

# Lezione 8:

## Livello 3 – Routing statico

Claudio Ardagna, Patrizio Tufarolo – Università degli Studi di Milano

Insegnamento di Laboratorio di Reti di Calcolatori



# Introduzione - 1

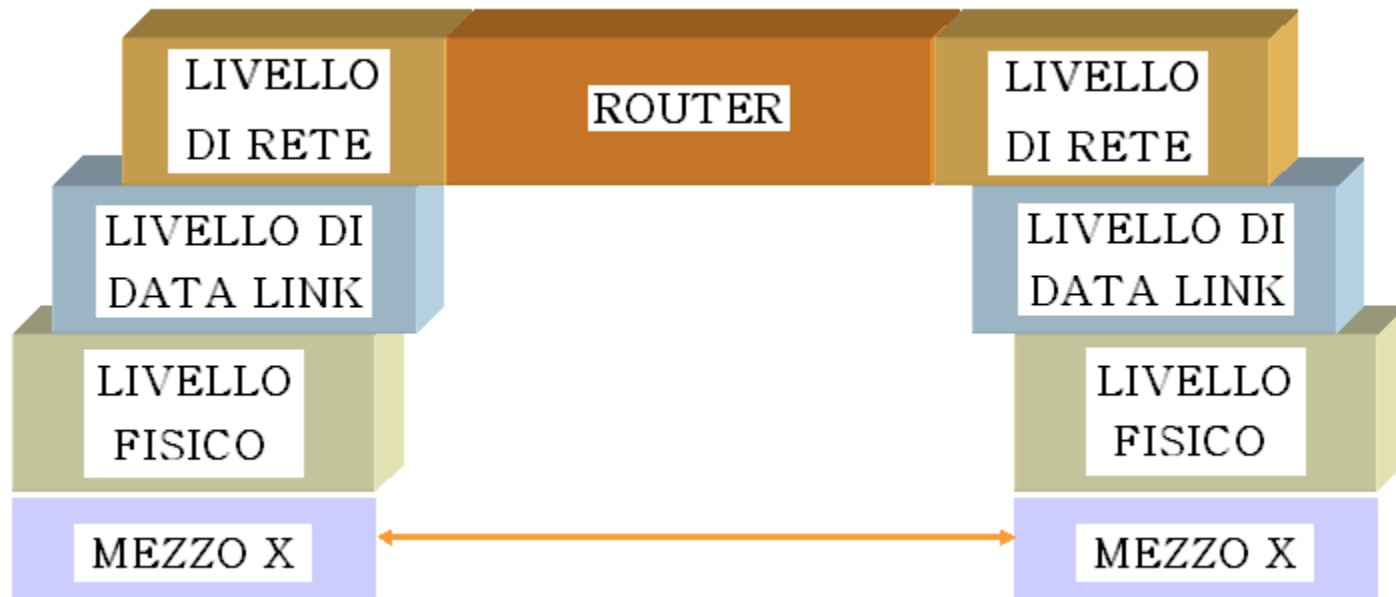
- ▶ Abbiamo visto come sia possibile realizzare topologie a stella a **livello fisico**, con gli **hub**
- ▶ Abbiamo visto come sia possibile realizzare topologie a stella a **livello data-link**, con gli **switch**
- ▶ Andremo ora a vedere come sia possibile **interconnettere reti eterogenee** con dispositivi di livello 3: i **router**

# Introduzione - 2

- ▶ Richiamo dalla teoria: differenza tra hub, switch e router
  - ▶ Hub
    - ▶ Forniscono semplicemente le **connessioni meccaniche ed elettriche** tra i nodi di una rete
  - ▶ Switch
    - ▶ **Esaminano il frame** ethernet per trovare l'indirizzo **MAC di destinazione**
    - ▶ **Non effettuano alcuna modifica** ai pacchetti
  - ▶ Router (instradatori)
    - ▶ **Esaminano** i pacchetti e **li modificano all'occorrenza**
    - ▶ Possono eseguire una *conversione* di protocollo
    - ▶ Rispetto agli switch richiedono di **maggiore capacità di elaborazione**

# Introduzione - 3

- ▶ Router: interfaccia del livello di rete



# Terminologia - 1

- ▶ ISO/OSI (Open System Interconnection)
  - ▶ Standard de iure che organizza l'architettura di una rete di calcolatori in una struttura composta da 7 livelli (stack di rete)
- ▶ MAC Address
  - ▶ Media Access Control Address, o indirizzo fisico, indirizzo a 48 bit che identifica univocamente un'interfaccia di rete
- ▶ ARP (Address Resolution Protocol)
  - ▶ Protocollo che si colloca tra livello 2 e livello 3, permettendo l'associazione di un indirizzo IPv4 al corrispondente indirizzo fisico (RFC 826)
  - ▶ Il protocollo analogo ad ARP per IPv6 è il Neighbor Discovery Protocol (RFC 4861)

# Terminologia - 2

- ▶ IPv4
  - ▶ Internetworking protocol, protocollo per interconnettere reti di tipologie eterogenee, versione 4
- ▶ Indirizzo IP
  - ▶ Identificatore numerico univoco a 4 byte (32 bit) suddiviso in *net id o prefisso* e un *host id*
- ▶ Subnet (Sottorete)
  - ▶ **Partizionamento** dello spazio di indirizzamento in sottoreti, realizzato tramite l'applicazione di una **subnet mask**

# Terminologia - 3

- ▶ Router (instradatore)
  - ▶ Dispositivi per l'interconnessione di reti eterogenee (o sottoreti differenti), che stabilisce un percorso logico di comunicazione, instradando i pacchetti
- ▶ Hop
  - ▶ Router che un pacchetto deve attraversare per arrivare a destinazione (salti)
- ▶ Tabella di routing
  - ▶ Database memorizzato in un router, che contiene le rotte di destinazione di una rete
  - ▶ Permette di memorizzare informazioni sulla topologia di rete

# Richiamo dalla teoria: instradamento statico

- ▶ Esistono due tipi di instradamento
  - ▶ Instradamento **diretto**
    - ▶ Il pacchetto può **essere inviato direttamente a destinazione**, **incapsulato in un frame ethernet**, poiché l'indirizzo di destinazione si trova nella stessa sottorete dell'indirizzo mittente
    - ▶ **Avviene tramite il protocollo ARP**
  - ▶ Instradamento **indiretto**
    - ▶ L'indirizzo di destinazione **non si trova nella stessa sottorete**, viene quindi utilizzato l'**indirizzo fisico del router**, che provvederà ad instradare il pacchetto verso la giusta direzione, in base a delle decisioni prese con varie tecniche
    - ▶ Una tecnica assolutamente razionale è l'utilizzo di una **tabella di routing**



# Instradamento basato su tabella di routing

- ▶ Tabella di routing
  - ▶ **Database** memorizzato in un router, che contiene **le rotte di destinazione di una rete**.
- ▶ È generalmente composta organizzata in almeno tre campi
  - ▶ Sottorete o host di destinazione
  - ▶ Costo / Metrica
    - ▶ **Costo del percorso** attraverso cui il pacchetto deve essere inviato
  - ▶ Salto successivo
    - ▶ **Indirizzo del router successivo** (o dell'host, nel caso in cui il pacchetto sia prossimo alla destinazione)

# Instradamento basato su tabella di routing - 2

- ▶ Nella realtà si compone di:
  - ▶ Destinazione da raggiungere (host o sottorete)
  - ▶ Gateway - Router che permette di raggiungere la destinazione
  - ▶ Maschera di sottorete della destinazione
  - ▶ Flags
    - ▶ U -> UP (la rotta è attiva)
    - ▶ G -> Gateway (la rotta è un gateway)
    - ▶ H -> La destinazione è un host (maschera di sottorete a 255.255.255.255, ovvero /32)
  - ▶ Metrica
  - ▶ Interfaccia di rete
- ▶ Tra le rotte è presente una rotta di default (**default gateway**) da utilizzare nel caso non sia possibile utilizzare le altre rotte. Questa è contrassegnata dall'indirizzo **0.0.0.0/0**, o dalla stringa «default» a seconda dell'implementazione.
- ▶ **NB: possono esistere più tabelle di routing**

# Tabella di routing: esempio

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
default	gw-219.eduroam.	0.0.0.0	UG	600	0	0	wlp4s0
10.20.0.0	*	255.255.255.0	U	0	0	0	virbr5
10.42.0.0	*	255.255.255.0	U	100	0	0	enp3s0
srv-035-107-dhc	gw-219.eduroam.	255.255.255.255	UGH	600	0	0	wlp4s0
159.149.219.0	*	255.255.255.0	U	600	0	0	wlp4s0
172.17.0.0	*	255.255.0.0	U	0	0	0	docker0
192.168.122.0	*	255.255.255.0	U	0	0	0	virbr4
192.168.150.0	*	255.255.255.0	U	0	0	0	virbr6



# Il comando Traceroute

- ▶ Il comando *traceroute* (tracert su Windows) permette di **vedere il percorso che un pacchetto intraprende per arrivare a destinazione**, evidenziando tutti gli **hop attraversati**, e **individuando gli eventuali punti di failure**
- ▶ Esso sfrutta una proprietà del protocollo IP, il campo **TTL** (time to live), il quale specifica il **numero degli apparati (hop) che il pacchetto può attraversare** prima di essere considerato «scaduto»
- ▶ Ogni router che riceve un pacchetto, prima di inoltrarlo, diminuisce il TTL di una unità
- ▶ Se il campo arriva a 0, viene generato un errore, propagato agli host tramite il protocollo **ICMP (ICMP Time Exceeded)** a cui possono seguire **tentativi di ritrasmissione**

# Routing su Linux

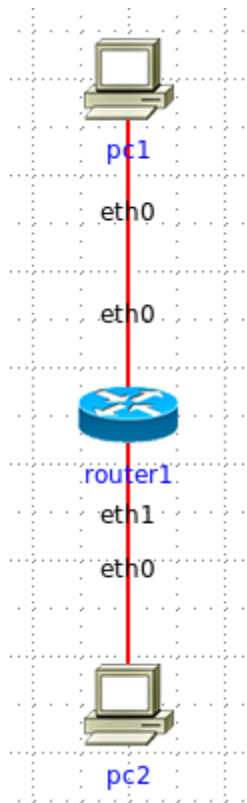
- ▶ Il comando Linux per manipolare e consultare la tabella di routing di default è
  - ▶ `ip route`
- ▶ Comandi alternativi a questo sono
  - ▶ `netstat -r`
  - ▶ `route`
- ▶ Il comando `ip route` restituisce le rotte nel seguente formato
  - ▶ `<destinazione> via <gateway> [dev <device>] proto {static,kernel} [src <sourceAddress>] [metric <metrica>] [link status]`
- ▶ Per far comportare una macchina Linux come router occorre
  - ▶ Disporre di due interfacce, connesse a due reti
  - ▶ Abilitare l'IP Forwarding a livello kernel scrivendo in `procfs` con il comando
    - ▶ `sysctl -w net.ipv4.ip_forward=1`
    - ▶ Comando alternativo:
      - `echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward`

# Routing su IMUNES – Esercizio 1

- ▶ Dopo aver disabilitato l'autoassegnamento degli indirizzi IP creare in IMUNES una topologia di rete composta da:
  - ▶ 2 PC, pc1 e pc2
  - ▶ 1 router, connesso ai due pc
- ▶ Configurare gli indirizzi IP per pc1, pc2 e router, in modo che pc1 e pc2 appartenga a due reti diverse e il router appartenga ad entrambe le reti
- ▶ Aggiungere una rotta che permetta a pc1 di contattare pc2 passando per il router, e viceversa
- ▶ Avviare un traceroute da pc1 a pc2
- ▶ Avviare un ping da pc1 a pc2, ed osservarlo tramite tcpdump sul router
- ▶ *Hint:* comando per aggiungere una rotta:
  - ▶ `ip route add <retedidestinazione> via <gateway> dev <interfaccia>`

# Routing su IMUNES – Esercizio 1 - Soluzione

## Topologia IMUNES



## Comandi eseguiti per la configurazione

- ▶ Router
  - ▶ `ip addr add 10.0.0.1/24 dev eth0`
  - ▶ `ip addr add 10.0.1.1/24 dev eth1`
- ▶ PC1
  - ▶ `ip addr add 10.0.0.2/24 dev eth0`
- ▶ PC2
  - ▶ `ip addr add 10.0.1.2/24 dev eth0`
- ▶ PC1
  - ▶ `ip route add 10.0.1.2 \`  
`via 10.0.0.1 dev eth0`
- ▶ PC2
  - ▶ `ip route add 10.0.0.2 \`  
`via 10.0.1.1 dev eth0`

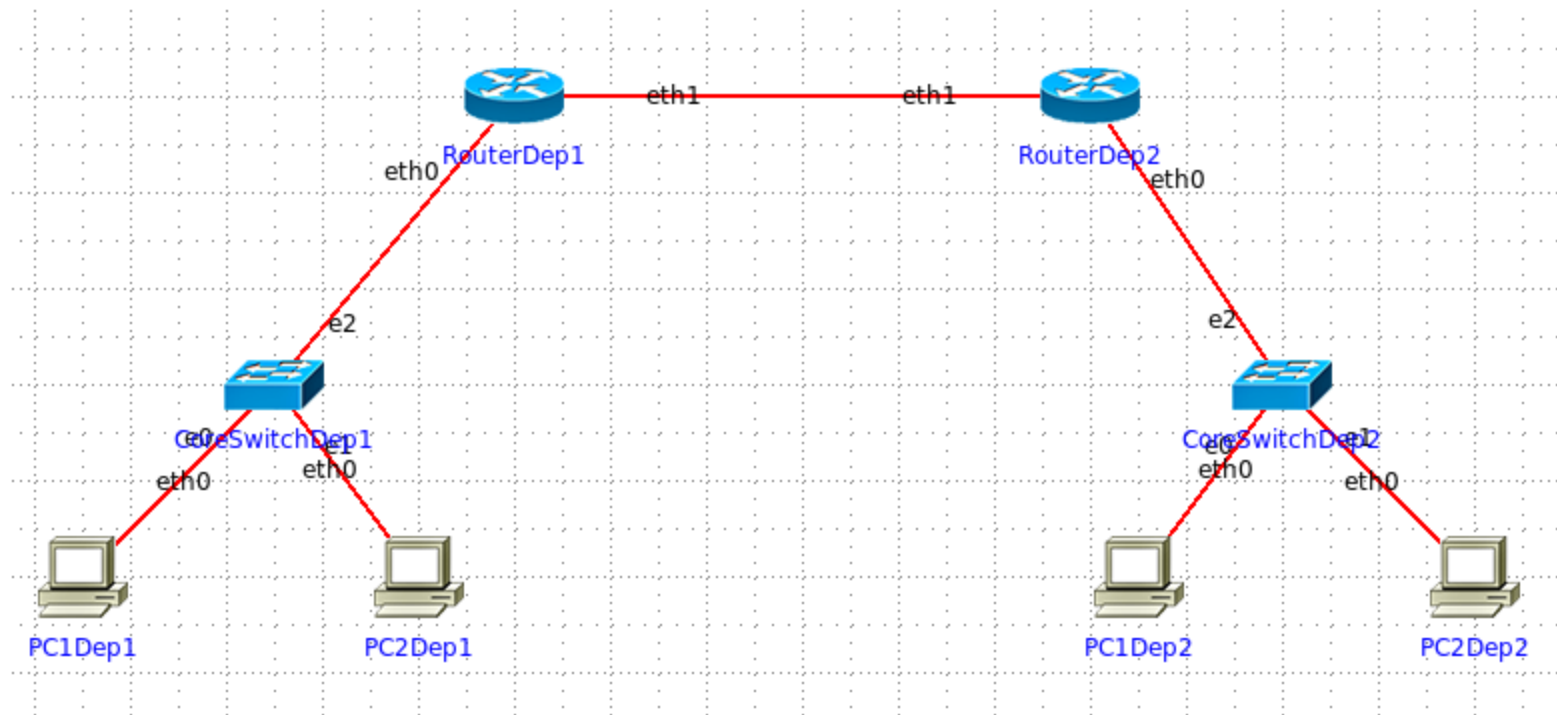
# Routing su IMUNES – Esercizio 2

- ▶ L'azienda «A» possiede due dipartimenti
- ▶ Il primo dipartimento contiene due postazioni collegate a un core switch, che hanno IP appartenente alla rete 172.16.5.0/24
- ▶ Il core switch è connesso tramite un router a una rete condivisa a un altro dipartimento che dispone di una topologia di rete esattamente analoga, con IP appartenenti alla rete 172.17.5.0/24
- ▶ La rete condivisa, composta dai due router connessi tra loro, ha IP appartenenti a 192.168.0.0/24
- ▶ Dopo aver disabilitato l'autoassegnamento degli indirizzi IP realizzare questa topologia con IMUNES e configurare opportunamente la tabella di routing per far dialogare i PC dei due dipartimenti
- ▶ Utilizzare traceroute per visualizzare il percorso dei pacchetti generati dai PC del primo dipartimento e destinati ai PC del secondo dipartimento, e viceversa



# Routing su IMUNES – Esercizio 2 - Soluzione

## Topologia IMUNES



# Routing su IMUNES – Esercizio 2 - Soluzione

## Comandi eseguiti – Assegnamento indirizzi

### ▶ PC1Dep1

- ▶ `ip addr add \`  
`172.16.5.2/24 dev eth0`

### ▶ PC2Dep1

- ▶ `ip addr add \`  
`172.16.5.3/24 dev eth0`

### ▶ RouterDep1

- ▶ `ip addr add \`  
`172.16.5.1/24 dev eth0`
- ▶ `Ip addr add \`  
`192.168.0.1/24 dev eth1`

### ▶ PC1Dep2

- ▶ `ip addr add \`  
`172.17.5.2/24 dev eth0`

### ▶ PC2Dep2

- ▶ `ip addr add \`  
`172.17.5.3/24 dev eth0`

### ▶ RouterDep2

- ▶ `ip addr add \`  
`172.17.5.1/24 dev eth0`
- ▶ `ip addr add \`  
`192.168.0.2/24 dev eth1`

# Routing su IMUNES – Esercizio 2 - Soluzione

## Comandi eseguiti – Assegnamento rotte

### ▶ PC1Dep1 e PC2Dep1

- ▶ `ip route add \`  
`172.17.5.0/24 via \`  
`172.16.5.1 dev eth0`

### ▶ RouterDep1

- ▶ `ip route add \`  
`172.17.5.0/24 via \`  
`192.168.0.2 dev eth0`

### ▶ PC1Dep2 e PC2Dep2

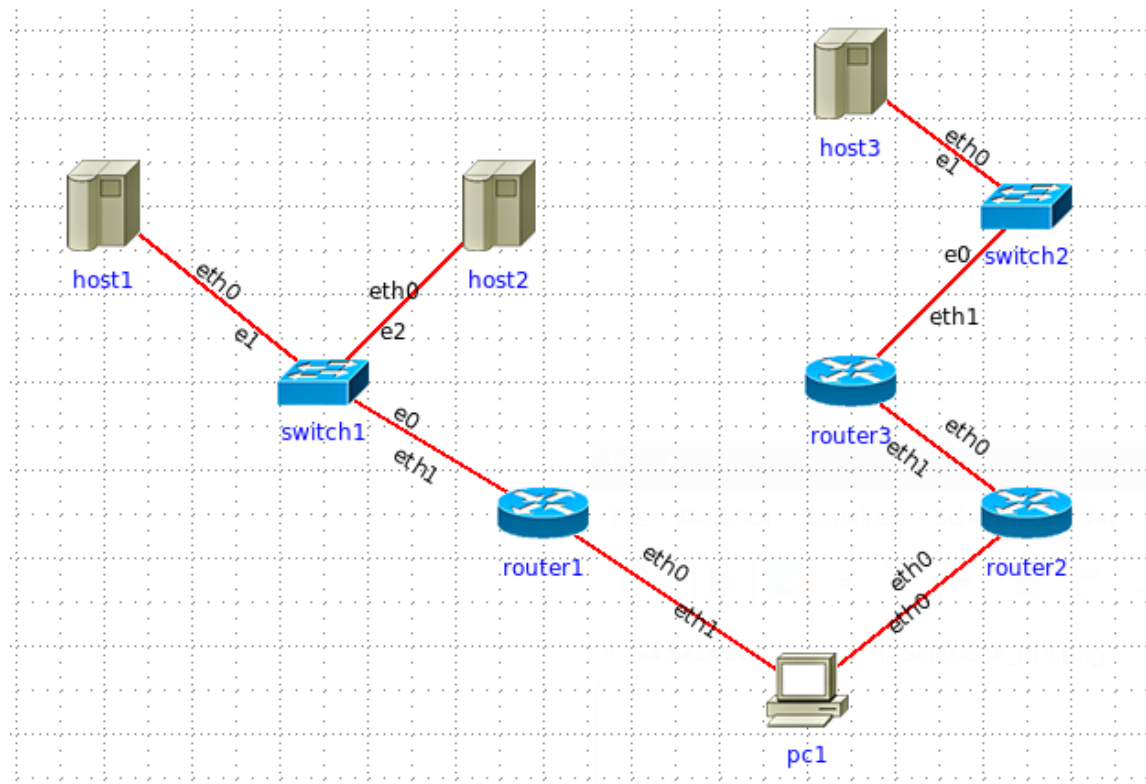
- ▶ `ip route add \`  
`172.16.5.0/24 via \`  
`172.17.5.1`

### ▶ RouterDep2

- ▶ `ip route add \`  
`172.16.5.0/24 via \`  
`192.168.0.1 dev eth0`

# Routing su IMUNES – Esercizio per casa

- Disegnare la seguente topologia, configurare la rete in modo opportuno per permettere la comunicazione tra PC1 e gli host (soluzione nella videolezione)



# Conclusioni

- ▶ Abbiamo ripassato alcuni concetti di teoria del routing statico
- ▶ Abbiamo visto come il routing è implementato in Linux
- ▶ Abbiamo configurato delle tabelle di routing su IMUNES in due topologie tipiche
- ▶ Abbiamo utilizzato lo strumento *traceroute* per diagnosticare l'instradamento dei pacchetti IP

