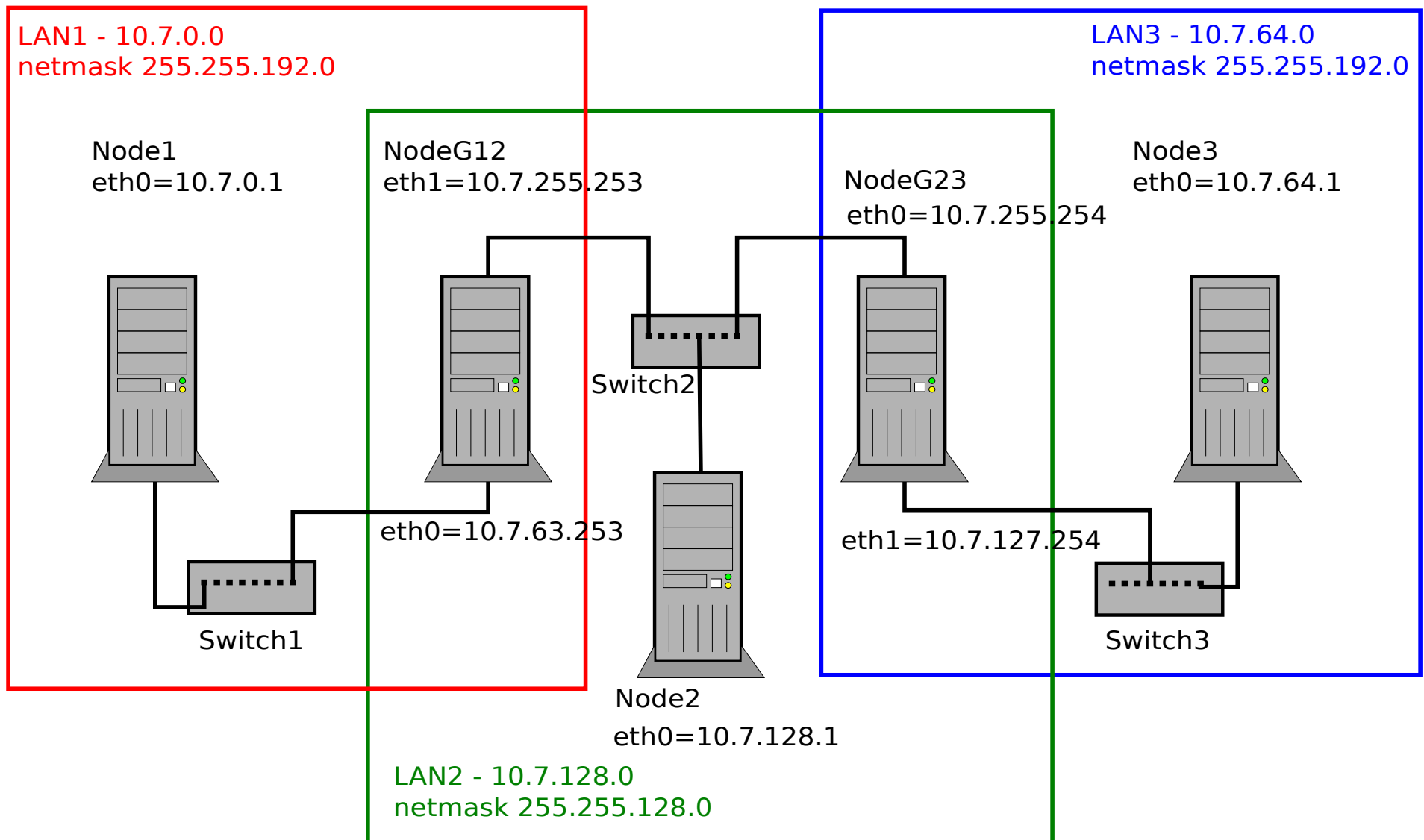
*“NodeG23 occupi gli ultimi indirizzi IP delle sottoreti di cui fa parte”*

- NodeG23, ultimi indirizzi delle subnet 10.7.64.0/18 e 10.7.128.0/17:

eth0: 10.7.255.254 /17

eth1: 10.7.127.254 /18

# Soluzione esercitazione



# Esercizio: configurazione

Si propone una soluzione che implementa regole di instradamento basate sulle subnet (e non sui singoli host) non permanenti. Per la configurazione permanente della rete modificare i file di configurazione come visto nelle lezioni precedenti.

**Node 1:** interfaccia di rete eth0 e regole di routing verso LAN2 e LAN3 attraverso NodeG12

- *Configurazione eth0:*

```
ifconfig eth0 10.7.0.1 netmask 255.255.192.0 up
```

- *Route verso LAN2:*

```
route add -net 10.7.128.0 netmask 255.255.128.0 gw 10.7.63.253
```

- *Route verso LAN3:*

```
route add -net 10.7.64.0 netmask 255.255.192.0 gw 10.7.63.253
```

# Esercizio: configurazione

**Node 2:** interfaccia di rete eth0 e regole di routing verso LAN1 attraverso NodeG12, e verso LAN3 attraverso NodeG23

- *Configurazione eth0:*

```
ifconfig eth0 10.7.128.1 netmask 255.255.128.0 up
```

- *Route verso LAN1 attraverso NodeG12:*

```
route add -net 10.7.0.0 netmask 255.255.192.0 \  
gw 10.7.255.253
```

- *Route verso LAN3 attraverso NodeG23:*

```
route add -net 10.7.64.0 netmask 255.255.192.0 \  
gw 10.7.255.254
```

# Esercizio: configurazione

**Node 3:** interfaccia di rete eth0 e tabella di routing verso LAN1 e LAN2 attraverso NodeG23

- *Configurazione eth0:*

```
ifconfig eth0 10.7.64.1 netmask 255.255.192.0 up
```

- *Route verso LAN1 attraverso NodeG23:*

```
route add -net 10.7.0.0 netmask 255.255.192.0 \  
gw 10.7.127.254
```

- *Route verso LAN2 attraverso NodeG23:*

```
route add -net 10.7.128.0 netmask 255.255.128.0 \  
gw 10.7.127.254
```

# Esercizio: configurazione

**Node G23:** interfacce di rete eth0 ed eth1 e tabella di routing verso LAN1 attraverso NodeG12

- *Configurazione eth0:*

```
ifconfig eth0 10.7.255.254 netmask 255.255.128.0 up
```

- *Configurazione eth1:*

```
ifconfig eth1 10.7.127.254 netmask 255.255.192.0 up
```

- *Route verso LAN1 attraverso NodeG12:*

```
route add -net 10.7.0.0 netmask 255.255.192.0 \  
    gw 10.7.255.253
```