Università degli Studi di Milano

Laboratorio di Reti di Calcolatori
Corso di Laurea in Informatica (F1X)

Dispensa di Cisco Packet Tracer

Giorgio Biacchi, Giulio Casella, Elena Pagani*

Ultima revisione: 6 dicembre 2021

Sommario

Questo documento è indirizzato agli studenti dell'insegnamento di Reti di Calcolatori per il Corso di Laurea triennale in Informatica (F1X) della Facoltà di Scienze e Tecnologie dell'Università degli Studi di Milano. Il documento illustra le caratteristiche del software Packet Tracer sviluppato da Cisco, in accordo a quanto spiegato nelle relative lezioni in aula del Modulo di Laboratorio del corso.

*Reference author; e-mail: elena.pagani@unimi.it

Indice

1 Introduzione

La presente dispensa è rivolta agli studenti dell'insegnamento di Reti di Calcolatori per il Corso di Laurea in Informatica (F1X), allo scopo di fornire supporto a quanto visto nelle lezioni del Modulo di Laboratorio, e per permettere agli studenti che non hanno avuto la possibilità di frequentare le lezioni di affrontare con successo la prova d'esame. Si assume dimestichezza con gli argomenti spiegati nella parte di Teoria. La dispensa fa riferimento alle caratteristiche di **Cisco Packet Tracer versione 7.3.1**.

Per acquisire ulteriore familiarità con il software, gli studenti possono visionare documentazione, tutorial ed esempi di utilizzo che si trovano nella pagina ufficiale Cisco Networking Academy, ovvero consultare i contenuti mostrati a lezione e opportuna documentazione resa disponibile nella sezione *"Materiale didattico"* del sito del corso su Ariel, o ancora tramite l'apposita documentazione prevista all'interno del software.
In questo modulo di Laboratorio si userà la versione per sistema operativo Windows 64 bit. Si avvisa che Packet Tracer in ambiente Unix risulta normalmente più instabile e quindi si preferisce non adottarlo nel corso

2 Uso di Cisco Packet Tracer

In questa sezione si forniscono nozioni introduttive sull'uso di Packet Tracer, che verranno successivamente dettagliate più approfonditamente nelle sezioni relative ai vari argomenti.

I file prodotti da Packet Tracer hanno suffisso `.pkt` e sono indicati con il termine di *Activity*.

Avvio del tool: alla partenza dell'applicazione è visualizzata una schermata in cui vengono richieste le credenziali di accesso. Gli studenti registrati possono usare le credenziali date in fase di registrazione. In alternativa è possibile accedere come *Guest*. In questo secondo caso, si apre una finestra del browser per il sito ufficiale del tool, e nella finestra di partenza è richiesta (dopo qualche secondo di attesa) la conferma dell'accesso come Guest.
N.B.: L'accesso *Guest* ha limitazioni quali un numero massimo di salvataggi di un file, cosa che può essere scomoda sia in fase di esercitazione sia durante l'esame.

Struttura finestra: come si osserva da Fig. 2.1, la finestra di Packet Tracer coinvolge diverse componenti; in dettaglio:

- la **barra principale** contiene menu per l'accesso alle funzionalità descritte nei punti successivi per le toolbar, e altre funzionalità avanzate aggiuntive.

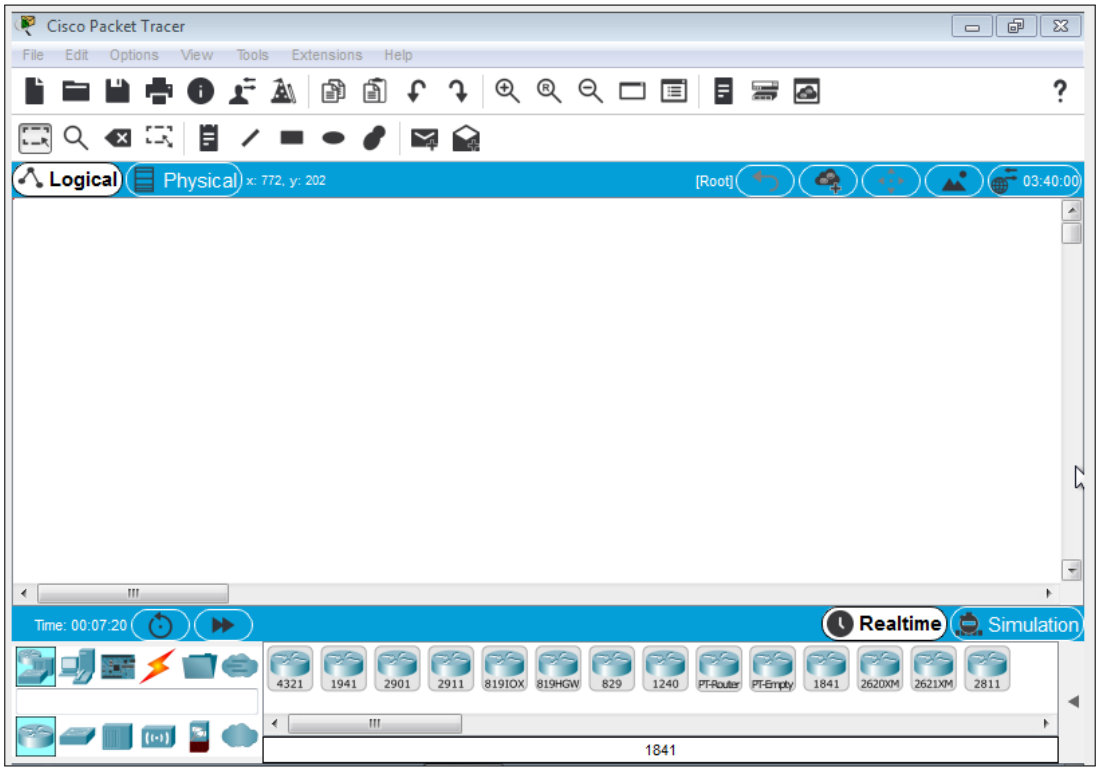


FIGURA 2.1: Interfaccia di Cisco Packet Tracer.

- nella **toolbar principale superiore** sono disponibili icone per l'accesso rapido a normali strumenti per creare un nuovo file `.pkt`, aprire un file esistente, salvare o stampare il file corrente, aggiungere informazioni al file corrente, aggiungere informazioni sull'utente creatore del file, copiare o incollare un componente della rete, cancellare o ripetere l'ultima azione, fare zoom della rete, mostrare l'elenco delle componenti nello spazio di lavoro, visualizzare gli ultimi comandi dati, gestire i template di apparati, e accedere alla documentazione.

- nella **toolbar secondaria superiore** sono presenti nell'ordine gli strumenti per (*i*) selezionare uno o un gruppo di oggetti per lo spostamento nell'area di lavoro o la rimozione; (*ii*) ispezionare alcune caratteristiche dell'oggetto su cui si clicca dopo la selezione dello strumento; (*iii*) cancellare lo/gli oggetto/i selezionato/i; (*iv*) modificare le dimensioni di una forma disegnata; (*v*) inserire una nota di commento; (*vi*) disegnare linee o forme che evidenzino zone della rete; (*vii*) generare pacchetti di ping per test di connettività.

- nella **parte superiore dell'area di lavoro** è attiva la label *Logical* che fa riferimento al fatto che si considera la struttura logica della rete, indipendentemente dalle caratteristiche fisiche del territorio/edificio in cui essa deve essere installata.

Questa è la modalità che verrà usata durante il corso; la label alternativa *Physical* – che permette di specificare la dislocazione dei dispositivi su una mappa o una planimetria – non verrà mai usata.

- nella **parte inferiore dell'area di lavoro** sono disponibili le label *Realtime* e *Simulation* che permettono di osservare il funzionamento della rete in tempo reale oppure passo passo, come si descriverà in dettaglio nel seguito.

- nella **parte sinistra della toolbar inferiore** sono presenti le componenti utilizzabili per la costruzione della propria rete. Ai fini del corso sono di interesse soltanto le categorie *Network Devices*, *End Devices* e *Connections* nella parte superiore. La selezione di una di queste alternative provoca la sottostante visualizzazione di vari apparati di rete quali *Routers*, *Switches*, *Hubs*, di vari tipi di end system o varie categorie di mezzi trasmissivi. Selezionata tra queste la componente desiderata, si ottiene – nella **parte centrale della toolbar inferiore** – la visualizzazione degli apparati, end system o connettori disponibili in dipendenza delle scelte di categoria effettuate nella parte sinistra. In particolare, gli apparati di rete per cui è mostrato un codice alfanumerico simulano i reali apparati Cisco con il medesimo codice.

- nella **parte destra della toolbar inferiore** è visualizzata una freccia, cliccando la quale si accede a strumenti per test di connettività della rete che verranno illustrati nel seguito.

Altre componenti dell'interfaccia grafica saranno via via introdotte nel seguito del documento.

DISEGNO DELLA RETE: per disegnare una topologia di rete è sufficiente selezionare – come sopra descritto – le componenti richieste, e trascinare il dispositivo desiderato, tra quelli mostrati nella parte centrale della toolbar inferiore, nell'area di lavoro della finestra. Per connettere i dispositivi: (*i*) si scelga l'opportuna categoria di link tra quelle mostrate nella parte centrale della toolbar inferiore a seguito della scelta della categoria *Connections*; (*ii*) si clicchi sul dispositivo di partenza del link e si scelga l'interfaccia a cui si desidera collegare il link tra quelle mostrate nella lista che compare; (*iii*) si clicchi sul dispositivo di arrivo del link e si scelga l'interfaccia a cui si desidera collegare il link tra quelle mostrate nella lista che compare. In caso si voglia interrompere l'azione di disegno di un cavo, usare il tasto *Esc*.
Riguardo alla scelta dei tipi di link, si veda la prossima sezione.
Suggerimento: si consiglia di salvare spesso il lavoro svolto (menu a tendina File in alto a sinistra); Packet Tracer a volte si chiude inaspettatamente quando si è lavorato a lungo e si sono accumulate molte informazioni di stato.

Analisi apparati: per un esame rapido della configurazione di un end system o apparato di rete è sufficiente fermarsi con il cursore del mouse sull'apparato stesso. In assenza di configurazione, l'unica informazione mostrata è l'indirizzo MAC (o hardware address) delle schede di rete di cui l'apparato è munito. Per un'analisi più approfondita e per la configurazione dell'apparato si vedano le prossime sezioni dedicate ai vari argomenti.

3 Esercizi preliminari

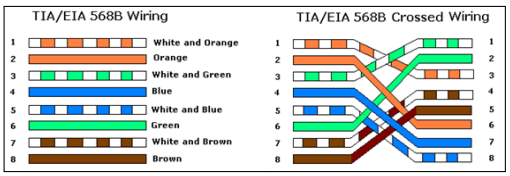
Le attività proposte in questa sezione sono riportate nelle Activity Cablaggi.pkt e Topologie.pkt disponibili nel sito del corso.

Host e apparati di rete: Per ogni componente della rete è possibile esaminare, ed eventualmente modificare, la struttura fisica dell'apparato.

Esempio 3.1. In Packet Tracer, creare nell'area di lavoro un PC, uno hub, un bridge, uno switch e un router. In modalità *selezione oggetto*, cliccare sul PC, scegliere il tab *Physical* e osservare l'elenco dei moduli presenti sul PC e come sia possibile spegnerlo e riaccenderlo: si noti l'*interruttore di accensione* dell'apparato con relativa spia.
Per i quattro apparati di rete è possibile visualizzare l'elenco delle interfacce di rete hardware di cui il dispositivo è dotato. Il tipo di modulo è identificato da un acronimo in cui le ultime lettere indicano nell'ordine se l'interfaccia è in rame (C) o fibra ottica (F), e quindi se la scheda è Ethernet (E), FastEthernet (FE) oppure Gigabit Ethernet (GE). Così ad es. una scheda FFE è una scheda per cavo in fibra e FastEthernet.
Per tutti gli apparati, si può modificare lo hardware dell'apparato come segue: (*i*) spegnere l'apparato; (*ii*) trascinare i moduli che si desiderano eliminare dall'apparato nella sotto-finestra a sinistra *Modules*; trascinare i moduli che si desidera aggiungere all'apparato selezionandoli dalla sotto-finestra *Modules* nello slot desiderato; (*iii*) riaccendere l'apparato. Si noti come sotto il tab *Config* siano riportate, e aggiornate dinamicamente, le interfacce montate sui dispositivi. Inoltre per bridge, switch e router, in questo tab è possibile accendere o spegnere una singola interfaccia di rete.
Suggerimento: Nelle Activity svolte durante il corso, si usino solo i router PT-Router e PT-Empty, opportunamente equipaggiati secondo le necessità. △

Cablature: I link di connessione di Packet Tracer riproducono i cavi attualmente usati, vale a dire cavi UTP composti da 4 coppie di fili, ma anche cavi in fibra ottica, connessioni con linee telefoniche o cavi coassiali o via porta seriale.
Per quanto riguarda i cavi UTP, si usano due coppie di fili per Ethernet e FastEthernet, e 4 coppie per Gigabit Ethernet. Gli 8 fili sono identificati da differenti colori, e sono collegati in un connettore RJ45 ad 8 pin, di cui i pin 1 e 2 servono per la trasmissione

dati e i pin 3 e 6 servono per la ricezione dati. Con queste cablature, a differenza ad es. dei cavi coassiali, non si possono più verificare collisioni sui cavi ma solo eventualmente negli apparati. I cavi si distinguono nelle due seguenti categorie (Fig. 3.1(a)):



	Hub	Switch	Router	Workstation
Hub	Crossover	Crossover	Straight	Straight
Switch	Crossover	Crossover	Straight	Straight
Router	Straight	Straight	Crossover	Crossover
Workstation	Straight	Straight	Crossover	Crossover

FIGURA 3.1: (a) Ordine collegamenti per cavi straight e cross. (b) Casi d'uso: bridge e switch hanno lo stesso comportamento.

cavo dritto (straight): l'ordine di collegamenti dei fili ai pin *di entrambi i connettori* è il seguente: (1) bianco/arancio; (2) arancio; (3) bianco/verde; (4) blu; (5) bianco/blu; (6) verde; (7) bianco/marrone; (8) marrone. I cavi dritti vengono usati per *connettere tipi diversi di dispositivi* (es. un PC con uno hub o con uno switch)

cavo cross: l'ordine di collegamento dei fili ai pin in un connettore (diciamo sx) è uguale a quello dato sopra per i cavi straight, mentre per il secondo connettore (diciamo dx) è: (1) bianco/verde; (2) verde; (3) bianco/arancio; (4) bianco/marrone; (5)marrone; (6) arancio; (7) blu; (8) bianco/blu. Praticamente 1sx $\leftrightarrow$ 3dx, 2sx $\leftrightarrow$ 6dx, 6sx $\leftrightarrow$ 2dx e 3sx $\leftrightarrow$ 1dx, ovvero si scambiano trasmissione e ricezione per permettere la *comunicazione tra dispositivi dello stesso tipo* (es. due PC, due hub, due switch). La motivazione è la seguente: se un PC trasmette sulla coppia bianco/arancio - arancio, collegando due PC attraverso un cavo straight essi trasmetterebbero sulla stessa coppia, collidendo (analogamente per gli altri casi); scambiando i collegamenti questo è evitato.

Si rammenta che, mentre gli standard Ethernet e FastEthernet sono tra loro compatibili, così non è per (Fast)Ethernet e Gigabit Ethernet *se non a Livello Fisico*. **Si ponga quindi sempre attenzione nella scelta delle interfacce a cui i cavi vengono connessi.**

Warning: *Si segnala che, tra router e switch, Packet Tracer prevede l'utilizzo di un cavo dritto, mentre con alcuni tipi di apparati reali andrebbe usato un cavo cross, benchè molti dei dispositivi moderni siano "auto-crossing", quindi si possano utilizzare entrambi i tipi di cavo. In Packet Tracer vi è inoltre una modalità di selezione rapida del link, rappresentato dall'icona del fulmine, per cui si lascia la scelta del tipo di cavo al tool (ma a volte è molto rischioso!).*

Esempio 3.2. In Packet Tracer, collegare coppie di apparati secondo tutte le combinazioni possibili usando l'appropriato tipo di cavo e verificare la corrispondenza con la Fig.3.1(b).

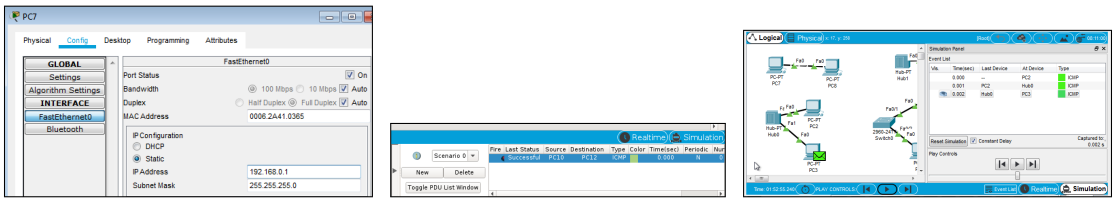
Per le combinazioni che non coinvolgono router, la correttezza è confermata dalle spie verdi sui due estremi del link.[1] △

SIMULAZIONE DI RETE: Negli esempi seguenti, per ottenere una maggiore comprensione del funzionamento della rete, si ricorrerà all'uso del comando `ping`; a tal fine è necessario assegnare degli indirizzi di rete ai PC. Poichè gli aspetti relativi all'indirizzamento a Livello 3 verranno trattati in sezioni successive, per ora si proceda come segue:

1. in modalità *selezione oggetto*, click sul PC di cui si vuole configurare l'indirizzo fa aprire una finestra di configurazione
2. scegliere il tab *Config*, e all'interno di questo, nella sotto-finestra di sinistra, selezionare il bottone *FastEthernet0*
3. nella sotto-finestra di destra mostrata, impostare un indirizzo *static* (scelta di default) nella forma *192.168.0.x* con x compreso tra 1 e 254 (Fig. 3.2(a)). *Il valore di x deve essere differente per ogni PC nella rete costruita.* Premere *Return* per avere la compilazione automatica del campo *Subnet Mask*.

Il test si esegue cliccando sull'icona della busta nella toolbar secondaria e cliccando successivamente sul PC di partenza e sul PC di arrivo nella rete. L'esito del test si può osservare cliccando sulla freccia all'estrema destra della barra inferiore, che fa aprire una nuova sotto-finestra (Fig. 3.2(b)). Se il *ping* va a buon fine, si ottiene uno stato *Successful* per il test.

La simulazione può essere eseguita in modalità *Realtime* o *Simulation*. Nel primo caso



(a) (b) (c)

FIGURA 3.2: (a) Impostazione dell'indirizzo di rete in un PC. (b) Modalità *Simulation*: si mostra anche la sotto-finestra di controllo della simulazione (c).

(default), gli apparati scambiano messaggi autonomamente come in una rete reale. Alla seconda modalità si passa scegliendo l'altro tab nell'angolo in basso a destra della finestra (Fig. 3.2(c)). Si può eseguire una **simulazione passo-passo** facendo procedere il tempo tramite click sul tasto *freccia avanti* più a destra (Next event). Il simbolo *freccia avanti*

[1] Per i router le spie rimangono rosse se le relative interfacce non sono configurate; si rimanda per questo al paragrafo successivo.

centrale fa progredire automaticamente la simulazione (Play), mentre il simbolo *freccia indietro* (Previous event) consente di riavvolgere la simulazione tornando indietro di uno o più passi. Cliccando su una busta si può esaminare il contenuto del messaggio (livello per livello), e vedere le azioni intraprese dall'apparato in cui il messaggio si trova, a fronte della ricezione. Per ripulire la simulazione, cliccare sul bottone *Delete* nella parte inferiore destra della finestra.
Per visualizzare solo alcuni tipi di messaggi si usi la sotto-finestra mostrata alla selezione della modalità *Simulation* (Fig. 3.2(c));[2] in particolare, nella parte alta della finestra sono visualizzati tutti i messaggi scambiati. Negli esempi in questa sezione può accadere di osservare anche messaggi ARP (Address Resolution Protocol) e STP (Spanning Tree Protocol) – protocolli che vengono trattati nelle lezioni di Teoria – o messaggi DTP (Dynamic Trunking Protocol, di Livello 2, non trattato a Teoria) o CDP (Cisco Discovery Protocol, di Livello 2, non trattato a Teoria).

Esempio 3.3. In Packet Tracer, dare un indirizzo a tutti i PC e router nelle topologie formate all'esercizio precedente, e verificare la corretta esecuzione di `ping`. Per i router, si ricorda di mettere a `on` l'interfaccia di rete connessa con l'altro apparato.
Warning: *è possibile eseguire `ping` solo tra due PC, due router, o un router e un PC.* Infatti `ping` è uno strumento di analisi rete di Livello 3, e quindi non opera su dispositivi di Livello 2 o 1.
Warning: *il primo `ping` in modalità Simulation può fallire perchè gli apparati stanno costruendo tabelle di stato interne. Per ripetere il medesimo test, è possibile cliccare due volte sull'icona nella colonna Fire nell'area inferiore destra (Fig. 3.2(b)).* △

Per iniziare, si creino delle semplici topologie di rete per poi verificarne il funzionamento, acquisendo dimestichezza con i vari dispositivi e le funzionalità previste da Packet Tracer (si faccia riferimento anche alla Activity Topologie.pkt sul sito del corso). Tra i vari esempi che si consiglia di realizzare riportiamo:

Esempio 3.4. *Due host collegati con cavo cross* (Figura 3.3(a)). Si verifichi che le interfacce coinvolte riportino una spia verde, che sta a significare che la trasmissione dei dati a livello 1 avviene correttamente. Si assegnino indirizzi ai due PC come illustrato sopra, e si esegua un test di ping. △

Esempio 3.5. *Quattro host collegati a uno hub* rappresentati come 4 host → 1 hub (Figura 3.3(b)). Si verifichi che le interfacce coinvolte riportino una spia verde. Si assegni un indirizzo ad ogni PC e si esegua un ping tra una coppia di host: si può notare dalla simulazione passo-passo come lo hub invii ogni pacchetto su tutte le interfacce tranne quella di ricezione.

[2] Si può riaprire in qualsiasi momento cliccando sul bottone *Simulation*.

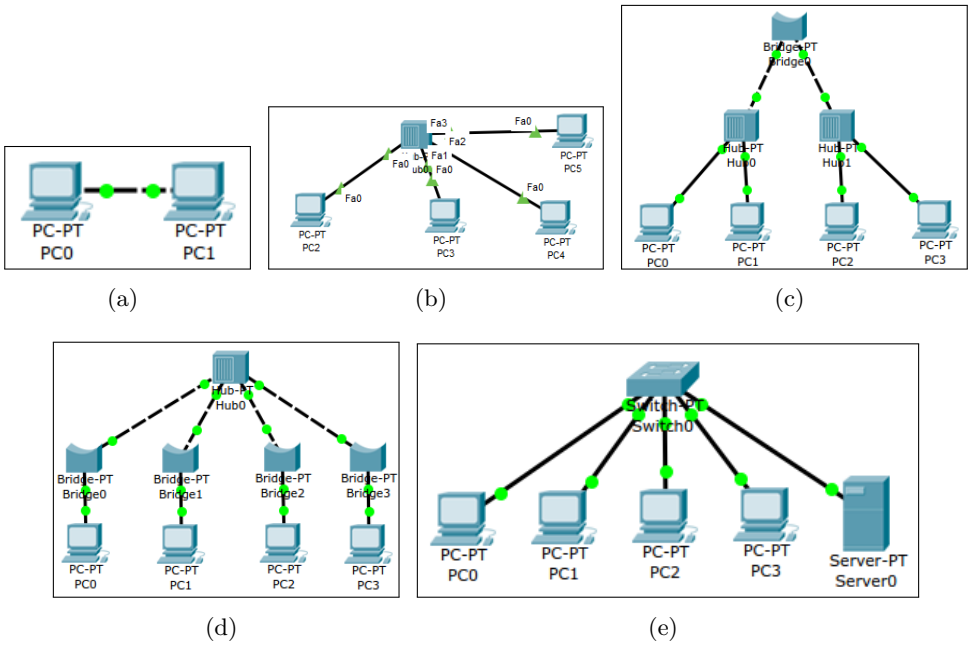


FIGURA 3.3: Topologie di rete utilizzate negli Esempi 3.4-3.8.

Si eseguano quindi – in modalità *Simulation* – due ping contemporanei tra due diverse coppie di host, e si faccia procedere la simulazione passo passo. Si noti come nello hub si verifica una collisione (messaggi a fuoco). △

Esempio 3.6. 4 *host* → 2 *hub* → 1 *bridge* (Figura 3.3(c)). Anzitutto si osservi come ogni volta che si è chiamati a collegare due apparati di rete di Livello 1-2 tra loro, sia necessario usare un cavo *cross*. Il processo di auto-configurazione a livello 2 del bridge si osserva dal fatto che inizialmente i link hanno spie arancioni dal lato del bridge, mentre dopo un certo intervallo le spie diventano verdi; si può accelerare il tempo impiegato per completare questa procedura, usando il bottone *Fast Forward Time* nella modalità *Realtime*.[3] Esaminando la configurazione del bridge (passandoci sopra il cursore del mouse) si può notare che **le interfacce di rete sono numerate come** x/y dove x è il numero di slot in cui la scheda è alloggiata nell'apparato, e y è il numero di interfaccia (o porta) sulla scheda. Entrambe le numerazioni iniziano da 0, e si conta da destra a sinistra, e se necessario dal basso all'alto. Si noti anche che i bridge hanno *due* interfacce.[4]
Si osserva inoltre come, grazie alla sua politica anti-flooding, il bridge apprende gli indirizzi di livello 2 dagli header dei frame, partizionando conseguentemente in due il dominio

[3] In alternativa si può esaminare in maggiore dettaglio cosa accade abilitando il tasto *Play* in modalità *Simulation* ma, come anticipato, sono coinvolti anche protocolli non illustrati nella parte di Teoria.
[4] Un *multi-port bridge* è in effetti uno switch.

di collisione: se si eseguono due ping successivi tra PC sotto lo stesso hub, la diffusione del secondo ping viene limitata dal bridge. Cliccando sul bridge con lo strumento *lente d'ingrandimento* e selezionando la *MAC table* si scopre che la tabella è inizialmente vuota, e i primi ping vengono diffusi broadcast. Ma ogni volta il bridge impara la collocazione del PC sorgente del ping; una volta completato l'apprendimento, più nessun messaggio è inviato broadcast dal bridge, che adopera solo le interfacce opportune per l'inoltro.[5] Per verifica della MAC table, si può **visualizzare l'identificatore di interfacce** o fermando il mouse su una spia dei link (e la visualizzazione scompare dopo qualche tempo), oppure da menu *Options* → *Preferences* scegliendo l'opzione *Always show port labels*. Si può re-inizializzare il bridge spegnendo e accendendo l'interruttore.
Infine, si può riprodurre l'occorrenza di una **collisione** all'interno di uno hub: in modalità *Simulation* si facciano partire due ping da due PC sotto lo stesso hub (verso PC qualsiasi) e si esegua una simulazione passo-passo. La visualizzazione di messaggi a fuoco evidenzia la rilevazione da parte degli apparati di messaggi danneggiati dalla collisione. △

Esempio 3.7. 4 *host* → 4 *bridge* → 1 *hub* (Figura 3.3(d)). Si riproduca un ping tra una qualsiasi coppia di PC. Scopo di questo esercizio è mostrare come, con una siffatta topologia, i bridge non re-inoltrino messaggi broadcast (nonostante si possano verificare collisioni ma solo all'interno dello hub). È quindi interessante notare come combinando i 4 bridge e lo hub si ottiene uno switch. △

Esempio 3.8. 5 *host (*4 *pc e* 1 *server)* → 1 *switch* (Figura 3.3(e)). Si riproduca un ping tra una qualsiasi coppia di PC e si osservi come il comportamento dello switch sia equivalente a quello ottenuto nella rete precedente. In entrambi i casi l'apprendimento di bridge e switch è ottenuto esaminando e ricordando gli indirizzi MAC sorgente indicati nei frame che passano per gli apparati, e le interfacce da cui i frame sono stati ricevuti. Si ricorda che il comando di shell `ping` provoca la generazione di pacchetti ICMP *Echo request* incapsulati in frame che contengo nello header di Livello 2 il MAC address scoperto corrispondere allo IP address.
Si ricorda che per poter aggiungere porte o interfacce sui vari dispositivi e, in ogni caso, per modificarne la configurazione hardware, è necessario prima spegnerli. △

Esempio 3.9. *3 bridge a triangolo.* Si colleghino 3 bridge in una struttura triangolare – ovvero ogni bridge è collegato agli altri due – e in modalità *Simulation* si osservi cosa accade con i messaggi di STP. In modalità *Realtime* si osservi che col progredire del tempo un'interfaccia di uno dei dispositivi non risulta mai abilitata, allo scopo di evitare la formazione di loop. Lo stesso si può osservare costruendo analoga topologia con 3 switch al posto dei 3 bridge. △

[5] Nella grafica l'apprendimento può essere un po' asincrono rispetto alla ricezione dei pacchetti nel bridge.

4 VLAN e Trunking IEEE 802.1Q

Si riprendono qui brevemente i concetti di VLAN illustrati nella parte di Teoria. Una LAN è allo stesso tempo un dominio di collisione – come visto nell'esempio 3.6 – e un *dominio di broadcast*, ovvero un segmento di rete all'interno del quale i dati inviati da uno degli host connessi possono essere diffusi ad ogni altro host attraverso un broadcast di Livello 2. Questo perchè gli host sono connessi al medesimo segmento di rete. Tuttavia a volte si può voler includere "semanticamente" nella stessa LAN host che sono fisicamente connessi a LAN differenti, oppure confinare il traffico entro gruppi di lavoro ad es. per ragioni di (blanda) sicurezza. Ad es., in Dipartimento gli host possono appartenere al gruppo di lavoro docenti o amministrazione, indipendentemente dal fatto che i docenti risiedano in piani diversi, e in qualche piano siano presenti sia docenti sia uffici amministrativi. Per quanto visto finora, questo potrebbe essere ottenuto attraverso la stesura di opportuni cavi che connettano via switch gli host appartenenti alla stessa LAN (esempio 3.8). Questa soluzione può essere costosa in caso di host fisicamente separati da rilevanti distanze, ed è poco pratica in caso di spostamento degli host che richiederebbe la ri-cablatura.
La tecnologia VLAN (Virtual LAN) consente di far comunicare host connessi a segmenti di rete differenti come se fossero nella stessa LAN, e di partizionare il dominio di broadcast di una LAN in più reti logicamente separate. La situazione è rappresentata in Fig.4.1, in cui si vuole che gli host in verde (PC0 e PC1) appartengano ad una VLAN, e gli host in celeste (Laptop0, Laptop1 e PC3) ad altra VLAN. Questo è ottenuto attraverso

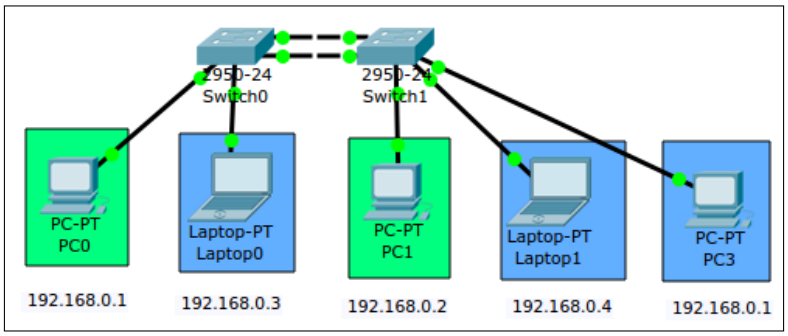


Figura 4.1: Topologia con 2 VLAN.

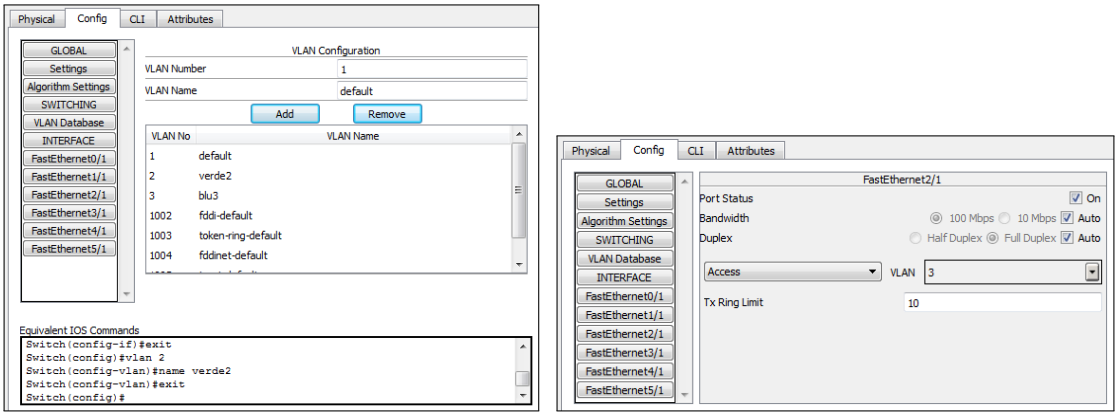
l'opportuna configurazione software degli switch, velocemente modificabile senza costi e senza intervenire sulla topologia fisica della rete. Più VLAN possono poi essere connesse tra loro con un apparato di Livello 3.

Esempio 4.1. 5 *host* $\rightarrow$ 2 *switch* (Figura 4.1). Si riproduca la rete rappresentata, con *una* connessione tra i due switch (senza ulteriori configurazioni) e indirizzi assegnati ai PC quelli raffigurati. Si verifichi che un ping tra una qualsiasi coppia di PC funziona – tranne per i PC con il medesimo indirizzo di rete per cui il PC sorgente in realtà fa

ping a se stesso – per cui la rete è un solo dominio di broadcast. **Warning:** *Si noti che NON è una buona pratica utilizzare il medesimo indirizzo per due host nella stessa rete. Quando si tratteranno gli aspetti di indirizzamento di rete si tornerà sull'argomento.* △

Gli step da seguire per poter definire più VLAN all'interno di una topologia di rete sono molto semplici. Si tratta di configurare gli switch in modo tale da aggiungere al database interno delle VLAN le reti virtuali di cui si vuole disporre. La gestione della VLAN su apparati reali si effettua collegandosi con uno host all'indirizzo MAC dello switch, oppure via seriale direttamente connessi alla porta console del dispositivo. In Packet Tracer, la configurazione per ogni switch si esegue cliccando sullo switch e selezionando il tab *Config* nella finestra che si apre. Selezionare poi dalla sotto-finestra di sinistra il bottone *VLAN Database*. (Si noti che nella parte inferiore della finestra vi è un'area che mostra i comandi che si dovrebbero dare da console equivalenti alle azioni eseguite nella GUI.) Da qui è possibile aggiungere, modificare o cancellare VLAN. Ognuna di queste possiede sia un identificativo simbolico (VLAN Name, ovvero un nome user-friendly) sia un identificativo numerico (VLAN Number, che sarà utilizzato per riferirsi alla VLAN all'interno dei dispositivi di rete).

Best Practice 4.1. *È consigliabile adottare un nome VLAN che contenga al proprio interno il relativo VLAN ID, così da evitare errori e facilitare la corretta configurazione dei dispositivi.*



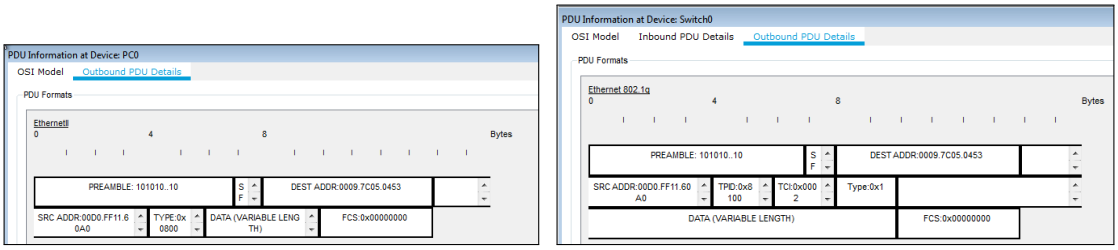
(a) (b)

FIGURA 4.2: Configurazione degli switch per separazione di VLAN.

Si proceda alla configurazione degli switch in modo da distinguere le due VLAN in Fig.4.1, come descritto di seguito:

Esempio 4.2. 5 *host* → 2 *switch* (Figura 4.1). Si modifichi la rete rappresentata ponendo *due* connessioni tra i due switch, i quali vanno configurati in modo da distinguere le

due VLAN. Nella finestra *VLAN Database* aggiungere due VLAN con numero 2 e nome verde2, e con numero 3 e nome blu3 rispettivamente (Fig.4.2(a)).[6] Successivamente, nella sotto-finestra di sinistra, dalla sezione *Interface*, scegliere l'interfaccia del cavo cross che si vuole associare alla VLAN 2, e nella sotto-finestra di destra indicare che tale interfaccia opera in modalità *Access* ed è associata alla VLAN 2 (Fig.4.2(b)); associare similmente l'interfaccia dell'altro cavo cross alla VLAN 3. Eseguire analoga configurazione per i cavi dritti che collegano lo switch ai PC, selezionando la VLAN opportuna volta per volta. Ripetere tutti i passaggi per l'altro switch, facendo attenzione che i due estremi di un dato cavo cross appartengano alla stessa VLAN.
Si verifichi che un ping tra una qualsiasi coppia di PC della stessa VLAN funziona, senza che i messaggi vengano inviati broadcast, mentre i ping tra PC appartenenti a VLAN differenti falliscono. La duplicazione di indirizzo su PC0 e PC3 in due VLAN differenti non crea conflitti. Cliccando sui pacchetti di ping, nella scheda *Outbound PDU Details*, verificare che il formato di frame Ethernet è quello standard mostrato a lezione e in fig.4.3(a). △



(a) (b)

FIGURA 4.3: Confronto tra formati frame inviati per ping in caso d'uso di IEEE 802.1Q: (a) formato frame Ethernet in uscita da uno host; (b) formato frame 802.1Q in uscita da uno switch.

Si faccia riferimento anche alla activity di Packet Tracer VLAN.pkt disponibile sul sito del corso.

IEEE 802.1Q: La soluzione adottata di aggiungere tanti cavi cross tra gli switch quante sono le VLAN attive, configurando all'interno dello switch ognuna delle interfacce a cui tali cavi sono collegati in modo che trasportino i pacchetti di una precisa VLAN, è improponibile nel caso si sia in presenza di un numero elevato di VLAN. In alternativa, è possibile utilizzare una estensione di Ethernet, il **protocollo IEEE 802.1Q** (trunk VLAN), che permette con un solo cavo di risolvere il problema. Con 802.1Q cambia il formato dei frame, che vengono etichettati con l'identificatore di VLAN grazie al quale gli switch sono in grado di smistarli opportunamente; in effetti può essere utile considerare

[6] Alcune VLAN sono definite di default nello switch. Per le proprie VLAN scegliere nomi e id differenti.

i pacchetti che transitano in una VLAN come se fossero opportunamente colorati del colore identificato dalla VLAN stessa.

Esempio 4.3. 5 *host* → 2 *switch* (Figura 4.1). Si modifichi la topologia lasciando un solo cavo cross tra i due switch. Le interfacce del cavo cross vanno modificate – nella scheda dell'opportuna interfaccia (Fig. 4.2(b)) – selezionando la modalità *Trunk* al posto della modalità *Access*, e selezionando quel sottoinsieme di VLAN che si vuole trasportare sul cavo. Le configurazioni delle interfacce degli switch ai PC vanno lasciate immutate rispetto all'esempio precedente.
Si ripetano i ping tra coppie di PC appartenenti alla medesima VLAN, e tra coppie di PC in VLAN differenti, verificando che i domini di broadcast sono effettivamente separati. Analizzare il formato dei frame Ethernet che incapsulano i pacchetti di ping, verificando che è differente rispetto a quello dell'esempio precedente (fig.4.3(b)). △

Esempio 4.4. 9 *host* → 3 *switch*. Si modifichi la topologia aggiungendo uno switch collegato in sequenza[7] agli altri due e suddividendo i 9 host in modo tale che per ogni switch abbia uno host per ognuna di 3 VLAN. Si configurino opportunamente gli switch (*a*) usando la modalità *Access* e ponendo tra gli switch tanti cavi quanti necessario; (*b*) ponendo un solo cavo tra ogni coppia di switch adiacenti e configurando opportunamente gli switch usando 802.1Q.
Si ripetano i ping tra coppie di PC appartenenti alla medesima VLAN, e tra coppie di PC in VLAN differenti, verificando che i domini di broadcast sono effettivamente separati. Analizzare il formato dei frame Ethernet che incapsulano i pacchetti di ping, verificando che nel caso (*b*) sono incluse le informazioni del protocollo 802.1Q (fig.4.3(b)), ovvero il tag protocol identifier di 16 bit `tpid=0x8100` a indicare un frame di tipo 802.1Q, seguito dal tag control information di 16 bit `TCI`, suddiviso in 3 bit di *user priority*, un flag di 1 bit che indica se i MAC address sono in forma canonica, e 12 bit di identificatore VLAN `VID`. Nell'esempio in figura il `VID` è pari a 2 mentre gli altri campi di `TCI` sono a 0. △

Per questo argomento si faccia riferimento anche alla activity di Packet Tracer VLAN_2.pkt disponibile sul sito del corso.

5 Breve introduzione alla Command Line Interface

Così come per il resto di questa dispensa, le informazioni riportate in questa sezione hanno carattere parziale e servono esclusivamente ad introdurre lo studente nell'uso della Command Line Interface (CLI) degli apparati Cisco. Per maggiori informazioni si

[7] Se si cerca di collegare gli switch a triangolo, un'interfaccia resta con spia arancione, disabilitata, grazie al meccanismo di costruzione di uno spanning tree – visto nelle lezioni di Teoria già a proposito dei bridge – che impedisce la formazione di loop tra apparati.

Tabella 5.1: Elenco delle principali modalità di accesso alla CLI.

Prompt	Descrizione
nomeapparato>	User mode
nomeapparato#	Privileged mode (o EXEC-level mode)
nomeapparato`(config)#`	Global configuration mode
nomeapparato`(config-if)#`	Interface sub-mode

possono consultare i lucidi della Prof.ssa Pagani sul sito del corso, e le guide alla CLI raggiungibili dai relativi link riportati tra i siti di interesse sul sito del corso.

Nel mondo reale, gli apparati vengono configurati da linea di comando, ovvero attraverso la Command Line Interface. Se in Packet Tracer la CLI (accessibile dall'omonima scheda del dispositivo) copre tutte le funzionalità fornite dalla GUI, non è vero l'opposto; in altre parole alcune funzioni e strumenti del dispositivo sono configurabili solo tramite CLI. Si può visionare come i comandi effettuati tramite GUI nella scheda *Config* siano tradotti automaticamente nel linguaggio della CLI, nella parte bassa della finestra di configurazione.

Nonostante sia opportuno conoscere alcuni comandi basilari, alcune funzionalità della CLI permettono anche a utenti meno esperti di interagire con lo strumento senza eccessiva difficoltà :

- tasto **?**: mostra una lista dei possibili comandi disponibili (e annessa descrizione) in un certo contesto operativo; inoltre, all'interno di un comando, mostra un elenco dei parametri disponibili per poterlo correttamente completare;
- tasto **tab**: permette di completare un comando immettendone i primi caratteri che lo compongono, purchè questi lo specifichino univocamente;
- uso di forme abbreviate: è possibile utilizzare un comando immettendone solo i primi caratteri; se la stringa immessa è lunga a sufficienza da evitare ambiguità con altri comandi, allora questo viene eseguito come fosse stato digitato nella sua interezza. Ad esempio `en` può sostituire `enable`, così come `config t` il comando `configure terminal`;
- stringa **no** *command*: serve per negare il comando attivo *command*;
- stringa **do** *command*: serve per forzare l'esecuzione di un comando anche quando si è in una modalità in cui quel comando non è disponibile (es. si vuole mostrare la configurazione delle interfacce con `show interfaces` mentre si è in submode *config*). In questo caso lo help '?' non è disponibile.

Diverse sono le modalità di accesso ai comandi della CLI. Nel caso si digiti un comando non ammesso all'interno di una certa modalità , è possibile che venga sollevato un messaggio di errore. Le principali modalità sono elencate in Tabella 8.1. Sebbene la modalità *utente* sia quella a cui si accede di default, i comandi utili agli scopi del corso in questa modalità sono limitati. È possibile accedere alla modalità *privilegiata* attraverso il comando `enable` (che in apparati reali richiede l'inserimento di una password).

Attenzione! Se al prompt della CLI viene scritto in modo errato un comando (es. "enbla"), il sistema cerca di interpretare la stringa come un nome simbolico da risolvere e mostra il messaggio "`Translating "enbla"...domain server (255.255.255.255)`", quindi si blocca per 60 secondi cercando inutilmente di raggiungere un server DNS. Per sbloccare la situazione usare la combinazione di tasti **Ctrl + Shift + 6**.

5.1 Configurazione switch mediante la CLI

Si riporta in questa sezione la modalità per configurare le VLAN su uno switch usando la CLI. Per prima cosa si entri sullo switch in modalità privilegiata. Si entri in modalità configurazione con il comando `configure terminal`.
Analogamente a quanto fatto nella GUI, la prima operazione da compiere è il popolamento del VLAN database. A questo fine, si usa il comando `vlan` seguito dall'identificatore numerico della VLAN da configurare. Si entra così in sub-mode *config-vlan*, dove si definisce il nome della VLAN con il comando `name` seguito dalla stringa del nome. Si ripetano queste operazioni per ogni VLAN (uscendo ogni volta dalla configurazione della VLAN corrente con `exit` per configurare la VLAN successiva).[8]
Successivamente bisogna configurare opportunamente le interfacce. Da modo *config* si entra in sub-mode *configurazione interfaccia* con il comando `interface` seguito dal tipo, il numero di slot e il numero di interfaccia che si desidera configurare; ad esempio `interface FastEthernet 0/1`. Da qui si può usare il comando `switchport` per configurare l'interfaccia in modo *access* o *trunk*.
Nel caso di configurazione in modo *access*, si specifica la modalità d'uso con il comando `switchport mode access`[9] e poi si configura l'interfaccia con il comando `switchport access vlan` seguito dal numero della VLAN. Nel caso di configurazione in modo *trunk* (ovvero usando il protocollo 802.1Q), si abilita l'interfaccia all'uso di 802.1Q con il comando `switchport mode trunk`; quindi si usa il comando `switchport trunk allowed vlan` per indicare quali VLAN possono essere inoltrate attraverso l'interfaccia. Quest'ultimo comando ha vari argomenti: con *add* e *remove* si possono aggiungere o togliere identificatori di VLAN dalla lista; con *all* si permette il passaggio di frame appartenenti a tutte le VLAN nel database; con *none* si disabilita il passaggio di tutte le VLAN; con *except* si permette il passaggio di tutte le VLAN tranne quella indicata.

[8] È possibile popolare il database direttamente da privileged mode con il comando `vlan database`, ma è deprecato. Probabilmente perchè non è evidenziato che ci si trova in submode config.
[9] Questo su molti apparati è il modo di default.

Esempio 5.1. Si ricostruiscano le reti di cui agli esempi 4.2 e 4.3, configurando questa volta tutte le interfacce degli switch unicamente mediante la CLI. Si verifichi la correttezza della configurazione eseguendo dei test di connettività. (Si veda anche la Activity VLAN-CLI.pkt sul sito del corso.) △

6 Indirizzamento a Livello 3 e Subnetting

Per gli esercizi di questa sezione, si ricorda che gli indirizzi usati da IPv4 sono composti da 32 bit, che si possono suddividere in un netID (bit più significativi) e uno hostID (bit meno significativi). Nella **notazione decimale puntata** i 32 bit sono raggruppati in 4 ottetti, di ognuno dei quali viene riportato il valore decimale.
Al fine di ridurre le dimensioni delle tabelle di routing, si usano le **netmask**, che sono sequenze con tutti bit 1 in corrispondenza del netID e bit 0 in corrispondenza dello hostID, che vengono messe in AND bit a bit con l'indirizzo di destinazione in un pacchetto da inoltrare: questo permette di collassare in un'unica riga della tabella di instradamento le informazioni per l'inoltro dei pacchetti a tutti gli host di una data rete, invece di avere una riga per ognuno di essi.
Le strutture di indirizzo particolari sono:

0.0.0.0	"this network"
255.255.255.255	broadcast di Livello 3 sulla rete 0.0.0.0; si propaga con TTL=1
⟨netID⟩.⟨00 . . . 0⟩	indirizzo base della rete
⟨netID⟩.⟨11. . . 1⟩	indirizzo di broadcast di Livello 3 sulla rete netID

Si propongono i seguenti esercizi per familiarizzare con il calcolo di indirizzi di rete.

Esercizio 6.1. Si trasformino da notazione puntata a rappresentazione binaria i seguenti indirizzi di rete:

127.128.129.192	Soluz: 01111111.10000000.10000001.11000000
218.160.179.60	Soluz: 11011010.10100000.10110011.00111100
20.148.67.123	Soluz: 00010100.10010100.01000011.01111011
164.172.205.82	Soluz: 10100100.10101100.11001101.01010010
87.194.104.77	Soluz: 01010111.11000010.01101000.01001101

△

Esercizio 6.2. Si trasformino da rappresentazione binaria a notazione puntata i seguenti indirizzi di rete:

01011011.01110110.00101111.10011111	Soluz: 91.118.47.159
00001100.10001000.01110010.00110111	Soluz: 12.136.114.55
11100111.00100110.01100100.01100000	Soluz: 231.38.100.96

10011011.00011011.01010001.00100010 Soluz: 155.27.81.34
00101011.10111001.01001001.10111111 Soluz: 43.185.73.191
△

Esercizio 6.3. Scrivere in notazione puntata indirizzo broadcast, netmask, e due indirizzi validi di dispositivi per le seguenti reti, facendo riferimento al *classful addressing*: 15.0.0.0; 137.149.0.0; 215.151.59.0.

Soluzione: La rete 15.0.0.0 ha il primo ottetto con valore compreso tra 1 e 127, quindi è una rete di classe A, in cui il netID occupa il primo ottetto e lo hostID gli ultimi 3 ottetti. Perciò l'indirizzo broadcast è 15.255.255.255; la netmask è 255.0.0.0, e due indirizzi validi sono ad esempio 15.0.0.3 e 15.27.84.126.
La rete 137.149.0.0 ha il primo ottetto compreso tra 128 e 191, perciò è una rete di classe B in cui i primi due ottetti costituiscono il netID e gli ultimi due rappresentano lo hostID. Quindi l'indirizzo broadcast è 137.149.255.255; la netmask è 255.255.0.0, e due indirizzi validi sono ad esempio 137.149.0.1 e 137.149.12.6.
La rete 215.151.59.0 ha il primo ottetto compreso tra 192 e 223, quindi è una rete di classe C in cui il netID è rappresentato dai primi 3 ottetti mentre l'ultimo ottetto rappresenta lo hostID. Perciò l'indirizzo broadcast è 215.151.59.255; la netmask è 255.255.255.0, e due indirizzi validi sono ad esempio 215.151.59.7 e 215.151.59.111. △

6.1 Notazione CIDR

Allo scopo di ottimizzare l'assegnamento degli indirizzi di rete minimizzando gli sprechi, col tempo si è superato il concetto di classi di indirizzamento (si veda la trattazione di CIDR nelle lezioni di Teoria), per cui un generico indirizzo di rete può essere frazionato in netID e hostID in qualsiasi posizione. Un indirizzo di rete è rappresentato in *notazione CIDR* come $x.y.w.z/k$ dove k è il numero di bit usati per il netID. Nelle reti dell'esercizio precedente, l'indirizzo di rete sarebbe quindi rappresentato rispettivamente come 15.0.0.0/8; 137.149.0.0/16; 215.151.59.0/24.

Warning! *Rappresentare un indirizzo di rete senza il suffisso /k è un* **errore!**

Esercizio 6.4. Dati i seguenti indirizzi di host – con rappresentata tra parentesi la notazione CIDR per la rete di appartenenza – scrivere (*i*) netmask; (*ii*) indirizzo di rete; (*iii*) indirizzo broadcast; (*iv*) massimo numero di apparati di livello ≥ 3 indirizzabili. Per semplicità di calcolo si passi per la rappresentazione binaria.[10]

```
111.162.136.87 (/12)     206.191.1.207 (/25)     127.172.119.205 (/26)
164.179.205.82 (/30)     171.189.24.102 (/18)    14.177.10.148 (/14)
184.172.171.86 (/23)     87.194.104.77 (/20)     58.85.84.104 (/21)
213.215.174.137 (/29)    57.47.77.159 (/11)      209.179.212.66 (/27)
```

[10] Esercizio tratto da A. Bianco, C. Casetti, P. Giaccone, "Esercitazioni di reti telematiche", CLUT, 2009.

Soluzione: **111.162.136.87 (/12)** → per netID si usa tutto il primo ottetto e metà del secondo. La rappresentazione in binario del secondo ottetto è 10100100; considerando solo i primi 4 bit di netID, essi rappresentano valore 160. La maschera su questo ottetto deve essere 11110000, che corrisponde a 240, mentre il broadcast è 10101111 che corrisponde a 175. Quindi (*i*) netmask = 255.240.0.0, (*ii*) indirizzo di rete= 111.160.0.0/12, (*iii*) indirizzo broadcast = 111.175.255.255, (*iv*) massimo numero apparati = $2^{20} - 2$.
206.191.1.207 (/25) → per netID si usano i primi 3 ottetti più un bit del quarto. La rappresentazione in binario del quarto ottetto è 11001111; considerando solo il primo bit, il quarto ottetto ha valore 128. La maschera su questo ottetto deve essere 10000000 che corrisponde a 128, mentre il broadcast è 11111111 che corrisponde a 255. Quindi (*i*) netmask = 255.255.255.128, (*ii*) indirizzo di rete= 206.191.1.128/25, (*iii*) indirizzo broadcast = 206.191.1.255, (*iv*) massimo numero apparati = $2^7 - 2$.
127.172.119.205 (/26) → per netID sono usati i primi 3 ottetti più due bit del quarto. La rappresentazione del quarto ottetto è 11001101; considerando solo i primi due bit l'ottetto ha valore 192. La maschera deve essere 11000000, mentre il broadcast è 11111111. Quindi (*i*) netmask = 255.255.255.192, (*ii*) indirizzo di rete= 127.172.119.192/26, (*iii*) indirizzo broadcast = 127.172.119.255, (*iv*) massimo numero apparati = $2^6 - 2$.
164.179.205.82 (/30) → per netID sono usati tutti gli ottetti tranne gli ultimi due bit. La rappresentazione del quarto ottetto è 01010010; considerando i primi sei bit l'ottetto ha valore 80. La maschera deve essere 11111100 (252); il broadcast deve essere 01010011 (83). Quindi (*i*) netmask = 255.255.255.252, (*ii*) indirizzo di rete= 164.179.205.80/30, (*iii*) indirizzo broadcast = 164.179.205.83, (*iv*) massimo numero apparati = $2^2 - 2 = 2$.
171.189.24.102 (/18) → per netID sono usati i primi due ottetti e due bit del terzo. La rappresentazione del terzo ottetto è 00011000; considerando i primi due bit l'ottetto ha valore 0. La maschera deve essere 11000000 (192); il broadcast deve essere 00111111 (63). Quindi (*i*) netmask = 255.255.192.0, (*ii*) indirizzo di rete= 171.189.0.0/18, (*iii*) indirizzo broadcast = 171.189.63.255, (*iv*) massimo numero apparati = $2^{14} - 2$.
14.177.10.148 (/14) → per netID si usa il primo ottetto e 6 bit del secondo. La rappresentazione del secondo ottetto è 10110001; considerando i primi 6 bit l'ottetto ha valore 176. La maschera deve essere 11111100 (252); il broadcast deve essere 10110011 (179). Quindi (*i*) netmask = 255.252.0.0, (*ii*) indirizzo di rete= 14.176.0.0/14, (*iii*) indirizzo broadcast = 14.179.255.255, (*iv*) massimo numero apparati = $2^{18} - 2$.
184.172.171.86 (/23) → per netID si usano i primi due ottetti e 7 bit del terzo. La rappresentazione del terzo ottetto è 10101011; considerando i primi 7 bit l'ottetto ha valore 170. La maschera deve essere 11111110 (254); il broadcast è 10101011 (171). Quindi (*i*) netmask = 255.255.254.0, (*ii*) indirizzo di rete= 184.172.170.0/23, (*iii*) indirizzo broadcast = 184.172.171.255, (*iv*) massimo numero apparati = $2^9 - 2$.
87.194.104.77 (/20) → per netID si usano i primi due ottetti e metà del terzo. La rappresentazione del terzo ottetto è 01101000: considerando i primi 4 bit l'ottetto ha valore 96. La maschera è 11110000 (240); il broadcast è 01101111 (111). Quindi (*i*) netmask = 255.255.240.0, (*ii*) indirizzo di rete= 87.194.96.0/20, (*iii*) indirizzo broadcast

= 87.194.111.255, (*iv*) massimo numero apparati = $2^{12} - 2$.
58.85.84.104 (/21) → per netID si usano i primi due ottetti e 5 bit del terzo ottetto. La rappresentazione del terzo ottetto è 01010100; considerando i primi 5 bit esso vale 80. La maschera è 11111000 (248); il broadcast è 01010111 (87). Quindi (*i*) netmask = 255.255.248.0, (*ii*) indirizzo di rete= 58.85.80.0/21, (*iii*) indirizzo broadcast = 58.85.87.255, (*iv*) massimo numero apparati = $2^{11} - 2$.
213.215.174.137 (/29) → per il netID si usano i primi 3 ottetti e 5 bit del quarto ottetto. La rappresentazione del quarto ottetto è 10001001; considerando i primi 5 bit esso vale 136. La maschera è 11111000 (248); il broadcast è 100001111 (143). Quindi (*i*) netmask = 255.255.255.248, (*ii*) indirizzo di rete= 213.215.174.136/29, (*iii*) indirizzo broadcast = 213.215.174.143, (*iv*) massimo numero apparati = $2^3 - 2$.
57.47.77.159 (/11) → per il netID si usa il primo ottetto e 3 bit del secondo. Il secondo ottetto ha rappresentazione 00101111; considerando i primi 3 bit esso vale 32. La maschera è 11100000 (224); il broadcast è 00111111 (63). Quindi (*i*) netmask = 255.224.0.0, (*ii*) indirizzo di rete= 57.32.0.0/11, (*iii*) indirizzo broadcast = 57.63.255.255, (*iv*) massimo numero apparati = $2^{21} - 2$.
209.179.212.66 (/27) → per il netID si usano i primi 3 ottetti più 3 bit del quarto. La rappresentazione del quarto ottetto è 01000010; considerando i primi 3 bit esso vale 64. La maschera è 11100000 (224); il broadcast è 01011111 (95). Quindi (*i*) netmask = 255.255.255.224, (*ii*) indirizzo di rete= 209.179.212.64/27, (*iii*) indirizzo broadcast = 209.179.212.95, (*iv*) massimo numero apparati = $2^5 - 2$. △

Esercizio 6.5. È data una rete che comprende 4 apparati a cui sono assegnati i seguenti indirizzi: 137.116.36.31; 137.116.32.205; 137.116.39.43; 137.116.35.112. Determinare (*i*) indirizzo base della rete di dimensione minima per ospitare tali apparati; (*ii*) netmask; (*iii*) indirizzo broadcast per la rete; (*iv*) massimo numero di *host* indirizzabili.

Soluzione: I quattro indirizzi sono uguali nei primi due ottetti e differiscono a partire dal terzo ottetto; qui dunque vi deve essere il confine tra netID e hostID. Rappresentando il terzo ottetto dei 4 indirizzi si ottengono: 00100100; 00100000; 00100111; 00100011. I primi cinque bit sono sempre uguali e fanno quindi verosimilmente parte del netID, mentre gli ultimi 3 bit variano. Di conseguenza: il valore della maschera sul terzo ottetto deve essere 11111000 (248). Il terzo ottetto nell'indirizzo base ha valore 00100000 (32); il broadcast deve essere 00100111 (39). Quindi l'indirizzo base è 137.116.32.0/21; la netmask è 255.255.248.0; l'indirizzo broadcast è 137.116.39.255. Il massimo numero di host indirizzabili in questa rete è $2^{11} - 3$, dove si sono sottratti l'indirizzo di rete, l'indirizzo di broadcast, e *un indirizzo che deve essere riservato al gateway che serve per connettere l'intera sottorete a Internet*, e non può quindi essere usato per uno end system.
Attenzione! Senza notazione CIDR *non* siamo certi di dove sia il confine tra netID e hostID. In questo esempio è anche possibile che i primi 4 bit facciano parte del netID e i rimanenti dello hostID, portandoci così a una situazione in cui l'indirizzo base è 137.116.32.0/20; la netmask è 255.255.255.240; l'indirizzo broadcast è 137.116.47.255,

e possono essere indirizzati $2^{12} - 3$ apparati. Ecco perchè la notazione CIDR è così importante: dice tutto sulla rete. △

Esercizio 6.6. È data una rete che comprende 4 apparati a cui sono assegnati i seguenti indirizzi: 20.148.67.123; 20.148.67.113; 20.148.67.126; 20.148.67.119. Determinare (*i*) indirizzo base della rete di dimensione minima per ospitare tali apparati; (*ii*) netmask; (*iii*) indirizzo broadcast per la rete; (*iv*) massimo numero di *host* indirizzabili.

Soluzione: I quattro indirizzi sono uguali nei primi tre ottetti e differiscono nel quarto ottetto; qui dunque vi deve essere il confine tra netID e hostID. Rappresentando il quarto ottetto dei 4 indirizzi si ottengono: 01111011; 01110001; 01111110; 01110111. I primi quattro bit sono sempre uguali e fanno verosimilmente parte del netID, mentre gli ultimi 4 bit variano. Di conseguenza: il valore della maschera sul quarto ottetto deve essere 1111000 (240). Il quarto ottetto nell'indirizzo base ha valore 01110000 (112); il broadcast deve essere 01111111 (127). Quindi l'indirizzo base è 20.148.67.112/28; la netmask è 255.255.255.240; l'indirizzo broadcast è 20.148.67.127. Il massimo numero di host indirizzabili in questa rete è $2^4 - 3$, per le ragioni viste sopra. △

Esercizio 6.7. Negli esempi proposti in Sez.3, determinare indirizzi IP e netmask da assegnare agli host coerentemente con i seguenti requisiti:

- Esempio 3.4: gli host appartengono alla rete 10.0.0.0/29
- Esempio 3.5: gli host appartengono alla rete 192.168.90.0/27
- Esempio 3.6: gli host appartengono alla rete 130.192.0.0/16
- Esempio 3.7: gli host appartengono alla rete 87.194.96.0/20
- Esempio 3.8: gli host appartengono alla rete 215.151.59.0/24

Verificare le proprie scelte nella rete realizzata in Packet Tracer, controllando il corretto funzionamento di ping tra gli host. △

6.2 Assegnazione indirizzi per subnetting: allineamento

È possibile dare una struttura logica e gerarchica alle reti, partizionando il range di indirizzi "piatto" assegnato all'organizzazione, in più sottoreti ognuna comprendente un sotto-insieme di apparati. La netmask *all'interno della rete* deve in tal caso isolare sia netID sia subnetID. Sull'argomento si vedano i lucidi della Prof.ssa Pagani tra i *Materiali didattici del corso*.

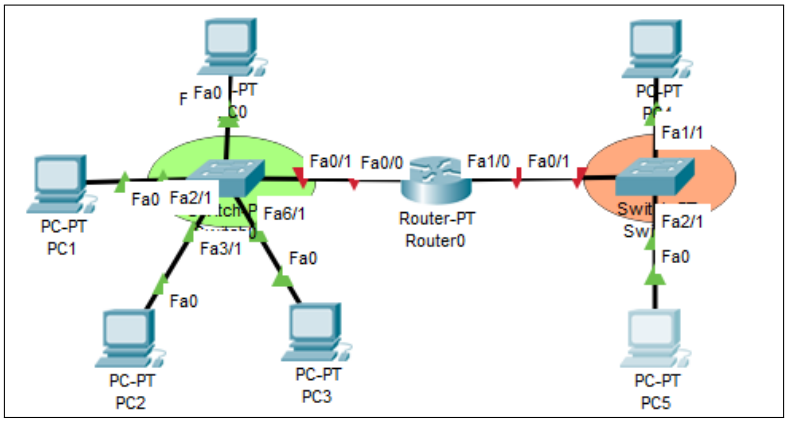


FIGURA 6.1: Un semplice esempio di subnetting (vedi Esempio 6.1).

Esempio 6.1. Si consideri la semplice topologia mostrata in Figura 6.1, dove i cerchi posti attorno agli switch identificano le due sottoreti. Supponiamo che si richieda di configurare la topologia tramite subnetting della rete 192.168.20.96/27. In particolare, si richiede di ricavare le sottoreti sprecando il minor numero possibile di indirizzi IP.

Con la maschera /27 fornita, per lo hostID sono disponibili gli ultimi 5 bit del quarto ottetto. Questo ha rappresentazione binaria **011**|00000 (dove si usa la barra verticale per separare i bit di subnetID – evidenziati – dai bit di hostID) nel quarto ottetto.
Per dimensionare la subnet di sinistra contenente i 5 apparati – ricordando che il primo e l'ultimo indirizzo IP all'interno di una sottorete sono riservati rispettivamente all'indirizzo di rete e a quello di broadcast – si ha bisogno almeno di 7 indirizzi IP, quindi al minimo 3 bit. La più piccola sottorete sufficiente a ospitare 5 apparati è perciò una /29, con subnet mask 255.255.255.248 (cioè quarto ottetto pari a **11111**|000), host address range (range di indirizzi IP riservato agli host) pari a [192.168.20.97 – 192.168.20.102], indirizzo di rete (*subnet ID*) 192.168.20.96/29, e broadcast address 192.168.20.103 (ultimo ottetto pari a **01100**|111).
Passando alla seconda sottorete, essendo composta da 3 apparati, servono 5 indirizzi, quindi si ha nuovamente bisogno di una /29, in cui il subnetID è 192.168.20.104/29 (quarto ottetto pari a **01101**|000), lo host address range risulta pari a [192.168.20.105 – 192.168.20.110], subnet mask ancora 255.255.255.248, broadcast address 192.168.20.111 (ultimo ottetto pari a **01101**|111). (Si veda la Activity Subnet_1.pkt sul sito del corso.) △

Un problema di non poco conto è quello dell'**allineamento**. Si consideri il seguente esempio.

Esempio 6.2. Supponiamo che si richieda di configurare una topologia tramite subnetting della rete 192.168.20.96/27 in due sottoreti: la prima (denominata s_1) che contenga 5 ap-

parati, la seconda (s_2) che ne contenga 14. Supponiamo ora di partire dalla subnet più piccola, di classe /29, con host address range di nuovo pari a [192.168.20.97 – 192.168.20.102], subnet mask 255.255.255.248, indirizzo di rete 192.168.20.96/29, e broadcast address 192.168.20.103. Per s_2 si ha bisogno di una /28, con subnet mask 255.255.255.240. Si potrebbe pensare di scegliere il range di indirizzi successivi adiacenti a s_1, che comportano un host address range pari a [192.168.20.105 – 192.168.20.118], indirizzo di rete 192.168.20.104/28, e broadcast address 192.168.20.119.

Il problema è che una tale configurazione *non è corretta*. Per comprenderne il motivo, consideriamo due host h_1 con indirizzo IP 192.168.20.105 e h_2 con indirizzo IP 192.168.20.118, inclusi nel range appena definito. In binario si ha:

IP address h_1: `11000000 10101000 00010100 01101001`
IP address h_2: `11000000 10101000 00010100 01110110`
subnet mask s_2: `11111111 11111111 11111111 11110000`

Ricordando che se due host stanno nella stessa sottorete, allora le due stringhe binarie ottenute applicando l'operatore AND logico tra indirizzo IP e subnet mask *devono* coincidere, si osserva:

IP address h_1 & subnet mask s_2: `11000000 10101000 00010100 011`**`0`**`0000`
IP address h_2 & subnet mask s_2: `11000000 10101000 00010100 011`**`1`**`0000`

In altri termini, i due host sembrano assegnati a due subnet differenti. △

In termini più rigorosi, per decidere l'indirizzo iniziale di ogni subnet è necessario applicare la seguente regola:

Regola 6.1. *Una rete di dimensione* 2^n *(ovvero che contenga* 2^n *indirizzi) può iniziare solo a intervalli regolari multipli di* 2^n *(a posizioni pari a* $k \cdot 2^n$ *per* $k \geq 0$*); ovvero il primo indirizzo disponibile nello host address range deve essere composto da tutti* 0 *negli ultimi* n *bit per qualsiasi sottorete.*

Il piazzamento ammette che vi possano essere "gap" di indirizzi non assegnati all'interno dello spazio disponibile; in casi reali può in effetti essere una strategia di progettazione (*i*) sovradimensionare le sottoreti rispetto al fabbisogno corrente nel caso se ne preveda una futura crescita di dimensioni; oppure (*ii*) lasciare dei gap di dimensioni congrue nel caso si preveda una futura necessità di aggiunta di nuove sottoreti.

Esistono diverse strategie per evitare di incorrere nel problema dell'allineamento; la più semplice è adottare il seguente algoritmo greedy:

Euristica 6.1. *Si esegue il subnetting allocando spazi di indirizzamento dapprima per le reti più grandi, via via procedendo per dimensione decrescente fino a trattare le reti più piccole, dando la precedenza, in caso di ambiguità , alle sottoreti i cui apparati hanno nome minore. La stessa politica dovrà poi essere adottata all'interno di ciascuna sottorete, configurando le interfacce in modo tale che all'apparato con nome minore sia associato un indirizzo IP minore (es. IP Router0 < IP Router2).*

Esercizio 6.8. Si ri-calcolino gli indirizzi delle sottoreti nell'Esempio 6.2 adottando la strategia proposta, ovvero gestendo dapprima la rete più grande. Si ripetano tutti i passaggi mostrati nell'esempio, e ci si assicuri che il primo e l'ultimo indirizzo nello host address range di entrambe le sottoreti non soffrano del problema dell'allineamento; ovvero *applicando la subnet mask a tutti gli indirizzi validi di host nella medesima sottorete, si deve ottenere lo stesso risultato.*

Soluzione: allochiamo per primo lo spazio per la rete di 14 apparati; considerando l'indirizzo base e l'indirizzo broadcast, servono 16 indirizzi ovvero 4 bit. L'ultimo ottetto dell'indirizzo base è **0110**|0000, per il broadcast diventa **0110**|1111 (111), quindi questa sottorete ha indirizzo base 192.168.20.96/28, indirizzo broadcast 192.168.20.111, indirizzi validi nel range [192.168.20.97 − 192.168.20.110] e subnet mask 255.255.255.240.
Per la seconda sottorete servono cumulativamente 7 indirizzi, ovvero 3 bit. L'ultimo ottetto delll'indirizzo base è **01110**|000, per il broadcast diventa **01110**|111 (119). Quindi questa sottorete ha indirizzo base 192.168.20.112/29, indirizzo broadcast 192.168.20.119, indirizzi validi nel range [192.168.20.113 − 192.168.20.118] e subnet mask 255.255.255.248.
△

Si osservi che tutti i prefissi di rete (in rappresentazione binaria) sono a due a due distinti in almeno un bit.

Per facilitare la progettazione e corretta configurazione delle sottoreti si consiglia di usare la *Tabella progettazione reti* fornita sul sito del corso nella sezione *"Materiale didattico"*.
In tale tabella compare anche la cosiddetta wildcard mask. Questo parametro è la maschera di bit che indica quale parte dell'indirizzo IP esaminare in taluni contesti quali, in Packet Tracer, per indicare la dimensione della sottorete per alcuni protocolli di routing, come OSPF (trattati in sezioni successive). In termini pratici, in maniera semplicistica, la wildcard mask altro non è se non una subnet mask invertita; ad es. con subnet mask 255.255.255.248 si ha wildcard mask pari a 0.0.0.7. Per il momento si può ignorare tale parametro.

Esercizio 6.9. La rete di una PMI ha indirizzi in 10.11.160.0/24 e deve essere suddivisa nelle seguenti 5 sottoreti: (A) amministrazione con 25 host; (G) gestione ordini con 14 host; (K) marketing con 28 host; (R) reparto di produzione con 58 host; (M) magazzino

con 9 host. Per ogni sottorete si indichino indirizzo base, indirizzo di broadcast, subnet mask, e range degli indirizzi possibili per gli host.[11]

Soluzione: Iniziamo col determinare il numero di bit necessari per rappresentare gli indirizzi in ogni sottorete. Per A e K servono 5 bit; per R servono 6 bit; per M servono 4 bit. Per G servono almeno $14 + 2 = 16$ indirizzi; in realtà, *se vogliamo collegare questa sottorete al resto, è necessario prevedere un indirizzo anche per il router*, il che porta a 17 indirizzi per cui di nuovo sono necessari 5 bit.[12]
Se si calcolano i dati delle sottoreti in accordo alla precedente euristica – osservando che le reti per cui è richiesto un uguale numero di bit per lo hostID sono intercambiabili – si ottiene la seguente allocazione degli indirizzi:

R: la netmask usa 2 bit dell'ultimo ottetto (**11**|000000=192), quindi si ha indirizzo base 10.11.160.0/26 (**00**|000000 nell'ultimo ottetto); netmask 255.255.255.192; indirizzo broadcast 10.11.160.63 (ultimo ottetto 00111111); indirizzi disponibili da 10.11.160.1 a 10.11.60.62

A: la netmask usa 3 bit dell'ultimo ottetto (**111**|00000=224), quindi si ha indirizzo base 10.11.160.64/27 (**010**|00000 nell'ultimo ottetto); netmask 255.255.255.224; indirizzo broadcast 10.11.160.95 (ultimo ottetto 01011111); indirizzi disponibili da 10.11.160.65 a 10.11.60.94

K: la netmask usa 3 bit dell'ultimo ottetto, quindi si ha indirizzo base 10.11.160.96/27 (**011**|00000 nell'ultimo ottetto); netmask 255.255.255.224; indirizzo broadcast 10.11.160.127 (ultimo ottetto 01111111); indirizzi disponibili da 10.11.160.97 a 10.11.60.126

G: la netmask usa 3 bit dell'ultimo ottetto, quindi si ha indirizzo base 10.11.160.128/27 (**100**|00000 nell'ultimo ottetto); netmask 255.255.255.224; indirizzo broadcast 10.11.160.159 (ultimo ottetto 10011111); indirizzi disponibili da 10.11.160.129 a 10.11.60.158

M: la netmask usa 4 bit dell'ultimo ottetto, quindi si ha indirizzo base 10.11.160.160/28 (**1010**|0000 nell'ultimo ottetto); netmask 255.255.255.240; indirizzo broadcast 10.11.160.175 (ultimo ottetto 10101111); indirizzi disponibili da 10.11.160.161 a 10.11.60.174

La soluzione illustrata implementa l'allocazione mostrata in Fig.6.2(a); si noti che i prefissi dell'ultimo ottetto appartenenti alla subnet mask non creano ambiguità: solo R ha i primi due bit pari a 00; nessuna tra A, G e K ha i primi 3 bit pari a 101 come ha M. La soluzione in Fig.6.2(b) è ugualmente corretta; per completezza si riportano qui i parametri delle reti con questa seconda soluzione (le netmask ovviamente non cambiano):

[11] Gli esercizi da 6.9 a 6.11 sono tratti da temi d'esame.
[12] Questo è vero anche per le altre reti, ma non fa cambiare la potenza di 2 necessaria per rappresentare tutti gli indirizzi.

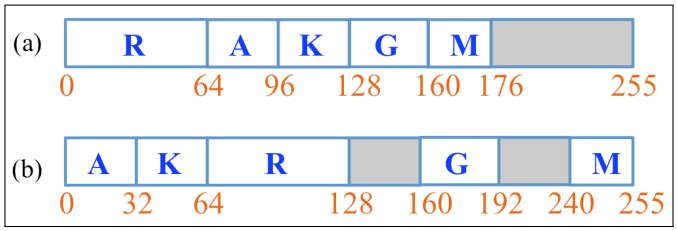


FIGURA 6.2: Due esempi corretti di allocazione sottoreti (vedi Esercizio 6.9).

R: indirizzo base 10.11.160.64 (01|000000 nell'ultimo ottetto); indirizzo broadcast 10.11.160.127

A: indirizzo base 10.11.160.0 (000|00000 nell'ultimo ottetto); indirizzo broadcast 10.11.160.31

K: indirizzo base 10.11.160.32 (001|00000 nell'ultimo ottetto); indirizzo broadcast 10.11.160.63

G: indirizzo base 10.11.160.160 (101|00000 nell'ultimo ottetto); indirizzo broadcast 10.11.160.191

M: indirizzo base 10.11.160.240 (1111|0000 nell'ultimo ottetto); indirizzo broadcast 10.11.160.255

△

Esercizio 6.10. A un'azienda è assegnata la rete 87.215.180.0/22, che deve essere suddivisa nelle seguenti sottoreti: (P) produzione con 74 host; (S) server con 14 host; (D) dirigenti con 20 host; (I) impiegati con 135 host; (O) ospiti con 32 host. Per ogni sottorete si indichino indirizzo base, indirizzo di broadcast, subnet mask, e range degli indirizzi possibili per gli host.

Soluzione: a titolo di esempio si riporta qui solo l'allocazione ottenibile applicando l'euristica enunciata; altre soluzioni corrette sono possibili come nell'esercizio precedente.

I: per gli indirizzi servono 8 bit (256 indirizzi), quindi si ha indirizzo base 87.215.180.0/24 (terzo ottetto 10110100), netmask 255.255.255.0 (terzo ottetto 11111111), indirizzo broadcast 87.215.180.255

P: per gli indirizzi servono 7 bit (128 indirizzi), quindi si ha indirizzo base 87.215.181.0/25 (terzo ottetto 10110101, quarto ottetto tutto a 0), netmask 255.255.255.128 (quarto ottetto 1|0000000), indirizzo broadcast 87.215.181.127 (quarto ottetto 01111111)

O: per gli indirizzi servono 6 bit (64 indirizzi, considerando anche indirizzo base, indirizzo broadcast, e un indirizzo per il router); quindi si ha indirizzo base 87.215.181.128/26 (quarto ottetto 10|000000), netmask 255.255.255.192 (quarto ottetto 11|000000), indirizzo broadcast 87.215.181.191 (quarto ottetto 10|111111)

S: per la ragione di cui all'esercizio 6.9, per gli indirizzi servono 5 bit (32 indirizzi), quindi si ha indirizzo base 87.215.181.192/27 (quarto ottetto 110|00000), netmask 255.255.255.224 (quarto ottetto 111|00000), indirizzo broadcast 87.215.181.223 (quarto ottetto 110|11111)

D: per gli indirizzi servono 5 bit (32 indirizzi), quindi si ha indirizzo base 87.215.181.224/27 (quarto ottetto 111|00000), netmask 255.255.255.224, indirizzo broadcast 87.215.181.255 (quarto ottetto 111|11111).

△

Esercizio 6.11. A un campus universitario è assegnata la rete 165.43.96.0/21, che deve essere suddivisa nelle seguenti sottoreti: (A) amministrazione con 232 host; (LB) Laboratori del Dip.Biologia con 14 host; (UB) uffici del Dip.Biologia con 78 host; (LI) Laboratori del Dip.Informatica con 92 host; (UI) uffici del Dip.Informatica con 115 host. Per ogni sottorete si indichino indirizzo base, indirizzo di broadcast, subnet mask, e range degli indirizzi possibili per gli host.

Soluzione: A titolo di esempio si riporta qui solo l'allocazione ottenibile applicando l'euristica enunciata; altre soluzioni corrette sono possibili come nell'esercizio precedente.

A: per rappresentare gli indirizzi servono 8 bit (256 indirizzi), quindi si ha indirizzo base 165.43.96.0/24, netmask 255.255.255.0, indirizzo broadcast 165.43.96.255

UI: per rappresentare gli indirizzi servono 7 bit (128 indirizzi), quindi si ha indirizzo base 165.43.97.0/25, netmask 255.255.255.128 (quarto ottetto 1|0000000), indirizzo broadcast 165.43.97.127 (quarto ottetto 0|1111111)

LI: per rappresentare gli indirizzi servono 7 bit (128 indirizzi), quindi si ha indirizzo base 165.43.97.128/25 (quarto ottetto 1|0000000), netmask 255.255.255.128, indirizzo broadcast 7 bit (128 indirizzi), quindi si ha indirizzo base 165.43.97.255 (quarto ottetto 1|1111111)

UB: per rappresentare gli indirizzi servono 7 bit (128 indirizzi), quindi si ha indirizzo base 165.43.98.0/25, netmask 255.255.255.128, indirizzo broadcast 165.43.98.127 (quarto ottetto 0|1111111)

LB: analogamente agli esempi precedenti, per rappresentare gli indirizzi servono 5 bit (32 indirizzi), quindi si ha indirizzo base 165.43.98.128/27 (quarto ottetto 100|00000), netmask 255.255.255.224, indirizzo broadcast 165.43.98.159 (quarto ottetto 100|11111)

Si noti che A, UI e UB sono disambiguate dal fatto che differiscono nel terzo ottetto (estratto dalla maschera); similmente LI e LB differiscono anche nel terzo ottetto (oltre che nel quarto rispetto alle altre 3 reti). △

6.3 Configurazione router in Packet Tracer

L'esempio seguente mostra come sia possibile configurare le interfacce dei router in Packet Tracer, in modo tale da suddividere la rete in sottoreti (*subnet*). In Packet Tracer è possibile configurare le interfacce direttamente tramite GUI. Prima di tutto si ricorda (Esempio 3.2) che è possibile configurare lo hardware di un apparato – router inclusi – aggiungendo e togliendo le interfacce di rete desiderate. Nei router questo si può fare attraverso la scheda *Physical* della finestra che si apre cliccando sul router. È inoltre necessaria un'ulteriore operazione per rendere funzionante un'interfaccia: nella scheda *Config*, sezione INTERFACE, si clicchi sull'interfaccia che si vuole accendere, e si marchi la checkbox in alto a destra *On* (nella sotto-finestra della CLI si può osservare che questo equivale a dare un comando di *no shutdown* per l'interfaccia). Nella stessa finestra, per configurare gli indirizzi delle interfacce, è sufficiente indicare, all'interno dei campi omonimi, indirizzo IP e subnet mask dell'interfaccia corrispondente.
Negli host va configurato l'indirizzo del router (gateway) nella scheda di configurazione generale; si faccia attenzione ad assegnare a PC e corrispondente interfaccia del router indirizzi appartenenti alla stessa sottorete, e ad indicare correttamente l'indirizzo di quell'interfaccia del gateway nello host.

Nel calcolo dei parametri delle rete si deve tenere conto, oltre che del numero degli host, della necessità di *prevedere un indirizzo aggiuntivo per il default gateway* che serve per collegare la sottorete con il resto del mondo (Esercizio 6.9). Tale indirizzo è assegnato in accordo ad una delle due seguenti politiche:[13]

Best Practice 6.1. *Per l'assegnamento di indirizzo al default gateway si adotti e si mantenga per coerenza una delle due seguenti politiche: (i) il default gateway ha assegnato* SEMPRE *l'indirizzo più alto tra quelli disponibili nella sottorete; oppure (ii) il default gateway ha assegnato* SEMPRE *l'indirizzo più basso tra quelli disponibili nella sottorete.*

Esercizio 6.12. Si realizzino in Packet Tracer le reti degli esercizi 6.9-6.11 e si verifichi la correttezza della propria soluzione eseguendo un ping tra host di ogni coppia di sottoreti. △

7 Protocollo ARP

Utilizzando l'activity di cui all'Esercizio 6.12 è possibile osservare il comportamento del protocollo ARP. Si faccia partire un ping tra due host appartenenti a LAN differenti, in modalità *Simulation*: si osserveranno le icone di due messaggi pronti a partire, il primo dei quali (come si può osservare fermandocisi sopra con il cursore del mouse) è di

[13] In questo corso si predilige la politica (i).

tipo ARP. Cliccando sul messaggio ARP ed esaminandone il contenuto, si vede che esso ha indirizzo di destinazione di livello 2 pari a tutti bit 1 (FFFF.FFFF.FFFF) – quindi sarà inviato in broadcast sulla LAN – mentre l'indirizzo IP del destinatario è quello del gateway. Questo avviene perchè lo host sorgente, considerando la propria configurazione di rete, ha individuato l'indirizzo IP del destinatario del ping come appartenente ad altra rete, e quindi deve inoltrare il messaggio ICMP al proprio gateway ma deve prima scoprire l'indirizzo MAC del gateway.
Si verifichi che il messaggio ARP viene diffuso broadcast dallo switch (in generale, i messaggi ARP passano attraverso gli apparati di Livello 1-2 senza essere processati); solo il router risponde con i propri dati. All'arrivo del messaggio ARP, usando l'applicazione Command Prompt sul PC (vedere sezione 12), si dia a prompt il comando arp -a (che chiede di visualizzare la ARP table dello host per tutte le entry) e si noti la riga creata per il gateway nella ARP cache dello host. All'arrivo della risposta, partirà il pacchetto ICMP che verrà correttamente instradato verso il gateway.
Il primo ping può non andare a buon fine perchè i dispositivi – in particolare il router – non hanno ancora tutti costruito le ARP table. Nell'esempio sopra: una volta che lo host sorgente ha scoperto il MAC address del gateway, incapsula il pacchetto ICMP Echo Request con l'indirizzo IP della destinazione dentro un frame con l'indirizzo MAC del gateway. Quando il gateway riceve tale frame e lo decapsula, la prima volta non possiede l'indirizzo MAC corrispondente all'indirizzo IP dello host destinazione. Scarta pertanto il pacchetto ICMP (provocando il fallimento del primo ping), ma fa partire un messaggio di ARP Request sull'interfaccia collegata alla rete in cui si trova il destinatario per scoprirne l'indirizzo MAC. Rieseguendo il medesimo test attraverso doppio clic, nella finestra più a destra della barra inferiore, sull'icona nella colonna *Fire* si otterrà successo perchè il router ha completato l'apprendimento.
Al fine di replicare l'esperimento, è possibile cancellare il contenuto della ARP table sullo host sorgente con il commando arp -d nel Command Prompt.

8 Uso della Command Line Interface nei router

L'uso della CLI per la configurazione dei router segue le linee generali descritte a proposito degli switch; le principali modalità sono elencate in Tabella 8.1.

In modalità utente si può eseguire un comando di ping; questo può essere utile per testare specifiche interfacce di router e verificare le configurazioni per l'instradamento e la conseguente raggiungibilità di reti (si veda sezione relativa).

In modalità privilegiata (comando `enable`), con il comando `show running-config` (risp. `show startup-config`) è possibile visualizzare la configurazione attuale (risp. iniziale, di default vuota) del router; con il comando `write` (abbreviazione di `write memory`) si provvede al **salvataggio della configurazione attuale del router**, cosicchè

Tabella 8.1: Elenco delle principali modalità di accesso alla CLI.

Prompt	Descrizione
nomeapparato>	User mode
nomeapparato#	Privileged mode (o EXEC-level mode)
nomeapparato`(config)#`	Global configuration mode
nomeapparato`(config-if)#`	Interface mode
nomeapparato`(config-subif)#`	Subinterface mode
nomeapparato`(config-line)#`	Line mode
nomeapparato`(config-router)#`	Router configuration mode

a seguito di spegnimento e riaccensione del dispositivo ne venga ripristinato lo stato corretto (di default una volta che il router viene spento, perde tutte le configurazioni); per finire, con il comando `configure terminal` si passa nella modalità di configurazione globale.

Tra i comandi più semplici in modalità configurazione globale, ricordiamo:

- `hostname RouterCisco`: assegna al Router il nome passato come argomento (RouterCisco nel caso in esame);
- `enable password cisco`: assegna una password per accedere alla modalità di configurazione. Usualmente l'accesso ad un router è protetto da una password, ed una password differente può essere associata alla modalità privilegiata, così ad esempio da permettere ad alcuni utenti di poter esplorare la configurazione di un router ma non di cambiarla;
- `exit`: esce dalla modalità di configurazione attuale, ritornando alla precedente;
- `no ip domain-lookup` disabilita permanentemente il tentativo di tradurre una stringa non riconosciuta come comando, ricorrendo alla traduzione da parte del DNS.

È possibile mostrare un elenco delle interfacce presenti sull'apparato e dei relativi parametri di funzionamento attraverso il comando `show interfaces`. Si ricorda che una interfaccia viene identificata da 3 parti: (i) parte alfabetica → tecnologia (es. fastEthernet); (ii) primo numero → slot in cui è inserita; e (iii) secondo numero → posizione all'interno dello slot; questi ultimi due sono separati dal carattere / (es. fastEthernet 0/0). Per configurare una interfaccia, ad esempio fastEthernet 0/0, è sufficiente digitare il comando `interface fastEthernet 0/0` mentre si è in modalità `config`: il prompt mostrerà la nuova modalità a cui si è acceduti (modalità interfaccia). In questa modalità è possibile, ad esempio:

– `ip address`: assegnare all'interfaccia l'indirizzo IP via DHCP o direttamente fornendone l'ottetto e la netmask (es. `ip address 192.168.1.0 255.255.255.252`);

– `no shutdown`: cambiare stato alla scheda ponendolo su up;

– `exit`: uscire dalla modalità di configurazione attuale, ritornando alla precedente.

Esistono tantissimi altri comandi: i più importanti saranno discussi nelle sezioni di competenza.

Esercizio 8.1. Si configuri il router dell'Esercizio 6.9 attraverso i comandi della CLI, come segue:
• rispondere *no* alla richiesta di configuration dialog
• digitare `enable` dal prompt di modalità non privilegiata
• digitare `conf t` per entrare in modalità configurazione da terminale
• cambiare nome al router con il comando `hostname` seguito dalla stringa desiderata
• entrare in modalità configurazione di un'interfaccia (es. `inter fastEthernet 0/0`)
• assegnare opportuni indirizzo IP e netmask a quell'interfaccia come mostrato sopra, e verificare da interfaccia grafica che ora passando con il mouse sul router quell'interfaccia risulta associata all'indirizzo desiderato
• attivare l'interfaccia con il comando `no shutdown`. L'estremo del link nell'interfaccia grafica deve diventare verde
• uscire con `exit` dalla configurazione di quell'interfaccia e ripetere per tutte le altre interfacce
• quando si è certi della correttezza della propria configurazione, si può uscire dalla modalità di configurazione, e salvare con il comando `copy running-config startup-config` – equivalente al comando `write memory` – così che il router all'accensione parta con la configurazione effettuata. △

9 Connessione tra VLAN

Negli esempi visti in Sez.4, ogni VLAN costituisce un *dominio di broadcast a Livello 2*, ma non è possibile la comunicazione tra host appartenenti a VLAN differenti. Nella maggior parte delle situazioni reali, è opportuno che host appartenenti a VLAN differenti possano dialogare tra loro. Affinchè ciò sia possibile non si può fare altro che (*i*) assegnare indirizzi agli host nelle VLAN così che ogni VLAN corrisponda ad una rete differente; (*ii*) *introdurre un apparato di Livello* 3, ovvero un router che – grazie alle sue capacità di instradare tra reti differenti – fungerà da gateway per host appartenenti a differenti VLAN.[14] Poichè host appartenenti a VLAN diverse necessitano di un gateway differente,

[14] Un gateway è un router che capisce anche dettagli di protocolli di Livello differente dal 3. In questo caso, capisce anche il protocollo IEEE 802.1Q.

tale router dovrebbe, in linea di principio, disporre almeno di tante interfacce quante sono le VLAN, comportando lo stesso problema di scalabilità delle cablature osservato in precedenza per gli switch.

Esempio 9.1. Si parta da una rete con topologia come nell'Esempio 4.3 (Figura 4.1) in cui due switch sono collegati a host appartenenti a due VLAN, e con un cavo tra loro. Gli end system della VLAN verde prendono indirizzi nello spazio 192.168.0.0/24, mentre gli host della VLAN azzurra prendono indirizzi nello spazio 192.168.1.0/24. Collegare un router su *due interfacce* del medesimo switch, con due cavi straight. Una delle due interfacce dello switch va configurata in modalità *Access* per una VLAN, e l'altra per la seconda VLAN. Configurare le due interfacce usate dal router con due indirizzi validi nelle due VLAN (coerentemente con la VLAN di appartenenza scelta nello switch per ogni interfaccia). Configurare ogni host di ogni VLAN con default gateway uguale all'indirizzo del router usato per l'interfaccia associata a quella VLAN. Verificare che ora funziona *ping* anche tra host di VLAN differenti grazie alle capacità di inoltro del router, eseguendo una simulazione passo-passo. (Si faccia riferimento anche alla activity *VLAN_router_no1Q.pkt* nella sezione *"Materiale didattico"* del sito del corso.) △

Warning! Quando si usa un router per collegare più VLAN, come appena mostrato è necessario che tali VLAN costituiscano sottoreti differenti, con spazi di indirizzamento distinti. Infatti un router *non può avere più interfacce sulla medesima rete* (allo scopo di evitare loop nelle rotte). Per verificare questa caratteristica, si tenti semplicemente di configurare un router associando a due sue interfacce indirizzi validi nello spazio di indirizzamento 192.168.0.0/24.

La soluzione che permette l'uso di un solo cavo consiste nel configurare il router così che possa trasportare 802.1Q, e nel creare tante **sotto-interfacce virtuali** quante sono le VLAN, corrispondenti alla medesima interfaccia fisica: in tal caso sarà sufficiente una sola interfaccia per poter permettere a host appartenenti a VLAN differenti di poter comunicare tra loro, ognuno con un proprio gateway (diverso nelle varie VLAN).

Esempio 9.2. Si consideri la topologia riportata in Figura 9.1 (si faccia riferimento, a tal proposito, alla activity *VLAN_router_1Q.pkt* nella sezione *"Materiale didattico"* del sito del corso). Si assegnino ai PC da 0 a 4 gli indirizzi IP da 192.168.0.1 a 192.168.0.4 (con set automatico della subnet mask), con gateway 192.168.0.254 (in sostanza l'ultimo indirizzo IP disponibile è riservato al gateway, mentre i primi sono assegnati agli host). Analogamente si assegnino ai PC da 5 a 8 gli indirizzi IP da 192.168.1.1 a 192.168.1.4, con gateway 192.168.1.254. Per poter avvalersi del protocollo 802.1Q, gli switch devono essere configurati in modo tale da aggiungere due VLAN all'interno del proprio database (es. Uffici con id 20 e Magazzino con id 30). Le interfacce che collegano host sulla VLAN Magazzino (risp. Uffici) devono essere configurate in modalità *Access* sulla VLAN 30 (risp. 20). L'interfaccia su cui è inserito il cavo cross che collega i due switch

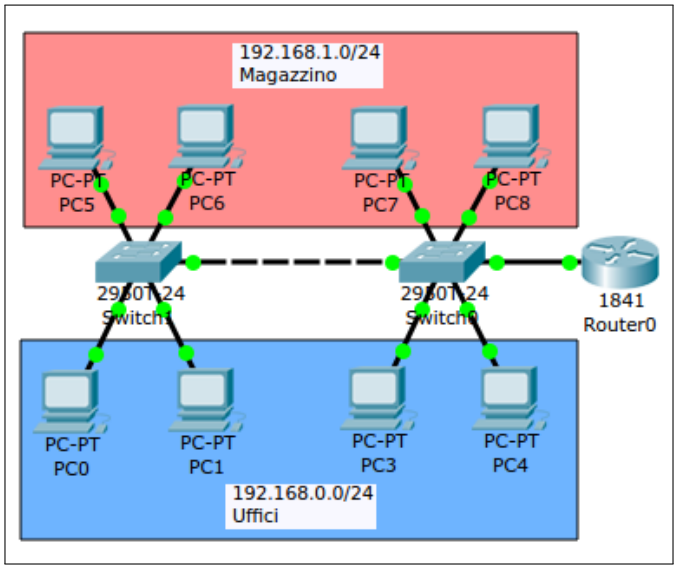


FIGURA 9.1: Topologia con 2 VLAN in cui tutti gli host possono comunicare tra loro.

deve invece essere configurata in modalità *Trunk*, eventualmente specificando lo ID delle due VLAN di cui si vuole permettere il transito dei pacchetti (di default il traffico è consentito a tutte le VLAN, sia quelle predefinite sia quelle user-defined). Analogamente deve essere configurata in modalità *Trunk* l'interfaccia dello switch collegata al router. Rimane solo da configurare il router. A tal fine, si usa il concetto di sotto-interfaccia (*subinterface*), una suddivisione logica e non fisica di una interfaccia. Usando la CLI, in modalità configurazione (enable → configure terminal), si digitano i comandi:

```
interface FastEthernet 0/0.20
encapsulation dot1Q 20
ip address 192.168.0.254 255.255.255.0
no shutdown
exit

interface fastEthernet 0/0.30
encapsulation dot1Q 30
ip address 192.168.1.254 255.255.255.0
no shutdown
exit

interface fastEthernet 0/0
no shutdown
exit
```

dove in sostanza nella prima riga viene creata la sotto-interfaccia responsabile dei pacchetti delle singole VLAN (*aggiungendo un punto ed un numero identificativo come suffisso al nome standard dell'interfaccia*); nella seconda viene specificato l'uso del protocollo Trunking 802.1Q per la specifica VLAN a cui la sotto-interfaccia appartiene; nella terza viene configurato l'indirizzo IP e subnet mask della sotto-interfaccia (alias il gateway configurato sugli host appartenenti alla VLAN in questione). Per finire, nell'ultima riga viene abilitata la sotto-interfaccia. Una simile sequenza di comandi è ripetuta per la sotto-interfaccia di ogni VLAN. Infine, è necessario accendere l'interfaccia fisica su cui sono definite le due sotto-interfacce.
Di tali istruzioni, l'unica che può essere compiuta direttamente tramite GUI è quest'ultima, cliccando sull'apposito checkbox Port Status nella scheda Config → FastEthernet0/0.
Verificare che funzioni *ping* anche tra host di VLAN differenti grazie alle capacità di inoltro del router, eseguendo una simulazione passo-passo. △

Best Practice 9.1. *È buona norma, al fine di aumentare la leggibilità delle configurazioni e di ridurre la possibilità di errore, utilizzare lo ID numerico della VLAN anche nel suffisso del nome della sotto-interfaccia del router.* △

Warning: Negli elaborati di esame capita di osservare due situazioni scorrette e pertanto foriere di errori e malfunzionamenti. Si consideri una semplice topologia VLAN – switch – router con VLAN ID `VID` diverso da 1.

I. Lo switch ha configurate entrambe le interfacce come *Access* per `VID`; il router *non* ha configurata subinterface per quel `VID`. Nei test con ping tutto funziona, perchè il router riceve frame standard Ethernet e non si accorge di nulla.
MA: se esiste la possibilità futura che quella VLAN venga suddivisa in più altre VLAN, non configurare appropriatamente da subito lo switch con un'interfaccia *Trunk* verso il router, e quest'ultimo con una sotto-interfaccia, può portare a errori di cofigurazione e comportamenti indesiderati.

II. Lo switch ha configurata l'interfaccia verso la VLAN come *Access* per `VID` e l'interfaccia verso il router come *Trunk*. Mentre il router non ha configurazione di sorta. In questo caso è evidente subito dal ping che qualcosa non funziona: il problema deriva dal fatto che il router in questo caso riceve dei frame incapsulati in accordo a 802.1Q, non li capisce perchè non ha l'interfaccia configurata con il medesimo incapsulamento, e quindi li scarta e genera una notifica di errore *ICMP Host Unreachable*.

10 DHCP

Il servizio DHCP serve per configurare in maniera automatica indirizzo IP, subnet mask e gateway negli host appartenenti a una certa sottorete. Ovviamente non è necessario avere

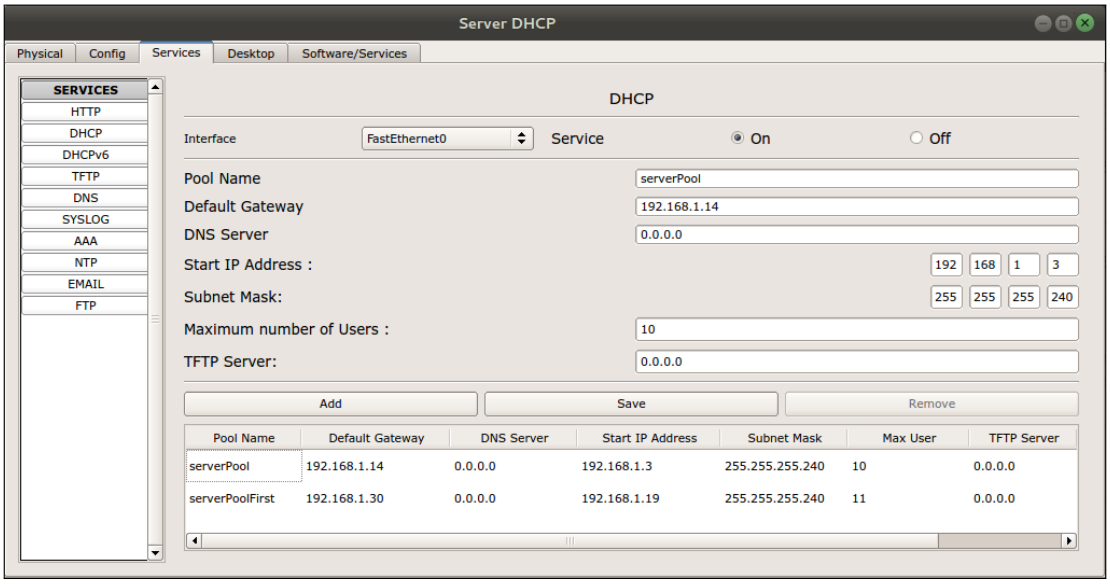


FIGURA 10.1: Esempio di configurazione dei serverPool.

un server DHCP per ogni sottorete, purchè si specifichino correttamente i serverPool, ovvero strutture dati definite all'interno del server DHCP che specificano come gestire gli host appartenenti a una certa sottorete.

Attenzione! se una rete include anche, o solo, indirizzi dinamici, è comunque necessario eseguire il calcolo dei parametri di indirizzamento considerando lo spazio necessario per *tutti* gli indirizzi necessari sia statici sia dinamici.

Per poter attivare il servizio DHCP, nella scheda Services → DHCP del server deputato a erogare detto servizio è necessario attivare il checkbox Service.
Si noti che in tale finestra è già presente un serverPool di default, di cui sono automaticamente riempiti i campi in base agli indirizzi della subnet in cui il server si trova. Verosimilmente sarà necessario modificare tale configurazione in base alle specifiche necessità. Nel caso in cui si configuri un ulteriore serverPool per la medesima subnet, quello di default sarà quello considerato da Packet Tracer. Tale caratteristica *non* è presente nelle reti reali.
Per configurare il serverPool è necessario indicare il gateway di default (Default Gateway), il primo indirizzo IP disponibile che può essere assegnato agli host (Start IP Address), la subnet mask (Subnet Mask) e per finire il numero massimo di host che possono avvalersi del servizio (Maximum number of Users). La Figura 10.1 mostra un esempio di configurazione compatibile con lo scenario in esame (si noti che alcuni parametri, quali Start IP Address e Maximum number of Users sono fissati in modo da rispettare specifiche di progetto non riportate nell'esempio). Si noti infine che per tutti gli host che si avvalgono del servizio DHCP si dovrà attivare il checkbox *DHCP* nella scheda *Config*.

Esempio 10.1. Nella subnet rosa di Figura 9.1, aggiungere un server DHCP, collegarlo allo switch di sinistra configurando il collegamento in modalità *Access* per la VLAN appropriata, e assegnargli un indirizzo IP statico con opportuna netmask e default gateway. Si osserva che *i server devono sempre avere un indirizzo statico, per poter essere contattabili*. Si modifichi uno host collegato allo switch di destra in modo che la sua configurazione di rete non sia statica ma determinata da DHCP. Configurare il server per fornire servizio DHCP per 20 host a partire da 192.168.1.10. Si esegua una simulazione passo-passo osservando lo svolgimento delle 4 fasi del protocollo DHCP viste nelle lezioni di Teoria.
Warning: se si passa con il mouse sopra lo host con indirizzamento dinamico, può capitare di osservare che ha assunto un indirizzo nella rete 169.254.0.0/16. Questo è semplicemente un meccanismo di auto-configurazione non visto nelle lezioni di Teoria,[15] e che non ostacola la corretta esecuzione della activity e della simulazione. Si può comunque eliminare in simulazione tale configurazione, spegnendo e riaccendendo lo host (tasto accensione nella scheda *Physical*). △

Si noti che in una delle due sottoreti non è presente un server DHCP. Si noti inoltre che una richiesta DHCP inviata in broadcast sarà confinata all'interno del broadcast domain della VLAN in cui lo host richiedente si trova.

È possibile usare il medesimo DHCP server anche per VLAN in cui tale server non risiede, ma solo sfruttando l'aiuto di un router che connetta le due VLAN e che sia adeguatamente istruito allo scopo, ovvero in grado di operare come *DHCP proxy*.
Nel caso il server debba operare per più VLAN, per ogni altra subnet che contiene host gestiti dal server DHCP – ad eccezione di quella in cui il server è inserito – si dovrà creare un ulteriore serverPool specificando il nome del pool (Pool Name), e come in precedenza il gateway di default (Default Gateway), il primo indirizzo IP disponibile che può essere assegnato agli host (Start IP Address), la subnet mask (Subnet Mask) e per finire il numero massimo di host che possono avvalersi del servizio (Maximum number of Users). Si faccia attenzione che *i suddetti parametri devono essere relativi allo spazio di indirizzamento della VLAN considerata*, non della VLAN in cui il server risiede.
Un ulteriore passaggio è necessario per specificare l'indirizzo IP del server DHCP che configurerà gli host nelle altre sottoreti: in sostanza si deve istruire il router per inoltrare una richiesta DHCP in broadcast all'indirizzo IP specificato. In generale, tale istruzione va specificata per ogni interfaccia la cui subnet non contiene il server DHCP ma in cui gli host si avvalgono di tale servizio di configurazione automatica.

Esempio 10.2. Si consideri la topologia riportata in Figura 10.2. In particolare sono definite due VLAN: First (abbreviazione di First Floor) con id 200 e Second (abbreviazione di Second Floor) con id 400; l'interfaccia GigabitEthernet0/0 di Router0 è

[15] Descritto nella RFC 3927, "Dynamic Configuration of IPv4 Link-Local Addresses".

inoltre configurata in modo tale da permettere agli host appartenenti alle due VLAN di poter comunicare tra loro in modalità *Trunk* (si faccia riferimento alla Sezione 9). Se 192.168.1.30 è l'indirizzo del gateway nella subnet First Floor e 192.168.1.14 quello

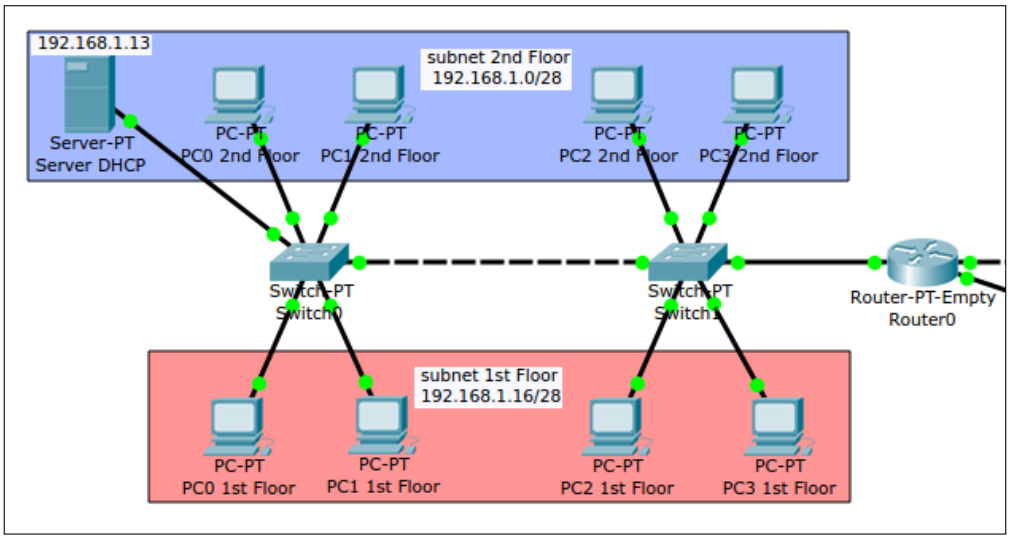


FIGURA 10.2: Topologia con 2 VLAN e servizio DHCP.

del gateway nella subnet Second Floor, Router0 viene dunque configurato con i seguenti comandi:

```
interface GigabitEthernet0/0.400
encapsulation dot1Q 400
ip address 192.168.1.14 255.255.255.240

interface GigabitEthernet0/0.200
encapsulation dot1Q 200
ip address 192.168.1.30 255.255.255.240
ip helper-address 192.168.1.13
```

Si noti come per la subinterface della VLAN *First* si istruisce il gateway ad "aiutare nella configurazione di indirizzi" re-inoltrando i pacchetti DHCP al server all'indirizzo indicato. Verificare che con tale configurazione si configurano con successo gli host in entrambe le VLAN. △

11 ROUTING

In questa sezione ci si prefigge di configurare correttamente i router in modo tale da permettere a host appartenenti a sottoreti differenti di poter comunicare tra loro.

Una possibilità consiste nel configurare manualmente delle **rotte statiche**; questa è una scelta indicata in particolare nel caso in cui la topologia di rete sia piccola e l'amministratore di rete desideri far passare il traffico lungo rotte particolari per ragioni di distribuzione di carico o di sicurezza. Una rotta statica può essere configurata sia da interfaccia grafica, sia da CLI con il seguente comando da modalità `config`:

```
ip route netID netmask next-hop
```

dove netID è l'indirizzo base della rete di destinazione, netmask è la sua maschera, e next-hop è l'indirizzo IP del router adiacente verso cui inoltrare i messaggi per gli apparati della rete destinataria. Si parta da questo semplice esempio.

Esempio 11.1. Si consideri la topologia illustrata in Figura 11.1. Si configurino i router

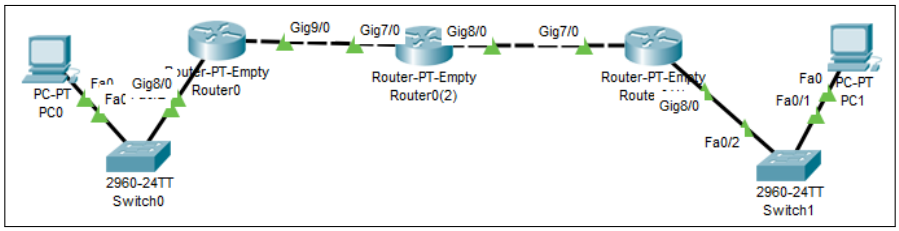


FIGURA 11.1: Algoritmi di routing (vedi Esempio 11.1).

in modo che le interfacce tra ogni coppia di router abbiano indirizzi appartenenti alla stessa sottorete. Nel caso di configurazione corretta, il *ping* tra router avrà esito positivo. Si operi ad esempio seguendo le indicazioni riportate in Tabella 11.1. Si configurino inoltre le interfacce che connettono i PC ai router, ad esempio fornendo indirizzo IP 192.168.1.2 e 192.168.10.2 rispettivamente a PC0 e PC1, ricordandosi di configurare correttamente subnet mask e gateway, in modo tale che tutte le coppie di dispositivi adiacenti possano comunicare tra loro.

TABELLA 11.1: Configurazione delle interfacce dei router presenti in Figura 11.1.

apparato	interfaccia	indirizzo IP	subnet mask
Router0	GigabitEthernet8/0	192.168.1.1/24	255.255.255.0
	GigabitEthernet9/0	192.168.0.1/30	255.255.255.252
Router1	GigabitEthernet8/0	192.168.10.1/24	255.255.255.0
	GigabitEthernet7/0	192.168.0.6/30	255.255.255.252
Router2	GigabitEthernet7/0	192.168.0.2/30	255.255.255.252
	GigabitEthernet8/0	192.168.0.5/30	255.255.255.252

Cliccando con lo strumento lente d'ingrandimento sui router e selezionando la voce *Routing table*, si nota come le tabelle di routing sui router contengano solo informazioni sulle reti direttamente connesse (Type = C): è necessario configurare i protocolli di routing su tali apparati affinchè sia possibile far comunicare tra loro anche host appartenenti a sottoreti tra loro distanti.

Si configurino rotte statiche per rendere raggiungibili ogni coppia di dispositivi in rete. In particolare:

- Router0 deve raggiungere la rete tra Router1 e Router2, e la rete tra Router1 e il suo host
- Router1 deve raggiungere la rete tra Router0 e Router2, e la rete tra Router0 e il suo host
- Router2 deve raggiungere la rete tra Router0 e il suo host, e la rete tra Router1 e il suo host

A valle della configurazione verificare con *ping* che ogni coppia di dispositivi (host e router) sia raggiungibile. (Si faccia riferimento, a tal proposito, alla activity *static.pkt* nella sezione *"Materiale didattico"* del sito del corso.) △

Esempio 11.2. Si consideri la topologia illustrata in Figura 11.2. Si configurino i router

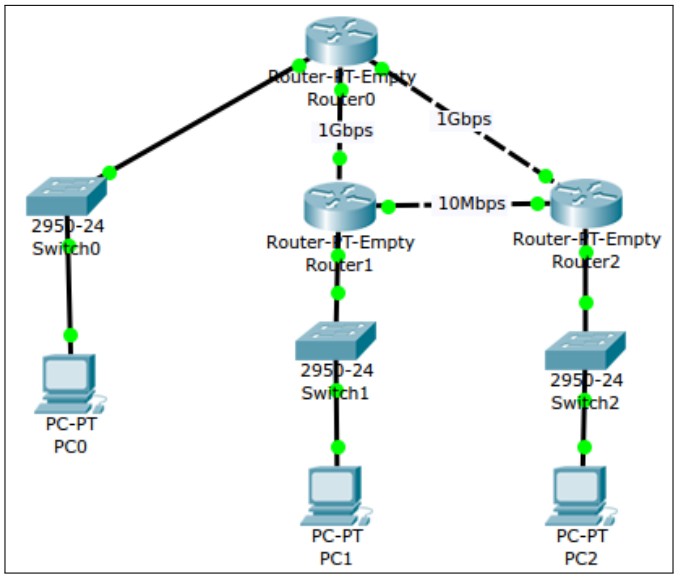


FIGURA 11.2: Algoritmi di routing (vedi Esempio 11.2).

in modo che le interfacce tra ogni coppia di router abbiano indirizzi appartenenti alla stessa sottorete. Nel caso di configurazione corretta, il *ping* tra router avrà esito positivo.

Si operi ad esempio seguendo le indicazioni riportate in Tabella 11.2. Si configurino inoltre le interfacce che connettono i PC ai router, ad esempio fornendo indirizzo IP 192.168.1.2, 192.168.10.100 e 192.168.20.100 rispettivamente a PC0, PC1 e PC2, ricordandosi di configurare correttamente subnet mask e gateway, in modo tale che tutte le coppie di dispositivi adiacenti possano comunicare tra loro. Come fatto per l'esempio precedente,

Tabella 11.2: Configurazione delle interfacce dei router presenti in Figura 11.2.

apparato	interfaccia	indirizzo IP	subnet mask
Router0	GigabitEthernet0/0	192.168.0.1/30	255.255.255.252
	GigabitEthernet1/0	192.168.0.5/30	255.255.255.252
	GigabitEthernet2/0	192.168.1.9/24	255.255.255.0
Router1	GigabitEthernet0/0	192.168.0.6/30	255.255.255.252
	GigabitEthernet1/0	192.168.10.1/24	255.255.255.0
	Ethernet2/0	192.168.0.9/30	255.255.255.252
Router2	GigabitEthernet0/0	192.168.0.2/30	255.255.255.252
	GigabitEthernet1/0	192.168.0.10/30	255.255.255.252
	Ethernet2/0	192.168.20.14/24	255.255.255.0

si configurino rotte statiche per rendere raggiungibili ogni coppia di dispositivi in rete. A valle della configurazione verificare con *ping* che ogni coppia di dispositivi (host e router) sia raggiungibile. (Si faccia riferimento, a tal proposito, alla activity *staticRoute.pkt* nella sezione *"Materiale didattico"* del sito del corso.) △

La modalità di routing statico, oltre ad essere noiosa e richiedere molta attenzione, ha un problema: in caso di guasto di un link o di un apparato, la rete non sa auto-riconfigurarsi, e due o più host possono rimanere scollegati fino ad intervento e ri-configurazione da parte dell'amministratore di rete.

Esistono due principali famiglie di algoritmi di routing **dinamici** (si faccia riferimento alle lezioni di Teoria):

- *Distance-vector* (es. RIP): ogni router non ha conoscenza completa della topologia di rete, ma solo del proprio vicinato. Come metrica per valutare il percorso migliore utilizza lo hop count, ovvero il numero di passi necessari per raggiungere la destinazione;
- *Link-state* (es. OSPF): richiede una conoscenza completa della rete. Utilizza altre metriche per valutare il percorso migliore, legate al costo della singola tratta, tipicamente inversamente proporzionale alla velocità della stessa.

11.1 RIP

Vediamo ora come sia possibile istruire i router per far sì che si scambino le informazioni sulle sottoreti conosciute, partendo dal protocollo RIP. In modalità configurazione globale si digiti il comando `router rip` così che il prompt diventi `config-router#`. Utilizzando il tasto `?` otteniamo al solito un elenco dei possibili comandi disponibili in questa modalità; tra questi citiamo:

- `distance`: definisce la distanza amministrativa. Come già anticipato, se RIP usa il concetto di distanza, OSPF usa quello di peso: si tratta di due metriche differenti che non sono confrontabili. La distanza amministrativa serve appunto per confrontare due rotte che usano metriche differenti, ovvero in topologie che fanno uso di un mix di protocolli di routing differenti; più è piccola, più la rotta è affidabile. Per le reti direttamente connesse, la distanza amministrativa è 0, il che la rende la rotta più affidabile. Per rotte statiche la distanza amministrativa è 1.

- `network`: premesso che il router deve rendere pubbliche le reti che conosce, il comando `network 192.168.1.0` permette di comunicare che il router conosce la rete 192.168.1.0. Si ottiene una configurazione completa di RIP ripetendo tale comando su ogni router per tutte le reti note. Da notare che per configurare RIP è sufficiente indicare il subnet address; manca in altre parole la subnet mask. Tale scelta deriva dal fatto che la versione 1 di RIP (default in Packet Tracer) è *classful*, ovvero non permette la suddivisione della rete in sottoreti, al contrario della versione 2 che dà la possibilità di suddividere la rete in sottoreti più piccole.

- `version`: permette di configurare quale versione di RIP si desidera utilizzare: la 1 (classful) o la 2 (classless inter-domain routing). Si tenga presente che in Packet Tracer di default RIP assume indirizzamento classful (cioè version 1), quindi la maschera viene dedotta dal valore del primo ottetto dell'indirizzo (Esercizio 5.3) e in alcune situazioni si possono raggruppare in una sola riga di configurazione più sottoreti che ricadrebbero nella stessa rete di una data classe.

- `passive-interface`: permette di indicare su quali interfacce *non* si vogliono inviare le varie notifiche. Ad esempio il comando

  ```
  passive-interface GigabitEthernet0/0
  ```

  evita che le notifiche sulle reti conosciute siano inviate sull'interfaccia passata come argomento. Il vantaggio è la riduzione del traffico generato dagli algoritmi di routing in sottoreti che contengono solo host (LAN), per i quali tali notifiche non hanno alcun significato e vengono ignorate.

Esempio 11.3. Riprendendo la topologia configurata nell'Esempio 11.2, si configurino le reti note ai router presenti nella topologia, utilizzando i comandi presenti in Tabella 11.3.

Tabella 11.3: Configurazione del protocollo RIP sui router presenti in Figura 11.2.

apparato	comando
Router0	`network 192.168.0.0` `network 192.168.1.0`
Router1	`network 192.168.0.0` `network 192.168.10.0`
Router2	`network 192.168.0.0` `network 192.168.20.0`

In Figura 11.3 si riporta la routing table presente in Router1 e accessibile, in Packet Tracer, attraverso lo strumento lente di ingrandimento. Se inizialmente la tabella di routing

Type	Network	Port	Next Hop IP	Metric
R	192.168.0.0/30	GigabitEthernet7/0	192.168.0.5	120/1
R	192.168.0.0/30	GigabitEthernet9/0	192.168.0.10	120/1
C	192.168.0.4/30	GigabitEthernet7/0	---	0/0
C	192.168.0.8/30	GigabitEthernet9/0	---	0/0
R	192.168.1.0/24	GigabitEthernet7/0	192.168.0.5	120/1
C	192.168.10.0/24	GigabitEthernet8/0	---	0/0
R	192.168.20.0/24	GigabitEthernet9/0	192.168.0.10	120/1

Figura 11.3: Tabella di instradamento di Router1 (vedi Esempio 11.4).

non mostra alcuna nuova informazione, ad eccezione delle reti che sono state configurate con il comando `network`, una volta terminata la dichiarazione delle rotte che ogni router conosce, in modalità Realtime si può osservare come questa venga aggiornata con tutte le informazioni disponibili congiuntamente. In particolare, nella tabella, il Type può essere S $\rightarrow$ rotta statica, R $\rightarrow$ RIP, O $\rightarrow$ OSPF. Nelle tabelle di routing viene indicata l'interfaccia per raggiungere una certa rete, il next hop IP, ovvero il gateway da utilizzare per andare verso la rete indicata, e per finire la metrica. Le metriche sono costituite da due numeri in forma x/y. Il primo valore x rappresenta la distanza amministrativa della rotta, definita sopra. Il secondo valore y rappresenta il costo del cammino secondo il protocollo di instradamento scelto. Ad es., per router direttamente connessi, il costo è 1. Quindi: Metric per reti direttamente connesse vale 0/0; per rotte statiche vale 1/0, per

Tabella 11.4: Configurazione del protocollo RIP version 2 sui router presenti in Figura 11.2.

Router0	Router1	Router2
`network 192.168.1.0`	`network 192.168.10.0`	`network 192.168.20.0`
`network 192.168.0.0`	`network 192.168.0.4`	`network 192.168.0.0`
`network 192.168.0.4`	`network 192.168.0.8`	`network 192.168.0.8`

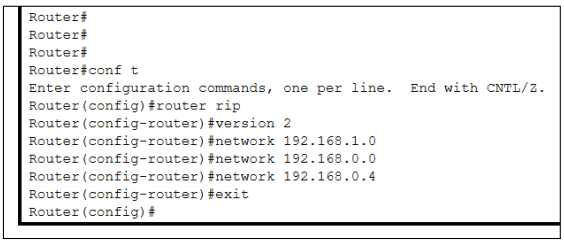
RIP dipende dal numero di hop e per OSPF primariamente dalla banda (es. 110/11 è più distante di 110/2).
Si noti infine la possibile ridondanza nelle entry presenti in tabella (vedi 192.168.0.0/30): possono infatti essere presenti due reti ripetute che non sono però uguali, differendo nel percorso. In particolare, con RIP vengono ripetute solo le reti che condividono uno stesso numero di hop nella colonna Metric; in caso contrario viene scelta la rotta con il minor numero di hop. Per test si può eseguire un *ping* da terminale sullo host che miri a raggiungere l'interfaccia di destinazione duplicata, e osservare come con tentativi ripetuti i pacchetti possono seguire l'una o l'altra rotta.
Si ricordi infine di configurare correttamente le passive-interface. △

Si faccia riferimento anche alla activity di Packet Tracer *RIP.pkt* nella sezione *"Materiale didattico"* del sito del corso.

RIP version 2 Nel caso in cui si voglia invece utilizzare l'indirizzamento *classless*, è necessario per prima cosa indicare che si vuole utilizzare la versione 2 di RIP, e quindi configurare il router in modo che annunci *tutte* le sottoreti a cui è connesso (pur senza netmask). La Tabella 11.4 mostra le reti da configurare per ognuno dei router in Figura 11.2. La Figura 11.4(a) mostra i comandi dati nella CLI per configurare Router0; in Figura 11.4(b) la routing table risultante dalla configurazione di tutti i router con RIPv2. Si evidenzia che *tutti i router della rete devono essere configurati con la medesima versione di RIP*, altrimenti non si parlano.

Con questa configurazione è anche possibile osservare il comportamento dinamico del protocollo di routing. A tal fine, eseguire il seguente test:

Esempio 11.4. Si esegua ping dallo host nella rete 192.168.1.0/24 allo host nella rete 192.168.20.0/24, e in modalità *Simulation* si osservi che i pacchetti ICMP percorrono la strada più corta passando per i due router che collegano le due reti.
Quindi si disabiliti (togliendo *on*) l'interfaccia di Router2 verso Router0, in modo da interrompere tale cammino (spegnendo l'interfaccia, la configurazione associata non è persa e può essere successivamente ripristinata), e in modalità *RealTime* si faccia avanzare



```
Router#
Router#
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 192.168.1.0
Router(config-router)#network 192.168.0.0
Router(config-router)#network 192.168.0.4
Router(config-router)#exit
Router(config)#
```

Routing Table for Router1

Type	Network	Port	Next Hop IP	Metric
C	192.168.0.0/30	GigabitEthernet8/0	---	0/0
C	192.168.0.4/30	GigabitEthernet7/0	---	0/0
R	192.168.0.8/30	GigabitEthernet7/0	192.168.0.6	120/1
R	192.168.0.8/30	GigabitEthernet8/0	192.168.0.2	120/1
C	192.168.1.0/24	GigabitEthernet9/0	---	0/0
R	192.168.10.0/24	GigabitEthernet7/0	192.168.0.6	120/1
R	192.168.20.0/24	GigabitEthernet8/0	192.168.0.2	120/1

FIGURA 11.4: (a) Comandi per la configurazione di Router0 con RIPv2. (b) Tabella di instradamento di Router1 risultante.

il tempo per permettere ai router di scoprire il cammino alternativo, più lungo. Quindi in modalità *Simulation* ripetere il ping precedente osservando che questa volta i pacchetti ICMP seguono il nuovo cammino. △

11.2 OSPF

Mostriamo ora come sia possibile configurare il protocollo OSPF. OSPF divide la rete in aree – come descritto nelle lezioni di Teoria – identificate come interi a 32 bit; tali identificatori per comodità possono essere espressi in dot notation benchè si sottolinea che non hanno nulla a che vedere con l'indirizzamento IP. Il motivo di tale suddivisione è individuare una struttura gerarchica grazie alla quale sia possibile ridurre i costi coinvolti dall'approccio Link State; in questo corso tuttavia considereremo solo reti piatte in cui tutti i router fanno parte della medesima area. All'interno di un dispositivo di rete possono girare più istanze di OSPF: una per ogni area. Per configurare OSPF è necessario usare la CLI; in modalità *configurazione* i comandi da seguire sono, nell'ordine, i seguenti:

1. `router ospf <process_id>`, dove il valore `process_id` è l'identificativo numerico – compreso tra 1 e 65535 – che serve per differenziare diverse istanze di OSPF in esecuzione sullo stesso router (ad esempio 1). Il prompt passa in modalità config-router#.

2. `area <area_id> stub`, dove `area_id` è l'identificativo numerico dell'area (ad esempio 1), mentre `stub` è un tipo di area che non riceve notifiche/annunci da reti al di fuori dell'organizzazione, ovvero non è usata come area di transito per l'instradamento di pacchetti non generati/destinati da/a host nell'area.

3. `network <IP_address> <wildcard_mask> area <area_id>`. Come in RIP, il comando `network` permette di configurare le reti connesse, ovvero annunciarle

nei pacchetti – in questo caso di Link State update – inviati agli altri router. Siccome però OSPF è classless, oltre a specificare il subnet address (`IP_address`) è necessario specificare la subnet fornendo la rispettiva *wildcard mask*, ovvero la negazione della subnet mask: ad esempio una /30 ha wildcard mask 0.0.0.3; una /24 0.0.0.255 (si veda anche Sez.5.2). Per finire, è necessario specificare l'area in cui l'istanza corrente di OSPF andrà ad operare, fornendone l'identificativo (`area_id`). Ad esempio, in riferimento alla Tabella 11.3, il comando `network 192.168.1.0 0.0.0.3 area 1` permette di configurare una delle reti note per Router0.

Valgono le stesse regole viste in RIP per selezionare su quali interfacce *non* si vogliono inviare notifiche nell'istanza corrente del protocollo di routing, settando opportunamente le passive-interface, così da evitare di invadere le LAN con pacchetti di controllo per i router che sono ignorati dagli host.

Esempio 11.5. Riprendendo la topologia configurata nell'Esempio 11.2, si configurino le reti note ai router presenti nella topologia, utilizzando i comandi introdotti sopra. Si assume una rete piatta e non di transito per altre reti. Si osservi il contenuto delle tabelle di routing popolate dal protocollo OSPF e se ne verifichi la completezza (ovvero la presenza di tutte le reti connesse e di tutte le reti non direttamente raggiungibili. Questo è un buon test anche in fase di esame per verificare di aver configurato tutto correttamente, o per individuare quali configurazioni siano state dimenticate). Per test si può eseguire il seguente esperimento:

1. porre a `shutdown` l'interfaccia tra Router0 e Router2 (l'interfaccia è spenta ma la configurazione non è persa)
2. eseguire con simulazione passo-passo un ping dallo host in rete 192.168.1.0/24 allo host in rete 192.168.20.0/24 e osservare come il pacchetto esegue il cammino di 2 hop tra Router0 e Router2 (può essere necessario eseguire prima il ping in modalità *Realtime* per permettere alle tabelle di popolarsi)
3. ripristinare (porre a `on`) l'interfaccia tra Router0 e Router2
4. eseguire un ping tra gli stessi due host ed osservare come OSPF impari il cammino migliore di un solo hop tra Router0 e Router2.

Si ricordi infine di configurare correttamente le passive-interface. △

Si faccia riferimento anche alla activity di Packet Tracer *OSPF.pkt* nella sezione *"Materiale didattico"* del sito del corso.

Si sottolinea infine che uno dei valori aggiunti nell'utilizzo di OSPF è che tiene traccia della capacità del link che connette due router e non si limita a considerare il numero di hop, come invece accade in RIP.

12 Configurazione Servizi

Già nella Sez.10 si è descritto come configurare il servizio DHCP. Si propongono in questa sezione ulteriori dettagli riguardo alla configurazione e fruizione di servizi di livello applicazione, limitatamente a quanto utilizzato nel corso.

Web server: a lato **server** il servizio HTTP è configurato da interfaccia grafica. Cliccando sul server in oggetto, e selezionando il tab Services e quindi il bottone HTTP, si accede alla finestra di configurazione in cui è possibile attivare (On) i protocolli HTTP e HTTTPS, ed editare i file (es. *index.hml*) della pagina web di test resa dal server.
Per il lato **client**, cliccando sul PC desiderato e selezionando il tab Desktop, si può scegliere il servizio *Web Browser* da cui inserire l'indirizzo IP del server web che si vuole testare.
Ricordare che, mentre in modalità RealTime la risposta del server è immediatamente mostrata, in modalità Simulazione è successivamente necessario iconizzare la finestra del PC e far procedere passo-passo la simulazione.

Command prompt: cliccando sullo end system desiderato e selezionando il tab Desktop, si può scegliere il servizio *Command prompt* da cui inserire i comandi di shell desiderati. Questo servizio può essere utile soprattutto per testare via *ping* la raggiungibilità di specifiche interfacce di router in altre reti, cosa impossibile da fare generando da interfaccia grafica un ping sul router stesso (la risposta sarebbe generata dall'interfaccia più vicina alla sorgente del ping, che non è necessariamente quella che si vuole testare).

13 Access Control List (ACL)

Le Access Control List (ACL) sono usate per filtrare (permettere o negare selettivamente) traffico di rete su apparati di livello 3 (router), controllando se i pacchetti possono essere inoltrati o se devono essere bloccati in ingresso o in uscita per una determinata interfaccia. Il router esamina dunque ogni pacchetto per determinare se inoltrarlo o bloccarlo sulla base dei criteri specificati nella ACL applicata all'interfaccia per cui questi transitano. Nel seguito si fornirà solo una breve panoramica delle ACL; si consiglia al lettore di riferirsi a materiale disponibile sul web per ulteriori approfondimenti.

Regola 13.1. *Si consideri la rete in Figura 13.1, in cui si vogliono imporre regole sul Router0 per filtrare il traffico rispetto alla "*rete interna*".*
Il traffico entrante nella rete interna è quello che arriva sul Router0 in **ingresso** *dalle interfacce FastEthernet 1/0 e FastEthernet 2/0, e in* **uscita** *dall'interfaccia FastEthernet*

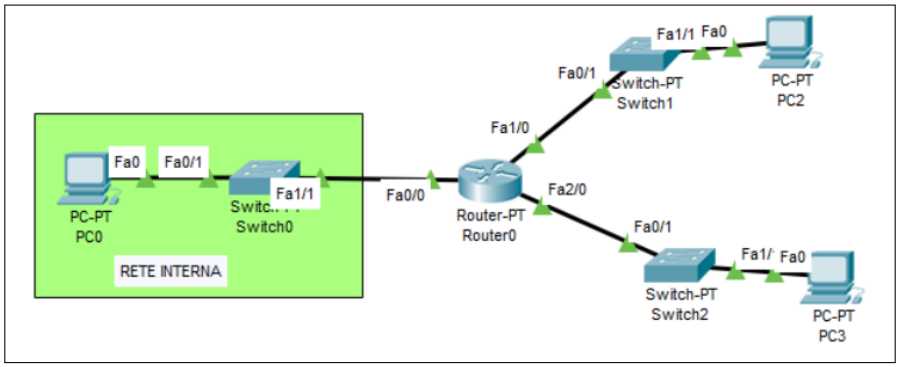


FIGURA 13.1: Topologia di riferimento per direzioni traffico.

0/0.
Il traffico uscente dalla rete interna è quello che arriva sul Router0 in **ingresso** *dall'interfaccia FastEthernet 0/0, e in* **uscita** *dalle interfacce FastEthernet 1/0 e FastEthernet 2/0.*

Esistono due tipi differenti di ACL:

1. *ACL standard*. Si tratta del primo tipo di ACL definito in Packet Tracer (attualmente deprecato). Il traffico può essere filtrato sulla base solamente dell'indirizzo IP sorgente dei pacchetti IP. Il numero identificativo di tali ACL deve essere compreso tra 1 e 99 o tra 1300 e 1999.

2. *ACL extended*. Si tratta di ACL più complesse che permettono il filtraggio del traffico IP sulla base di una combinazione di diversi criteri: indirizzo IP sorgente, indirizzo IP destinazione, porte TCP o UDP, protocollo, ... Il numero identificativo di tali ACL deve essere compreso tra 100 e 199 o tra 2000 e 2699.

Essendo più flessibili e potenti, nel seguito si farà riferimento esplicito alle sole ACL extended.[16] I passi di base per configurare una ACL sono due:

1. **creazione di una nuova ACL** tramite il comando `access-list` (accessibile in modalità *configurazione globale*) che permette di definirne le politiche (criteri di selezione dei pacchetti e decisione da intraprendere: permettere o negare);

2. **applicazione di una ACL** esistente sull'interfaccia di interesse, tramite il comando `ip access-group` (accessibile in modalità *configurazione interfaccia*).

Per quanto riguarda la prima fase, si riporta di seguito la sintassi del comando `access-list`:

[16] Nel seguito per riferirsi alle ACL extended si utilizzerà semplicemente l'acronimo ACL.

```
access-list access-list-number permit | deny protocol source-wildcard
source [operator port] destination-wildcard destination [operator
port] [established] [log]
```

una cui sommaria descrizione dei parametri è fornita in Tabella 13.1.

Tabella 13.1: Comando `access-list` usato per definire una ACL e descrizione sommaria dei suoi argomenti.

Comando / Parametri	**Descrizione**
`access-list`	Comando principale
`access-list-number`	Identifica la lista usando un numero compreso tra 100 e 199 o tra 2000 e 2699 (solo extended)
`permit | deny`	Indica se questa entry permette o blocca l'indirizzo specificato; essendo le ACL *first match* – cioè si applica la prima linea incontrata nella ACL per cui il pacchetto in esame soddisfa i parametri – è sempre più opportuno premettere i `permit` ai vari `deny`
`protocol`	IP, TCP, UDP, ICMP, OSPF, o EIGRP
`source and destination`	Identifica indirizzi IP sorgente e destinazione
`source-wildcard and destination-wildcard`	`any` identifica *ogni* sorgente o destinazione rispettivamente; `host` identifica un singolo host invece di una rete
`operator port`	L'operatore può essere `lt` (less than), `gt` (greater than), `eq` (equal to), `neq` (not equal to), oppure `range` (intervallo di numeri di porta). Il numero di porta può riferirsi o alla porta sorgente o alla porta destinazione, in funzione del punto in cui il parametro `port number` è configurato nella ACL. In alternativa al numero di porta, è possibile utilizzare il nome del protocollo di riferimento scelto tra alcuni servizi standard predefiniti (`telnet`, `ftp`, `www`, `smtp`, `pop3`, ...)
`established`	Solo per *inbound TCP connection*. Permette al traffico TCP di passare se il pacchetto è una risposta alla *outbound-initiated TCP session*. Questo tipo di traffico ha il bit ack settato
`log`	Invia un messaggio di log alla console

In sostanza dopo essere entrati in modalità configurazione (config terminal), digitando il comando `access-list id`, con `id` numero compreso tra 100 e 199, è possibile: aggiungere commenti (`remark`), permettere (`permit`) o negare (`deny`) il transito di pacchetti. Fa seguito l'identificativo del protocollo che deve essere filtrato (TCP, UDP, ICMP, ..., oppure IP per riferirsi ad ogni protocollo Internet). Segue un descrittore degli host sorgenti da monitorare: any (qualsiasi host sorgente), oppure host (un singolo host), o ancora fornendo direttamente l'indirizzo di rete e *wildcard mask* di una intera sottorete.

Attenzione! *Di default le access list bloccano il passaggio di tutti i pacchetti che non hanno fatto match con qualche linea* `permit` *della ACL; in sostanza è come se alla fine di*

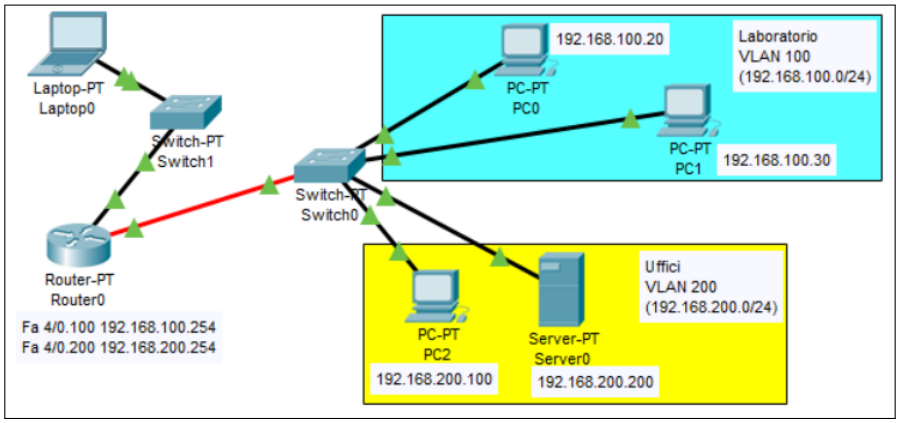


FIGURA 13.2: Topologia con 2 VLAN per introdurre il concetto di ACL.

ogni ACL vi fosse una riga implicita `deny ip any any`*. Una conseguenza è che potrebbe non essere più permesso neppure il traffico per configurare il router da remoto, ed è necessario configurarlo collegandosi ad una sua porta seriale.*

Esempio 13.1. Consideriamo la topologia in Figura 13.2. Supponiamo che a Server-PT Server0 sia assegnato l'indirizzo IP 192.168.200.200 e supponiamo di voler permettere l'accesso al server web ma non ICMP. I comandi da digitare sono:

```
access-list 110 permit TCP any host 192.168.200.200 eq 80
access-list 110 deny ICMP any any
```

dove il primo consente di far passare traffico TCP proveniente da any diretto allo host 192.168.200.200 sulla porta 80; il secondo vieta ICMP (ping) da qualunque sorgente a qualunque destinazione.

Per quanto riguarda invece la seconda fase, ovvero l'attivazione di una ACL su una specifica interfaccia, è necessario specificare a quale interfaccia si vuole che venga applicata. Nel caso in esame o la si applica all'interfaccia che contiene lo host 192.168.200.200 o all'altra. Sebbene per quanto concerne la configurazione del router i due approcci siano interscambiabili, a livello operativo bisogna prestare attenzione ad aspetti funzionali. In altre parole, se si applicasse la ACL all'altra interfaccia e in futuro dovessero essere aggiunte nuove VLAN, queste non si troverebbero applicata la regola; in questo caso è dunque di indubbio vantaggio l'applicazione della ACL alla interfaccia del router che contiene lo host 192.168.200.200. I comandi per ottenere questo risultato sono:

```
interface fastEthernet 4/0.200
ip access-group 110 out
```

Una possibile fonte di confusione è la scelta del parametro `in` o `out` nell'ultima istruzione, ovvero nell'applicazione del filtro ai pacchetti che entrano o che escono dall'interfaccia. In sostanza cambia la direzione in cui il traffico viene filtrato: siccome nel caso in esame si sta filtrando il traffico diretto verso la rete che contiene lo host 192.168.200.200, la scelta corretta è `out`, visto che si vuole filtrare il traffico che esce dall'interfaccia e va verso la rete che contiene lo host. Attenzione, dunque, alla prospettiva con cui considerare la direzione in entrata o in uscita del traffico: prospettiva che è quella del router su cui il filtro viene attivato. In altre parole, è necessario mettersi nei panni del router: il traffico da filtrare è sì in ingresso alla rete che contiene lo host, ma per il router è diretto alla rete che contiene lo host e dunque è in uscita. (si veda la **Regola 13.1**).

Si può testare la configurazione come segue: si assegni al Laptop0 un indirizzo dalla rete 10.0.0.0/24 (es.10.0.0.1), e all'interfaccia FastEthernet 0/0 di Router0 un indirizzo in accordo (es. 10.0.0.254). Si verifichi che è possibile eseguire ping tra Laptop0 ed entrambi gli host nella VLAN Laboratorio. Al contrario, ping fallisce tra Laptop0 e un qualsiasi host nella VLAN Uffici. Aprendo il servizio Web Browser su Laptop0, è possibile raggiungere il sito web con indirizzo 192.168.200.200. Si faccia riferimento anche alla Activity ACL_1.pkt sul sito del corso. △

È possibile vedere quali ACL sono configurate sull'apparato con il comando `show access-lists` oppure `show ip access-lists`; si noterà che le linee di configurazione di una ACL sono numerate, di default a intervalli di 10. Questo consente in alcuni casi di inserire a posteriori ulteriori linee in una ACL nel corretto ordine in cui devono essere prese in considerazione.
A tal fine, è possibile configurare delle **named extended access list**: dalla modalità *configure*, introdurre il comando

```
ip access-list extended <access-list-number> | <ACLname>
```

(dove `ACLname` è una qualunque stringa che prende il posto dell'identificativo numerico) con cui si entra in sub-mode config-ext-nacl. I comandi per la configurazione sono i medesimi visti in precedenza. Vi è però in aggiunta l'interessante possibilità di *inserire in tempi successivi linee intermedie* tra quelle stabilite in una configurazione iniziale, semplicemente premettendo ai comandi di configurazione il numero di posizione in cui si vuole che quella linea compaia. Ad esempio, si può inserire una linea in posizione 15 con un comando tipo: `15 permit udp any any`.

Esempio 13.2. Si consideri ora la topologia illustrata in Figura 13.3. Si supponga che la intranet aziendale disponga di due siti web: uno accessibile a tutto il mondo (Server0), l'altro riservato ad uso interno privato (Server1). Si supponga inoltre che nessuno host esterno all'azienda abbia accesso alla intranet. Il router dispone di due interfacce: FastEthernet 1/0 (interna) con indirizzo IP 192.168.0.254 e FastEthernet 0/0 (esterna) con indirizzo IP 10.0.0.254.

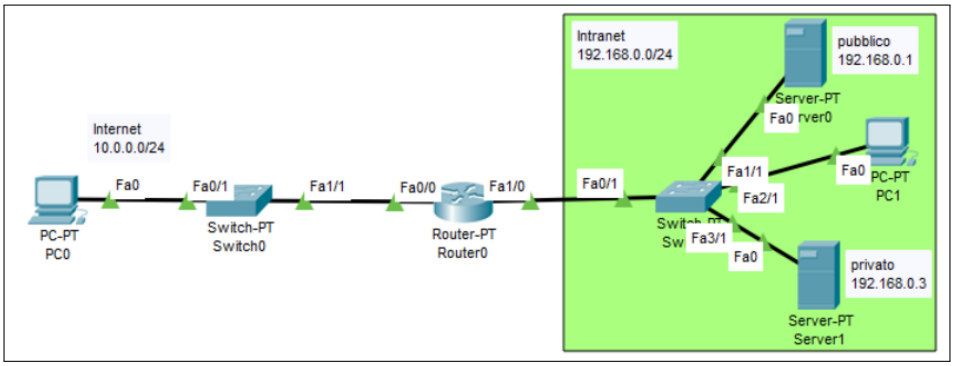


FIGURA 13.3: Topologia con accesso limitato alla intranet (vedi Esempio 13.2).

La prima fase consta nella definizione del filtro. Anzitutto la scelta ricade su una ACL (named) extended, dovendo filtrare solo il traffico web e non solo sull'indirizzo IP sorgente.

```
ip access-list extended 100
permit TCP any host 192.168.0.1 eq www
deny IP any any
```

In sostanza, nella ACL con id 100 si permette il traffico TCP diretto da qualunque sorgente allo host 192.168.0.1 sulla porta 80. L'ultima istruzione serve per vietare l'accesso di tutto il traffico Internet: adottando le ACL una politica *first match*, lasciando in ultima posizione il comando `deny` non si rischia di bloccare il traffico TCP diretto a Server0.

La seconda fase consiste nell'applicazione della ACL alla giusta interfaccia. È necessario anzitutto entrare in modalità configure interface per poi attivare l'ACL:

```
interface FastEthernet 0/0
ip access-group 100 in
```

Si noti come sia possibile impartire questi comandi anche all'interfaccia interna. In tal caso la modalità di accesso non sarebbe `in`, ma `out`, per i motivi spiegati in precedenza. Si verifichi, usando il servizio Web Browser su PC0, che è possibile raggiungere il web server pubblico, ma il tentativo di raggiungere il web server privato fallisce con timeout. Si faccia riferimento anche alla Activity ACL_2.pkt sul sito del corso. △

Esempio 13.3. Si consideri infine la topologia rappresentata in Figura 13.4 che rappresenta una variante dello scenario adottato nell'esempio precedente. Si verifichi che Server2, attraverso il servizio Web Browser, è correttamente in grado di accedere al Server0 pubblico, e non al Server1 privato. Si noti inoltre come Server2 non sia in grado di fare ping verso il Router0 (interfaccia FastEthernet 0/0), poichè la ACL definita

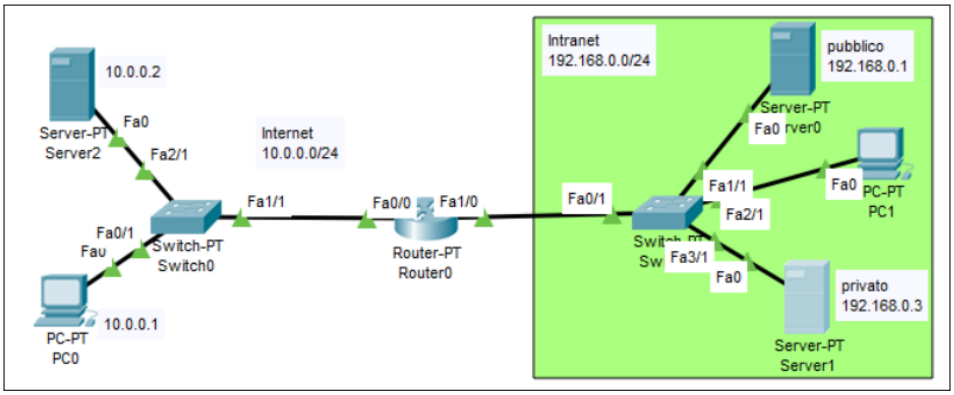


FIGURA 13.4: Topologia con accesso limitato alla intranet (vedi Esempio 13.3).

nell'Esempio 13.2 è applicata all'interfaccia esterna.
Simmetricamente si verifica un problema: nessuno dei tre end system nella Intranet è in grado di accedere via servizio Web Browser al Server2 *pubblico*. Per capirne la ragione, si passi in modalità *Simulazione*, abilitando i filtri per i protocolli ARP e ICMP (sotto il tab IPv4) e HTTP e TCP (sotto il tab Misc). Eseguendo la simulazione passo-passo si osserva la seguente evoluzione: Server0 inizia una procedura di three-way handshake per connettersi a Server2; il segmento TCP della seconda fase della procedura torna al Router0, e qui viene eliminato perchè destinato al processo che esegue il browser, associato ad una porta differente dalla porta 80; la regola della ACL che fa match è quindi `deny IP any any`.

Per risolvere il problema entra in gioco TCP e, in particolare, le caratteristiche del protocollo di *three-way-handshake* visto nelle lezioni di Teoria. In sostanza, volendo permettere a tutte le richieste di servizi esterni (basati su TCP) generate nella Intranet di ottenere la relativa risposta in ingresso alla Intranet – senza specificare nel dettaglio tutte le porte destinatario – è sufficiente osservare che questo genere di PDU ha il flag ack dello header del segmento TCP settato a 1 a partire dalla seconda fase del three-way-handshake; ovvero, a partire dal primo messaggio ri-entrante nella Intranet. Operativamente, in aggiunta al filtro già applicato, bisogna inserire nella ACL il comando:

```
15 permit TCP any any established
```

dove `established` significa appunto "segmenti con ack pari a 1" ovvero a connessione TCP stabilita. Questo permette la ricezione di risposta da ogni servizio basato su TCP. Il numero di rga `15` permette l'inerimento prima della regola finale che blocca qualunque pacchetto IP non faccia match con le regole precedenti.
Si faccia riferimento anche alla Activity ACL_3.pkt sul sito del corso. △

14 Network Address Translation (NAT)

Sebbene svariati siano i compiti del NAT (si faccia riferimento alle lezioni di Teoria), in queste dispense ci si limiterà a nascondere gli indirizzi interni quando alcuni pacchetti escono dalla Intranet. Si usi a tal fine la stessa topologia illustrata in Figura 13.4, togliendo le ACL correntemente attive (è sufficiente a tal fine togliere ip access-group nell'interfaccia, con il comando `no ip access-group`).

Nella configurazione via CLI dell'interfaccia interna (quindi, da prompt di modalità privilegiata e mode `config`, entrando nel sub-mode `interface FastEthernet 1/0`) si digita:

```
ip nat inside
```

analogamente si configura l'interfaccia esterna (FastEthernet 0/0) con il comando:

```
ip nat outside
```

A questo punto, tornando in modalità configurazione, digitando il comando `ip nat inside source ?` si osservano due possibili argomenti:

- `static`: richiede una corrispondenza (indirizzo interno - indirizzo esterno);
- `list`: richiede una ACL che specifichi quali indirizzi siano idonei ad essere tradotti.

In modo `config` si dovrà dunque prima creare una ACL che permetta a tutti gli indirizzi coinvolti di essere tradotti (la ACL *non* deve essere applicata ad alcuna interfaccia):

```
access-list 110 permit ip any any
```

In alternativa, si può mascherare anche solo un subset di indirizzi per lasciare dei server con indirizzo pubblico. Ad esempio, se all'interno fosse usata la rete 10.0.0.0 con netmask 255.0.0.0, si potrebbe usare il comando `access-list 110 permit ip 10.0.0.0 0.0.0.255 any` per dire che il NAT maschera solo gli indirizzi da 10.0.0.1 a 10.0.0.254. Tornando al NAT, con il comando da modo `config`:

```
ip nat inside source list 110 ?
```

viene proposta la scelta di una politica di traduzione, che può essere:

- `pool`: un pool di indirizzi tra cui scegliere un indirizzo pubblico per la traduzione dell'indirizzo privato;
- `interface`: l'indirizzo IP dell'interfaccia in questione.

Il comando completo:

```
ip nat inside source list 110 interface fastEthernet 0/0
```

si legge: “quando si fa NAT degli indirizzi interni si usi come criterio degli indirizzi da tradurre quello indicato nella ACL 110 e si traducano gli indirizzi usando l’indirizzo IP dell’interfaccia FastEthernet 0/0”.

Esempio 14.1. Per testare la configurazione fatta sul router, si esegua un ping da uno host interno alla Intranet su un end system in Internet, mediante una simulazione passo-passo, e si osservi che un pacchetto ICMP entra nel router con indirizzo sorgente nella rete 192.168.0.0/24 (Inbound PDU Details) e ne esce con indirizzo sorgente quello dell’interfaccia FastEthernet 0/0 del router (Outbound PDU Details). Similmente, la risposta al ritorno reca come indirizzo destinazione – all’ingresso nel router – quello dell’interfaccia FastEthernet 0/0 del router, e all’uscita dal router l’indirizzo del generatore del ping nella rete 192.168.0.0/24.

Ovviamente nell’esempio non è più possibile ad uno host in Internet accedere neppure al web server pubblico della Intranet.
Si faccia riferimento anche alla Activity NAT_1.pkt sul sito del corso. △

È anche possibile fare in modo che il router NAT accetti richieste di un servizio standard al proprio indirizzo esterno e sulla propria porta definita per quel servizio, e provveda poi a re-inoltrare la richiesta all’opportuno server interno con indirizzo non pubblico (ovvero mascherato da NAT).

Esempio 14.2. Si configuri su uno dei due server interni un server HTTP (Sez.12). Sul router, da CLI in modo `config` si inserisca la configurazione

`ip nat inside source static tcp` $IP_{interno}$ $porta_{interna}$ $IP_{esterno}$ $porta_{esterna}$

sostituendo opportunamente i 4 parametri: $\text{IP}_{interno}$ $\text{porta}_{interna}$ con indirizzo del server interno (nell’esempio, 192.168.0.1) e porta 80, e $\text{IP}_{esterno}$ $\text{porta}_{esterna}$ con indirizzo dell’interfaccia esterna del router (nell’esempio, 10.0.0.254) e porta 80. La riga di configurazione si legge: “quando arriva una richiesta all’indirizzo esterno e porta indicata, si traduca nell’indirizzo interno e porta indicata”. Questa modalità di configurazione si chiama *port forwarding*. Si faccia partire una richiesta HTTP dal PC esterno per la porta 80 del router e, come prima, attraverso una simulazione passo-passo si verifichi il corretto re-indirizzamento dei messaggi di richiesta e risposta quando attraversano il router.
Si faccia riferimento anche alla Activity NAT_2.pkt sul sito del corso. △

Esistono altre modalità di utilizzo del NAT; ad esempio si potrebbero avere nella intranet diversi server web con indirizzi privati che si vuole rendere pubblici. Sul router si dovrebbe definire un mapping statico tra porta e indirizzo privato. Tali argomentazioni esulano dagli obiettivi del presente corso.