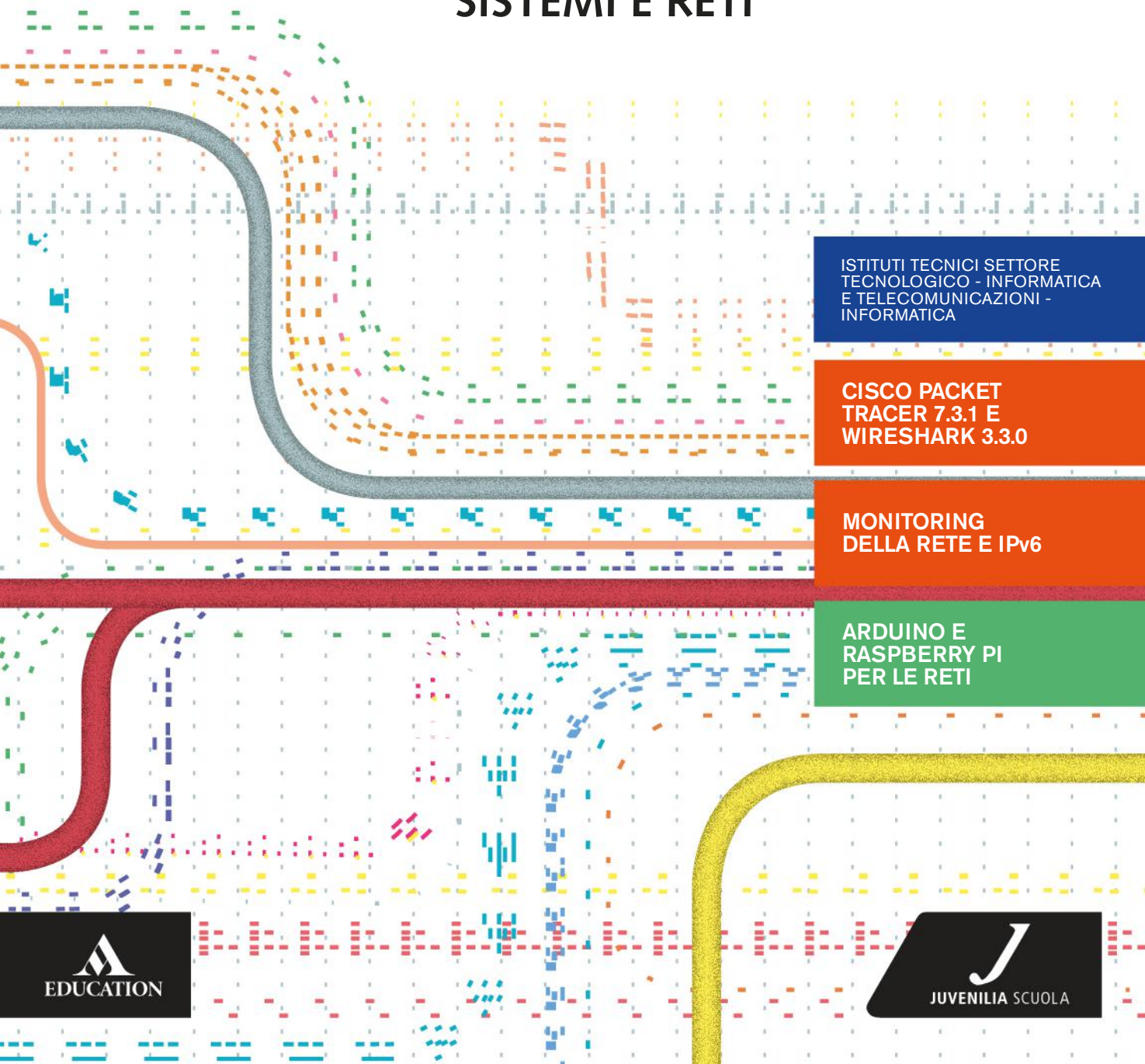
Terza edizione
4° anno

DDI+ Didattica Digitale Integrata Plus

Elena Baldino, Renato Rondano,
Antonio Spano, Cesare Iacobelli

Internetworking

SISTEMI E RETI



ISTITUTI TECNICI SETTORE TECNOLOGICO - INFORMATICA E TELECOMUNICAZIONI - INFORMATICA

CISCO PACKET TRACER 7.3.1 E WIRESHARK 3.3.0

MONITORING DELLA RETE E IPv6

ARDUINO E RASPBERRY PI PER LE RETI

EDUCATION

JUVENILIA SCUOLA

E. Baldino, R. Rondano, A. Spano, C. Iacobelli

INTERNET WORKING

SISTEMI E RETI

JUVENILIA SCUOLA

Il Sistema Qualità di Mondadori Education S.p.A. è certificato da Bureau Veritas Italia S.p.A. secondo la Norma UNI EN ISO 9001.

www.mondadorieducation.it

Redazione	GEM Milano
Progetto grafico	Angela Garignani
Impaginazione	GEM Milano
Proofreading	GEM Milano
Art direction del progetto grafico della copertina	46xy studio
Realizzazione della copertina	MOST - Themost.it Milano
Disegni	duDAT S.r.l. - Bologna, Edistudio Milano
Ricerca iconografica	duDAT S.r.l. - Bologna

Si ringrazia la Prof.ssa Sara Riccardo per la rilettura critica del testo.

Contenuti digitali

Progettazione	Fabio Ferri, Vincenzo Belluomo
Redazione	duDAT S.r.l. - Bologna (lezioni, esercizi)
Realizzazione	duDAT S.r.l. - Bologna (lezioni, mappe ed esercizi), Studio Frigo (audio)

PRESENTAZIONE

PREFAZIONE ALLA TERZA EDIZIONE

La terza edizione di *Internetworking – Sistemi e Reti* si presenta rinnovata nella sua articolazione su **tre volumi** per la classe terza, quarta e quinta. Nel primo e secondo volume i contenuti sono stati riorganizzati con l'obiettivo di proporre gli argomenti cardine previsti per quell'anno, inserendo nelle Unità finali i contenuti che possono essere affrontati anche nell'anno successivo.

L'opera offre numerose risorse testuali, laboratoriali e multimediali, che si integrano per realizzare un percorso ben articolato nel mondo complesso delle reti e dei sistemi. Alcuni argomenti fondamentali sono **introdotti e poi ripresi più volte nel corso dei tre anni**, per fornire agli studenti, al termine del percorso triennale, una solida preparazione teorica e pratica per affrontare l'Esame di Stato e il mondo del lavoro.

Sono stati rivisti e potenziati molti degli apparati didattici che costellano il progetto: mappe concettuali e sintesi di fine unità per una **didattica inclusiva**, attività esercitative a fine unità, attività progettuali per lo **sviluppo delle competenze**, laboratori ed esempi svolti nella teoria, schede per la didattica CLIL. La presentazione in PowerPoint dell'Unità, la **flipped classroom**, la **mappa concettuale** modificabile e le **risorse audio e multimediali** accessibili anche da smartphone tramite QR Code sono un importante corredo a disposizione del docente per un percorso strutturato di **Didattica Digitale Integrata**.

Internetworking affronta le problematiche del mondo delle reti e dei sistemi basando la trattazione sulle indispensabili **basi teoriche** e sullo **stato dell'arte** delle tecnologie presenti sul mercato e future.

L'obiettivo che ci poniamo è dunque non solo di fornire le conoscenze e le competenze utili attualmente, ma di permettere al futuro perito informatico di **integrarle con quelle emergenti**, offrendo strumenti che torneranno utili nella sua professione. Tra questi si evidenziano: la capacità di lavorare con gli standard internazionali, la comprensione di testi in lingua inglese, l'abilità nell'uso degli strumenti di analisi e di simulazione.

Per orientarsi in questo mondo complesso e offrire una chiave di lettura del contenuto dei volumi dell'opera, all'inizio di ogni volume è stata inserita una **mappa generale del volume** stesso, che presenta in modo organico le tematiche trattate e il modo in cui si correlano le une alle altre. Questo strumento permette al docente di personalizzare i diversi momenti didattici sulle specificità della propria classe, avendo sempre presente la **visione d'insieme**, e allo studente di capire come gli argomenti trattati nel corso dell'anno non siano isolati gli uni dagli altri ma **strettamente correlati tra loro**, acquisendo così maggiore consapevolezza del suo percorso di apprendimento.

L'intero volume del quarto anno è dedicato alla trattazione dell'architettura di rete TCP/IP. Vengono scalati i 4 livelli che la costituiscono e viene affrontata nel dettaglio l'intera suite di protocolli, configurandone i relativi servizi sul simulatore Cisco Packet Tracer.

ELEMENTI DI NOVITÀ NEL VOLUME PER IL 4° ANNO

- La prima Unità è propedeutica a tutte quelle successive, in quanto presenta il modello OSI e l'architettura TCP/IP, gli enti internazionali che definiscono gli standard per le reti, il simulatore Cisco Packet Tracer 7.3.1 e l'analizzatore di protocolli Wireshark 3.x.
- Le esercitazioni con Packet Tracer sono state completamente rifatte nella nuova versione, che prevede una nuova interfaccia grafica, e aumentate di numero. Sono state introdotte nuove esercitazioni con Wireshark.
- È stata introdotta una nuova Unità in cui si approfondiscono i protocolli IPv6 e ICMPv6, portando così a 3 le Unità dedicate al Network Layer, con un sostanziale incremento anche degli esercizi sull'indirizzamento IP.
- È stata ripresa dal volume del quinto anno della precedente edizione l'Unità dedicata a DHCP e DNS, con l'obiettivo di rendere così completa la trattazione dei protocolli e servizi dello stack TCP/IP nel quarto anno.
- L'Unità sull'Application Layer è stata aggiornata e un'intera lezione è stata dedicata alle applicazioni relative al trasporto della voce su IP (VoIP).
- Le esercitazioni che richiedono l'uso dei sistemi Windows e Linux sono state aggiornate alle release Windows 10 e Linux Ubuntu 20.04 LTS.
- L'ultima Unità è dedicata ad attività con Arduino e Raspberry Pi che prevedono l'uso della rete, con nuovi esercizi.

Gli Autori

INDICE

CONTENUTI DIGITALI INTEGRATIVI

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

FILE SORGENTI
Scarica i file

TEST
Svolgi il test interattivo

AUDIO
Ascolta le risposte

AUDIO
Ascolta la pronuncia del testo

LETTURA
Verifica la traduzione

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

FILE SORGENTI
Scarica i file

TEST
Svolgi il test interattivo

AUDIO
Ascolta le risposte

AUDIO
Ascolta la pronuncia del testo

LETTURA
Verifica la traduzione

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

CASE STUDY
Progettare una rete e assegnare indirizzi IP

CONTENUTI DIGITALI INTEGRATIVI

ESERCIZIO COMMENTATO
Il subnetting
FILE SORGENTI
Scarica i file
TEST
Svolgi il test interattivo
AUDIO
Ascolta le risposte
AUDIO
Ascolta la pronuncia del testo
LETTURA
Verifica la traduzione

PRESENTAZIONE
Guarda la presentazione dell'Unità
MAPPA MODIFICABILE
FILE SORGENTI
Scarica i file
TEST
Svolgi il test interattivo
AUDIO
Ascolta le risposte
AUDIO
Ascolta la pronuncia del testo
LETTURA
Verifica la traduzione

PRESENTAZIONE
Guarda la presentazione dell'Unità
MAPPA MODIFICABILE
CASE STUDY
Routing statico
FILE SORGENTI
Scarica i file
TEST
Svolgi il test interattivo
AUDIO
Ascolta le risposte
AUDIO
Ascolta la pronuncia del testo
LETTURA
Verifica la traduzione

CONTENUTI DIGITALI INTEGRATIVI

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

ESERCIZIO COMMENTATO
I segmenti TCP

TEST
Svolgi il test interattivo

AUDIO
Ascolta le risposte

AUDIO
Ascolta la pronuncia del testo

LETTURA
Verifica la traduzione

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

FILE SORGENTI
Scarica i file

TEST
Svolgi il test interattivo

AUDIO
Ascolta le risposte

AUDIO
Ascolta la pronuncia del testo

LETTURA
Verifica la traduzione

CONTENUTI DIGITALI INTEGRATIVI

PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

FILE SORGENTI
Scarica i file

ESERCIZIO COMMENTATO
Trasferimento di una cartella con FTP

TEST
Svolgi il test interattivo

AUDIO
Ascolta le risposte

AUDIO
Ascolta la pronuncia del testo

LETTURA
Verifica la traduzione

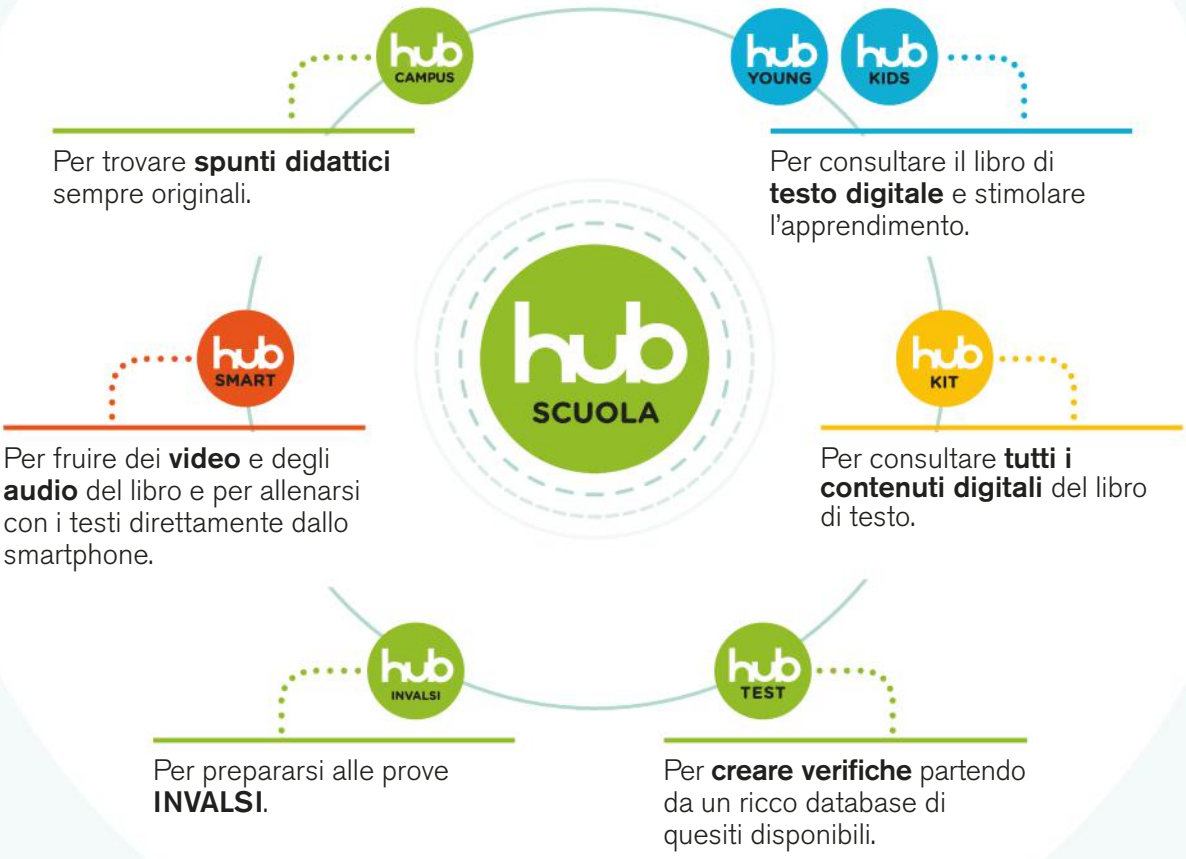
PRESENTAZIONE
Guarda la presentazione dell'Unità

MAPPA MODIFICABILE

FILE SORGENTI
Scarica i file

HUB Scuola: per una didattica digitalmente aumentata

HUB Scuola è la piattaforma che permette a studenti e docenti di **consultare il libro digitale, esplorare le risorse multimediali** integrate nel libro e **condividere i contenuti disponibili**.



Per accedere a hubscuola.it
Sei registrato? Usa le tue credenziali Mondadori Education e inizia a consultare i contenuti.
Non sei registrato? Clicca su **registrati** e compila il form.

Per Scaricare HUB Young o HUB Kids
L'App è scaricabile direttamente da **hubscuola.it** oppure dai principali store on line.
Lancia l'App, effettua il login e nella libreria troverai tutti i libri che hai attivato.

Link utili
- La piattaforma per la didattica digitale: **hubscuola.it**
- Il sito web con le nostre novità: **mondadorieducation.it**
- L'assistenza per tutti: **assistenza.hubscuola.it**

DDI+

Didattica Digitale Integrata Plus

Scopri i vantaggi della DDI Plus di Mondadori Education

È Integrata perché

grazie ad **HUB Scuola** in un unico ambiente potrai trovare

- i vantaggi del **libro di testo**
- **contenuti digitali** complementari
- **servizi specifici** per la progettazione didattica
- **lezioni digitali** per tutte le aree disciplinari

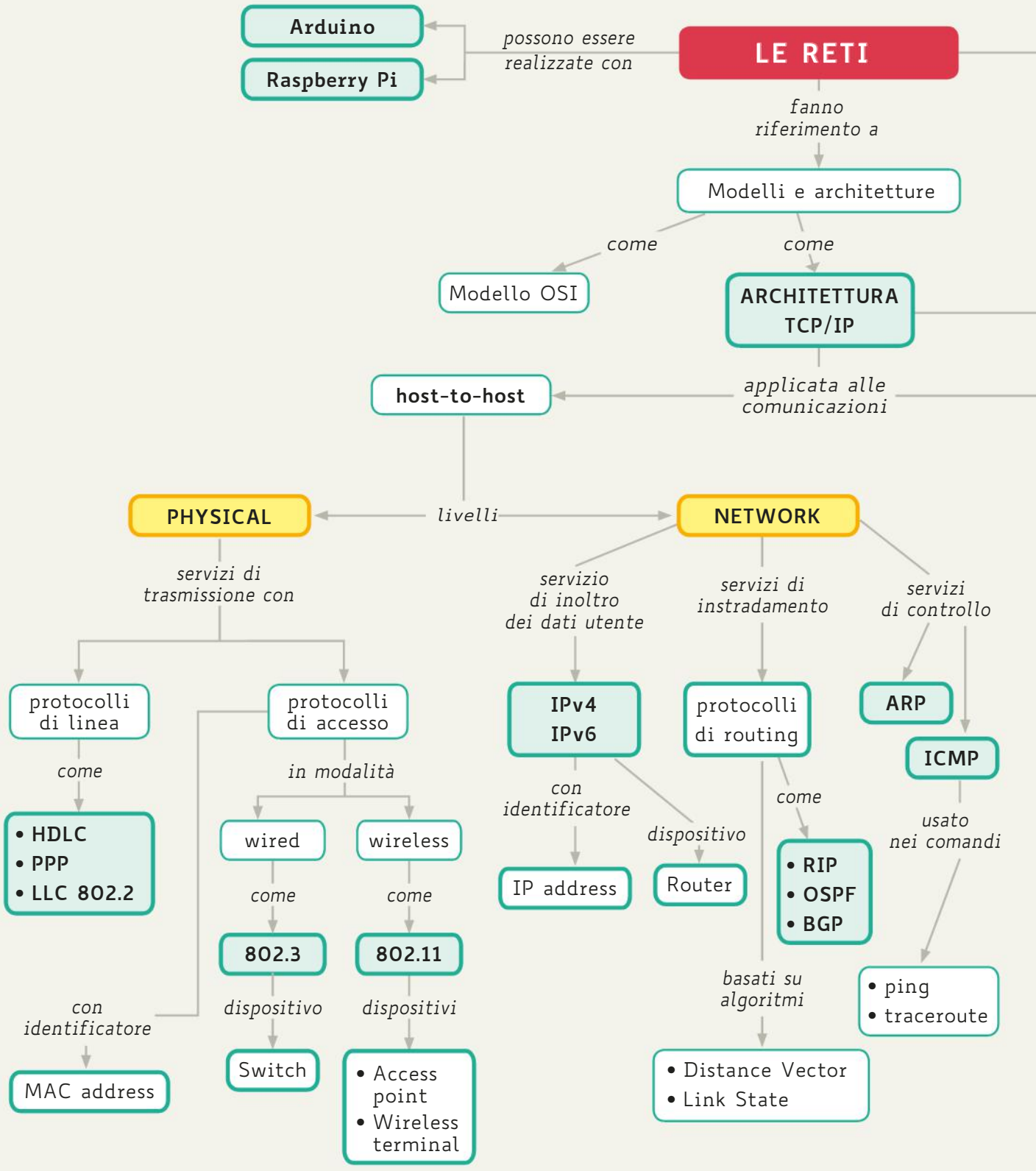
Scopri di più su
mondadorieducation.it

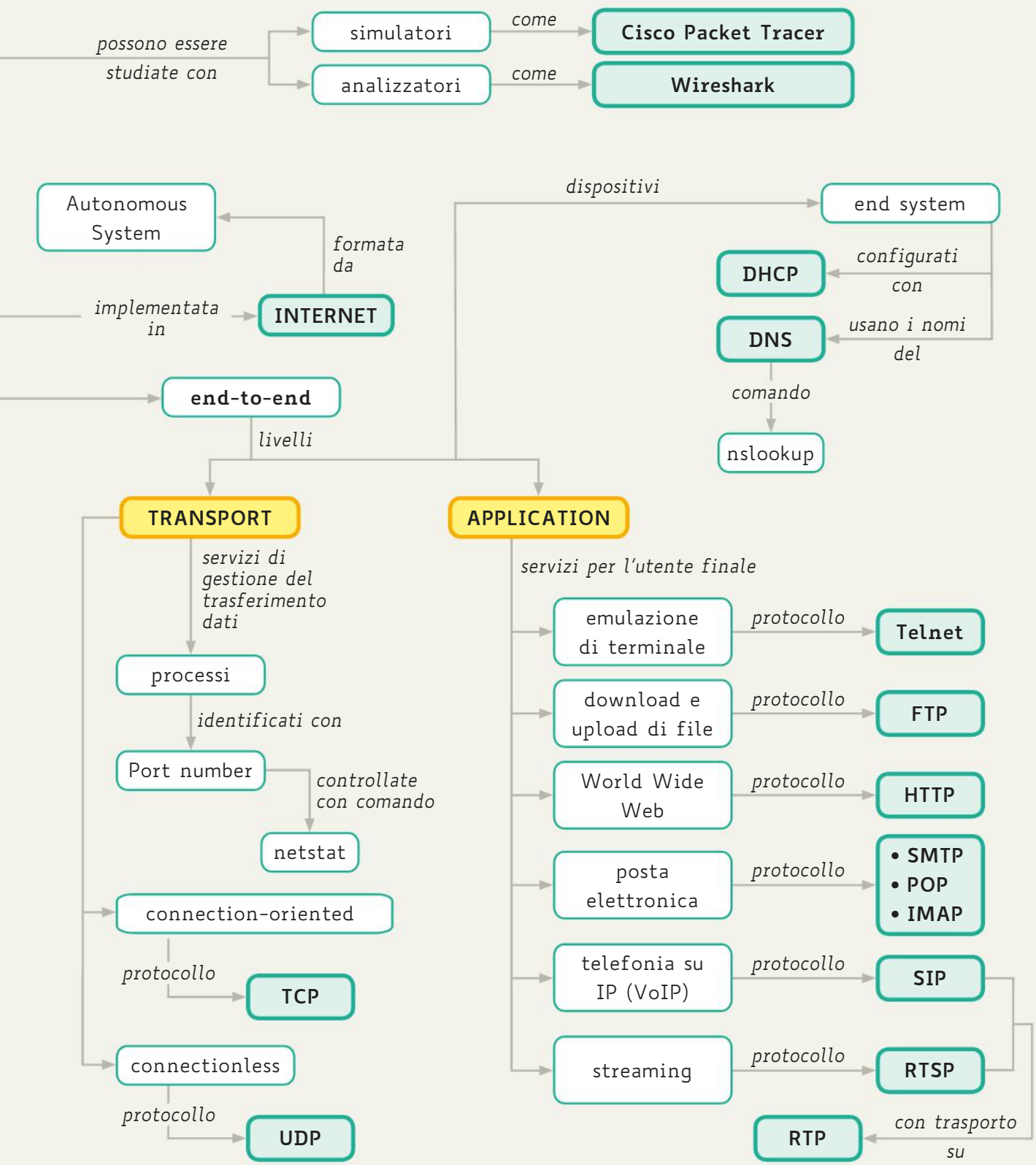
È Plus perché garantisce

- più **innovazione** grazie a **nuove metodologie** didattiche e allo sviluppo di nuove competenze: *blended learning*, classe capovolta, *debate*, apprendimento cooperativo, contesti di realtà, sviluppo delle competenze interdisciplinari
- più **inclusione** promuovendo le abilità e le competenze di ciascun alunno
- più **personalizzazione** grazie a contesti di apprendimento, adatti a diversi stili cognitivi per la promozione dell'autonomia e della creatività
- più **ingaggio motivazionale** grazie a un apprendimento attivo
- più **feedback e valutazione** grazie a HUB Test e a griglie e strumenti anche per l'autovalutazione
- più **formazione qualificata** per i docenti per accompagnarli nella progettazione e nella pratica didattica

MAPPA DEGLI ARGOMENTI DEL VOLUME

Le reti e l'architettura TCP/IP




possono essere studiate con
simulatori
come
Cisco Packet Tracer
analizzatori
come
Wireshark
Autonomous System
formata da
implementata in
INTERNET
dispositivi
end system
DHCP
configurati con
DNS
usano i nomi del
comando
nslookup
end-to-end
livelli
TRANSPORT
APPLICATION
servizi di gestione del trasferimento dati
processi
identificati con
Port number
controllate con comando
netstat
connection-oriented
protocollo
TCP
connectionless
protocollo
UDP
servizi per l'utente finale
emulazione di terminale
protocollo
Telnet
download e upload di file
protocollo
FTP
World Wide Web
protocollo
HTTP
posta elettronica
protocollo
• SMTP
• POP
• IMAP
telefonia su IP (VoIP)
protocollo
SIP
streaming
protocollo
RTSP
RTP
con trasporto su

UNITÀ

1 LE ARCHITETTURE DI RETE

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

conoscenze	abilità	competenze
Conoscere come è organizzato il software di rete in livelli.	Saper distinguere i servizi offerti da ogni livello del modello di riferimento.	Gestire le reti secondo la normativa.
Conoscere il significato di Protocol Data Unit.	Saper reperire le informazioni sugli standard.	Classificare una rete e i servizi offerti con riferimento agli standard tecnologici.
Conoscere i principali organismi internazionali che rilasciano standard per le telecomunicazioni.	Saper usare un analizzatore di protocollo e un simulatore di rete.	Monitorare il traffico della rete con un analizzatore di protocollo.
Conoscere strumenti di analisi e di simulazione della rete		Creare scenari di rete con un simulatore di rete.

DDI+ Didattica Digitale Integrata Plus

FLIPPED CLASSROOM

A casa

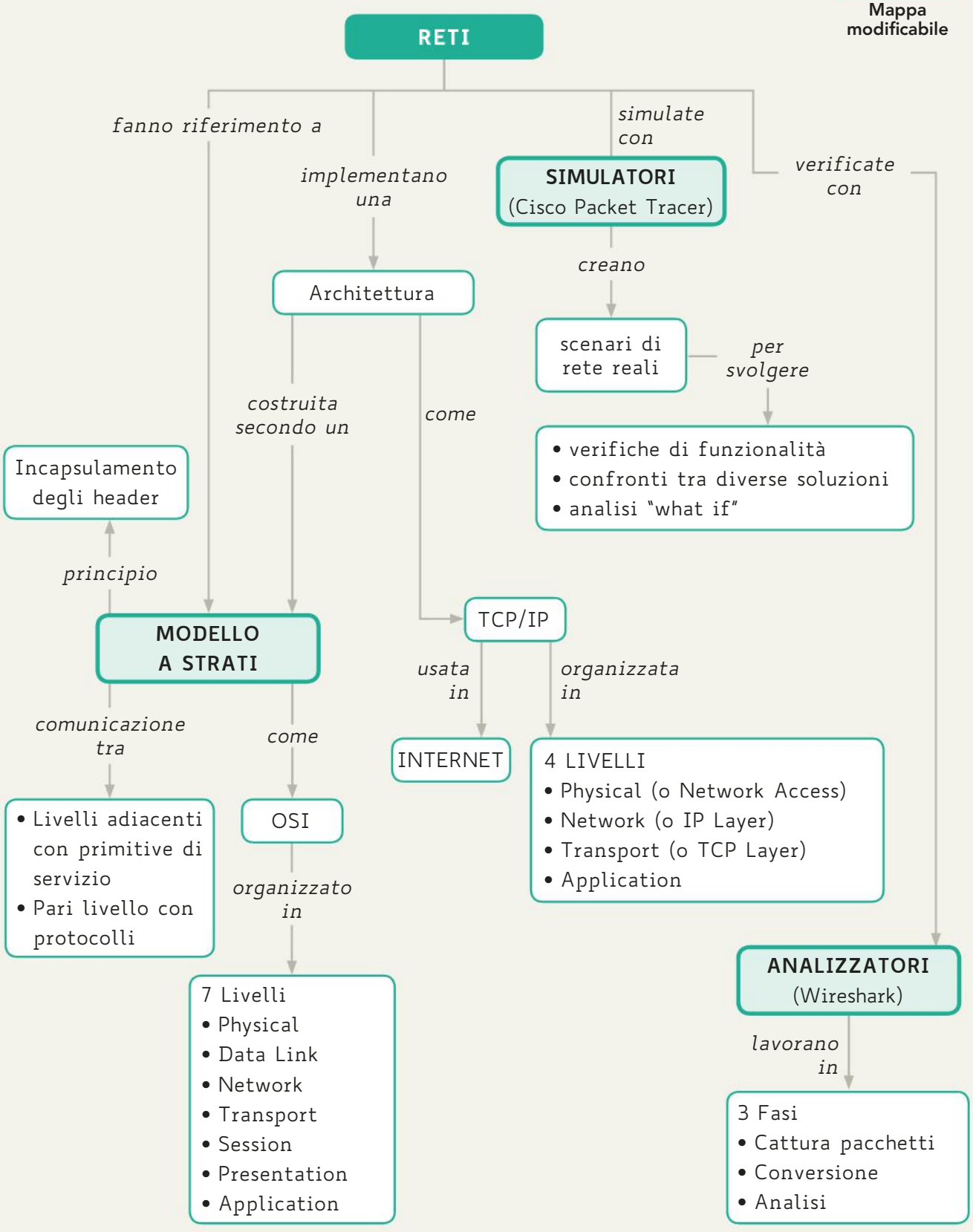
- Leggi la prima Lezione di questa Unità;
- applica i concetti del modello a strati al caso di una ditta di spedizioni;
- descrivi il processo di prelievo di un pacco dal cliente e la successiva consegna al destinatario, applicando la metodologia "a livelli" in trasmissione e in ricezione.

In classe

- Confrontate le procedure di descrizione del processo di spedizione/ricezione pacchi;
- discutete le eventuali differenze;
- concordate una descrizione finale del processo che applichi al meglio il modello a strati.

MAPPA CONCETTUALE

Mappa modificabile



1 I MODELLI E LE ARCHITETTURE DI RETE

1.1 Organizzare la complessità

Le reti sono nate per permettere la trasmissione di dati tra due macchine, anche molto distanti tra loro, e per facilitare la condivisione delle risorse. La realizzazione di una rete è un compito complesso: dal mezzo fisico su cui trasmettere il segnale alla gestione degli errori che si possono verificare durante la trasmissione, dall'identificazione dei sistemi che comunicano alle tecniche per garantire affidabilità e sicurezza. È difficile pensare di poter affrontare tutte queste problematiche con un'unica soluzione, più semplice è affrontare le problematiche in modo separato, sviluppando moduli indipendenti che cooperano per raggiungere l'obiettivo di trasmettere un messaggio da un host mittente a un host destinatario.

I progettisti di **#architetture di reti** usano come riferimento il **#modello di rete a strati** (o a **livelli**) per suddividere la complessità della comunicazione tra sistemi in funzioni elementari, assegnate a strati diversi.

#techwords

Una **architettura di rete** definisce le specifiche con cui viene realizzata una rete, nei suoi componenti hardware e software e nelle funzionalità svolte. Specifica anche i protocolli da usare nella comunicazione.
Un esempio è l'*architettura TCP/IP*.

#techwords

Un **modello di rete** definisce le modalità per interconnettere le entità che devono comunicare. Un modello non specifica i protocolli, ma solo i servizi che devono essere offerti dalla rete.
Un esempio è il *modello OSI* definito dall'ISO.

Modello a livelli – Analogia con la programmazione informatica

Nel mondo dell'informatica spesso ci troviamo a dover affrontare problemi molto complessi, per la risoluzione dei quali diventa difficoltoso tenere conto contemporaneamente di tutti gli aspetti coinvolti, fin nei minimi particolari. Una metodologia che si applica in questi casi è quella **top-down**: il problema da risolvere viene scomposto in sottoproblemi più semplici (FIGURA 1).
Così, anziché realizzare un software monolitico che svolga tutte le funzioni, si implementano più funzioni di complessità minore, ciascuna delle quali risolve una parte del problema.

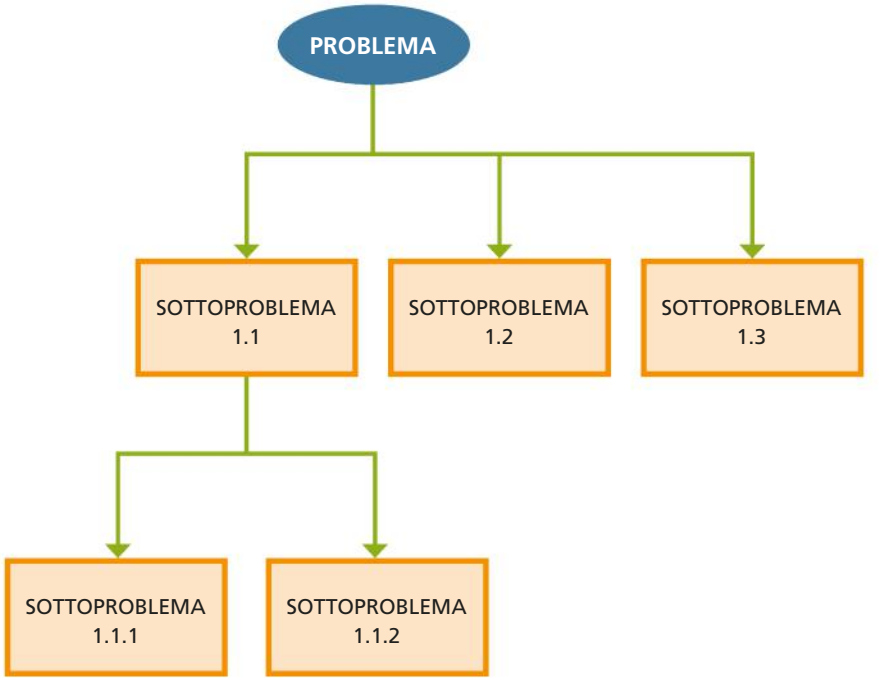


FIGURA 1 Scomposizione in sottoproblemi

Si aggiunge, però, una complessità: è necessario definire le modalità di comunicazione tra le varie funzioni in termini sia di dati scambiati, sia di "come" questi sono passati in input e in output.
I modelli a livelli definiti per le reti si basano sullo stesso principio: ridurre la complessità del problema di realizzare un sistema di comunicazione tra due macchine.

Modello a livelli – Analogia con la gestione di una richiesta di spedizione
Per comprendere la complessità della comunicazione tra i sistemi e come si riduca suddividendo il lavoro in compiti più semplici, si consideri la seguente situazione (FIGURA 2): il responsabile dell'ufficio acquisti di una società italiana con sede a Roma vuole avere informazioni sui nuovi dischi XT, prodotti da un'azienda americana famosa per i suoi supporti di storage.

La comunicazione avviene tra due persone che si trovano in luoghi molto lontani e che parlano una lingua diversa: vediamo allora come si può suddividere in compiti più semplici:

- individuare le persone nell'azienda che svolgono determinati compiti (il traduttore, l'addetto alle spedizioni, il fattorino);
- definire i servizi che ciascuna persona fornisce ad altre persone (la traduzione della lettera, la consegna/ritiro dall'ufficio postale);
- stabilire delle regole per lo svolgimento dei vari compiti, che devono essere seguite da ogni persona responsabile dell'attività (la lettera deve essere inserita in una busta e su questa va messo il francobollo).

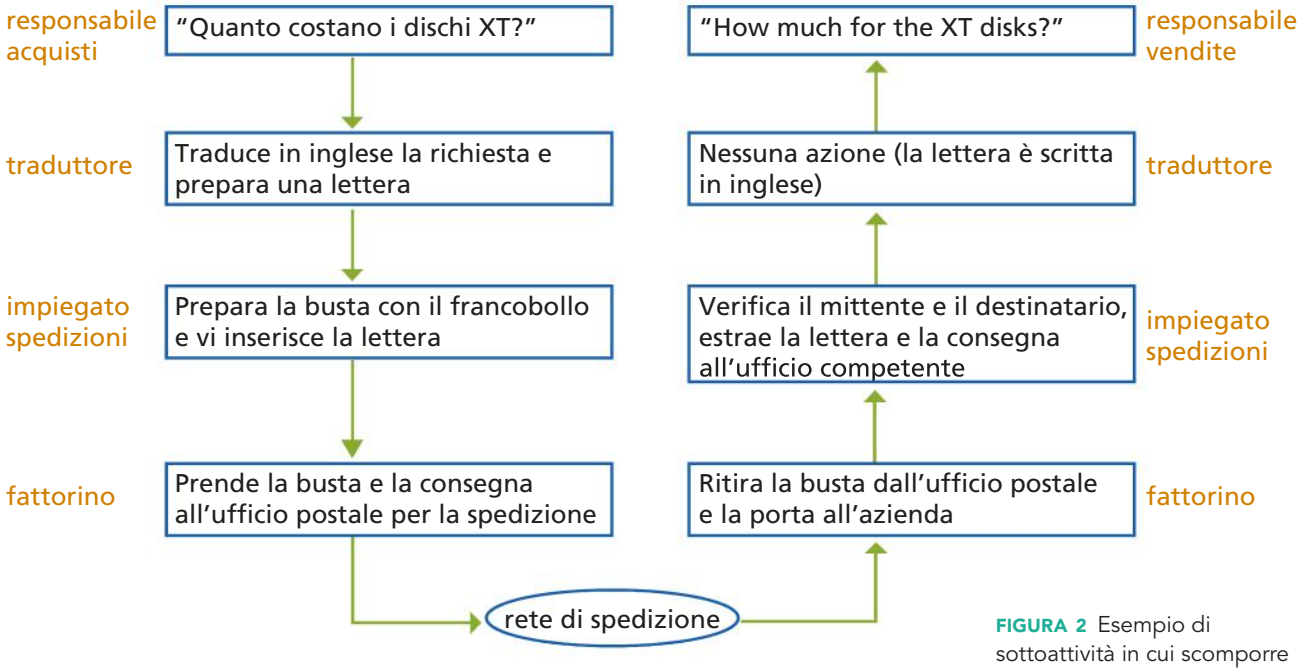


FIGURA 2 Esempio di sottoattività in cui scomporre la comunicazione

1.2 Il modello a strati o livelli

Tradizionalmente il modello a strati è rappresentato in verticale, come mostrato nella FIGURA 3, dove si presenta un ipotetico modello a cinque livelli.

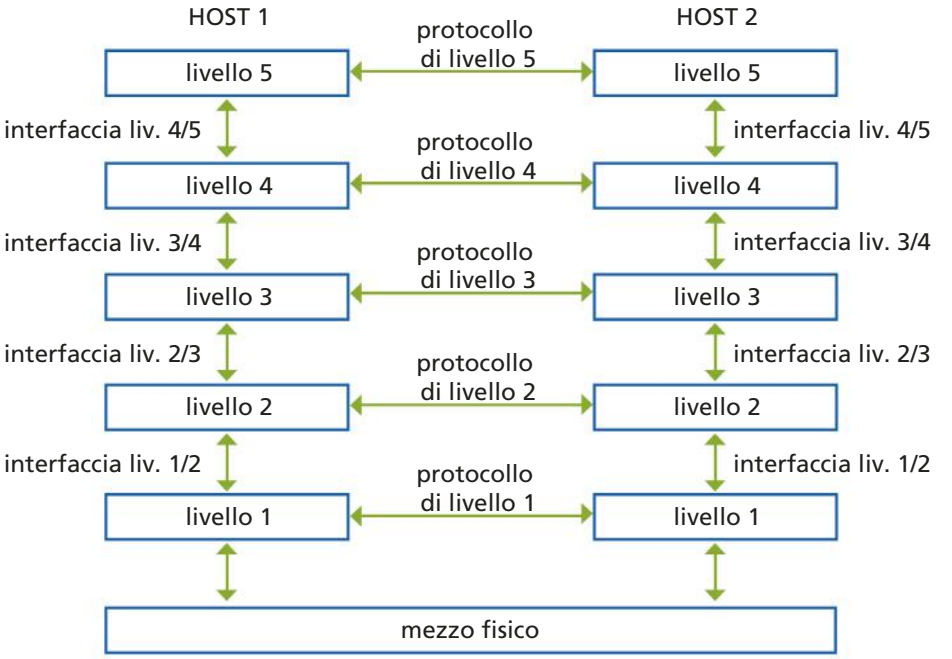


FIGURA 3 Il modello a strati

Due host che vogliono comunicare realizzano la stessa architettura a strati e implementano uno o più protocolli a ogni livello per consentire lo scambio di informazioni tra *pari*: il livello *N* mittente comunica con il livello *N* destinatario. Questi livelli vengono chiamati **peer level**.
Ogni elemento attivo, in grado cioè di inviare e ricevere informazioni, in un livello si dice **entità** (entity). Come i livelli paritari vengono chiamati peer level, così le entità paritarie sono chiamate **peer entity**.

> Nel primo volume si è più volte accennato al concetto di protocollo, ora possiamo darne una definizione più precisa, riferita al modello a livelli: **un protocollo è un insieme di regole che definiscono la comunicazione tra due peer entity**.

Quindi, il protocollo definisce le modalità con cui due entità di pari livello comunicano e, di conseguenza, deve specificare anche le informazioni di controllo da utilizzare per la gestione della comunicazione affinché il trasferimento dei dati vada a buon fine. Un esempio di informazione di controllo è il messaggio di conferma che il destinatario invia al mittente quando ha ricevuto i dati.

Il seguente elenco mostra alcune problematiche che si possono trovare nei vari livelli; nelle prossime Unità si esamineranno nel dettaglio i protocolli usati nelle reti TCP/IP come Internet e si vedrà come questi problemi sono stati affrontati nelle specifiche dei vari protocolli:

- identificazione delle peer entity;
- modalità di trasferimento dei dati (simplex, half-duplex, full-duplex);
- controllo degli errori di trasmissione;
- mantenimento dell'ordine di invio dei dati;
- adattamento della velocità di trasmissione alla capacità di ricezione del destinatario (controllo di flusso);
- gestione della dimensione dei pacchetti che transitano in rete;
- instradamento dei pacchetti nella rete.

Ogni livello fornisce un **#servizio** più astratto man mano che si procede dal basso (hardware) verso l'alto (applicazioni), svolgendo ciascuno compiti diversi dagli altri; tutti insieme concorrono a realizzare la **comunicazione tra i sistemi**.

#techwords

Servizio (*service*)
Il concetto di servizio nel modello a strati può essere considerato alla stregua di un rapporto di tipo client-server: il client è lo strato superiore e il server è lo strato sottostante. Un esempio di servizio è la consegna affidabile di un messaggio.

Nel modello a strati si possono quindi individuare due modalità di comunicazione:

1. la **comunicazione logica tra peer entity**: il messaggio che viene trasmesso da un livello al suo pari in realtà passa attraverso la comunicazione con il livello inferiore, che a sua volta lo consegna al suo pari e così via fino ad arrivare al canale fisico; fa eccezione il primo livello, dove i peer comunicano direttamente tramite il canale fisico;
2. la **comunicazione fisica tra livelli adiacenti**: ogni strato interagisce solo con i due adiacenti:
   - in **trasmissione**: il livello N riceve il messaggio dal livello N+1, lo elabora aggiungendovi le informazioni necessarie allo svolgimento delle proprie funzioni, che saranno utili alla peer entity, infine lo invia al livello N–1;
   - in **ricezione**: il livello N riceve il messaggio dal livello N–1, elabora le informazioni che erano state aggiunte dalla peer entity, successivamente le elimina e invia i restanti dati al livello N+1.

L'interazione tra due livelli adiacenti avviene tramite un'**interfaccia**, come indicato nella Figura 3.

L'interfaccia di comunicazione tra due strati definisce le regole secondo le quali un livello accede ai servizi offerti dal livello sottostante.
La suddivisione a strati delle funzioni e il concetto di interfaccia rendono le reti **modulari**: è possibile intervenire sulle caratteristiche specifiche di uno strato senza dover modificare anche gli altri, purché l'interfaccia resti immutata.

Riassumendo, i vantaggi principali del modello a strati, che ne hanno decretato il successo come modello per le reti, sono:

- **riduzione della complessità** nella costruzione di architetture protocollari tramite l'introduzione di livelli di astrazione;
- **indipendenza dei vari strati**: ogni strato deve svolgere un compito diverso dagli altri e la sua struttura non è vincolata da quella degli altri livelli; quindi strati differenti possono anche essere sviluppati da enti diversi;
- **interazione tramite servizi**: i livelli sono disposti a pila, uno sopra l'altro. Ogni livello fornisce servizi al livello superiore e usufruisce di servizi dal livello sottostante, comunicando tramite un'interfaccia; come vengono implementate le varie funzioni all'interno di un livello non è di interesse per gli altri;
- possibilità di sviluppare un **progetto modulare**: se in un secondo tempo si decidesse di aggiungere un nuovo servizio, si modificherebbe solo il software (o l'hardware) del livello coinvolto, lasciando inalterate le funzionalità svolte dagli altri livelli;
- possibilità di utilizzare **differenti protocolli** per compiti specifici.

1.3 L'incapsulamento

La FIGURA 4 mostra il principio alla base delle architetture di rete che utilizzano il modello a strati: l'**incapsulamento** (encapsulation).

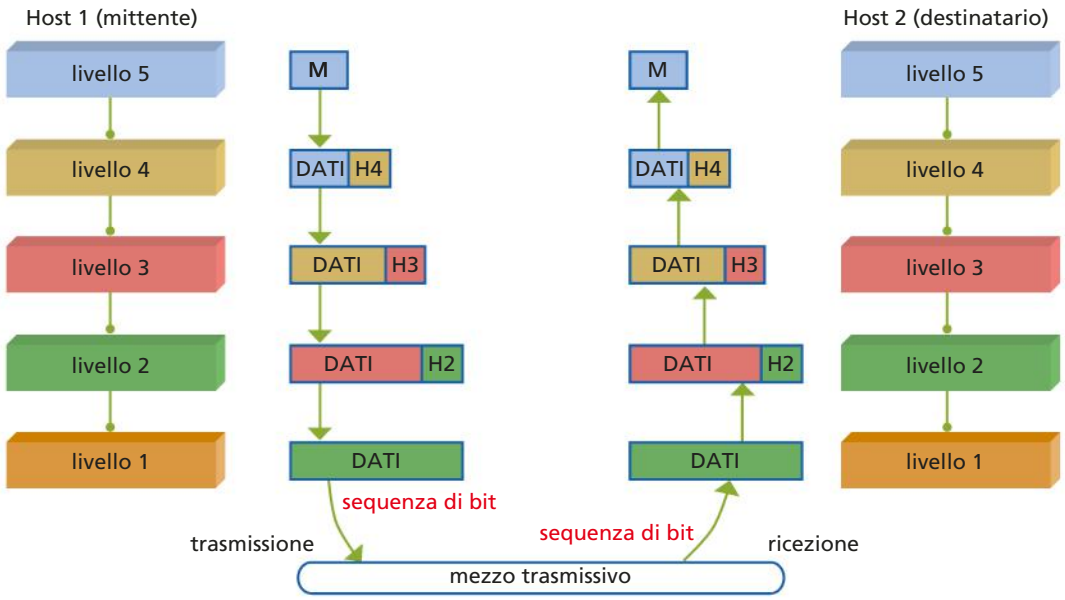


FIGURA 4 Il principio dell'incapsulamento del modello a strati

Quando il messaggio (M) inviato dal mittente passa al livello inferiore viene modificato con l'aggiunta di un insieme di informazioni utili per lo svolgimento delle funzioni specifiche di quel livello; questo insieme di dati viene detto **header** ed è collocato all'inizio, prima dell'unità dati. Per esempio, nella Figura 4 il messaggio che verrà inviato dal livello 4 al livello 3 è formato dai dati ricevuti dal livello 5 (M) e dall'header aggiunto dal livello 4 (H4). L'insieme M + H4 diventa un unico pacchetto dati inviato al livello 3.

Vediamo ora cosa succede lato ricezione: il livello 4 riceve un pacchetto dati dal livello 3, estrae i byte che compongono l'header H4, contenenti informazioni utili all'elaborazione del messaggio, informazioni che gli sono state inviate dalla sua peer entity dell'host mittente. I dati rimanenti, senza H4, sono inviati al livello 5.

Unica eccezione al principio dell'incapsulamento sono il primo e l'ultimo livello, che non modificano l'unità dati aggiungendo un header.

1.4 Le caratteristiche delle architetture di rete

Quanto descritto sinora consente di specificare meglio il concetto di architettura di rete per la comunicazione tra computer. Per realizzare un'**architettura di rete** occorre:

- definire il **modello di riferimento** in termini di schema concettuale, numero di strati coinvolti, definizione delle funzioni svolte da ciascuno strato e delle relazioni tra essi;
- definire il **servizio**, ossia ciò che viene fornito da ciascun strato di rete a quelli adiacenti;
- specificare formalmente i **protocolli** e le **interfacce** per descrivere *come* viene fornito un servizio dal singolo strato.

Finora abbiamo usato il termine generico "messaggio" o "pacchetto" per indicare l'insieme dei dati scambiati tra un host mittente e un host destinatario. Introduciamo un termine più formale, utilizzato nelle specifiche delle architetture di rete a livelli: **Protocol Data Unit** (**#PDU**). Lo standard che definisce un protocollo deve specificare il formato della sua PDU.

#techwords

PDU

Con Protocol Data Unit si identifica il messaggio scambiato tra peer entity, formato da header+dati. Una PDU è specifica di uno strato del modello, quindi la PDU del generico strato N sarà indicata come N-PDU.

La specifica dell'interfaccia avviene attraverso la descrizione delle **#primitive** da usare per la comunicazione. Tramite le primitive il livello superiore effettua la richiesta del servizio al livello inferiore, in trasmissione, oppure la ricezione delle informazioni, lato destinatario.
Le primitive di servizio sono caratterizzate da parametri quali: informazioni da trasferire, indicazione del destinatario, caratteristiche del servizio richiesto, ecc.

La FIGURA 5 mostra lo scambio delle primitive tra livelli adiacenti sullo stesso host e tra peer level.

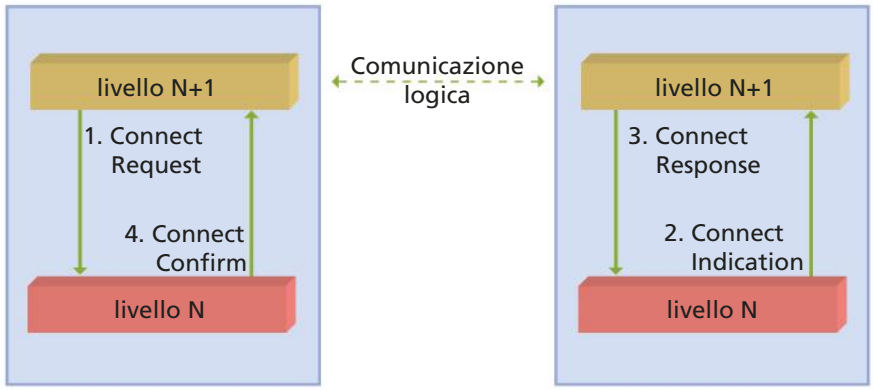


FIGURA 5 Esempio di primitive

Le primitive utilizzate nella Figura 5 sono:
1. **Connect Request**: richiesta del servizio di connessione. Specifica alcuni parametri: l'host a cui connettersi, la dimensione massima dei pacchetti usati nella connessione una volta stabilita, ecc.;
2. **Connect Indication**: la segnalazione che riceve l'host destinatario di richiesta di connessione;
3. **Connect Response**: specifica se il destinatario della richiesta l'accetta o meno;
4. **Connect Confirm**: segnalazione ricevuta dall'host sorgente che riporta l'esito della richiesta di connessione.

Non si devono confondere funzioni, servizi e primitive: le funzioni sono operazioni svolte all'interno di un determinato livello, i servizi sono offerti su un'interfaccia tra livelli adiacenti, mentre le primitive permettono di attivare i servizi.

IN ENGLISH PLEASE

Two levels on different computers (peer-entities) communicate to each other using a *protocol*. Two adjacent levels on the same computer communicate to each other through an *interface* using some *primitives*.

#techwords

Primitive
Servono a richiedere un servizio e a essere informati dell'esito della richiesta.

FISSA LE CONOSCENZE

- Descrivi il modello a strati per la comunicazione tra i sistemi.
- Perché è utile avere un modello suddiviso in livelli separati piuttosto che un modello unico?
- Che ruolo svolgono i protocolli?
- Spiega il concetto di peer entity.
- Spiega che cos'è l'header e il principio dell'incapsulamento.
- Come avviene la comunicazione tra livelli adiacenti su uno stesso host?

2 IL MODELLO ISO/OSI

2.1 Un modello di riferimento per le reti di computer

L'**ISO** (International Organization for Standardization) è l'organismo di standardizzazione che per primo cercò di definire in modo formale una modalità per interconnettere i computer. Nel 1978 ISO specificò un modello, chiamato **OSI** (Open Systems Interconnection), che divenne il modello standard di riferimento per le reti di computer. Nel documento **ISO 7498**, dal titolo *Basic Reference Model*, sono descritti i principi base di questo standard. Negli anni successivi sono state emesse versioni aggiornate con nuove funzionalità, quali, per esempio, quelle relative alla sicurezza.

#prendinota

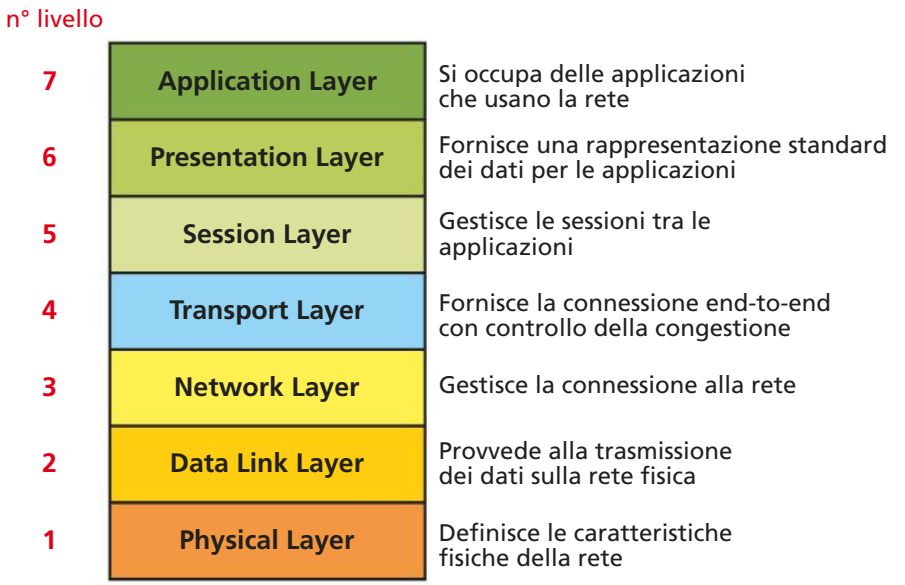
Il modello OSI non è una suite di protocolli. Lo standard non specifica i protocolli, ma come devono essere organizzati i livelli, le interfacce e i servizi che ogni livello deve offrire ai livelli adiacenti.

Il **modello ISO/OSI** è un modello a strati che suddivide in **sette livelli** le funzionalità necessarie a realizzare reti di computer.

La FIGURA 6 mostra i sette strati del modello OSI:

- i primi tre livelli fanno riferimento alla rete (sono detti **lower layers** o **network oriented layers**);
- il quarto livello separa l'ambiente rete dall'ambiente applicazione;
- gli ultimi tre livelli fanno riferimento all'applicazione (sono detti **upper layers** o **application oriented layers**).

FIGURA 6 I livelli del modello OSI



#prendinota

L'evoluzione delle reti e in particolare di Internet ha reso difficile inquadrarne l'infrastruttura in paradigmi ben definiti. Sono quindi possibili implementazioni alternative rispetto al modello OSI teorico, che però mantiene tutta la sua validità come modello di riferimento.

La FIGURA 7 mostra come avviene la comunicazione tra due host utilizzando il modello OSI. In trasmissione, i dati inviati dall'applicazione presente nell'host mittente passano da un livello al successivo e ognuno aggiunge il proprio header secondo il principio dell'incapsulamento. Quando arriva al computer di destinazione, il messaggio subisce l'operazione inversa: ogni livello esamina l'header di propria competenza e invia al livello superiore solo la parte dati (detta **#payload**).

#techwords

Payload
È la parte dati della PDU (escluso l'header).

I primi tre livelli del modello OSI sono coinvolti nella comunicazione di rete e di inter-rete (tra router), mentre solo i livelli superiori controllano la comunicazione a livello end-to-end.

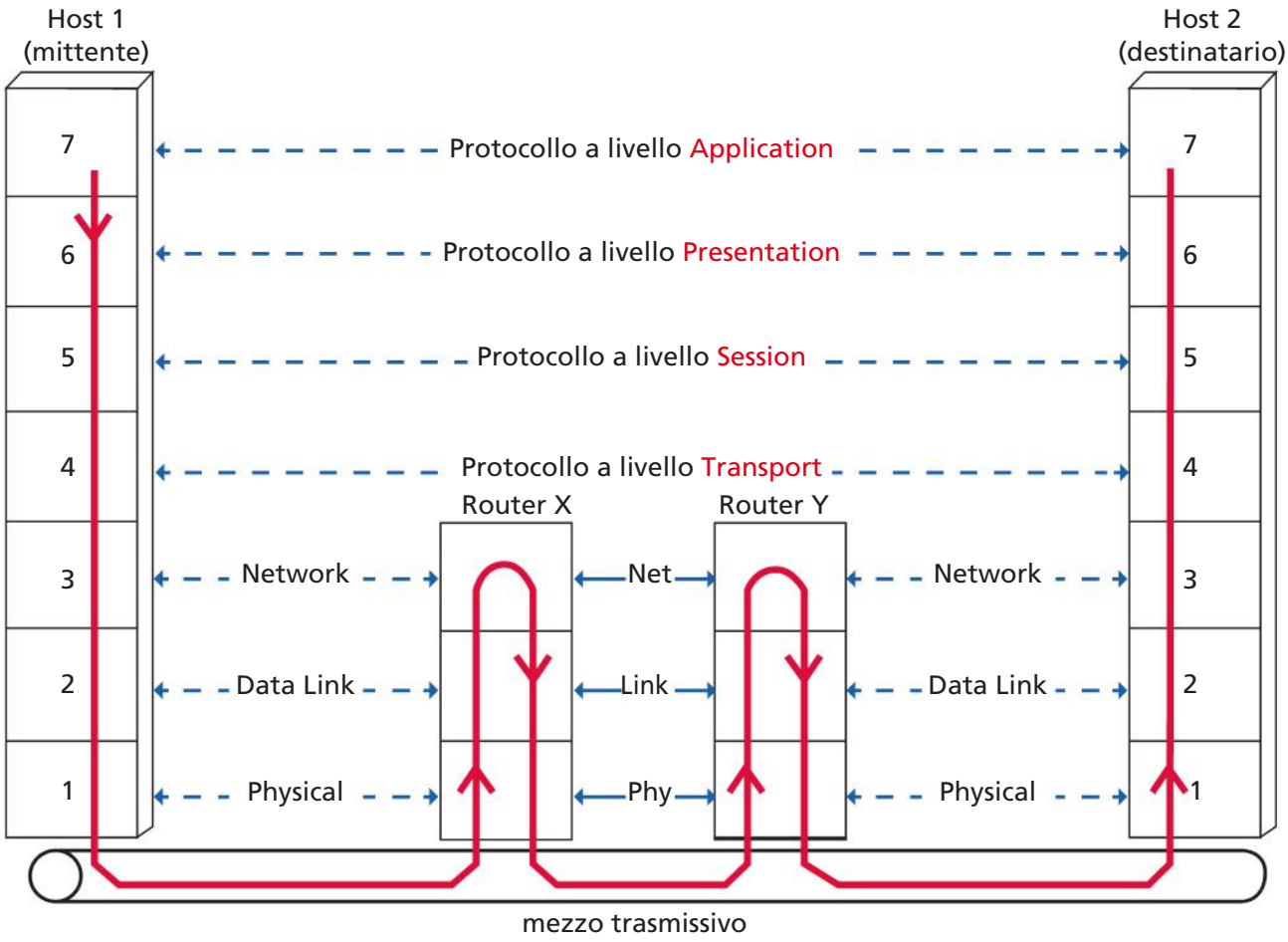


FIGURA 7 Gli strati del modello OSI

2.2 I sette livelli del modello OSI

Nel seguito si fornisce una breve descrizione dei sette strati del modello OSI.

PHYSICAL LAYER

Si occupa della trasmissione di una **sequenza di bit** attraverso un mezzo fisico.

I suoi compiti principali sono:

- definire le caratteristiche fisiche delle interfacce tra gli apparati e il mezzo trasmissivo;
- rappresentare i bit, ossia come la sequenza di 0 e di 1 viene trasformata in un segnale (elettrico, ottico, radio) da trasmettere sul mezzo fisico;
- definire la velocità di trasmissione (quanti bit sono inviati in un secondo?), sincronizzando mittente e destinatario;
- realizzare la topologia fisica della rete in base alla quale connettere i dispositivi che formano la rete.

APPARATI: i dispositivi che operano a livello 1 sono le **schede di rete** (NIC) e gli **hub**.

DATA LINK LAYER

La sua funzione principale è di rendere affidabile il collegamento instaurato a livello fisico, in modo che appaia privo di errori al livello Network. Si occupa della trasmissione tra due host della stessa rete utilizzando l'**indirizzamento fisico**: un esempio di indirizzo fisico è il **MAC Address**.

I suoi compiti specifici sono:

- in trasmissione: suddividere il flusso di bit proveniente dal livello Network in PDU dette **frame** (trame), aggiungendo a ciascun frame l'header con le informazioni su mittente e destinatario;
- controllare il flusso al fine di prevenire la congestione del dispositivo ricevente;
- controllare gli errori al fine di garantire l'affidabilità del livello fisico, individuare i frame che arrivano danneggiati, quelli persi, ecc.;
- controllare l'accesso al mezzo trasmissivo nel caso più dispositivi siano connessi allo stesso canale di comunicazione.

APPARATI: i dispositivi che operano a livello 2 sono i **bridge** e gli **switch**.

NETWORK LAYER

Si occupa dell'instradamento verso il destinatario del pacchetto inviato dal mittente attraverso reti diverse (se mittente e destinatario appartengono alla stessa rete è sufficiente il servizio offerto dal livello Data Link).

I suoi compiti principali sono:

- in trasmissione: suddividere il messaggio proveniente dal livello Transport in PDU dette **packet** o **datagram**;
- gestire l'**indirizzamento logico** in quanto l'indirizzo fisico usato a livello Data Link va bene solo a livello locale, se il pacchetto dati deve attraversare più reti è necessario usare un sistema di indirizzamento che permetta di individuare univocamente il mittente e il destinatario dei dati su rete WAN: un esempio è l'**indirizzo IP**;
- instradare (**routing**) i pacchetti: se il percorso che deve fare un pacchetto per raggiungere il destinatario attraversa più reti, i vari collegamenti indipendenti devono essere connessi per formare una **Internetwork** (una rete di reti, FIGURA 8).

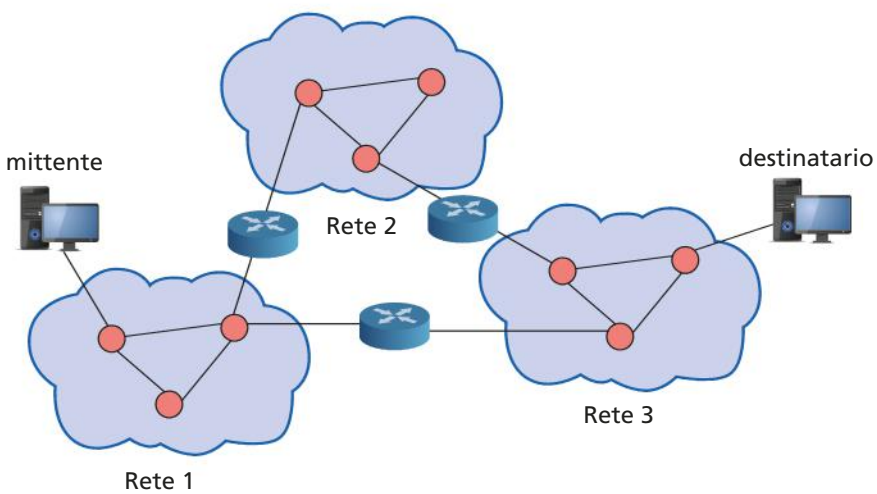


FIGURA 8 Esempio di Internetwork

APPARATI: i dispositivi che operano a livello 3 sono i **router**.

TRANSPORT LAYER

È responsabile della consegna dell'intero messaggio da mittente a destinatario, infatti si occupa della comunicazione **end-to-end** (E2E), cioè tra gli end system. Grazie al lavoro svolto da questo strato, il livello superiore Session opera come se ci fosse una linea diretta tra i due computer mittente e destinatario.

I suoi compiti principali sono:

- consegnare il messaggio al processo destinatario: su un computer ci sono molti processi attivi ed è necessario individuare quello corretto; a questo scopo si usano i numeri di **porta**: ogni porta identifica un processo sull'host mittente/destinatario;
- segmentare e riassemblare: ogni messaggio proveniente dal livello Session viene suddiviso in **segmenti** e a ognuno è associato un numero di sequenza; tale numero è quello che permetterà allo strato Transport del computer destinatario di ricostruire il messaggio nella sua interezza, inoltre consente di rilevare se durante la trasmissione si sono persi dei segmenti o sono arrivati duplicati;
- controllo di connessione: lo strato Transport offre sia il servizio **#connection-oriented** sia quello **#connectionless**, in quest'ultimo caso non viene garantita la consegna corretta del messaggio, ogni pacchetto proveniente dal livello Network è trattato in modo indipendente;
- controllo di flusso: tra i compiti svolti dallo strato Data Link c'è già il controllo di flusso, svolto a livello di singolo collegamento, nello strato Transport invece agisce end-to-end, quindi tra host mittente e host ricevente;
- controllo d'errore: anche in questo caso l'analogo controllo effettuato dallo strato Data Link è a livello di singolo collegamento, lo strato Transport invece assicura che l'*intero* messaggio arrivi al destinatario senza errori (pacchetti persi, duplicati o danneggiati).

#techwords

Un servizio è **connection-oriented** quando la comunicazione tra i due host prevede l'instaurazione di una connessione prima di inviare i dati. Un esempio è la telefonia. Un servizio è **connectionless** quando i dati sono trasmessi senza sapere se il destinatario è pronto a riceverli. Ogni pacchetto viaggia in rete in modo indipendente dagli altri. Un esempio è il tradizionale servizio postale.

SESSION LAYER

È il controllore del dialogo svolto in rete: apre, gestisce, sincronizza le interazioni tra i diversi sistemi coinvolti nella comunicazione.

I suoi compiti principali sono:

- controllo del dialogo: il dialogo viene suddiviso in unità logiche, dette **sessioni**; una sessione può essere interrotta e poi ripresa in base alle necessità del momento;
- sincronizzazione: permette ai processi coinvolti nella comunicazione di inserire dei **checkpoint** (punti di sincronizzazione) in un flusso dati; questo consente di suddividere il flusso in unità più piccole che vengono riscontrate in modo indipendente così che, nel caso di mancata ricezione, verrà inviata nuovamente solo l'unità mancante.

PRESENTATION LAYER

Offre un servizio di controllo della correttezza sintattica e semantica delle informazioni scambiate tra i due host.

I suoi compiti principali sono:

- traslazione: le sequenze di informazioni alfanumeriche che si scambiano i processi applicativi devono essere convertite in flussi di bit. Lo strato di presentazione

cambia, in trasmissione, il formato dei dati da quello del computer mittente (*sintassi locale*) a un formato comune (*sintassi di trasferimento*); in ricezione effettua l'operazione opposta: cambia i dati dal formato comune a quello proprio del computer destinatario;

- crittografia: se richiesto, lo strato di Presentazione si occupa di crittografare i dati prima di inviarli allo strato sottostante; in ricezione effettuerà l'operazione di decrittografare il messaggio per riportare l'informazione al suo stato originario, comprensibile dall'applicazione destinataria;
- compressione: quando il flusso di bit da trasmettere è di grandi dimensioni, come nel caso di invio di file audio o video, lo strato di Presentazione si occuperà di ridurre la quantità di bit da inviare.

APPLICATION LAYER

Offre un'interfaccia utente con la rete (non necessariamente utente umano, può anche essere un processo software). Questo strato fornisce il supporto ai servizi di rete come la posta elettronica, il trasferimento file, il terminale remoto, ecc. Si noti che questo strato non aggiunge un header al messaggio da inviare in rete. La PDU a questo livello viene detta **message**.

Curiosità: per ricordare, nella sequenza corretta, i nomi dei livelli del modello OSI sono stati inventati vari mnemonici.

I più noti sono:

- dal basso: "**P**lease **D**o **N**ot **T**hrow **S**ausage **P**izza **A**way";
- dall'alto: "**A**ll **P**eople **S**eem **T**o **N**eed **D**ata **P**rocessing".

#techwords

Reference Model
È una struttura astratta che permette di capire le relazioni tra le entità presenti in un certo ambito e agevola lo sviluppo degli standard a supporto di quell'ambito. È usato come riferimento per le implementazioni.

2.3 L'uso di OSI nelle reti

Il modello OSI è stato universalmente adottato come riferimento (**#reference model**) per organizzare le architetture di protocolli, mentre non hanno avuto successo i protocolli che sono stati definiti a livello Network e Transport.

Nel frattempo Internet si è diffusa velocemente e con essa l'architettura **TCP/IP**, anch'essa a strati.

I protocolli fondamentali di TCP/IP (IP per il livello Network e TCP per il livello Transport) sono incompatibili e in concorrenza con quelli definiti per OSI. Per il resto i due modelli si equivalgono.

FISSA LE CONOSCENZE

- Tutti i livelli prevedono di inserire un header nel messaggio prima di inviarlo al livello sottostante?
- Quale strato del modello OSI è implementato nella scheda di rete (NIC)?
- Quali sono gli apparati che svolgono funzioni di livello 2 e di livello 3?
- Spiega cosa significa che il livello Transport opera in modo end-to-end.
- Alcuni compiti svolti dal livello Data Link si ritrovano nel Transport: quali? In che cosa si differenziano?
- Spiega come i livelli Session e Presentation aiutano la comunicazione tra utente e rete.

3 LO STACK TCP/IP

3.1 I livelli di TCP/IP

L'**architettura TCP/IP** prende il nome dai suoi due protocolli più importanti: IP e TCP. Spesso si fa riferimento a essa con il nome di **Internet protocol suite** o **Internet protocol stack** (letteralmente "pila protocollare di Internet") per indicare la struttura a strati dell'architettura e i protocolli definiti a ogni livello.

Come si nota dalla FIGURA 9, questa architettura si basa solo su quattro strati che, con le dovute differenze, corrispondono ai sette livelli del modello OSI. Questa diversità ha ragioni storiche, infatti i protocolli IP e TCP sono antecedenti al modello OSI. Quest'ultimo specifica in modo rigoroso le funzioni che devono essere svolte da ciascuno strato, mentre i livelli del TCP/IP contengono protocolli relativamente indipendenti che possono essere usati a seconda delle necessità.
Nella Figura 9 sono presenti anche i nomi con cui si identificano le PDU ai vari livelli.

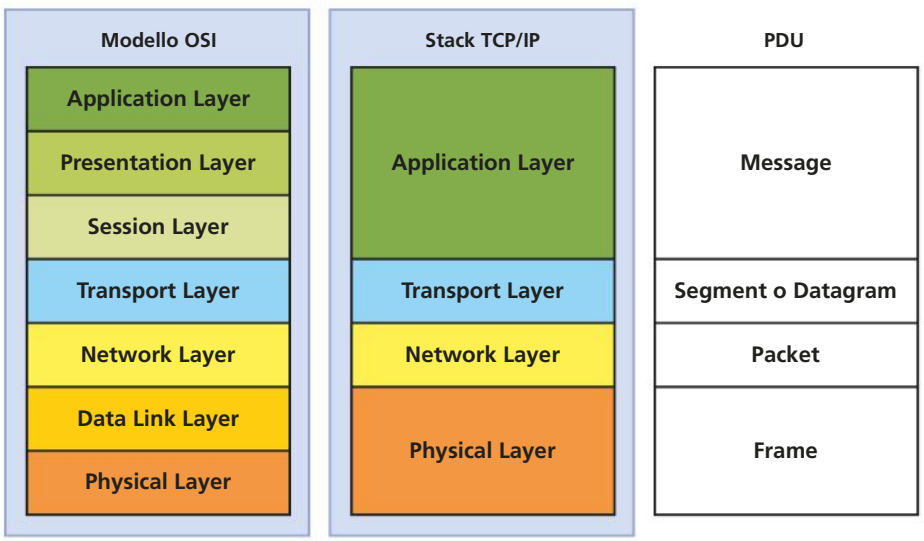


FIGURA 9 Confronto tra livelli OSI e TCP/IP

Per alcuni anni si è pensato che TCP/IP fosse una soluzione transitoria e che alla fine si sarebbe adottato l'OSI, tanto che il governo degli Stati Uniti aveva nominato una commissione per gestire il passaggio da TCP/IP a OSI. La diffusione di Internet però ha superato ogni aspettativa e finito per rendere sempre più difficoltoso il passaggio da TCP/IP a OSI e ormai da anni non si lavora più in questa direzione.
OSI rimane come modello di riferimento soprattutto per i livelli bassi (Physical, Data Link e Network).
Nel seguito si descrive l'architettura protocollare TCP/IP, su cui si basa la rete Internet.

#prendinota

A volte lo stack TCP/IP è raffigurato con cinque strati, dove il primo (Physical) viene suddiviso in due livelli: Physical e Data Link, per renderlo più corrispondente al modello OSI.

PHYSICAL LAYER

Viene spesso identificato come **Network Access Layer** perché consente di utilizzare risorse di rete diverse tra loro. Include le funzioni degli strati Physical, Data Link e, in parte, Network del modello OSI (del Network solo gli aspetti legati al funzionamento della singola rete, non dell'Internetwork).

NETWORK LAYER

È detto anche **IP Layer** dal nome del protocollo che consente il funzionamento della rete Internet. Infatti si occupa dell'instradamento dei pacchetti nella rete e dell'interconnessione delle reti. Include quindi le funzioni dello strato Network di OSI che si occupano dell'Internetwork. Offre un servizio connectionless.

TRANSPORT LAYER

È detto anche **TCP Layer** dal nome del protocollo principale che permette di gestire le connessioni a livello end-to-end. TCP infatti fornisce un servizio connection-oriented. Corrisponde al livello Transport dell'OSI. In questo livello è stato standardizzato anche un altro protocollo, denominato UDP, che offre un servizio connectionless.

APPLICATION LAYER

Corrisponde agli ultimi tre strati del modello OSI e realizza i servizi di livello applicativo per Internet, quali la posta elettronica, il web, il trasferimento dei file, ecc.

3.2 L'evoluzione di TCP/IP

Con la suite di protocolli Internet è possibile connettere tra loro vari tipi di reti, senza richiedere che esse soddisfino particolari requisiti, si assume solo che siano in grado di trasferire dati, senza entrare nel merito di come effettuino tale trasferimento. Chiaramente le prestazioni generali dipendono dalla tecnologia e tipologia delle sottoreti su cui appoggia TCP/IP.

Nel 1983 lo stack TCP/IP è diventato l'architettura ufficiale di Internet e nel corso degli anni è evoluto arrivando alla versione 6:

- **versione 4**: è ancora oggi molto diffusa, si basa sulla suddivisione in classi degli indirizzi IP che sono in un formato a 32 bit; la diffusione rapidissima di Internet ha portato, però, alla carenza di indirizzi e alla difficoltà di soddisfare nuove richieste;
- **versione 5**: è una proposta basata sul modello OSI che non è mai stata portata avanti a causa delle molte modifiche da effettuare sugli strati, che comportavano costi elevati;
- **versione 6**: le modifiche riguardano principalmente lo strato Network: IPv4 diventa IPv6 e utilizza indirizzi a 128 bit che consentono di gestire un numero elevatissimo di utenti.

FISSA LE CONOSCENZE

- Che cosa si intende per pila protocollare?
- Spiega la corrispondenza tra i livelli del modello OSI e quelli dell'architettura TCP/IP.
- Le funzioni svolte dal livello Network di OSI dove si possono ritrovare in TCP/IP?
- Il livello Network di TCP/IP è in qualche modo influenzato dalla tecnologia con cui sono realizzate le sottoreti?
- Descrivi quali versioni, nel corso degli anni, si sono avute di TCP/IP.

4 GLI ENTI DI STANDARDIZZAZIONE

4.1 Chi specifica gli standard per le reti?

Data l'importanza degli standard nelle telecomunicazioni, vediamo ora le principali organizzazioni internazionali che sviluppano standard in quest'ambito.
Anche se questi organismi lavorano in modo autonomo nel loro specifico ambito di competenza, non possono certamente ignorare quello che viene svolto dagli altri enti di standardizzazione; quindi negli anni, con l'evolvere della tecnologia e delle richieste degli utenti, si sono sviluppate collaborazioni e sinergie.

ITU-T

International Telecommunication Union - Telecommunication Standardization Sector
Sito web: *www.itu.int/ITU-T*

- 1865: i rappresentanti dei principali governi europei si riuniscono a Parigi per dare vita a un'organizzazione (ITU, International Telegraph Union) per la definizione di standard nel settore delle telecomunicazioni;
- 1947: ITU diventa un ente supportato dalle Nazioni Unite;
- 1956-1993: ITU-T è identificato con l'acronimo **CCITT** (Comité Consultatif International Télégraphique et Téléphonique);
- ITU viene suddiviso in tre settori:
  - ITU-R Radiocommunication Sector;
  - ITU-T Telecommunication Standardization Sector;
  - ITU-D Development Sector;
- l'autonomia nazionale è salvaguardata (in teoria) dal fatto che le specifiche ITU-T sono solamente *recommendation*, non sono standard;
- importanti raccomandazioni nell'ambito delle telecomunicazioni sono:
  - serie V Telephone Communications, per esempio V.90 e V.92 sono gli standard per il modem analogico a 56 kbps;
  - serie X Network Interface and Public Networks, per esempio la serie X.400 specifica gli standard per la posta elettronica.

ISO

International Standards Organization, venne in seguito denominato:
International Organization for Standardization
Sito web: *www.iso.org*

- ente fondato nel 1946 come organizzazione volontaria con lo scopo di stabilire accordi sugli standard internazionali;
- i membri sono i rappresentanti degli enti di standardizzazione designati da ciascun Paese aderente (il rappresentante italiano è l'UNI, Ente Nazionale Italiano di Unificazione) e di molte industrie che vi partecipano con l'obiettivo di definire nuovi modelli di compatibilità, migliore qualità, maggiore produttività e costi più bassi;
- si occupa di una vasta gamma di standard nei campi scientifico, tecnologico ed economico. Lo sviluppo degli standard è controllato da un Comitato Tecnico separato. Nell'ambito delle telecomunicazioni, importante è stato il suo contributo con la definizione del modello a strati di riferimento: modello a sette livelli OSI (Open System Interconnection).

#prendinota

Nella seconda metà degli anni Novanta i **modem 56k** erano lo standard per i modem analogici. Si trattava di dispositivi che si collegavano alla linea telefonica domestica, sulla quale si effettuava la chiamata verso l'operatore telefonico (e se la linea era occupata per la trasmissione dati non era possibile telefonare). Così ci si connetteva a Internet con una velocità di 56 kbps!

ANSI

American National Standards Institute

Sito web: *www.ansi.org*

- è un organismo americano nato con l'obiettivo di promuovere l'adozione degli standard come mezzo per il progresso dell'economia negli Stati Uniti;
- vi partecipano organizzazioni professionali, associazioni industriali, enti governativi e gruppi di consumatori;
- in ambito ANSI sono nati alcuni importanti standard, poi ratificati dall'ISO, tra questi ricordiamo:
  - ASCII (American Standard Code for Information Interchange);
  - FDDI (Fiber Data Distributed Interface).

IEEE

Institute of Electrical and Electronics Engineering

Sito web: *www.ieee.org*

- organizzazione professionale (è la più grande corporazione di ingegneri del mondo), dedicata al processo di standardizzazione e non solo;
- definisce standard nei settori dell'ingegneria elettronica e informatica/telecomunicazioni, per le reti ricordiamo:
  - progetto 802.x, definisce un insieme di standard per le LAN e le MAN relativamente al livello Physical di TCP/IP.

ETSI

European Telecommunications Standards Institute

Sito web: *www.etsi.org*

- è un'organizzazione non-profit il cui obiettivo è la definizione degli standard di telecomunicazione a livello europeo;
- riunisce oltre 700 membri provenienti da più di 60 nazioni europee e non: amministrazioni pubbliche, operatori di telecomunicazione, industrie manifatturiere, fornitori di servizi, centri di ricerca e organizzazioni di utenti;
- dal 1998 è tra le organizzazioni partner dell'associazione 3GPP (3rd Generation Partnership Project) che produce specifiche nell'ambito delle comunicazioni mobili.

IETF

Internet Engineering Task Force

Sito web: *www.ietf.org*

- è l'organismo operativo dello **IAB** (Internet Architecture Board), l'ente che si occupa della supervisione del processo di creazione degli standard Internet;
- è una comunità internazionale di progettisti, costruttori, enti di ricerca, che si occupa dell'evoluzione di Internet;
- i suoi documenti (**RFC**, Request For Comment) sono delle linee guida che rappresentano degli *standard de facto* per la loro ampia diffusione.

4.2 Una proposta di attività

Ognuno degli enti elencati in precedenza ha un sito web tramite il quale diffondere le proprie attività. Si tratta di siti in inglese (alcuni dei quali però offrono la traduzione in altre lingue), dove si possono reperire informazioni interessanti.
Infatti essendo queste le fonti autorevoli a cui si devono adeguare produttori di apparati, fornitori di servizi di rete e applicativi, e chiunque abbia a che fare con la comunicazione e le reti, è importante imparare a conoscere le fasi di sviluppo di uno standard, dove reperire la documentazione che interessa, quali nuove tecnologie e servizi sono in corso di definizione, ecc.

La seguente è una proposta di attività da svolgere analizzando i siti web dei principali organismi di standardizzazione elencati in precedenza, che potrà essere poi ripresa di volta in volta quando nelle unità successive si affronterà lo studio in dettaglio dei principali standard attualmente utilizzati in Internet.

1. **Norme sulla pubblicazione degli standard**: numerazione, lingua, periodo di validità, modalità di pubblicazione, i documenti sono a pagamento o si possono scaricare gratuitamente dal sito?
2. **Processo di standardizzazione**: gruppi di lavoro, modalità di lavoro (mailing list o riunioni periodiche), qual è l'iter che devono seguire i documenti per passare da bozza a standard?
3. **Nuovi standard**: ci sono lavori in corso per evolvere gli attuali standard? Se ne stanno sviluppando di nuovi? Quali provider/produttori/utenti/... partecipano allo sviluppo?
4. **Collaborazioni**: sono previste collaborazioni con altri organismi di standardizzazione, forum, organizzazioni di utenti?

esercizio

➜ Problema

Prendere in esame l'organizzazione IETF, rispondere ai punti elencati sopra e scegliere un Working Group di IETF del quale esaminare l'attività.

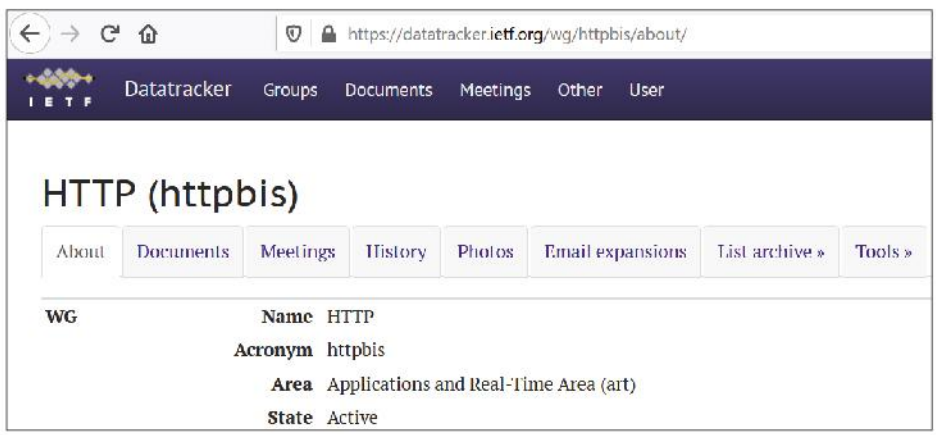
➜ Svolgimento

1) **Norme sulla pubblicazione degli standard**: le specifiche IETF vengono pubblicate come RFC (*Request For Comment*), si tratta di documenti tecnici che forniscono le linee guida in ambito Internet. Essi sono numerati in modo progressivo. Tutti i documenti IETF, in lingua inglese, sono di pubblico dominio e scaricabili gratuitamente dal sito web.
2) **Processo di standardizzazione**:
   – *Internet Draft* (I-D): documento allo studio, in via di definizione all'interno di un WG (*Working Group*);
   – *Proposed Standard Protocol*: il protocollo viene preso in considerazione per un'eventuale standardizzazione, devono esistere almeno due implementazioni indipendenti per passare alla fase successiva e devono trascorrere almeno 6 mesi di valutazione;
   – *Draft Standard Protocol*: verifica della reale operatività del protocollo su implementazioni indipendenti interoperabili, devono trascorrere almeno 4 mesi di valutazione;
   – *Standard Protocol*: documento finale ratificato.

3) **Nuovi standard**: per capire quali argomenti e problematiche sono oggetto di studio in ambito IETF è utile analizzare i WG attivi (*https://datatracker.ietf.org/wg/*) nelle varie aree:
   – Applications and Real-Time Area (art)
   – General Area (gen)
   – Internet Area (int)
   – Operations and Management Area (ops)
   – Routing Area (rtg)
   – Security Area (sec)
   – Transport Area (tsv)

Passiamo ora a esaminare l'attività svolta da un gruppo di lavoro, attivo in una delle sette aree di lavoro elencate (questo compito potrebbe essere affidato a gruppi di studenti, ognuno dei quali sceglie un IETF WG da esaminare).
La FIGURA 10 mostra la parte alta della pagina web del WG scelto: **HTTP** (httpbis).

FIGURA 10 IETF WG HTTP



Working Group: HTTP (httpbis)
Area: Applications and Real-Time Area
Chairs: Mark Nottingham (Fastly), Tommy Pauly (Apple Inc.)
Area Director: Barry Leiba (FutureWei Technologies)
Mailing List: ietf-http-wg@w3.org
Data creazione: ottobre 2007

Un'informazione utile per capire quali enti o aziende stanno lavorando su nuovi standard è individuare le persone che svolgono ruoli di chairman (presidente) del gruppo o sono autori delle specifiche e trovare per quale azienda lavorano: in ogni documento IETF, oltre al nome dell'autore è sempre specificata l'azienda o l'ente di appartenenza. Nel caso di questo WG, il chairman Mark Nottingham lavora per Fastly, un'azienda che offre una piattaforma in cloud e servizi per sviluppatori di siti web, il secondo chairman, Tommy Pauly, è dipendente della Apple. Quindi i presidenti del gruppo di lavoro sono persone che rappresentano aziende fortemente interessate alle specifiche emesse in ambito HTTP, perché interessate a implementarle nei loro prodotti, e la loro presenza nel WG è quindi determinante per capire come evolve lo standard. Spesso ciò avviene ancor prima che siano ratificate e questo si traduce spesso in un vantaggio competitivo per le aziende presenti in un WG.

Descrizione dello scopo del WG
Obiettivo del WG è il mantenimento e lo sviluppo delle specifiche "core" di HTTP ed eventuali estensioni di tipo generale, non specifiche di un'applicazione.

Il lavoro del WG sarà quindi volto a:
- revisionare le specifiche di HTTP 1.1 (documenti RFC 7230-7235);
- definire eventuali estensioni.

Documenti che devono essere prodotti dal WG

Il WG HTTP sta lavorando su un numero elevato di documenti (Internet Draft) e ha prodotto parecchi RFC.
Tre Internet Draft sono definiti "core", ossia fondamentali per le specifiche dello standard del protocollo HTTP:
- **HTTP/1.1 Messaging**
- **HTTP Caching**
- **HTTP Semantics**

Descrizione di un documento

Sotto si riporta l'abstract del secondo documento sopra elencato, prodotto dal WG. Si tratta di un Internet-Draft (I-D) e come tale ha una durata di 6 mesi. Prima della scadenza il WG dovrà scrivere un nuovo I-D, revisione del precedente, oppure, se è giunto a una versione definitiva, dovrà procedere con la richiesta di approvazione della specifica che diventerà un documento RFC (Request For Comments).
La specifica **HTTP Caching** definisce le caratteristiche e il funzionamento della cache associata al web browser. La cache è l'area di memoria locale dove il browser (HTTP client) memorizza le risposte ricevute alle richieste di pagine web. Questo meccanismo consente di ridurre i tempi di risposta: se una pagina è presente nella cache, il browser la visualizza subito, evitando di inoltrare la richiesta al server HTTP.

IN ENGLISH PLEASE

HTTP Working Group — R. Fielding, Ed.
Internet-Draft — Adobe
Obsoletes: 7234 (if approved) — M. Nottingham, Ed. Fastly
Intended status: Standards Track — J. F. Reschke, Ed. greenbytes
Expires: January 13, 2021 — July 12, 2020

HTTP Caching
draft-ietf-httpbis-cache-10

Abstract

The Hypertext Transfer Protocol (HTTP) is a stateless application-level protocol for distributed, collaborative, hypertext information systems. This document defines HTTP caches and the associated header fields that control cache behavior or indicate cacheable response messages.

FISSA LE CONOSCENZE

- Quale organizzazione internazionale emette le linee guida per Internet?
- Spiega cosa sono gli RFC.
- Quale organizzazione ha definito gli standard del primo livello dello stack TCP/IP per le reti LAN e MAN?
- Quando un documento è definito draft?

5 WIRESHARK: UN ANALIZZATORE DI PROTOCOLLO

5.1 L'uso di analizzatori di protocollo nelle reti

La complessità delle reti moderne ha reso difficile il controllo delle condizioni della rete, tipico compito svolto dall'amministratore di rete che deve garantire che sia funzionante e con ottime prestazioni.
L'analisi del traffico dati è di grande utilità per rilevare la causa di rallentamenti o potenziali problemi che potrebbero, in futuro, tradursi in malfunzionamenti della rete.

#techwords

Sniffer / Packet sniffer / Traffic sniffer / Packet Analyzer
Sono termini usati per riferirsi a strumenti di osservazione, cattura e analisi del traffico trasmesso e ricevuto dalle applicazioni e dai protocolli in esecuzione sul computer.

Un **analizzatore di protocollo**, o **#sniffer**, è uno strumento molto utile per gli amministratori di rete e per chi si occupa di sicurezza, aiutandoli nel monitoraggio della rete e delle sue prestazioni e nell'individuazione di anomalie o guasti.

Esistono vari tipi di analizzatori, i più diffusi sono applicazioni software che catturano i pacchetti entranti e uscenti dall'interfaccia di rete (NIC). Un esempio è **Wireshark**, descritto in seguito.
Un amministratore di rete, però, ha la necessità di raccogliere dati anche da molti altri dispositivi, non solo dalla NIC locale. In passato nelle LAN si usavano gli hub per realizzare reti con topologia a stella. Un hub quando riceve un pacchetto su una porta lo copia su tutte le altre porte, quindi è sufficiente collegare un analizzatore, software o hardware (#probe), a una porta per "vedere" tutto il traffico che transita attraverso l'hub.

#techwords

Probe
Nell'ambito delle reti, è un dispositivo hardware (in italiano, sonda) che si inserisce nella connessione tra due host per catturare il traffico che passa sul cavo.

Nelle reti moderne gli hub sono stati sostituiti dagli switch, che inoltrano i dati ricevuti alla sola porta di destinazione e non più a tutte (a meno di pacchetti inviati in broadcast), impedendo così a un analizzatore connesso a un'altra porta di intercettare tutto il traffico.
Una soluzione a questo problema è la tecnica del **port mirroring**: tutti i dati che transitano nello switch sono copiati e inviati su una porta specifica, alla quale è collegato l'analizzatore (FIGURA 11). In questo modo l'amministratore può catturare e analizzare il traffico proveniente da diverse sorgenti presenti nella rete.

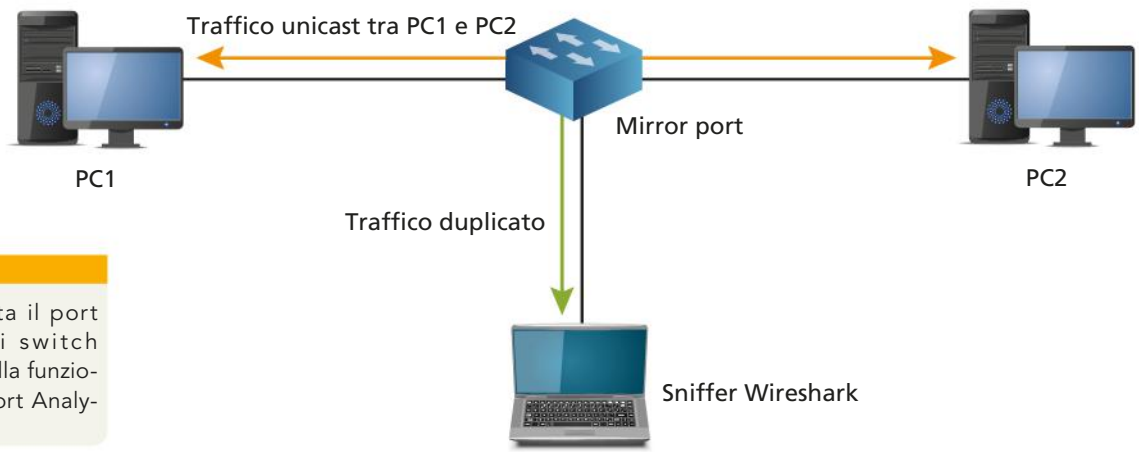


FIGURA 11
Switch con port mirroring

#prendinota

Cisco implementa il port mirroring negli switch Catalyst dotati della funzionalità Switched Port Analyzer (SPAN).

I pacchetti che l'analizzatore cattura (**#packet capture**) sulla NIC locale, o sullo switch tramite la mirror port, possono essere analizzati in modalità:

- "live", mentre avviene la cattura dei dati;
- a posteriori, analizzando il file che contiene il traffico catturato (**PCAP file**).

La seconda modalità è quella più utilizzata, perché consente di salvare i dati raccolti ed eventualmente inviarli a un gruppo di esperti per un'analisi più approfondita (è il caso delle analisi in ambito forense). La sequenza di passi solitamente è:

1. eseguire l'applicazione dell'analizzatore e selezionare le opzioni di cattura in base a quali dati si vogliono raccogliere;
2. avviare la cattura e raccogliere un buon numero di pacchetti (1.000-2.000 pachetti);
3. fermare la cattura e salvare il PCAP file nel formato appropriato;
4. analizzare il traffico catturato, un pacchetto alla volta o nel suo complesso.

#techwords

Packet Capture (PCAP)
"Afferrare" una copia dei pacchetti transitanti su un'interfaccia di rete, prima che essi siano processati dal sistema operativo. I dati catturati sono salvati in un file che viene chiamato **PCAP file**.

5.2 Le caratteristiche di Wireshark

Wireshark è un analizzatore di protocollo (rilasciato sotto licenza open source) in grado di esaminare il contenuto di tutti i pacchetti dati in transito sulle interfacce di rete utilizzate. Il programma raccoglie l'eredità del software *Ethereal*, dal 2006 rinominato Wireshark dal suo principale sviluppatore Gerald Combs.
Wireshark fornisce una "fotografia" dettagliata di tutto ciò che sta accadendo sulla rete locale mediante una semplice interfaccia grafica.
Consente di analizzare la struttura di una rete alla ricerca di eventuali errori di configurazione, inoltre è in grado di identificare molti tipi di incapsulamento e di isolare e visualizzare tutti i campi che compongono un pacchetto.

#prendinota

Wireshark è un software open source rilasciato con licenza GNU/GPL, scaricabile liberamente dal sito: *www.wireshark.org/#download*. Qui si trovano i programmi di installazione per Windows e macOS, mentre la maggior parte delle distribuzioni di Linux includono già il package di questo software.

Punto di forza di Wireshark è sicuramente la flessibilità: grazie a una serie di criteri di ordinamento e filtraggio diventa facile isolare i dati che interessano dalla gran quantità di traffico registrato.
Wireshark offre un valido ausilio per aiutare gli esperti nell'individuazione di eventuali vulnerabilità dei sistemi utilizzati in azienda (credenziali di accesso trasmesse in chiaro, attività sospette perpetrate dai client delle LAN, transito di informazioni sensibili e così via).

Wireshark può analizzare anche il traffico catturato da numerosi altri tool grazie alla funzionalità di Import, come per esempio Microsoft Network Monitor, e salvare i pacchetti raccolti nei formati usati da altri programmi di cattura, utilizzando la funzionalità di Export.

COME FUNZIONA WIRESHARK?

Il processo di **sniffing** attuato da Wireshark si può dividere in tre fasi:

1. **raccolta**: sono catturati i dati grezzi (raw binary data) che fluiscono attraverso l'interfaccia di rete selezionata;
2. **conversione**: i dati raccolti sono trasformati in un formato leggibile e i pacchetti sono riassemblati in base alla loro sequenza;
3. **analisi**: si inizia con l'identificare il tipo di protocollo, il numero di porta, ecc. per arrivare a un'analisi più approfondita esaminando i campi degli header dei vari protocolli.

La FIGURA 12 mostra uno schema molto generale dei blocchi che compongono il software Wireshark: un **core** che fa da collante tra tutti i moduli specifici, quali l'interfaccia grafica verso l'utente (**GUI**) e l'interfaccia **capture** verso le librerie esterne.
Oltre all'interfaccia utente grafica, basata sul framework **Qt**, è stata sviluppata anche un'interfaccia console, detta **Tshark.**
Per la cattura dei pacchetti, Wireshark utilizza le seguenti librerie esterne:

- **dumpcap**: è il motore che gestisce la cattura dei pacchetti e scrive i dati (raw data) in un file sul disco, il formato di default è **.pcapng**;
- **Npcap**: è la libreria di cattura e filtraggio dei pacchetti per i sistemi Windows, ha sostituito la precedente winpcap (è scaricabile da: *nmap.org/npcap*);
- **libpcap**: è la libreria di cattura e filtraggio dei pacchetti per i sistemi Unix/Linux (è scaricabile da: *www.tcpdump.org*).

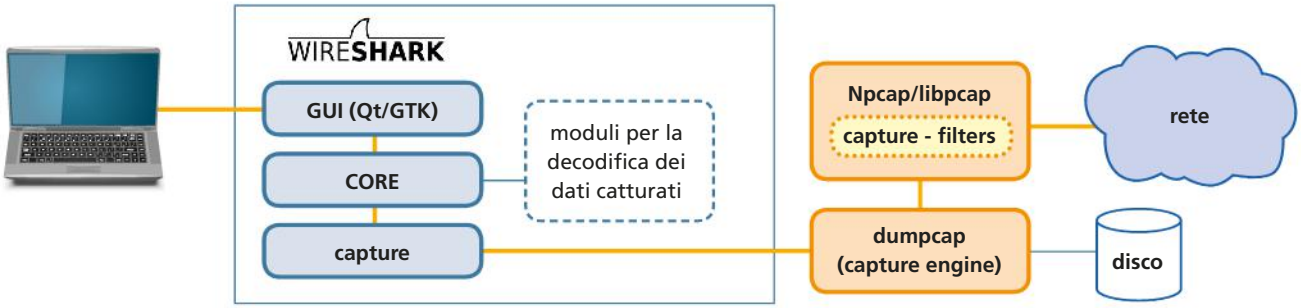


FIGURA 12 Schema del funzionamento di Wireshark

AVVIO DI WIRESHARK

La FIGURA 13 mostra l'interfaccia grafica di Wireshark versione 3 come appare quando si avvia l'applicazione ed è alla ricerca delle interfacce di rete presenti sul computer locale. La schermata successiva, visualizzata in FIGURA 14, è quella principale, in cui si elencano le interfacce trovate.

#prendinota

Il progetto Wireshark è migrato su **GitLab**, dove si trova il repository ufficiale: *https://gitlab.com/wireshark/wireshark*
Qui è anche disponibile un **wiki** molto utile per gli utenti.

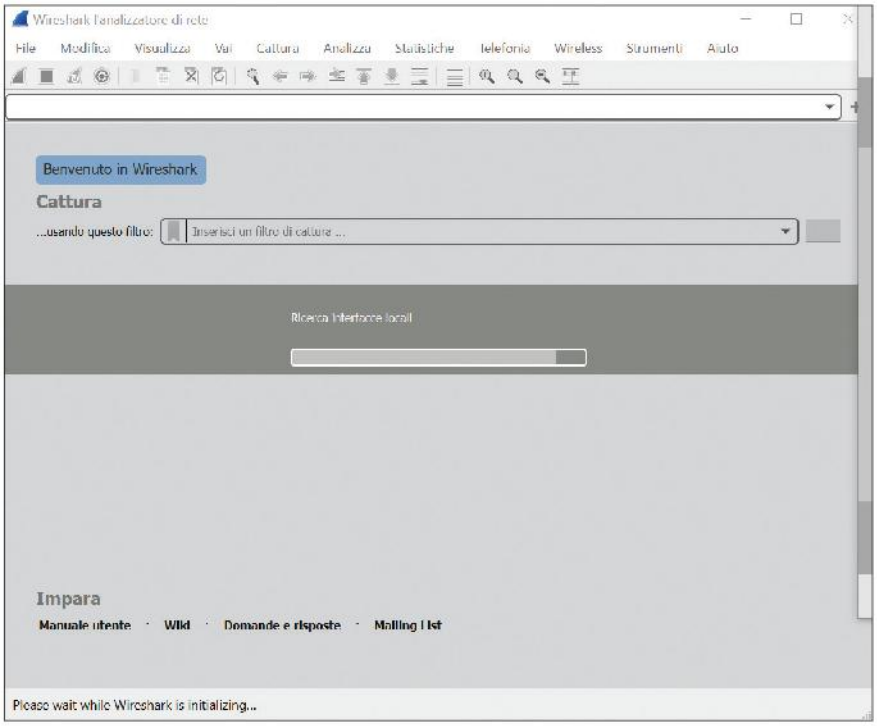


FIGURA 13 La pagina iniziale di Wireshark versione 3

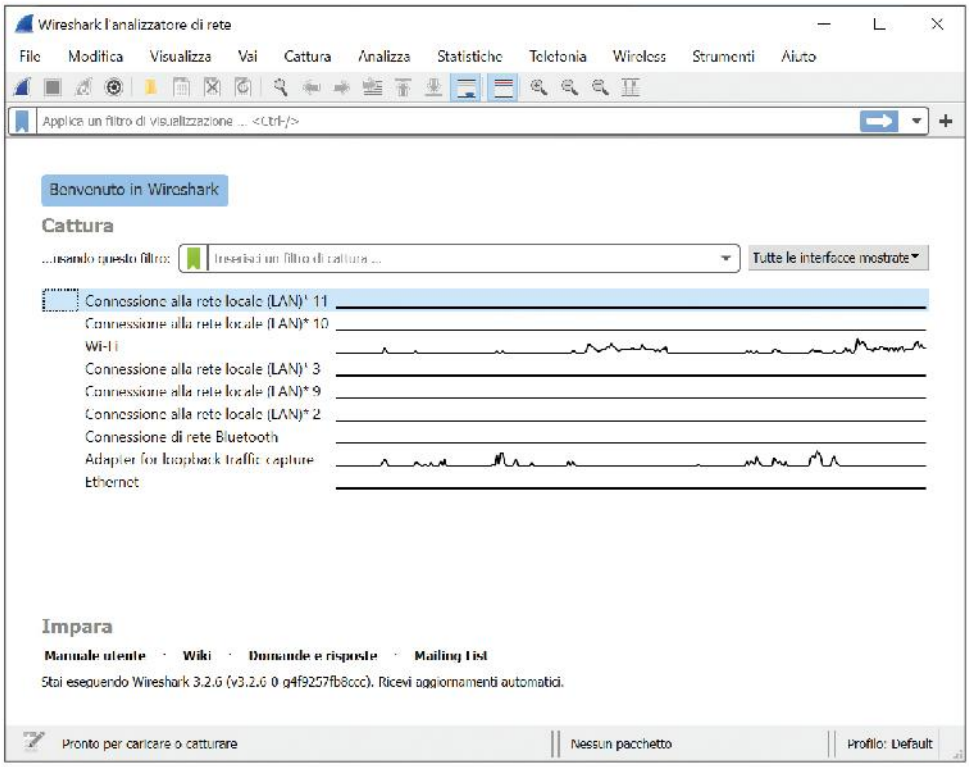


FIGURA 14 La pagina iniziale di Wireshark versione 3 con le interfacce di rete

LA CATTURA DEI PACCHETTI

Per effettuare con successo la cattura dei pacchetti è necessario tener presente alcuni fattori:

- per avviare una cattura in modalità "live" il sistema potrebbe richiedere i privilegi di amministratore;
- è importante scegliere l'interfaccia di rete adeguata alla cattura dei pacchetti;
- per vedere il traffico desiderato, è fondamentale dove ci si posiziona nella rete.

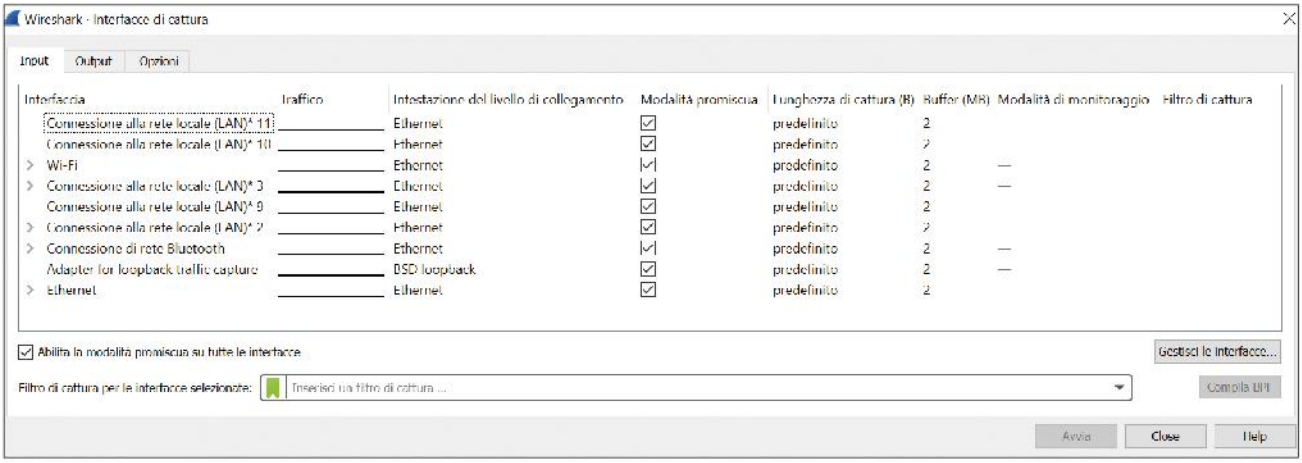
Per iniziare la cattura dei pacchetti con Wireshark si può:

- cliccare sull'interfaccia desiderata nella finestra principale;
- selezionare la voce **Opzioni** dal menu **Cattura**, compare una finestra in cui si seleziona l'interfaccia e poi si clicca sul pulsante **Avvia** (FIGURA 15).

#prendinota

Per conoscere tutte le modalità di impostazione di Wireshark per la cattura dei pacchetti è disponibile una guida al link: *gitlab.com/wireshark/wireshark/-/wikis/CaptureSetup*.

FIGURA 15 Finestra Opzioni – scheda Input



Nella Figura 15 si ha la possibilità di selezionare o meno la **modalità promiscua** su tutte le interfacce. Quando si seleziona la modalità promiscua, la scheda di rete invia alla CPU tutto il traffico che riceve, non solo i frame indirizzati al computer su cui è installata.

Prima di avviare la cattura, è possibile impostare dei **filtri** sui dati da catturare scegliendo tra quelli elencati (FIGURA 16) o cliccando su **Gestisci filtri di cattura**, per operazioni di cancellazione o inserimento di un nuovo filtro.

FIGURA 16 Elenco dei filtri applicabili alla cattura dei dati

Una volta avviata la cattura dei pacchetti sull'interfaccia selezionata nella scheda di Input delle Opzioni, compare una schermata che visualizza i pacchetti che transitano su questa interfaccia. Un esempio è mostrato nella FIGURA 17.

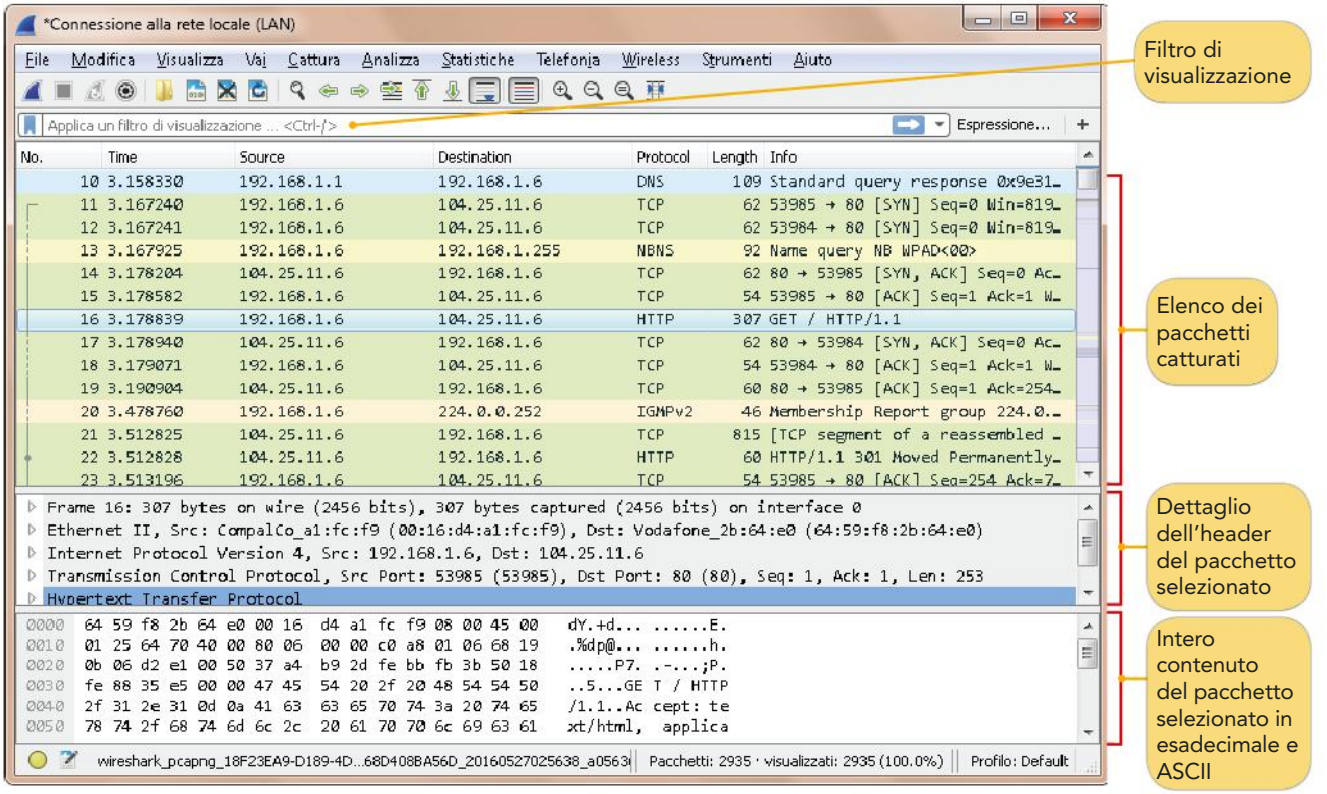
Come si può notare Wireshark utilizza un meccanismo di colorazione dei pacchetti per facilitarne l'individuazione.

La schermata mostrata nella Figura 17 presenta tre zone dall'alto verso il basso:

1. l'area colorata dei pacchetti: qui si può selezionare con il mouse una riga e analizzare il pacchetto in dettaglio nelle due zone successive;
2. l'area col dettaglio del pacchetto selezionato, nella quale si possono leggere i campi del frame Ethernet e del pacchetto IP; nell'ultima riga è riportato il protocollo del livello superiore. Si noti che eventuali dati calcolati da Wireshark, che non fanno parte del reale contenuto del pacchetto, sono racchiusi tra parentesi quadre;
3. l'area che visualizza il contenuto del pacchetto selezionato, in esadecimale e raw (ASCII).

In particolare, nella prima zona (quella colorata del punto 1) le colonne forniscono informazioni su:

- il numero progressivo del pacchetto dall'inizio della cattura (esempio: 10);
- il tempo trascorso dall'avvio della cattura, mostrato in secondi, con una risoluzione al microsecondo (esempio: 3.158330);
- gli indirizzi IP della sorgente e del destinatario;



- il protocollo in uso (è sempre indicato il protocollo di livello più alto, per esempio DNS);
- la lunghezza in byte;
- le informazioni principali estratte dall'header del pacchetto.

FIGURA 17 Esempio di cattura di pacchetti sull'interfaccia LAN

In base all'analisi che si vuol fare sui pacchetti, è anche possibile cambiare l'ordinamento, crescente o decrescente, dei dati visualizzati, cliccando sul nome della colonna interessata.

Nell'esempio mostrato nella Figura 17, il pacchetto selezionato è un messaggio GET del protocollo HTTP e nella zona centrale sono elencati tutti i protocolli utilizzati nei vari livelli dello stack TCP/IP: Ethernet, IP, TCP e, infine, HTTP. Selezionando la freccia a fianco di ciascun protocollo, compare il dettaglio delle informazioni a esso relative.

FISSA LE CONOSCENZE

- Descrivi lo schema di funzionamento di Wireshark.
- Quali librerie sono usate per la cattura dei pacchetti?
- Quali interfacce offre Wireshark all'utente?
- Come funziona la cattura in modalità promiscua?
- Spiega la differenza tra un filtro di cattura e un filtro di visualizzazione.
- Quali informazioni sono visualizzate per ciascun pacchetto catturato?

6 LAVORARE CON WIRESHARK

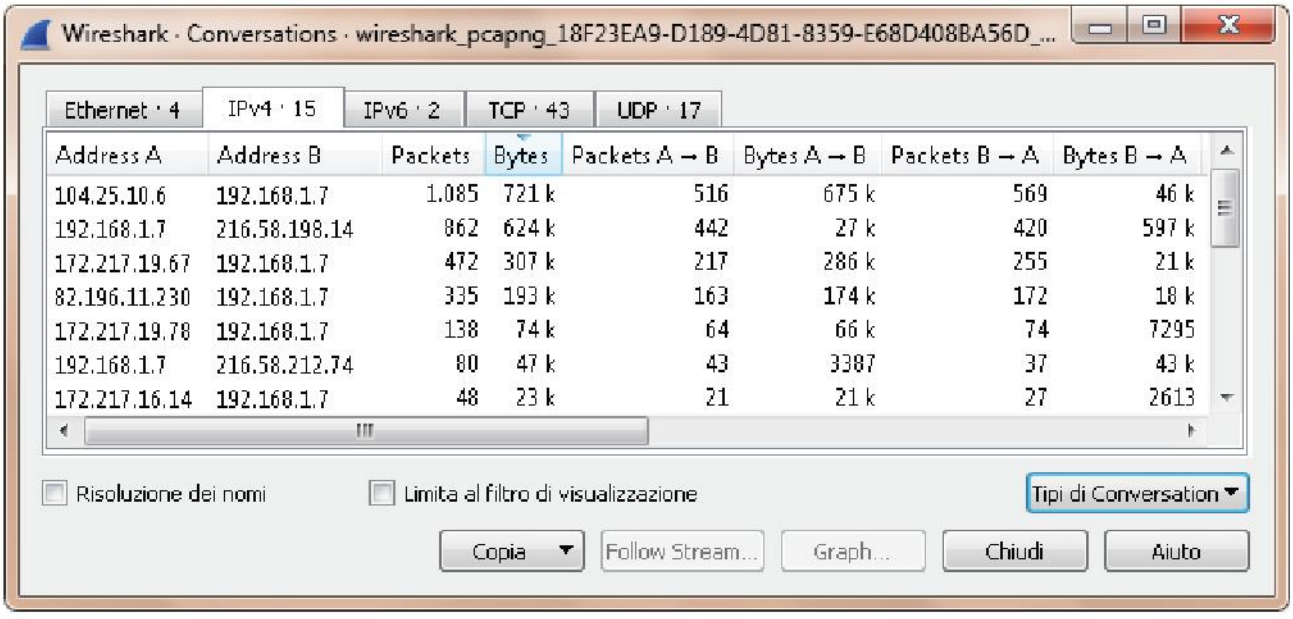
6.1 L'uso dei filtri

Supponiamo di aver attivato la cattura dei pacchetti scambiati tra il nostro computer, che ha indirizzo IP 192.168.1.7, e il web server che ospita il sito di Wireshark. Una volta avviata la cattura, abbiamo digitato nel browser l'url: *www.wireshark.org* e appena visualizzata la pagina abbiamo fermato la cattura.

Per focalizzare la nostra analisi solo sui pacchetti che riguardano il caricamento di questa pagina web, la cosa più semplice da fare è trovare l'indirizzo IP del web server e filtrare tutti i pacchetti tranne quelli che contengono questo indirizzo. Un metodo semplice è identificare tra le tante "conversazioni" catturate quella che è più attiva (ovvero che ha generato più traffico) e rilevare gli indirizzi IP coinvolti.

Dal menu di Wireshark selezionare: **Statistiche** → **Conversazioni**, quindi cliccare su IPv4 nella finestra che si apre. Dovendo cercare la conversazione più attiva (cioè i cosiddetti **top talkers**) cliccare due volte su **Bytes** così da ottenere i dati ordinati in ordine decrescente. Il risultato è mostrato nella FIGURA 18. La conversazione che ha generato più traffico sull'interfaccia LAN oggetto dell'analisi è quella con indirizzo sorgente 104.25.10.6, in particolare si osserva che il numero di byte più elevato è nella direzione A → B, quindi potremmo ragionevolmente dedurre che la macchina A, che ha indirizzo IP 104.25.10.6, sia proprio il web server dal quale abbiamo scaricato la home page del sito Wireshark.

FIGURA 18 Analisi delle conversazioni relative alla cattura effettuata



Per avere, però, la certezza che quella al top dell'elenco sia proprio la conversazione che cerchiamo, l'unico modo è risolvere l'indirizzo IP 104.25.10.6 e questo può essere fatto da Wireshark attivando l'opzione **Network Name Resolution** che permette di interrogare il DNS (Domain Name System) effettuando la cosiddetta "risoluzione inversa". Seguire quindi il percorso da menu: **Visualizza** → **Risoluzione dei nomi** e selezionare **Risolvi gli indirizzi di rete**.

Riaprire la finestra delle Conversazioni come fatto prima: ora compare la spunta nella casella in basso **Risoluzione dei nomi**.
Il risultato è mostrato nella FIGURA 19; come si può notare, la conversazione che ha generato maggior traffico è proprio la prima, quindi possiamo usare l'indirizzo IP 104.25.10.6 per filtrare i pacchetti da analizzare, sapendo che corrisponde al web server del sito *wireshark.org*, come mostrato nella FIGURA 20 (usare il tasto destro del mouse per visualizzare il menu contestuale).

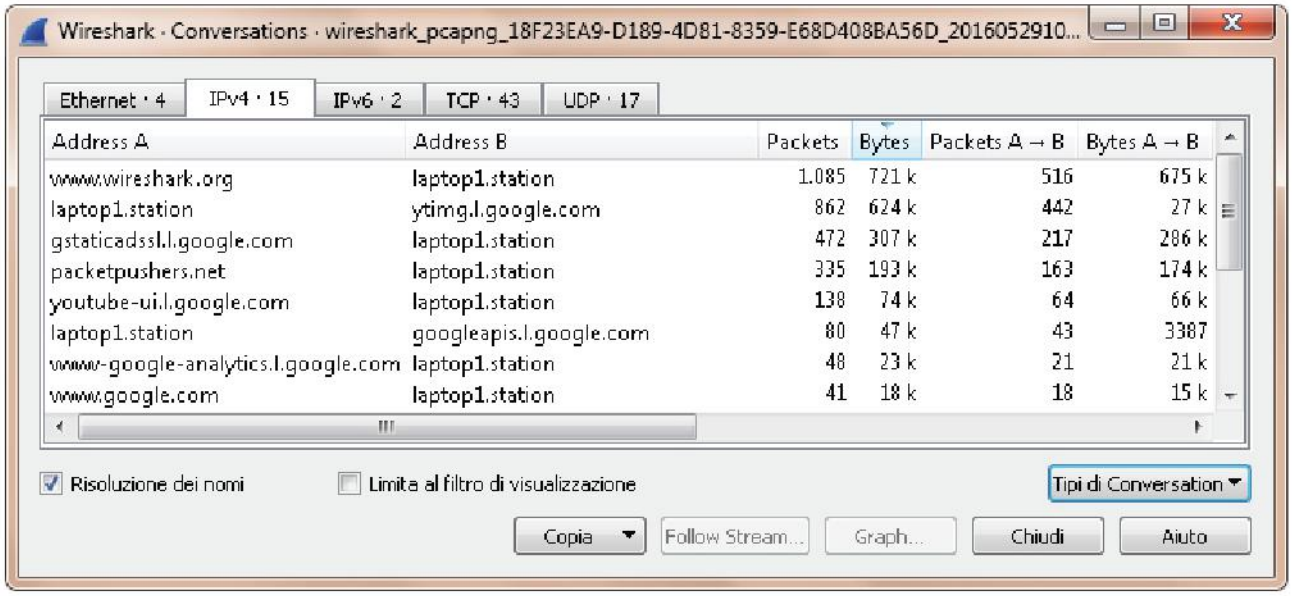


FIGURA 19 Analisi delle conversazioni con i nomi al posto degli indirizzi IP

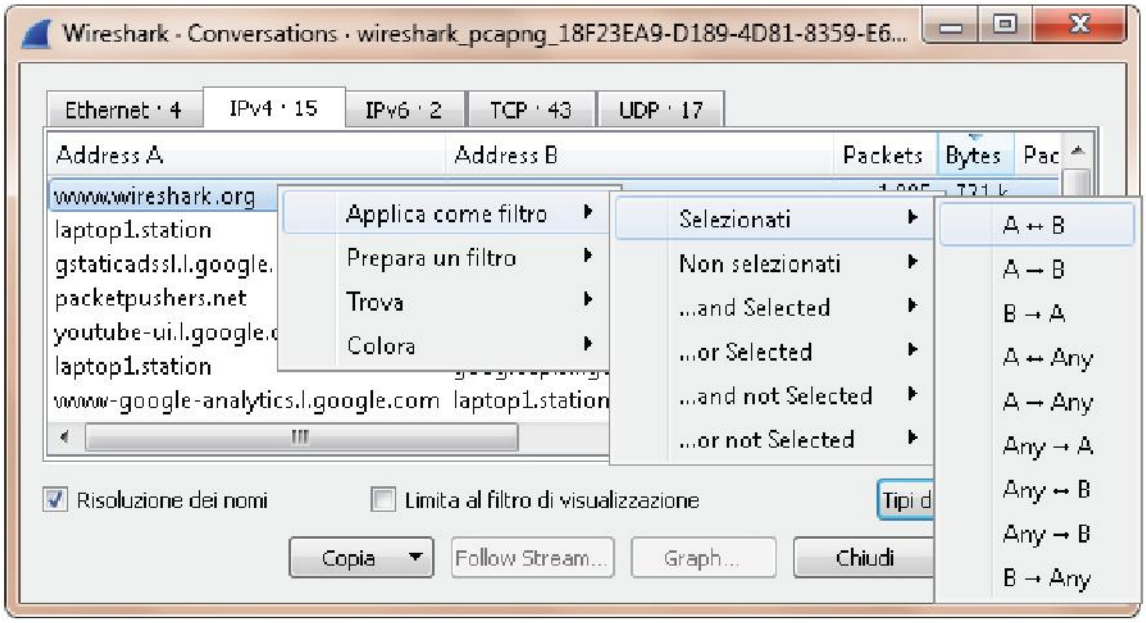


FIGURA 20 Applicazione del filtro per la conversazione selezionata

Wireshark applica il filtro, così definito, al traffico catturato e nella finestra verranno visualizzati solo i pacchetti scambiati tra il nostro computer e l'indirizzo IP di *wireshark.org*. Da notare nella FIGURA 21 la stringa del filtro generata automaticamente sulla base delle scelte effettuate: **ip.addr==104.25.10.6 && ip.addr==192.168.1.7**.

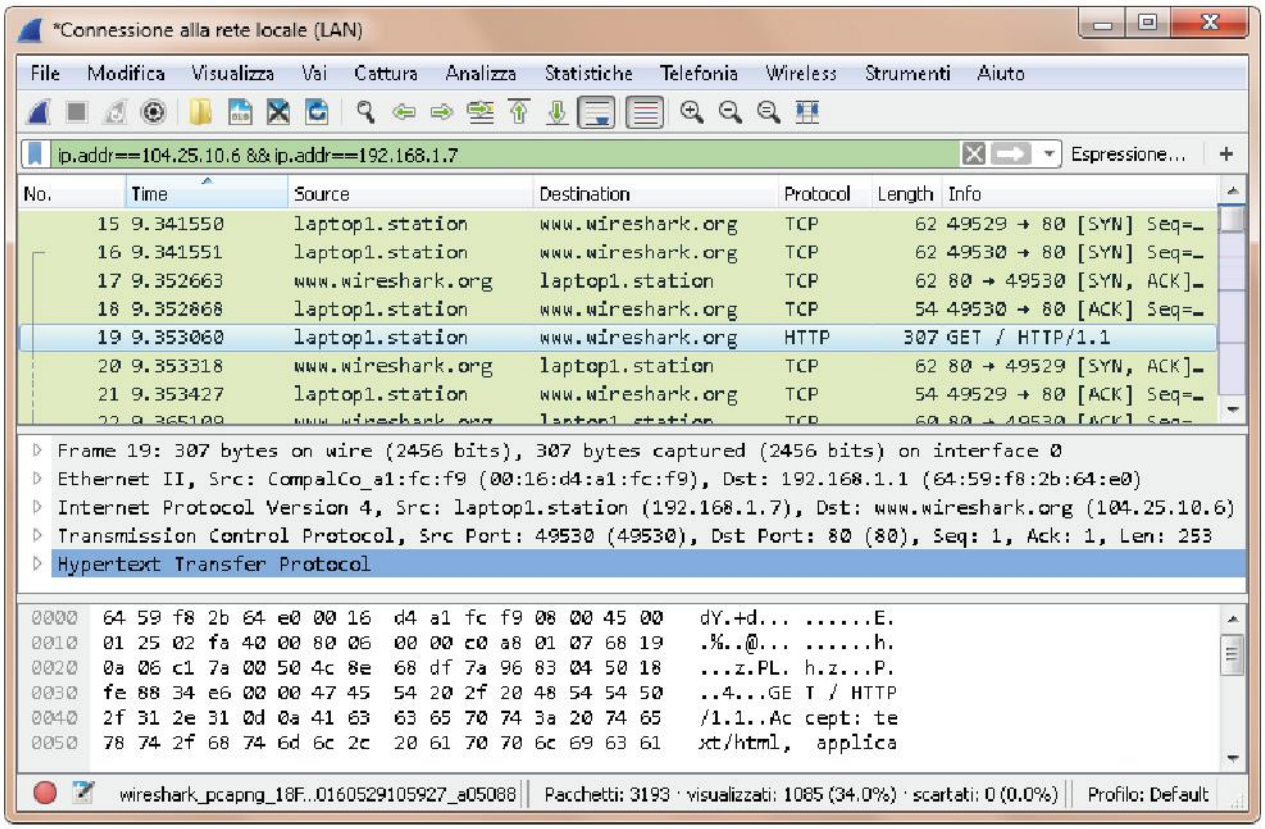


FIGURA 21 Risultato dell'applicazione del filtro ai pacchetti catturati

Per salvare i soli pacchetti filtrati e avere il tracciato della conversazione di interesse da poter analizzare in un secondo tempo, si deve selezionare **File → Esporta i pacchetti specificati**. Il file verrà salvato con l'estensione di default **.pcapng**.

6.2 La gestione del timestamp

#techwords

Timestamp
Letteralmente "marcatura temporale", è una sequenza di caratteri che rappresentano un orario (o una data), usata per segnare l'avvenimento di un certo evento.

Nella videata di Wireshark relativa alla cattura, a ogni pacchetto è assegnato un **#timestamp**, un'informazione particolarmente utile quando si vuole analizzare il flusso dei pacchetti e individuare eventuali anomalie legate al tempo.
Come scritto in precedenza, di default Wireshark visualizza nella colonna Time i secondi che sono trascorsi dall'avvio della cattura dei pacchetti, con risoluzione al microsecondo. In questo caso, il primo pacchetto avrà un timestamp pari a 0.000000.
Se si vuole modificare questa vista, si seleziona dal menu **Visualizza → Formato di visualizzazione del tempo**, il formato desiderato, scegliendo tra quelli proposti.
In alternativa, si può aggiungere una seconda colonna che usa un timestamp diverso da quello di default:

1. dal menu **Modifica → Preferenze**, selezionare la voce **Colonne** e cliccare su + ;
2. si crea così una nuova riga, cliccare su **Nuova colonna** e dare un titolo alla nuova colonna, quindi selezionare dal menu a discesa **Number** il tipo desiderato, per esempio il giorno e l'ora in cui è stato catturato il pacchetto (tempo assoluto e non più relativo all'inizio della cattura);
3. cliccare sulla nuova riga e trascinarla verso l'alto nella posizione desiderata.

La **FIGURA 22** mostra un esempio di questa procedura, è stata infatti creata una nuova colonna **Data e Ora** del tipo **UTC date and time**.

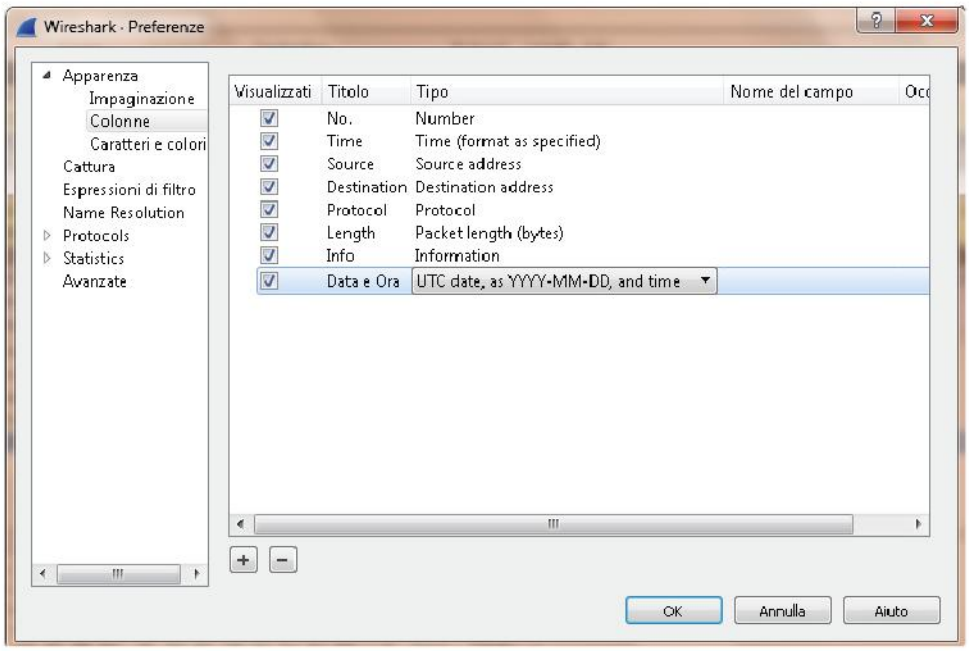


FIGURA 22 Aggiunta di una nuova colonna per data e ora "assoluta"

6.3 Le regole di colorazione dei pacchetti

Wireshark ha alcune regole predefinite per la colorazione dei pacchetti visualizzati nella schermata di cattura. Queste regole sono visibili e modificabili selezionando **Visualizza → Regole di Colorazione** (FIGURA 23).

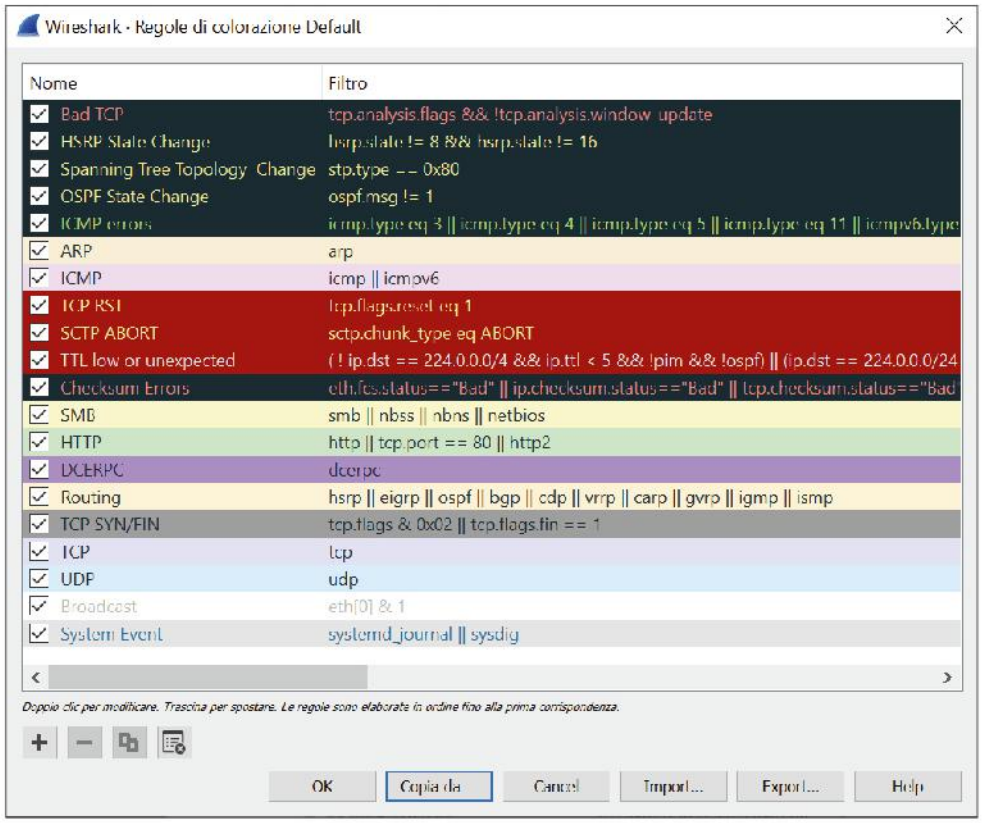


FIGURA 23 Regole di colorazione

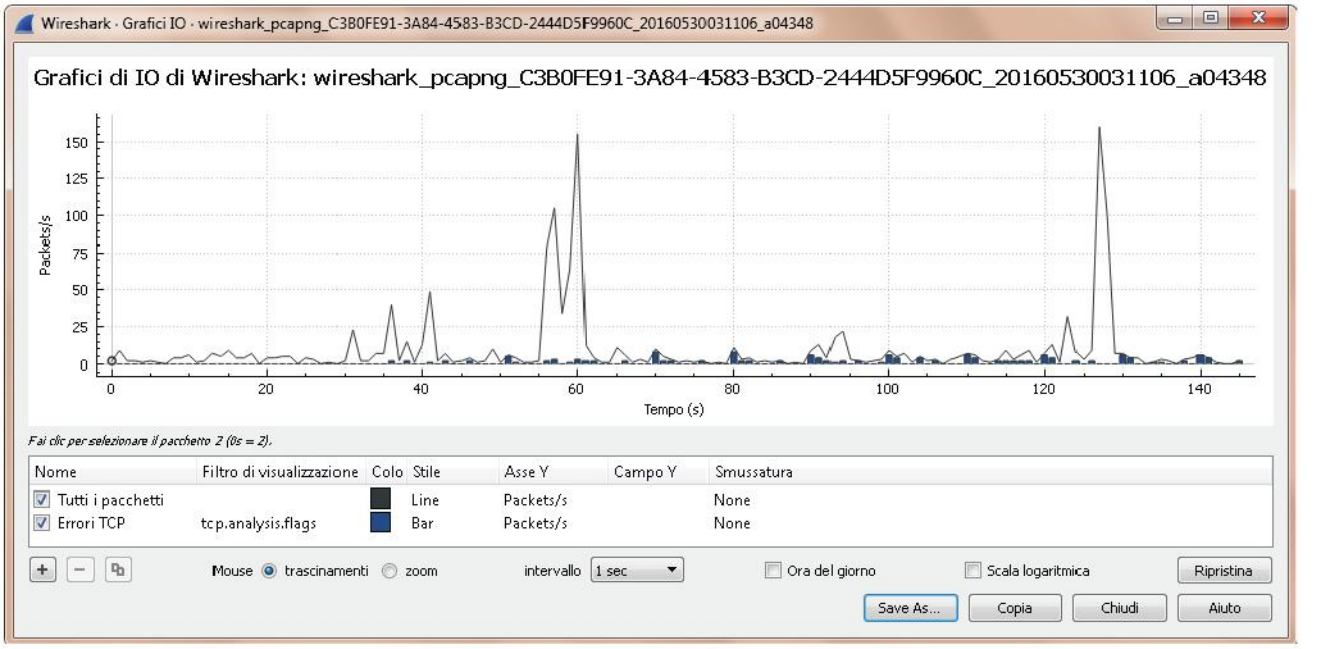
È possibile importare ed esportare le regole di colorazione, esse infatti sono memorizzate in un file chiamato **colorfilters** che si trova nella directory contenente i file di Wireshark.

6.4 I grafici di Input/Output

Wireshark fornisce un'interessante funzionalità che permette una visione grafica dei pacchetti catturati, particolarmente utile per controllare la quantità di dati che transita attraverso la rete.

Per generare questa vista grafica si seleziona dal menu **Statistiche** → **Grafici I/O** e si ottiene una videata come quella mostrata nella FIGURA 24.

FIGURA 24 Grafici I/O



Sull'asse X è rappresentato il tempo in secondi e sull'asse Y è visualizzato il numero di pacchetti rilevati in un secondo.

L'analisi del grafico è interessante per rilevare l'andamento del traffico; in particolare i picchi possono fornire indicazioni utili a individuare anomalie.

FISSA LE CONOSCENZE

- Spiega come si gestiscono le conversazioni.
- Che cosa sono i *top talkers*? come si possono visualizzare con Wireshark?
- A che cosa serve l'opzione *Network Name Resolution*?
- Che cos'è il *timestamp*? Come è gestito in Wireshark?
- Perché Wireshark ha definito delle regole di colorazione?
- Descrivi la funzione *Grafici I/O* del menu Statistiche.

7 CISCO PACKET TRACER: UN SIMULATORE DI RETE

7.1 La simulazione di una rete

Per poter svolgere delle attività pratiche, che consentano di capire il funzionamento di una rete, le problematiche da affrontare e le possibili soluzioni, sarebbe necessario avere a disposizione un laboratorio ben attrezzato con tutte le tipologie di apparati, mezzi trasmissivi e software, in numero tale da poter realizzare topologie fisiche e logiche significative, sia in ambito locale sia geografico. In mancanza di un'attrezzatura reale, si può usare un software per la simulazione di reti.

> Un **simulatore di rete** è un'applicazione che dispone di una serie di componenti, protocolli, meccanismi di gestione utili a riprodurre il funzionamento di una rete reale complessa. È in grado di **emulare** varie tipologie di apparati e sistemi di rete, permette di **creare** complesse topologie con schemi grafici e **verificare** come fluisce il traffico nella rete realizzata.

I simulatori sono strumenti molto utili non solo per gli studenti che devono acquisire competenze sulle reti, ma anche per progettisti e amministratori, in quanto permettono di testare topologie e configurazioni prima di implementarle sulla rete reale. È un'attività essenziale da svolgere quando, per esempio, si deve espandere la rete o offrire un nuovo servizio: verificare se la rete, così com'è, è in grado di supportarlo o, al contrario, è necessario attuare modifiche e riconfigurazioni, permette di risolvere molti problemi prima che i cambiamenti diventino operativi.

Come scritto sopra, le applicazioni per la simulazione di una rete emulano anche gli elementi reali che compongono la rete. Ci sono software che svolgono esclusivamente questa funzione e sono detti **emulatori**.

> Un **emulatore di rete** è un'applicazione che "imita" l'apparato reale e consente di svolgere configurazioni e test del sistema come se si lavorasse sul dispositivo reale.

È particolarmente utile quando si vuole provare un nuovo apparato prima di inserirlo in rete. A questo scopo, molti produttori mettono a disposizione nei loro siti web programmi di emulazione dei loro prodotti.
Esempi di emulatori si trovano anche al di fuori delle reti, basti pensare agli emulatori Android che permettono di utilizzare questo sistema operativo, per smartphone e tablet, su un computer Windows (per esempio: *Android Studio* di Google).
La simulazione al computer di scenari di rete reali permette di analizzarne le potenzialità e di caratterizzarne i comportamenti secondo vari indicatori. Infatti, aiuta a rispondere a molte domande che possono emergere sia nella fase di progettazione della rete sia nella fase operativa:

- quante linee di trasmissione si devono allocare per soddisfare le richieste degli utenti?
- quali prestazioni richiede alla rete l'implementazione di un nuovo protocollo?
- quale impatto avrà sulla rete l'aggiunta di un nuovo servizio?
- quali benefici porterà agli utenti l'ammodernamento della rete con nuovi apparati?

#prendinota

Emulatori vs simulatori
Le applicazioni attualmente utilizzate per creare topologie di rete e verificarne la correttezza e le prestazioni, hanno spesso sia funzioni di emulazione che di simulazione. *Packet Tracer* di Cisco ne è un esempio.
La sostanziale differenza tra i due strumenti è che nell'emulatore il software dell'apparato emulato è lo stesso di quello reale, mentre nel simulatore è completamente riscritto.

In generale, con l'attività di simulazione un amministratore può:

- verificare un ipotetico scenario di rete prima di metterlo realmente in funzione;
- verificare il comportamento della rete sulla base di indici prestazionali;
- confrontare soluzioni di rete a parità di condizioni, attività non semplice da effettuare nella realtà in quanto sono molte le variabili che potrebbero rendere inattendibile il confronto (numero di utenti, stato degli apparati, ecc.);
- svolgere analisi di tipo **what if** (*che cosa accadrebbe se...?*), per fare previsioni sul comportamento della rete in condizioni che si potrebbero verificare durante la sua normale operatività;
- conoscere l'evoluzione del traffico di rete in specifiche condizioni.

Queste attività vengono svolte, grazie alla simulazione, con una complessità inferiore rispetto a uno studio teorico, ma anche con tempi e costi inferiori rispetto a un'analisi effettuata direttamente su una rete reale.

7.2 Alcuni simulatori di rete utili per la didattica

I seguenti software offrono funzionalità sia di emulazione di apparati hardware, quali router e switch, sia di simulazione del funzionamento della rete.

Packet Tracer nasce come software didattico, studiato per mettere in pratica quanto appreso nei corsi Cisco. In realtà è usato anche da molti amministratori di rete per testare topologie di rete e nuovi apparati e servizi. In passato era disponibile solo per gli studenti iscritti al programma Network Academy di Cisco, ora è utilizzabile gratuitamente da chiunque, previa registrazione. Packet Tracer è disponibile anche in versione mobile per iOS e Android.

Boson NetSim Network Simulator è un simulatore per reti con router e switch Cisco. L'applicazione, creata per aiutare a superare gli esami di certificazione Cisco, è a pagamento, ma è disponibile una licenza demo, limitata, scaricabile dal sito *www.boson.com*.

NetSim™
NETWORK SIMULATOR™

GNS3 (**G**raphical **N**etwork **S**imulator-**3**) è un software open source composto da una parte client con GUI, per creare le topologie di rete, e da una parte server, la GNS3 Virtual Machine. Il suo punto di forza è la possibilità di lavorare con le immagini dei sistemi reali (intesi come i "sistemi operativi" degli apparati), che possono poi essere usate in un ambiente virtuale.
Oltre a quelli Cisco, GNS3 è in grado di emulare apparati di rete anche di altri produttori. L'applicazione, nelle versioni per Windows, Linux e Mac-OS, è scaricabile dal sito *www.gns3.com*.

7.3 Cisco Packet Tracer

Le esercitazioni che vedremo in alcune unità di questo volume saranno svolte con il simulatore **Packet Tracer** (**PT**) di Cisco, **versione 7.3.1** eseguito sul sistema Windows. Packet Tracer può essere eseguito anche su Linux e Mac-OS. Come accennato in precedenza, c'è anche l'applicazione per i dispositivi mobili Android e iOS.
Il software è gratuito, ma per averlo è necessario essere registrati a un corso della **Cisco Network Academy**.

Attualmente, il corso di introduzione a Packet Tracer è accessibile gratuitamente a questo link: *www.netacad.com/courses/packet-tracer*.
Effettuata la registrazione, è possibile eseguire il download del software. Ogni volta che si avvia l'applicazione sono richieste le credenziali di accesso. Nella **FIGURA 25** si mostra la pagina di login: l'accesso a tutto il mondo Cisco è stato unificato con il servizio Cisco OneID. Cliccando sul simbolo ⊕ è possibile cambiare la lingua da inglese (default) a italiano.

FIGURA 25 Cisco One Identity, pagina di accesso

LE CARATTERISTICHE PRINCIPALI

Packet Tracer permette di creare scenari di rete su cui sviluppare svariati tipi di analisi: le più interessanti sono quelle legate al verificarsi di eventi, per esempio l'invio di un messaggio, lo scadere di un timer, la ricezione di un pacchetto.
Infatti, Packet Tracer consente di:

- creare topologie fisiche, scegliendo tra i vari apparati disponibili quelli di interesse, realizzando i cablaggi e configurando l'hardware dei dispositivi;
- creare topologie logiche, con schemi di indirizzamento IP e VLAN (Virtual LAN);
- configurare i dispositivi (terminali di accesso e apparati di rete) tramite maschere grafiche e finestre di dialogo;
- svolgere una configurazione avanzata sugli apparati Cisco, da linea di comando (CLI);
- simulare operazioni eseguibili con Personal Computer e server, come l'ambiente Command Prompt di Windows;
- simulare servizi lato server (DHCP, web server, email server, ecc.);
- simulare e tracciare i percorsi dei pacchetti nella rete (packet sniffer) relativi ai principali servizi e protocolli di rete.

Packet Tracer ha un'**interfaccia grafica** semplice da utilizzare ed è **multiutente** per permettere la collaborazione Peer-to-Peer tra più studenti.

Fino alla versione 7.2.2, Packet Tracer non era in grado di interagire con apparati reali, funzionalità disponibile, invece, in altri simulatori come GNS3. Da questa versione in poi, è possibile interagire con i router virtuali **Cisco CSR**.
La comunicazione tra i router avviene tramite un nuovo protocollo, denominato **PTTP**, che Cisco ha registrato presso IANA a giugno 2019.

#prendinota

Cisco Cloud Service Router (CSR)
Sono router realizzati su una macchina virtuale in cloud. Possono quindi essere configurati secondo le esigenze del cliente e usati come router reali.

LE TECNOLOGIE DI RETE CHE POSSONO ESSERE SIMULATE

Packet Tracer offre la possibilità di lavorare con varie tecnologie di rete.
Nella TABELLA 1 sono elencati i protocolli supportati, suddivisi secondo lo stack protocollare TCP/IP.

TABELLA 1 I protocolli supportati in Packet Tracer

LAYER	PROTOCOLLI SUPPORTATI
Application	FTP, SMTP, POP3, HTTP, TFTP, Telnet, SSH, DNS, DHCP, NTP, SNMP, AAA, ISRVOIP, MQTT, SCCP config and calls ISR command support, Call Manager Express.
Transport	TCP, UDP, TCP Nagle Algoritm & IP Fragmentation, RTP.
Network	BGP, IPv4, ICMP, ARP, IPv6, ICMPv6, IPSec, RIPv1/v2/ng, Multi-Area OSPF,OSPFv3, EIGRP, EIGRPv6, Static Routing, Route Redistribution, Multilayer Switching, L3 QoS, NAT, CBAC, Zone-based policy firewall and Intrusion ProtectionSystem on the ISR, GRE VPN, IPSec VPN, HSRP, CEF, SPAN/RSPAN, L2NAT, PTP, REP, LLDP.
Physical	Ethernet (802.3), 802.11, HDLC, Frame Relay, PPP, PPPoE, STP, RSTP, VTP, DTP, CDP, 802.1q, PAgP, L2 QoS, SLARP, Simple WEP, WPA, EAP, VLANs, CSMA/CD, EtherChannel, DSL, 3/4 G network support.

7.4 L'ambiente di sviluppo: i menu di Packet Tracer

All'avvio del programma, si apre all'interno della classica finestra di Windows un'area di lavoro dove è possibile creare un'infrastruttura di rete e simularne il funzionamento.
Dalla versione 7.2, Packet Tracer ha una nuova interfaccia grafica che analizziamo tramite una serie di screenshot.
La FIGURA 26 mostra la pagina iniziale di presentazione dell'applicazione Packet Tracer.

FIGURA 26 Cisco Packet Tracer

A seguire, compare la finestra principale (FIGURA 27) con il menu e le barre degli strumenti (**#toolbar**):

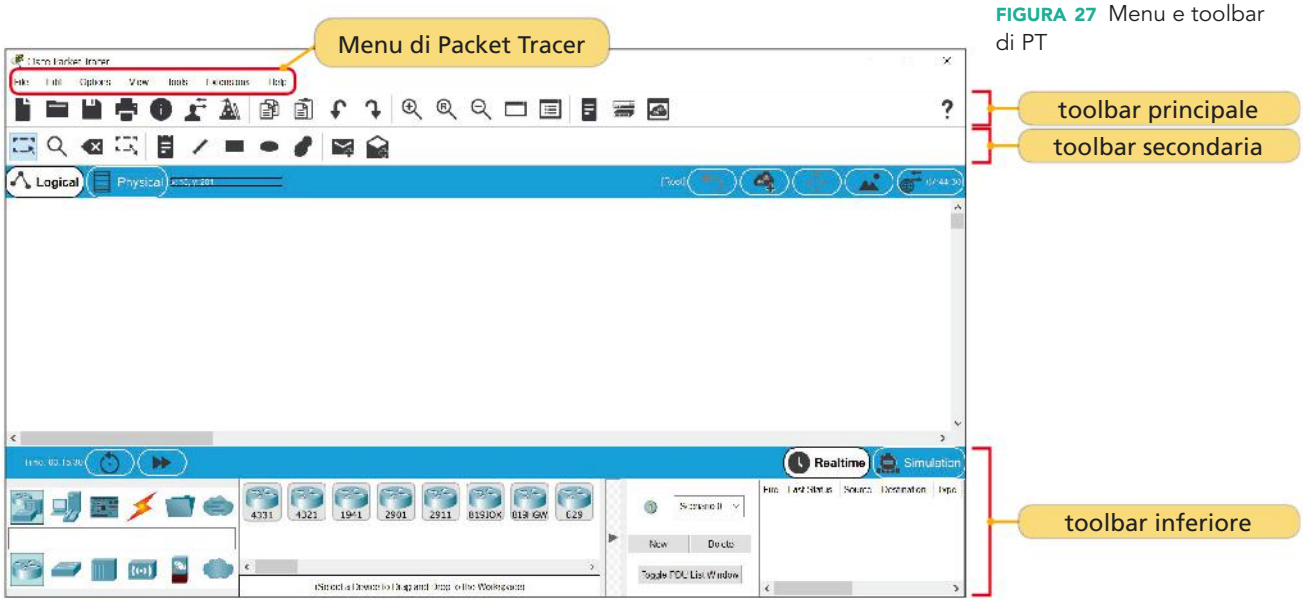


FIGURA 27 Menu e toolbar di PT

Il **menu** di PT, evidenziato dal contorno rosso nella Figura 27, prevede le seguenti voci:

- **File**: sono presenti i comandi per aprire, salvare e stampare i file generati con PT;
- **Edit**: include i comandi per duplicare i dispositivi, annullare le ultime modifiche e ripetere le modifiche annullate;
- **Options**: comprende i comandi per scegliere impostazioni personalizzate;
- **View**: offre la possibilità di scegliere le barre degli strumenti che si vogliono visualizzare; permette di passare dalla vista Realtime a quella Simulation; inoltre include i comandi per lo zoom;
- **Tools**: include uno strumento per aggiungere disegni sullo sfondo (utile per evidenziare le sottoreti) e uno per personalizzare i dispositivi;
- **Extensions**: comprende i comandi per l'utilizzo multiutente e l'Activity Wizard;
- **Help**: fornisce i link per visualizzare i file di aiuto e i tutorial.

Subito al di sotto del menu principale troviamo la **toolbar principale** che presenta, sotto forma di icone, alcuni comandi veloci già proposti nel menu precedente:

- creare, aprire, salvare e stampare il file;
- aggiungere una descrizione alla rete;
- copiare e incollare un oggetto;
- annullare e ripetere le ultime modifiche;
- aggiustare lo zoom secondo le proprie necessità;
- aggiungere disegni sullo sfondo;
- modificare un dispositivo nell'area di lavoro.

Al limite destro di questa toolbar c'è il simbolo ? che fornisce l'accesso veloce alla sezione Help.

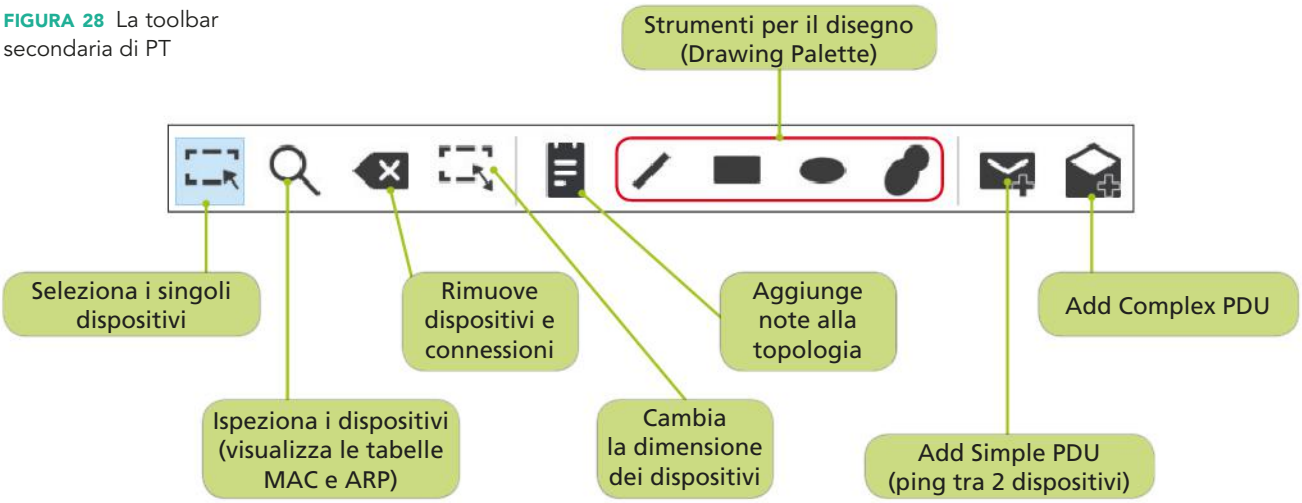
#techwords

Toolbar
È la barra degli strumenti a disposizione dell'utente di un'applicazione. La barra può essere orizzontale o verticale e raccoglie, sotto forma di icone, i collegamenti alle funzioni più usate. L'interfaccia grafica di un'applicazione può offrire una o più toolbar.

LA TOOLBAR SECONDARIA

Sotto la toolbar principale troviamo la toolbar secondaria (FIGURA 28) con i pulsanti per selezionare, spostare e cancellare gli oggetti presenti nell'area di lavoro, inserire note e tracciare disegni.

FIGURA 28 La toolbar secondaria di PT



#techwords

Ping
È un software di amministrazione molto usato nelle reti di computer per verificare la raggiungibilità di un dispositivo. Da un host viene inviato un messaggio Echo Request a un host di destinazione, il quale risponderà, se è in rete, con un messaggio Echo Reply. I messaggi appartengono al protocollo ICMP, descritto nell'Unità 4.

Nella toolbar secondaria, cliccando sulle icone a forma di busta è possibile simulare il test di connettività comunemente conosciuto come **#ping**, nelle due versioni semplice e avanzato.
Per eseguire un test semplice, si clicca sull'icona a forma di **busta chiusa (Simple PDU)**, poi si seleziona all'interno dell'area di lavoro la sorgente del test (apparato di rete o host) e infine si seleziona la destinazione di cui si vuole verificare la raggiungibilità (interfaccia, apparato di rete oppure host).
L'icona a forma di **busta aperta (Complex PDU)** permette di impostare alcuni parametri per il test ping e scegliere se svolgerlo una sola volta oppure se la PDU deve essere inviata periodicamente secondo un intervallo di tempo specificato. È anche possibile cambiare il tipo di PDU, selezionandolo da un elenco di protocolli di livello Application (vedi Tabella 1).

LOGICAL E PHYSICAL WORKSPACE

#techwords

Workspace
È un termine usato per indicare uno spazio in cui organizzare gli elementi su cui si lavora insieme ad altre risorse utili, così da avere tutto in un unico ambiente di progettazione.

Nelle FIGURE 29 e 30 è messa in evidenza la barra di colore azzurro, posizionata sotto la toolbar secondaria, che permette di personalizzare la visualizzazione dello schema di rete in progetto e di passare dalla topologia logica a quella fisica.
In Packet Tracer sono disponibili due ambienti di lavoro (#workspace) in cui creare le topologie logiche e fisiche della rete e in cui visualizzare le simulazioni.
Nella **topologia logica (Logical Workspace)** si crea lo scenario di rete inserendo i dispositivi e collegandoli con i cavi opportuni o in wireless. Questo è l'ambiente di lavoro tipico di chi usa Packet Tracer, in cui creare e configurare la rete oggetto della simulazione.

FIGURA 29 Barra della topologia logica

FIGURA 30 Barra della topologia fisica

Nella **topologia fisica (Physical Workspace)** lo scenario che si crea è calato nella realtà di edifici e città. In questo caso, nella valutazione delle prestazioni della rete, Packet Tracer terrà conto anche delle reali distanze tra i dispositivi. Ogni dispositivo creato è inserito in un contenitore che può essere il rack dell'armadio di piano o la scrivania presenti in un edificio.

LA TOOLBAR INFERIORE

La FIGURA 31 mostra la toolbar posta nella parte inferiore della finestra principale di PT.

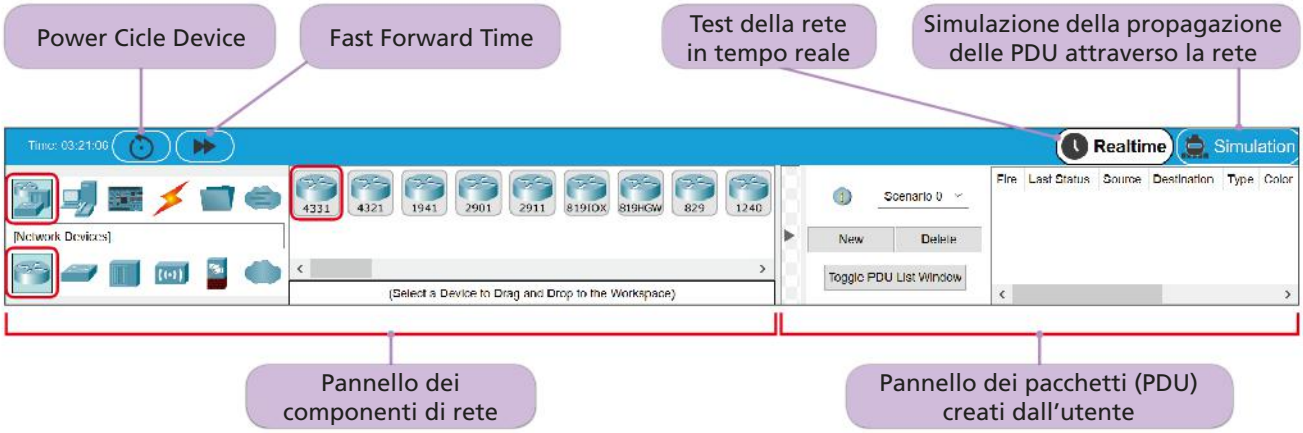


FIGURA 31 La toolbar inferiore di PT

La toolbar inferiore è divisa in due parti principali:

- a sinistra c'è il pannello per la selezione dei diversi componenti (**device**) presenti su una rete informatica, composto da due box affiancati:
  - il primo box, in alto, permette di selezionare la **categoria** del device da creare e, in basso, il **tipo**; nell'esempio di Figura 31 abbiamo selezionato la categoria "Network Device" e come tipo abbiamo scelto il router;
  - nel box a lato si sceglie il **device specifico** tra quelli mostrati, nel nostro esempio abbiamo selezionato il router 4331; per posizionare l'oggetto selezionato nel workspace si può usare la tecnica drug-and-drop oppure si clicca sull'oggetto e poi si clicca nel punto dell'area di lavoro dove lo si vuole posizionare;
- a destra c'è il pannello in cui sono visualizzati i pacchetti (PDU) creati dall'utente; qui è possibile gestire i diversi scenari di test creati con le PDU (simple e complex).

Nella barra superiore di colore azzurro, all'estrema destra, troviamo i due pulsanti per selezionare la modalità di gestione dei test della rete creata: **Realtime** e **Simulation**. Nella Figura 31 è selezionato Realtime, quindi, i pulsanti che vediamo all'estrema sinistra sono quelli disponibili in questa modalità:

- **Power Cycle Device**: la selezione di questo pulsante comporta il reboot immediato di tutti i device della rete realizzata, con la cancellazione delle configurazioni effettuate su router e switch (a meno che siano state salvate in modo permanente); è disponibile anche nella modalità Simulation, in cui cancella anche tutti gli eventi;
- **Fast Forward Time**: permette di far convergere la rete più velocemente: a ogni clic il realtime clock avanza di 30 secondi; è particolarmente utile quando nella rete ci sono molti switch connessi in loop: la rete diventa stabile in pochi secondi anziché impiegare qualche minuto.

LE MODALITÀ REALTIME E SIMULATION

L'attività di test della connettività tra i device e di simulazione del traffico nella rete realizzata avviene in Packet Tracer usando due differenti modalità:

- **Realtime**: si può svolgere un test (ping) in tempo reale tra i vari dispositivi coinvolti che rispondono automaticamente alla richiesta; quando la rete è in realtime è possibile ispezionare un device cliccando sulla relativa icona con lo strumento della lente;
- **Simulation**: consente la visualizzazione passo passo del flusso delle PDU legate ai vari protocolli di rete (ICMP, UDP, ecc.); lavorare con questa modalità è molto utile con topologie di rete complesse, in quanto permette di analizzare come i dati sono trasferiti nella rete e ispezionare un pacchetto nel dettaglio.

La FIGURA 32 mostra un semplice scenario di rete con due switch a cui sono collegati alcuni PC e una stampante. Abbiamo selezionato la modalità Simulation e sono comparse nuove finestre e pulsanti. La simulazione avviata consiste nell'analizzare il traffico generato da un ping tra il PC1 e il PC5.

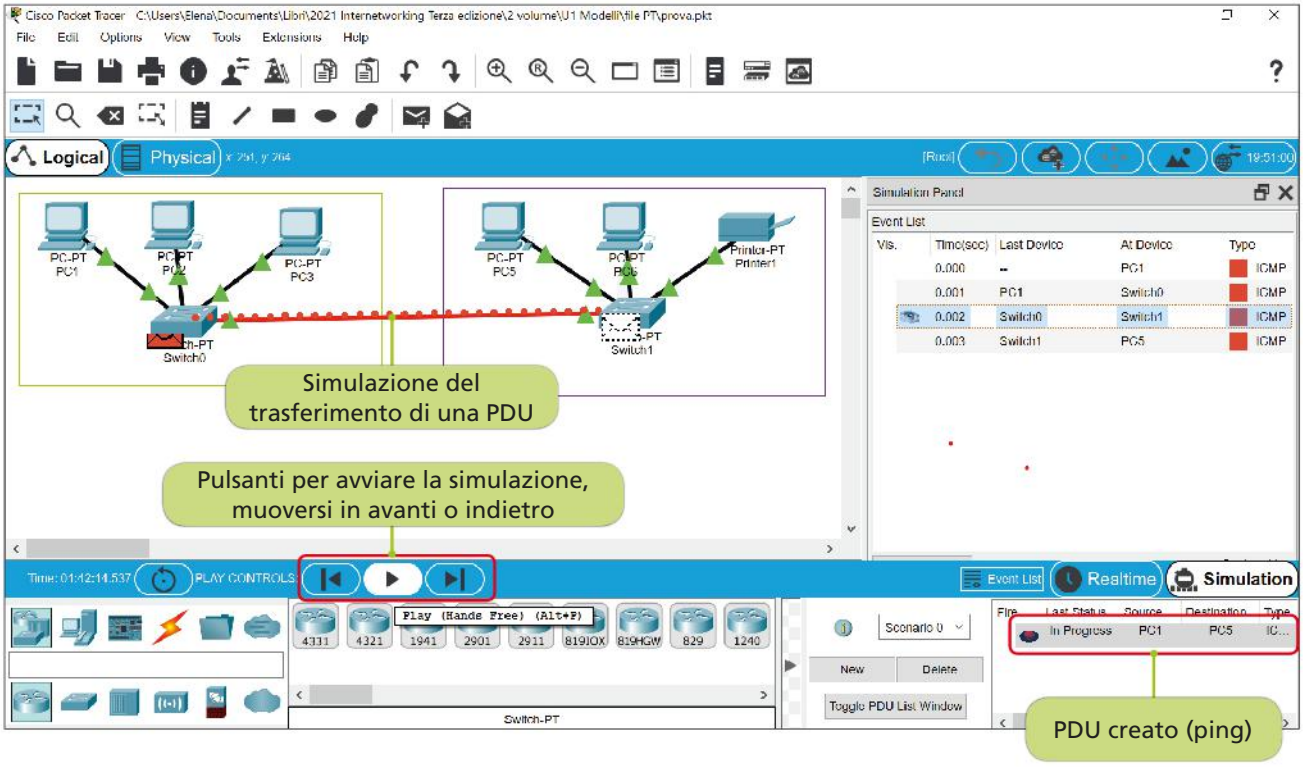


FIGURA 32 La modalità Simulation di PT

Nella parte destra troviamo il **Simulation Panel** (FIGURA 33) in cui è presente l'area **Event List** che visualizzerà le PDU generate a seguito dell'invio del ping.
Nella parte centrale ritroviamo gli stessi **pulsanti di controllo** presenti nella barra azzurra nella toolbar inferiore:

- **Go back to previous event**: permette di esaminare la situazione immediatamente precedente;
- **Play**: consente di catturare tutto il traffico della rete in base ai filtri impostati; si interrompe quando viene nuovamente cliccato;
- **Capture then forward**: permette di catturare e analizzare una PDU alla volta; ogni volta che si clicca su questo pulsante, il pacchetto si sposta da un device al successivo.

In basso, il pulsante **Edit Filters** permette di filtrare le PDU visualizzate nella Event List. Poiché abbiamo scelto di effettuare un semplice ping, scegliamo di visualizzare solo il protocollo ICMP, così da poter seguire il traffico generato dalle PDU di richiesta e di risposta.

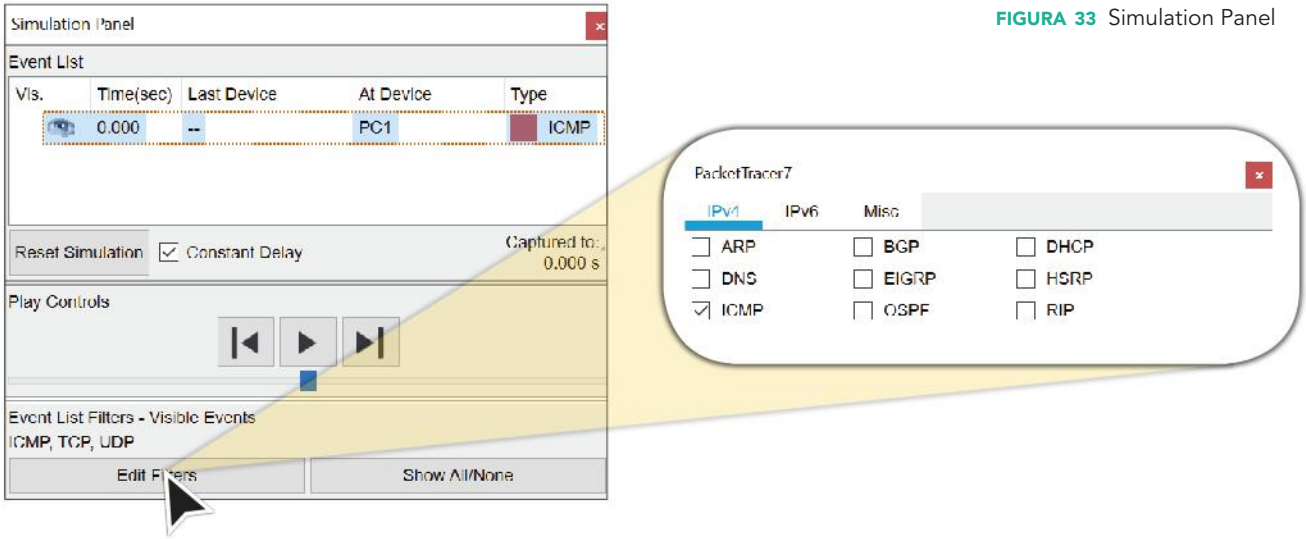


FIGURA 33 Simulation Panel

Con la modalità Simulation è possibile analizzare la singola PDU. La schermata presentata in **FIGURA 34** è ottenuta cliccando nel workspace sulla busta colorata, che rappresenta una PDU, mentre si sposta da un device all'altro.

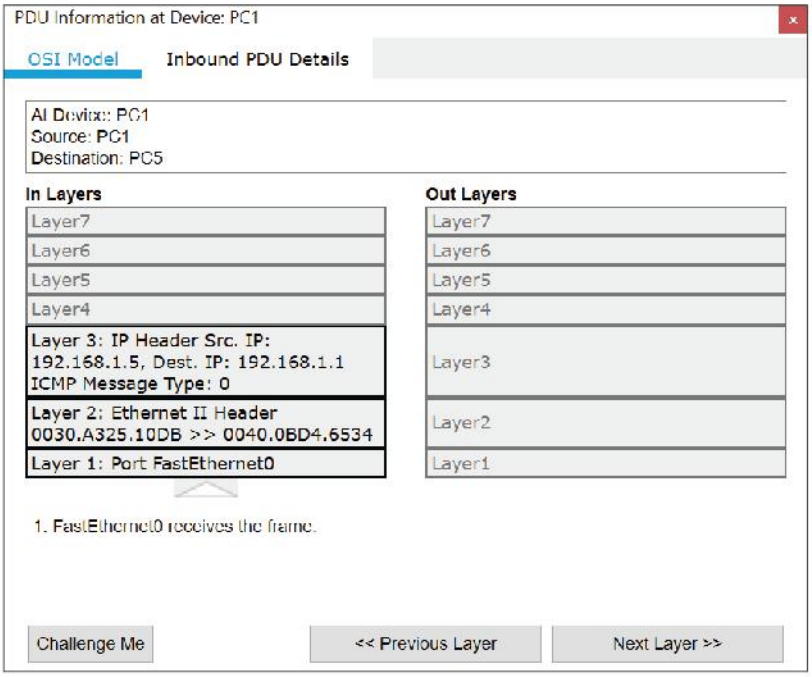


FIGURA 34 Analisi di una PDU in modalità Simulation

7.5 Lavorare con Packet Tracer

Packet Tracer prevede la creazione di **scenari di rete** su cui sviluppare analisi di tipo **what if**, permettendo quindi di studiare il comportamento della rete nel complesso, e dei singoli dispositivi nello specifico, al verificarsi di determinati eventi. Nel seguito si propongono alcuni semplici esempi per capire come lavorare con PT.

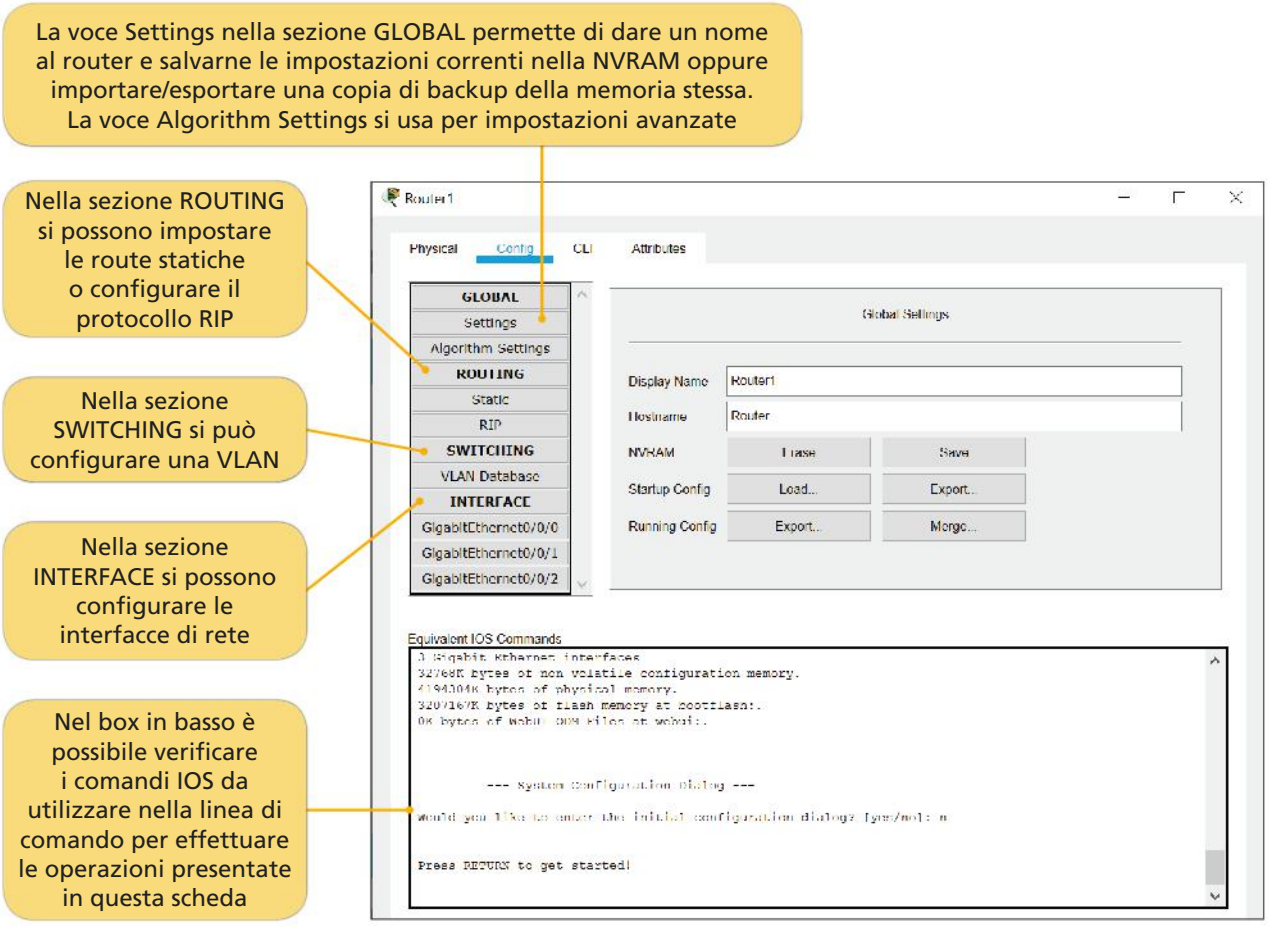
CREARE E CONFIGURARE I DISPOSITIVI

Nella Figura 31 abbiamo visto come selezionare un router, scegliendolo tra i modelli proposti nella toolbar inferiore e inserendolo nell'area di lavoro. La stessa procedura può essere utilizzata per qualunque altro dispositivo si voglia inserire nello schema di rete. Per le connessioni si usa un metodo analogo, l'unica differenza è che bisogna fare clic sui dispositivi da unire e sceglierne le interfacce corrette.
Per la configurazione di un dispositivo si deve entrare nell'apposita finestra di dialogo, con un clic sull'oggetto presente nella nostra area di lavoro.
A titolo di esempio, vediamo ora le finestre di configurazione di un router e di un PC.

Se apriamo la finestra relativa a un **router** standard ci troviamo di fronte a quattro schede (FIGURA 35):

- **Physical**: permette di aggiungere/rimuovere moduli e accendere/spegnere l'apparato. Per inserire nell'apparato la scheda di rete voluta, occorre prima spegnere l'apparato con un clic sull'icona del pulsante di accensione. Poi si deve selezionare dall'elenco di sinistra la scheda voluta; questa apparirà rappresentata nella parte bassa a destra della finestra di dialogo. Infine si trascina (drag-and-drop) la scheda nello slot sul retro del router. Si procede in egual modo per altri dispositivi personalizzabili (per esempio: Personal Computer);
- **Config**: permette di effettuare la configurazione di base dell'apparato tramite box, menu a tendina, spunte e finestre di dialogo;

FIGURA 35 La configurazione del router



- **CLI** (Command Line Interface): simula la linea di comando del Sistema Operativo degli apparati di rete Cisco (**IOS, Internetwork Operating System**) e permette l'impostazione completa;
- **Attributes:** mostra alcune caratteristiche del dispositivo come MTBF (mean time between failure), costo, potenza.

Nel caso di **router wireless integrato**, la scheda CLI è sostituita dalla scheda GUI che simula l'interfaccia grafica tipica per l'impostazione di tali dispositivi.

Apriamo ora la finestra relativa a un **computer** (clic sull'immagine dell'oggetto nel workspace):

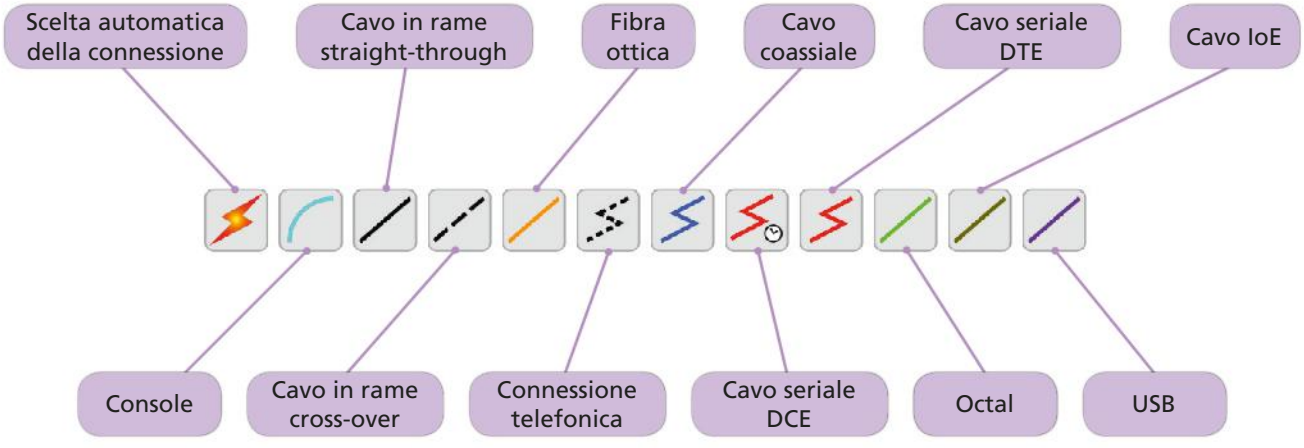
- **Physical** e **Config** funzionano come abbiamo già visto;
- **Desktop** mostra una simulazione di alcune funzionalità del PC per le impostazioni IP, per l'uso della finestra del terminale, per aprire un browser, ecc. (FIGURA 36);
- **Programming**: Packet Tracer supporta tre linguaggi di script per scrivere dei programmi in questo ambiente: Javascript, Python e Visual Scripting. Cliccando sul pulsante New si crea un nuovo progetto che poi si apre con il pulsante Open e si scrive lo script.

FIGURA 36 La scheda Desktop del PC

CREARE UNA CONNESSIONE

La FIGURA 37 mostra le tipologie di connessioni disponibili in Packet Tracer ver. 7.3.1.

FIGURA 37 Connessioni in PT 7.3.1



Ciascun tipo di cavo può essere collegato solo a determinati tipi di interfaccia.

- **Console**: una connessione console può essere fatta tra un PC e un router o switch; nella realtà viene usata per lavorare direttamente sul router, anziché attraverso la rete, come avviene solitamente con le interfacce web.
- **Cavo in rame straight-through** (dritto): è il cavo utilizzato per collegare due dispositivi che operano a livelli differenti del modello OSI, per esempio PC-switch, switch-router. Si può connettere alle porte Ethernet, Fast Ethernet e Gigabit Ethernet.
- **Cavo in rame crossover** (incrociato): è il cavo utilizzato per collegare due dispositivi che operano allo stesso livello del modello OSI, per esempio PC-PC, switch-switch. Si può connettere alle porte Ethernet, Fast Ethernet e Gigabit Ethernet.
- **Fibra ottica**: cavo usato per creare un collegamento tra porte in fibra a 100 Mbps e 1.000 Mbps.
- **Connessione telefonica**: si usa per il collegamento tra un dispositivo e un modem.
- **Cavo coassiale**: si usa per collegamenti sulle porte coassiali di un modem.
- **Cavo seriale DCE e DTE**: sui collegamenti WAN si usano spesso le porte seriali; infatti questo è il tipico collegamento tra due router usati per reti WAN. Il simbolo dell'orologio sul cavo DCE indica che deve essere abilitato il clock affinché il protocollo di linea sia funzionante, mentre il lato DTE è opzionale.
- **Octal**: è un cavo usato per connettere un access server o un terminal server a più router e switch; il cavo da un lato ha un connettore ad alta densità a 68 pin e dall'altro otto connettori RJ45.
- **Cavo IoE** (Internet of Everything): cavo usato per connettere componenti, microcontrollori e computer single-board (SBC);
- **USB**: cavo per porte USB usato come il cavo IoE in ambienti in cui connettere "things".

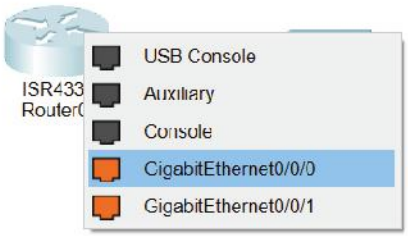
La FIGURA 38 mostra il workspace di PT con alcuni dispositivi: un router, uno switch e due PC.

FIGURA 38 Workspace con dispositivi da connettere

I due computer e il router devono essere connessi allo switch. Si tratta di dispositivi dotati di porte Ethernet che operano tutti a livelli diversi del modello OSI, quindi scegliamo di usare il cavo in rame straight-through per creare i collegamenti. I passi da seguire sono:

1. dalla toolbar inferiore selezionare l'icona delle connessioni  e scegliere il cavo dritto  ;

2. cliccare sul router (si noti la forma diversa del cursore quando è stato spostato nel workspace). Compare una finestra con l'elenco delle interfacce disponibili; scegliere la prima Gigabit Ethernet:



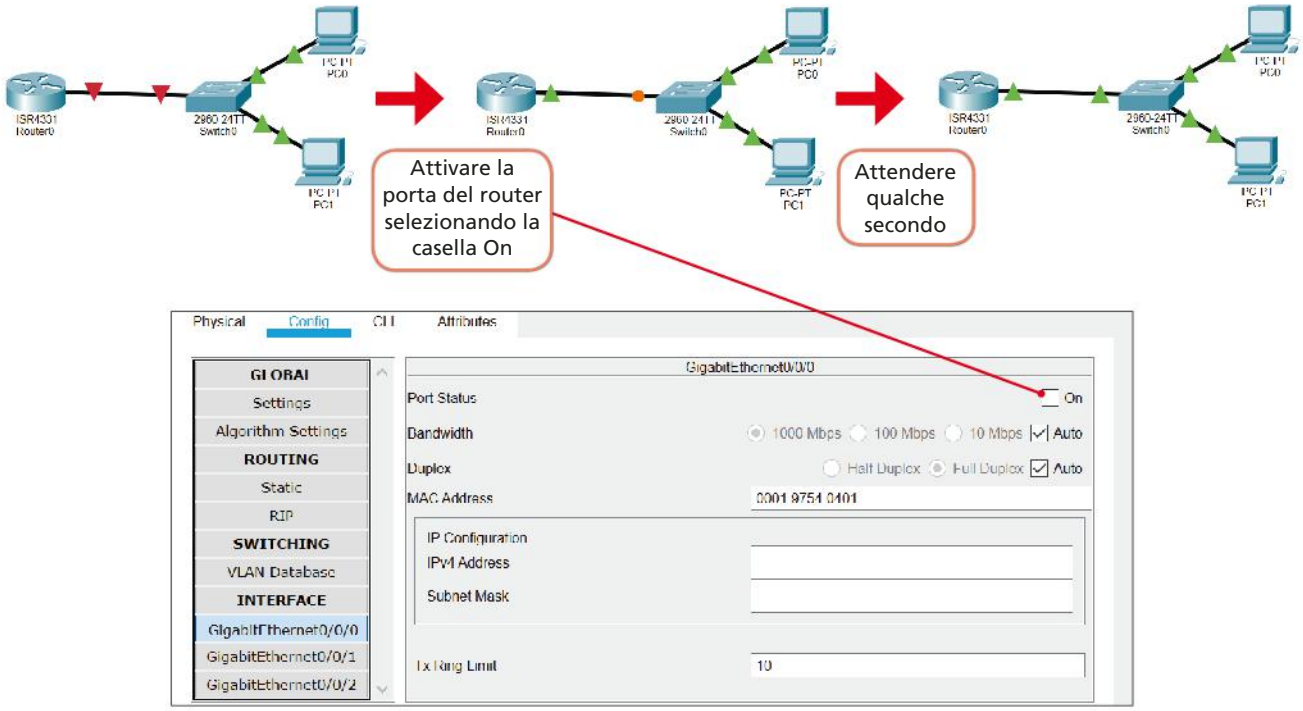
3. cliccare sullo switch e scegliere anche qui l'interfaccia Gigabit Ethernet; la connessione tra router e switch apparirà come una linea nera diritta;
4. ripetere i passi precedenti per collegare i due PC allo switch; questa volta si sceglieranno le interfacce FastEthernet.

Sulle estremità di ciascun collegamento è presente un simbolo che simula il **led di stato dell'interfaccia** (Physical Layer del modello OSI):

- triangolino **rosso**: l'interfaccia è down;
- triangolino **verde**: l'interfaccia è up;
- pallino **ambra**: l'interfaccia è in stato blocked in attesa che termini il processo di verifica di loop nella topologia di rete; si verifica sugli switch.

In FIGURA 39 si presenta la successione degli stati delle interfacce relative alla connessione router-switch: all'inizio non c'è il collegamento attivo perché è necessario mettere up l'interfaccia del router, selezionando la relativa casella nella finestra di configurazione del router (di default le interfacce sono down). Il passaggio da down a up richiede alcuni secondi; se si vuole velocizzare l'operazione si può cliccare sul pulsante Fast Forward Time, presente nella toolbar inferiore.

FIGURA 39 Stato delle interfacce



VERIFICARE LA CONNETTIVITÀ

Quando avremo creato la nostra rete e configurato i dispositivi assegnando un indirizzo IP alle interfacce di rete, dovremo testarne il funzionamento con gli strumenti messi a disposizione da Packet Tracer e presentati in precedenza.
Proviamo quindi un ping semplice con 3 veloci passi (FIGURA 40):

1. fare clic sull'icona della busta chiusa (Simple PDU) nella toolbar secondaria;
2. fare clic sui 2 dispositivi di cui vogliamo verificare la connessione;
3. verificare i risultati del ping nella toolbar inferiore nel box a destra.

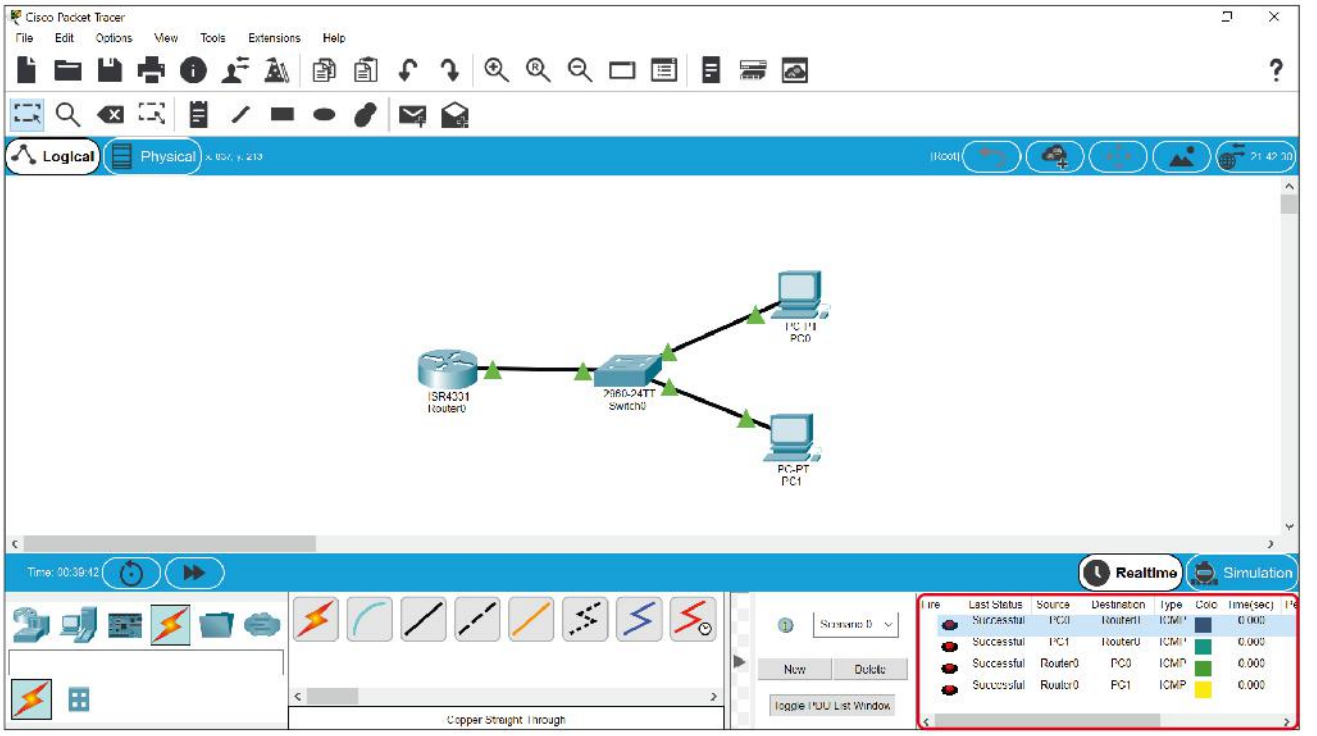


FIGURA 40 Simulazione del comando ping

FISSA LE CONOSCENZE

- In che cosa consiste la simulazione di reti?
- Spiega l'analisi *what if*.
- Quali sono le caratteristiche di PT?
- Quali tipi di tecnologie di rete (apparati, protocolli, ecc.) possiamo simulare con PT?
- Descrivi alcune configurazioni di rete che possono essere simulate con PT.
- Spiega come usare lo strumento Simulation di Packet Tracer per verificare la consegna di un messaggio.
- Come si crea un dispositivo con PT?
- Descrivi la configurazione di un router con PT.
- Come si crea una connessione con PT?
- Una volta creato lo scenario di rete, come si può verificare la connettività a livello fisico tra i dispositivi?

8 CISCO PACKET TRACER: SCENARI PEER-TO-PEER

In questa lezione sono proposte un paio di esercitazioni iniziali da sviluppare in laboratorio con Packet Tracer, allo scopo di prendere confidenza con lo strumento. Infatti, si useranno alcuni menu e strumenti di simulazione presentati nella lezione precedente. Sarà necessario anticipare alcuni concetti dei livelli Physical e Network, descritti nelle Unità 2 e 3, riguardanti l'identificazione delle interfacce di rete con indirizzi fisici (MAC Address) e con indirizzi logici (IP address). Non si richiede, però, una conoscenza di questi elementi, infatti in ogni esercizio è presente una tabella con le indicazioni di che cosa scrivere nei vari campi.

8.1 Connessione Peer-to-Peer tra due computer desktop

esercizio

➜ PROBLEMA

Configurare due PC direttamente, così che possano scambiarsi informazioni e visualizzare cartelle condivise.

➜ ANALISI DEL PROBLEMA

I due Personal Computer devono essere collegati tramite un cavo UTP di tipo crossover. Le schede di rete dispongono già di un MAC address, fornito dal produttore, che è univoco e fisso. L'indirizzo IP, invece, deve essere inserito in fase di configurazione e modificato se il PC si sposta da una rete a un'altra. La scelta di questi valori è affidata all'amministratore di rete e alle politiche di indirizzamento definite per la rete.
La TABELLA 2 mostra gli indirizzi IP e le subnet mask da assegnare ai due PC.
Una volta collegati e configurati i due PC è necessario procedere con un test di connettività (ping).

TABELLA 2 Indirizzi IP e subnet mask dei PC

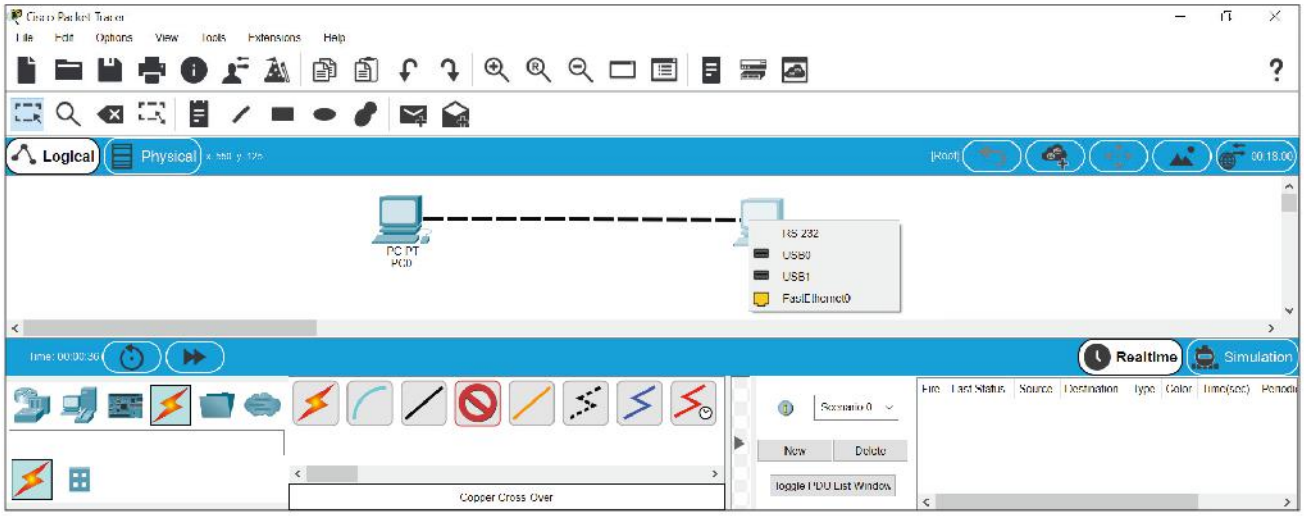
PC0	
IP address	192.168.1.1
subnet mask	255.255.255.0

PC1	
IP address	192.168.1.2
subnet mask	255.255.255.0

➜ SVOLGIMENTO

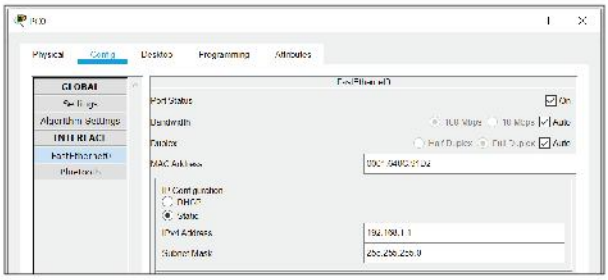
1. Aprire il programma Packet Tracer e inserire due PC nel workspace.
2. Collegare le schede di rete (FastEthernet) con il cavo UTP "Copper Cross-Over" (FIGURA 41). La comparsa dei due triangolini verdi all'estremità del cavo indica che la connessione a livello Physical è funzionante.

FIGURA 41 Collegamento tra i due PC

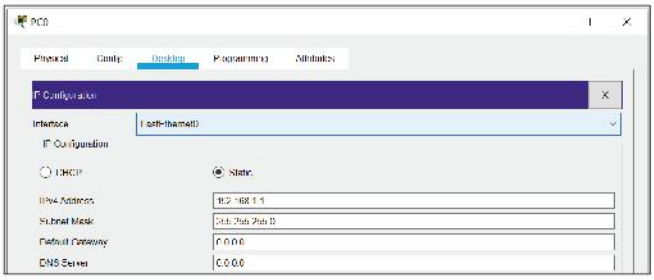


3. Cliccare sull'icona di PC0 nel workspace, si aprirà la finestra di configurazione; esistono due modalità per inserire i dati di configurazione della scheda di rete:
   a. scegliere il tab **Config** e, nelle schede a sinistra, selezionare **FastEthernet0**;
   b. scegliere il tab **Desktop** e selezionare **IP Configuration**.
4. Selezionare la configurazione statica (Static) e inserire nei campi IPv4 Address e Subnet Mask i valori indicati nella tabella 2 per PC0 **(FIGURA 42)**.
5. Ripetere i passi 2-4 per il secondo computer PC1.

FIGURA 42
a. Configurazione IP add. da Config
b. Configurazione IP add. da Desktop



a.



b.

Test di connettività

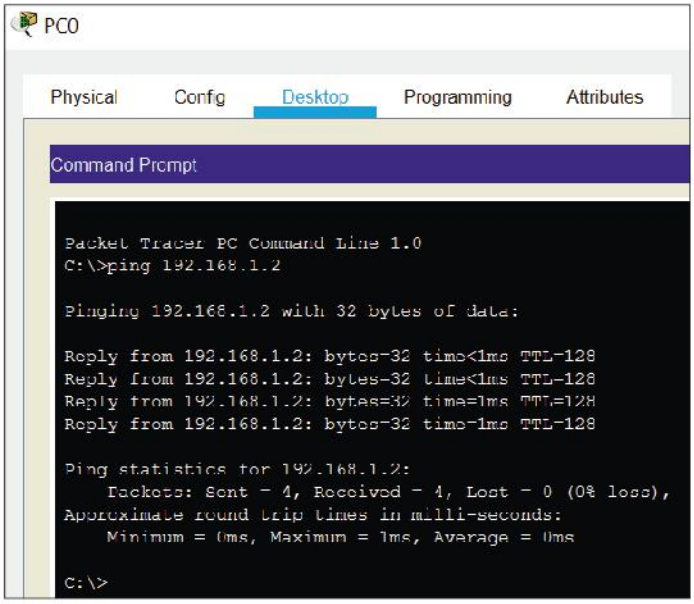
Per eseguire questo test si può procedere in due modi:

1. modalità grafica: si clicca sulla busta chiusa presente nella toolbar secondaria e poi si seleziona prima un PC e dopo l'altro, il risultato è visualizzato nel box in basso a destra;
2. da linea di comando: selezionare il tab **Desktop** di un PC, per esempio PC0, e poi **Command Prompt**, si apre la simulazione della finestra per la linea di comando di Windows, a quel punto digitare: ping 192.168.1.2. La **FIGURA 43** mostra il risultato del ping da PC0 verso PC1.

Eseguire il test di connettività:

- da PC0 verso PC1;
- da PC1 verso PC0.

FIGURA 43 Ping eseguito da Command Prompt



Come abbiamo visto nella lezione precedente, oltre a questo test eseguito in modalità Realtime, si può effettuare una simulazione di come fluisce il traffico nella rete usando la modalità Simulation. La vedremo nel prossimo esercizio.

8.2 Connessione Peer-to-Peer con hub

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Configurare una piccola rete formata da 4 computer connessi tra loro tramite un hub.

➜ ANALISI DEL PROBLEMA

Ricordiamo che l'hub è un dispositivo che opera in modalità broadcast: non è in grado di inoltrare i pacchetti solo al computer di destinazione, ma li invia a tutti i computer collegati, tranne quello sorgente da cui proviene il pacchetto.

L'hub crea quindi un unico dominio di collisione.

La TABELLA 3 mostra gli IP address e la subnet mask da assegnare alle interfacce di rete dei 4 PC da collegare tramite l'hub.

TABELLA 3 Indirizzi IP e subnet mask dei PC

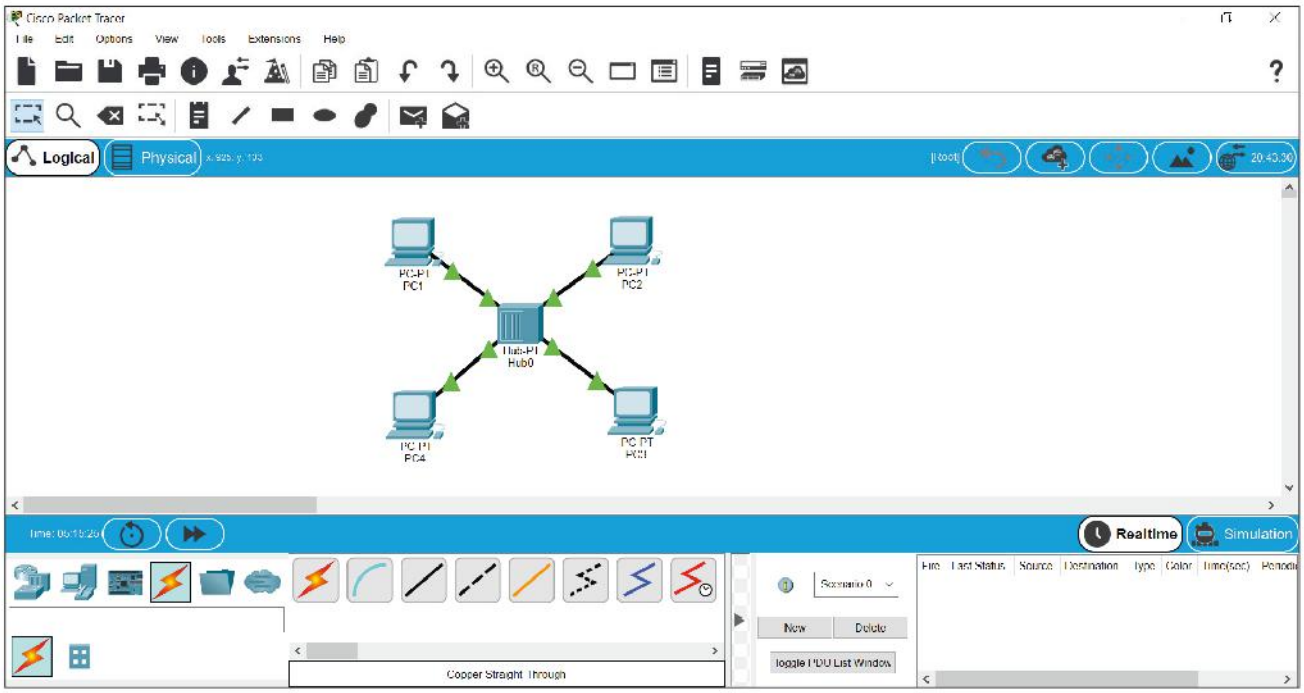
PC1		PC2		PC3		PC4	
IP address	192.168.1.1	IP address	192.168.1.2	IP address	192.168.1.3	IP address	192.168.1.4
subnet mask	255.255.255.0	subnet mask	255.255.255.0	subnet mask	255.255.255.0	subnet mask	255.255.255.0

Si noti che l'hub, come lo switch, non richiede la configurazione IP, in quanto è un apparato che opera a livello due del modello OSI ed è quindi identificato con il solo indirizzo fisico (MAC address).

➜ SVOLGIMENTO

1. Aprire il programma Packet Tracer e inserire 1 hub e 4 PC nel workspace.
2. Collegare all'hub le schede di rete (FastEthernet) di ciascun PC con il cavo UTP "Copper Straight-through". La comparsa dei due triangolini verdi all'estremità di ciascun cavo indica che la connessione a livello Physical è funzionante (FIGURA 44).

FIGURA 44 Scenario di rete con 4 PC e 1 hub



3. Configurare ogni PC in modalità statica, inserendo nei campi **IPv4 Address** e **Subnet Mask** i valori indicati nella tabella 3.

Test di connettività

Restando in modalità Realtime, eseguire il test di connettività con il comando ping, come visto nell'esercizio precedente.

Simulare il comportamento della rete

Aprire l'ambiente di simulazione selezionando la modalità Simulation. Come visto nella precedente lezione, è opportuno usare un filtro così da visualizzare nella Event List le PDU di interesse.

È importante sottolineare che il filtro agisce solo sulla visualizzazione, rendendo più veloce la simulazione, che non deve mostrare tutti gli eventi. Le PDU che sono state filtrate, però, continuano a essere presenti nella rete e a influenzarne il traffico.

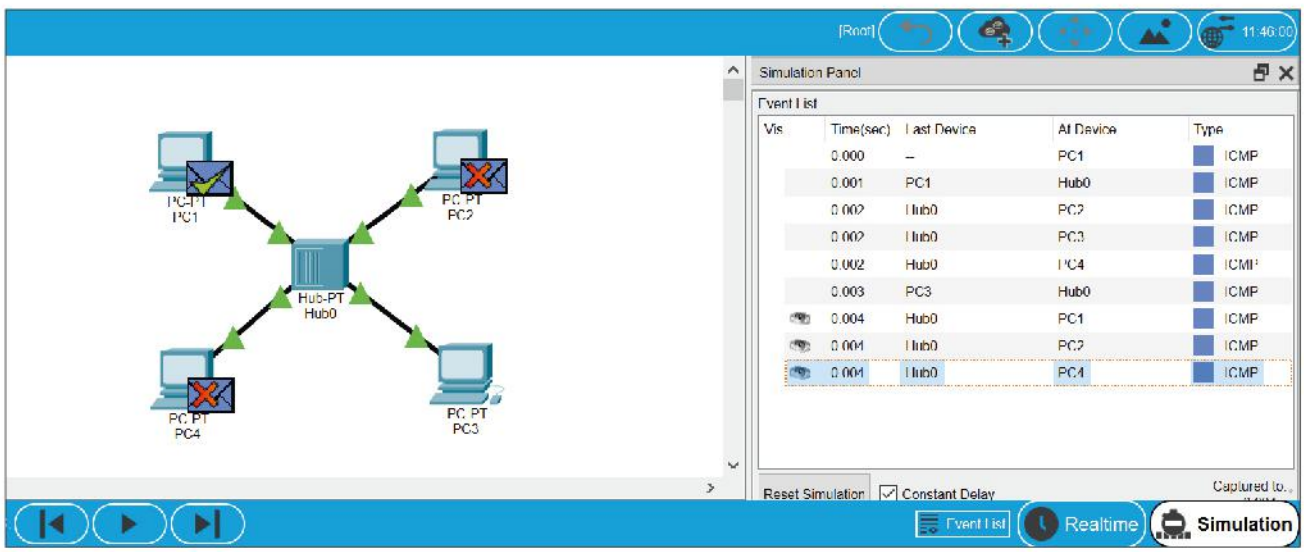
Nel nostro scenario vogliamo simulare il comportamento della rete quando PC1 invia un ping a PC3, quindi:

1. nel Simulation Panel cliccare su Edit Filters e selezionare solo ICMP, togliendo l'eventuale spunta a tutte le altre PDU;
2. nella toolbar secondaria selezionare "Add Simple PDU" e cliccare prima su PC1 e poi su PC3;
3. clic su Play per avviare la simulazione; i pulsanti "Play Controls" sono presenti sia nella toolbar inferiore sia nel Simulation Panel; in questo pannello è anche disponibile uno slider che permette di variare la velocità con cui avviene l'animazione.

Il pacchetto inviato da PC1 è ricevuto da PC2, PC3 e PC4, ma solo PC3 accetta il pacchetto, gli altri dispositivi lo cestinano. Analogamente il pacchetto di risposta che PC3 invia a PC1, viene ricevuto anche da PC2 e PC4, che, però, lo scartano.

Nella **FIGURA 45** è mostrata la ricezione della risposta inviata da PC3, che termina il flusso di dati transitati in rete a seguito dell'invio del ping (Simple PDU) da parte di PC1.

FIGURA 45 Simulazione del ping da PC1 a PC3



Nella fase di simulazione è importante analizzare il contenuto della finestra **Event List**, che mostra i pacchetti catturati durante la propagazione nella rete della PDU. In Packet Tracer il termine "event" individua una PDU che è stata generata dalla rete.

Per ogni PDU catturata, in Event List sono visualizzate le seguenti informazioni:

- **Visible**: la presenza dell'icona di un occhio indica che un evento sta accadendo in quel momento e l'icona sarà presente per tutti i pacchetti che sono visibili nell'animazione in quel dato istante; nell'animazione di Figura 45 l'icona è presente per gli ultimi tre pacchetti inviati dall'hub e visualizzati nel workspace;
- **Time**: il tempo, in secondi, in cui si è verificato l'evento è calcolato in modo relativo, a partire dall'ultimo restart della simulazione; inoltre, il tempo avanza solo quando ci sono eventi da catturare, in caso contrario si ferma e riparte quando si verifica il prossimo evento;
- **Last Device**: indica il dispositivo che aveva il pacchetto prima dell'attuale;
- **At Device**: indica il dispositivo dove si trova il pacchetto in quel momento;
- **Type**: visualizza il tipo di pacchetto catturato, ossia la PDU relativa al protocollo coinvolto (ICMP, HTTP, ecc.), è presente anche un riquadro colorato: a ogni protocollo corrisponde un colore diverso.

Per cancellare gli eventi presenti nella Event List e riportare lo scenario allo stato iniziale, si usa il pulsante Reset Simulation. Se, invece, si vuol eliminare la simulazione della PDU in esame, il ping nel nostro caso, si usa il pulsante Delete presente nella toolbar inferiore.

Simulare una collisione

Si provi contemporaneamente a fare due ping, uno da PC1 a PC3 e l'altro da PC2 a PC4: avviene una collisione nell'hub che è segnalata graficamente dalle due fiamme rosse sui pacchetti **(FIGURA 46)**.

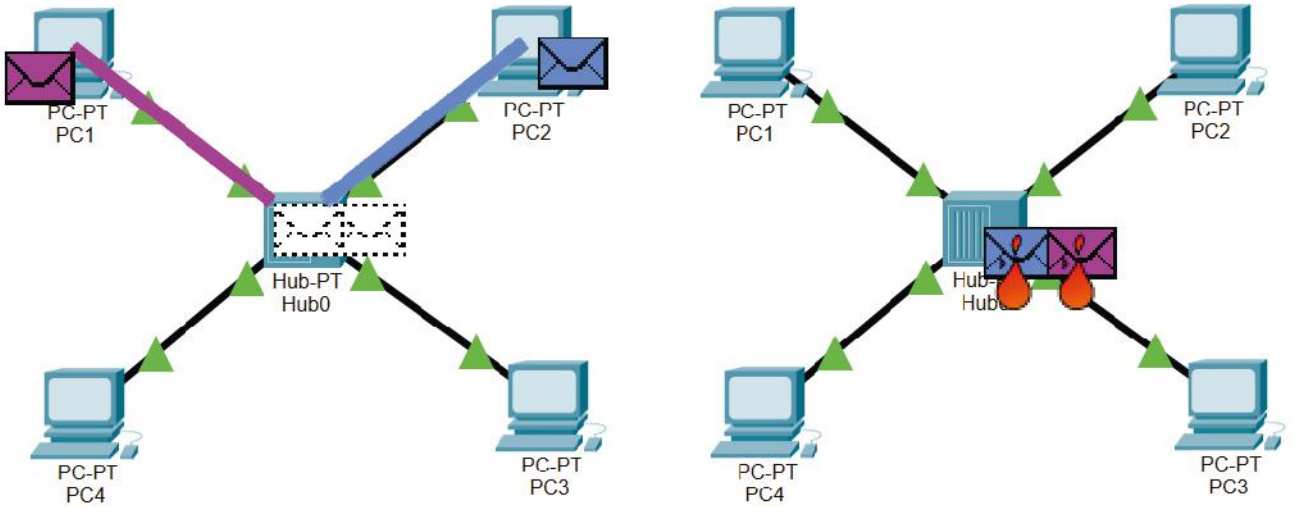
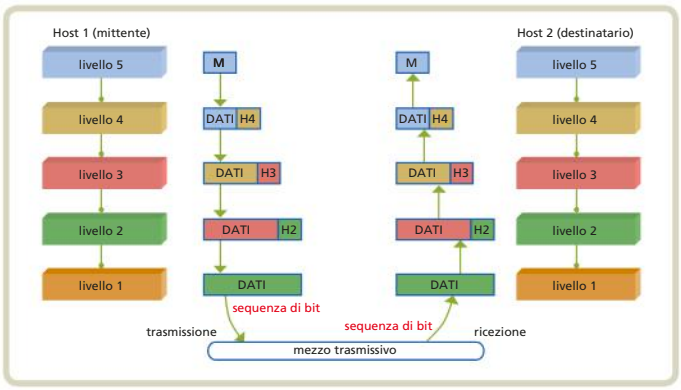


FIGURA 46 Simulazione di una collisione

FISSA LE CONOSCENZE

- Perché per collegare tra loro due PC si usa un cavo UTP crossover?
- Descrivi il comportamento dell'hub quando riceve un pacchetto da uno dei PC collegati.
- A differenza dei PC, per l'hub non è stata fatta la configurazione IP; perché?
- Come avviene il test di connettività tra due computer?
- Descrivi quali informazioni sono visualizzate nella finestra Event List.
- Cosa avviene quando due PC inviano un pacchetto all'hub in contemporanea?

1 I modelli e le architetture di rete



I progettisti di architetture di reti hanno scelto come riferimento il **modello a strati** (o a *livelli*): una struttura logica che consente di suddividere la complessità della comunicazione tra sistemi in funzioni elementari e di assegnarle a strati diversi. Un livello deve definire le interfacce per passare i dati ai livelli adiacenti e i protocolli per la comunicazione tra pari livello (*peer level*). Nella definizione di un simile modello occorre specificare il numero di strati coinvolti e le funzioni svolte in ciascuno di essi. Ciò che fornisce lo strato N allo strato N+1 viene detto *servizio*. Il servizio si richiede attraverso primitive e spesso implica l'elaborazione del messaggio, detto PDU (*Protocol Data Unit*), con l'aggiunta di un header.

2 Il modello ISO/OSI

L'ISO ha definito un modello, denominato ***Open System Interconnection*** **(OSI)**, allo scopo di specificare le modalità di comunicazione tra sistemi differenti. Il modello prevede **sette livelli**, ognuno dei quali svolge un insieme di funzionalità specifiche, i primi tre in supporto alla rete e gli ultimi quattro all'utente.
OSI è da considerarsi un modello di riferimento, essendo ormai universalmente adottata l'architettura TCP/IP.

n° livello		
7	Application Layer	Si occupa delle applicazioni che usano la rete
6	Presentation Layer	Fornisce una rappresentazione standard dei dati per le applicazioni
5	Session Layer	Gestisce le sessioni tra le applicazioni
4	Transport Layer	Fornisce la connessione end-to-end con controllo della congestione
3	Network Layer	Gestisce la connessione alla rete
2	Data Link Layer	Provvede alla trasmissione dei dati sulla rete fisica
1	Physical Layer	Definisce le caratteristiche fisiche della rete

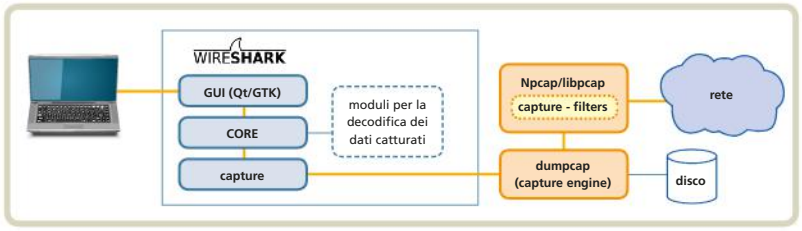
3 Lo stack TCP/IP

L'architettura TCP/IP (o stack TCP/IP o pila protocollare TCP/IP) prevede quattro livelli: il Physical Layer, ulteriormente scomposto in 2 sottolivelli, il Network Layer, il Transport Layer e l'Application Layer. Quest'ultimo raggruppa le funzionalità svolte dagli ultimi tre livelli del modello OSI. Nello stack TCP/IP le PDU usate nei vari livelli sono spesso individuate con un nome specifico: ***frame*** (Physical), ***packet*** (Network), ***segment*** o datagram (Transport) e ***message*** (Application).
Sviluppata prima della creazione del modello OSI e destinata a essere sostituita da questo, l'architettura TCP/IP ha avuto negli anni sempre maggiore diffusione. I suoi protocolli fondamentali, IP per il livello rete e TCP per il livello trasporto, sono alla base del funzionamento della rete Internet.
La versione attuale di TCP/IP è la 6, i cambiamenti hanno riguardato soprattutto il Network Layer con la specifica del protocollo IPv6, che, però, continua a coesistere con il precedente IPv4.

4 Gli enti di standardizzazione

Gli standard sono importanti nelle reti perché permettono l'interoperabilità di prodotti e servizi. Autorevoli enti di riferimento sono ITU-T e ISO. IETF è l'organismo che emette le linee guida per la rete Internet, esse rappresentano degli *standard de facto* per la loro ampia diffusione. I documenti specificati da IETF prendono il nome di RFC (Request for Comments) e sono numerati in modo progressivo. Sul sito di IETF si può prendere visione dello stato dell'arte dei nuovi standard in via di definizione nell'ambito dei vari gruppi di lavoro (WG, Working Group).

5 Wireshark: un analizzatore di protocollo



Un analizzatore di protocollo, o *sniffer*, è uno strumento molto utile per gli amministratori di rete e per chi si occupa di sicurezza, aiutandoli nel monitoraggio della rete, delle sue prestazioni e nell'individuazione di anomalie o guasti. In questo libro useremo un analizzatore molto diffuso: **Wireshark**. È un'applicazione open source, che opera in tre fasi: **raccolta** dei dati grezzi (*capture*) che fluiscono attraverso una data interfaccia di rete, **conversione** dei dati in un formato leggibile e **analisi** dei campi degli header dei vari protocolli.

6 Lavorare con Wireshark

Wireshark offre strumenti molto utili per agevolare l'analisi della grande mole di dati che cattura sulla rete. Tra questi fondamentali sono i **filtri di cattura**, che operano durante la fase di raccolta dei dati, e i **filtri di visualizzazione**, che consentono di mostrare a video solo alcuni dati su cui focalizzare l'indagine.

7 Cisco Packet Tracer: un simulatore di rete

Un **simulatore** di rete permette di riprodurre su un computer il funzionamento di una rete reale complessa e di fare previsioni sul comportamento del sistema a fronte di eventi che potrebbero verificarsi durante la normale operatività della rete (analisi **what if**). Spesso i simulatori offrono funzioni di **emulazione** degli apparati di rete, così da poter svolgere configurazioni e test come se si lavorasse sul sistema reale. In questo libro useremo il simulatore **Packet Tracer** sviluppato da Cisco per essere utilizzato nei corsi di certificazione. Packet Tracer offre due distinte modalità per svolgere test sulla rete progettata: Realtime e Simulation. Quest'ultima fornisce un'animazione del percorso che seguono i pacchetti nella rete e permette di analizzare le singole PDU.

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Il modello a strati semplifica la gestione della comunicazione. V F
2. I modelli a strati hanno sempre sette livelli. V F
3. La comunicazione fisica nel modello a strati avviene tra livelli adiacenti. V F
4. *Peer entities* sono entità che si trovano su due livelli adiacenti (per esempio, N e N–1). V F
5. Il livello 5 dell'host mittente e il livello 4 dell'host ricevente si possono definire *peer level*. V F
6. Nel modello a strati, un livello superiore può essere considerato client del livello sottostante. V F
7. La PDU del livello Data Link del modello OSI viene chiamata frame (trama). V F
8. La scheda di rete di un computer svolge funzioni di livello Physical. V F
9. Il livello Transport nel modello OSI si occupa di instradare i pacchetti in rete. V F
10. Il livello Presentation nel modello OSI svolge funzioni di compressione dei dati. V F
11. In un router sono implementate tutte le funzioni che si possono svolgere nei sette livelli del modello OSI. V F
12. Nell'architettura TCP/IP il livello Application corrisponde al livello Application del modello OSI. V F
13. La versione 5 di IP è quella attualmente in uso su Internet. V F
14. L'ente europeo che definisce standard in ambito telecomunicazioni è ANSI. V F
15. I documenti pubblicati da IETF sono a pagamento. V F
16. Un analizzatore di protocollo permette di modificare il contenuto di un pacchetto. V F
17. Nelle reti con hub è possibile intercettare tutto il traffico che transita su una rete locale. V F
18. La *mirror port* è un'interfaccia presente sui router. V F
19. Tshark è l'interfaccia console di Wireshark. V F
20. Npcap è la libreria di cattura usata per i sistemi Linux. V F
21. Con Wireshark non è possibile leggere i campi dell'header dei vari protocolli di rete. V F
22. L'analisi "what if" fornisce informazioni di tipo predittivo. V F
23. Un simulatore di rete permette di provare vari scenari di rete. V F
24. Cisco Packet Tracer non supporta i protocolli dell'Application Layer. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **Nel modello a strati, con quale termine si definisce il principio secondo il quale l'header che un livello aggiunge ai dati ricevuti dal livello superiore viene "nascosto" al livello inferiore?**
   - A Ereditarietà
   - B Astrazione
   - C Incapsulamento
   - D Associazione

2. **Nel modello a strati, con il termine "peer entities" si identificano entità che si trovano:**
   - A su due livelli adiacenti
   - B sullo stesso livello
   - C nel livello Application
   - D nel livello Physical

3. **In quale di questi servizi i pacchetti di uno stesso messaggio sono trasmessi in rete in modo indipendente gli uni dagli altri?**
   - A Direct delivery
   - B Connectionless
   - C Connection-oriented
   - D Packet release

4. **Gli switch a quale livello del modello OSI operano?**
   - A Livello Data Link
   - B Livello Physical
   - C Livello Transport
   - D Livello Network

5. **Nel modello OSI, l'header di livello 5 inserito dall'host mittente viene letto da quale livello nell'host destinatario?**
   - A Livello Presentation
   - B Livello Network
   - C Livello Transport
   - D Livello Session

6. **Nel modello OSI, quale livello svolge un ruolo intermedio tra i livelli che offrono servizi di rete e i livelli che offrono servizi applicativi?**
   - A Livello Presentation
   - B Livello Network
   - C Livello Transport
   - D Livello Session

7. **Nel modello OSI, qual è la funzione principale svolta dal livello Transport?**
- A Consegna del messaggio al processo sull'host destinatario
- B Sincronizzazione
- C Consegna del pacchetto al nodo di rete successivo
- D Gestione delle tabelle di instradamento dei pacchetti in rete

8. **La PDU a livello Network viene chiamata:**
- A frame
- B segment
- C packet
- D message

9. **Di quanti bit è composto un indirizzo IPv6?**
- A 32
- B 64
- C 96
- D 128

10. **Quale dei seguenti organismi di standardizzazione definisce gli standard per Internet?**
- A ITU
- B IETF
- C IEEE
- D ISO

11. **Il termine PDU significa:**
- A Process Data Utility
- B Process Destination Unit
- C Protocol Data Unit
- D Protocol Data Utility

12. **Alcuni documenti disegnano l'architettura TCP/IP su 5 strati, così da renderla più simile al modello OSI. Quale livello è diviso in due parti?**
- A Physical
- B Network
- C Transport
- D Application

13. **I documenti di specifica di uno standard pubblicati da IETF sono:**
- A Internet Draft
- B Working Draft
- C Proposed Draft
- D Request For Comment

14. **Quale dei seguenti standard è stato definito dall'ANSI?**
- A 802.x
- B 3GPP
- C ASCII
- D OSI

15. **Come si chiama il file che contiene i dati che un analizzatore raccoglie nella rete?**
- A DATA file
- B FCAP file
- C PCAP file
- D COLLECT file

16. **Quale dei seguenti tab non è presente nella finestra di configurazione di un PC in Packet Tracer?**
- A Physical
- B Config
- C Desktop
- D CLI

17. **In Packet Tracer che simbolo è usato per indicare che un'interfaccia è down?**
- A un pallino arancione
- B un triangolino verde
- C un triangolino rosso
- D un pallino blu

18. **In Packet Tracer l'icona della busta aperta indica:**
- A Simple PDU
- B Complex PDU
- C Event
- D Filter

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Quali vantaggi offre un modello strutturato a livelli?**

2. LEZIONE 1 **Descrivi il principio dell'incapsulamento tipico del modello a strati.**

3. LEZIONE 1 **Spiega il significato di protocollo.**

4. LEZIONE 1 **Che cos'è una primitiva di servizio?**

5. LEZIONE 2 **Descrivi come avviene la comunicazione tra due host con riferimento al modello OSI.**

6. LEZIONI 2 E 3 **Metti a confronto lo stack TCP/IP e il modello OSI.**

7. LEZIONE 4 **Descrivi il ruolo degli enti di standardizzazione nell'ambito delle reti.**

8. LEZIONE 5 **Perché gli analizzatori di protocollo sono spesso utilizzati dagli amministratori di rete?**

9. LEZIONE 5 **Spiega la tecnica "port mirroring" implementata sugli switch.**

10. LEZIONE 6 **Descrivi alcune funzionalità di Wireshark utili per l'analisi dei pacchetti.**

11. LEZIONE 7 **Spiega l'utilità dei simulatori di rete.**

12. LEZIONE 7 **Spiega la differenza tra emulatore e simulatore di rete.**

13. LEZIONE 7 **Quali sono le caratteristiche dell'applicazione Packet Tracer?**

14. LEZIONE 7 **Descrivi come inserire i dispositivi nell'area di lavoro di Packet Tracer e creare i collegamenti tra essi.**

IN ENGLISH, PLEASE

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

Network Architectures

In using communication networks it is essential to follow standards which ensure interoperability between different systems. OSI is a model defined by the ISO which specifies the means of communication between remote systems and which has structure at various layers, each of which performs a set of specific functions. A layered model typically defines the service which each layer provides and the protocols required for communication between peer layers. The Internet uses a different structure, which is simpler than the OSI model and which is called TCP/IP from the name of its two most important protocols.
It's very important to know what's happening on a network, so we can use a network protocol analyzer such as Wireshark.
Another very useful tool is a network simulator: it is able to emulate various types of devices and network systems and to create complex topologies based on graphic schemes. Packet Tracer is a learning tool developed by Cisco that provides a visual simulation of equipment and network processes

EXERCISES

Use the appropriate number to match words and meanings.

...	Physical Layer	1	Encryption and decryption
...	Data Link Layer	2	End-to-end connection
...	Network Layer	3	User interface
...	Transport Layer	4	Frame creation
...	Session Layer	5	Bit transfer through a media
...	Presentation Layer	6	Path discovery
...	Application Layer	7	Checkpoint setting

GLOSSARY

Architecture: in networking, it refers to the overall structure, topology, protocols and framework of a network.
CLI (Command Line Interface): a textual user interface of the router's Operating System.
Encapsulation: it adds headers before the start of a PDU and takes place at each layer of OSI reference model.
Header: it is the control part of a PDU and contains information that specifically addresses layer-to-layer communication (peer layers). In transmission, each layer can add a header to the data coming from the upper layer. This header will be read and then removed by the peer receiving layer.
IOS (Internetwork Operating System): software used on Cisco routers and switches that provides routing, switching, internetworking and telecommunications functionalities
Model: it characterizes and standardizes the functions of a system. It is not directly tied to any technologies (different equipment, applications by different vendors).
Packet/Protocol Analyzer: network monitoring software that is also known as a *sniffer*, *packet sniffer*, or *traffic sniffer*.
Payload: it is the data part of a PDU.
PDU (Protocol Data Unit): it is the information unit (data or control) that peer entities use to communicate.
Protocol Stack or Protocol Suite: a set of protocol layers that work together.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper utilizzare un simulatore di rete.
- Realizzare scenari di rete costituiti da dispositivi collegati tra loro usando i cavi corretti e le interfacce di rete adatte.
- Verificare il funzionamento della rete con test e simulazioni.
- Svolgere analisi del tipo "what if".

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Acquisire piena consapevolezza dell'utilità di un simulatore nel progettare una rete e verificarne il funzionamento.
- Esporre i risultati del proprio lavoro alla classe.

tempi

- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Applicazione Cisco Packet Tracer.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

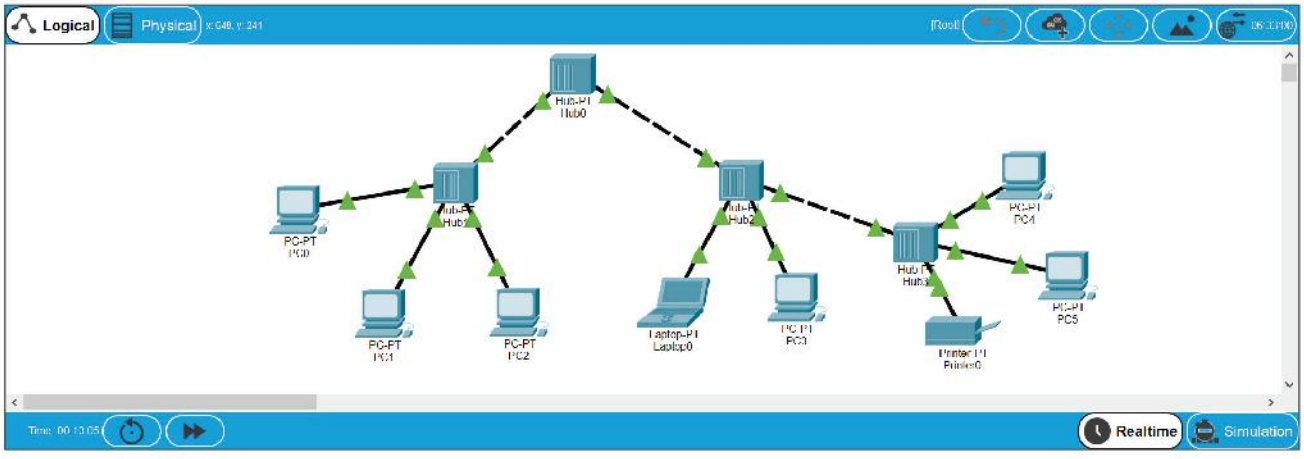
Realizzare una topologia a stella estesa utilizzando apparati hub come centro stella

Con un simulatore di rete creare uno scenario in cui siano presenti alcuni hub collegati in modo da creare una topologia di tipo gerarchico o stella estesa. Collegare agli hub alcuni computer e stampanti. Verificare che tutti i PC della rete siano raggiungibili, svolgendo gli opportuni test, e analizzare il traffico generato.

File sorgenti
Scarica il file

SVOLGIMENTO

Installare sul proprio computer l'applicazione Cisco Packet Tracer seguendo le indicazioni fornite nella Lezione 7 di questa Unità.
Avviare Packet Tracer, selezionare la categoria "Network Devices" e scegliere l'hub da inserire nel workspace per progettare la rete. Nella categoria "End Devices" selezionare alcuni PC, laptop, stampanti da collegare agli hub.
Selezionare i cavi UTP opportuni, straight-through o crossover, per collegare tra loro i dispositivi.
La figura seguente mostra uno scenario di rete realizzato con topologia a stella estesa, in cui ogni hub ha il ruolo di centro stella.



Passiamo ora a configurare gli indirizzi IP e la subnet mask su ogni computer e stampante. Nella finestra di configurazione di ciascun dispositivo inserire i valori indicati nelle tabelle seguenti:

PC0		PC1		PC2	
IP address	192.168.1.1	IP address	192.168.1.2	IP address	192.168.1.3
subnet mask	255.255.255.0	subnet mask	255.255.255.0	subnet mask	255.255.255.0

Laptop0		PC3	
IP address	192.168.1.4	IP address	192.168.1.5
subnet mask	255.255.255.0	subnet mask	255.255.255.0

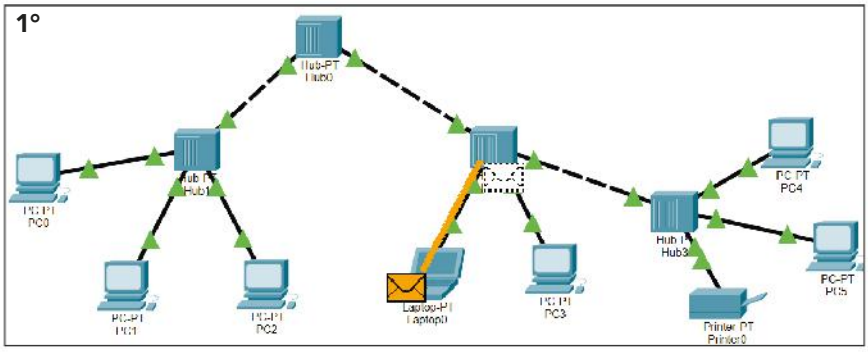
Printer00		PC4		PC5	
IP address	192.168.1.6	IP address	192.168.1.7	IP address	192.168.1.8
subnet mask	255.255.255.0	subnet mask	255.255.255.0	subnet mask	255.255.255.0

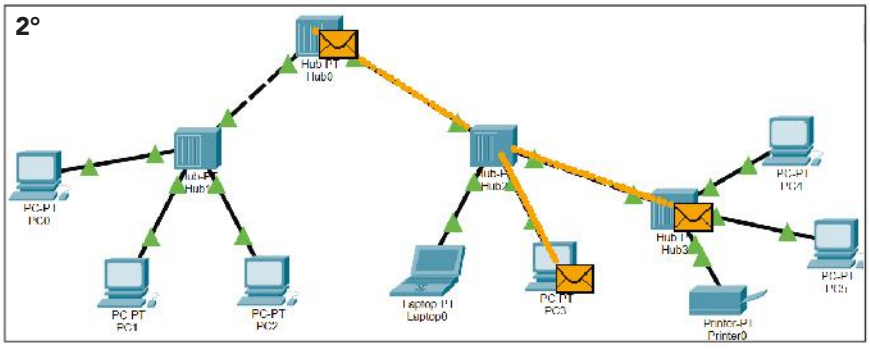
Per verificare il funzionamento della rete, rimanendo nella modalità Realtime, si può fare un test veloce con il ping, usando una Simple PDU e verificando, per esempio, che la stampante (Printer0) sia raggiungibile da ogni PC.

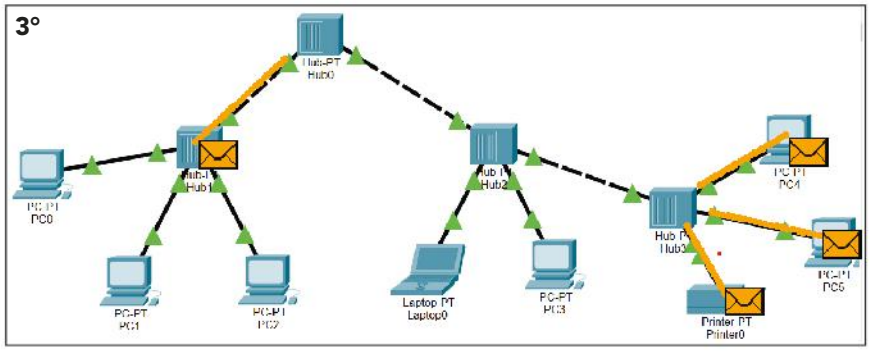
Interessante è l'analisi del traffico che l'invio di un pacchetto, come il messaggio ICMP del ping, genera nella rete:

- selezionare la modalità Simulation e filtrare il traffico catturando solo i pacchetti ICMP, come visto nella Lezione 7;
- scegliere una Simple PDU da inviare dal Laptop0 a PC0;
- avviare la cattura dei pacchetti cliccando sul tasto Play.

La successione di eventi generati dal ping è mostrata nelle schermate seguenti: 4 fasi successive che mostrano il percorso seguito dal pacchetto ICMP attraverso Hub2, Hub0 e Hub1 per giungere al PC0 di destinazione. Ogni volta che il pacchetto arriva a un hub, questi lo invia in output su tutte le sue interfacce. I dispositivi che lo ricevono verificano l'indirizzo IP

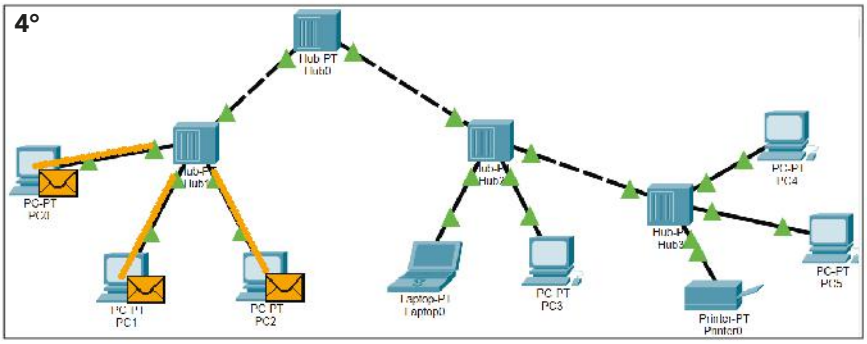






di destinazione e, non corrispondendo al proprio, scartano il pacchetto.

La simulazione ha permesso di verificare che, in questo scenario di rete, tutti i dispositivi presenti ricevono una copia del pacchetto inviato da Laptop0.



A CASA

- Ipotizza una tua soluzione al tema proposto.
- Leggi lo svolgimento per verificare se le tue ipotesi si adattano al caso preso in esame e se la trattazione proposta risulta completa.
- Modifica la rete **sostituendo Hub0 con uno switch** e verifica con la simulazione come cambia il flusso dei pacchetti.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i progetti di rete presentati.
- Trovate una spiegazione al differente comportamento di hub e switch nella rete.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho creato la topologia di rete e simulato il suo funzionamento senza difficoltà?	Ho avuto difficoltà nel selezionare i dispositivi e trovare i cavi corretti. Non sono riuscito ad avviare la simulazione. ☐	Con la guida del docente e la collaborazione dei compagni ho simulato il funzionamento della rete. ☐	Ho creato la rete e simulato il suo funzionamento autonomamente. ☐	Ho sostituito l'hub con lo switch e, grazie alla simulazione, ho compreso il loro diverso modo di lavorare. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ 2

IL PHYSICAL LAYER DEL TCP/IP

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

DDI+ Didattica Digitale Integrata Plus

conoscenze

Conoscere il livello Physical e i suoi sottolivelli.
Conoscere il Progetto 802.
Conoscere le modalità di accesso ai mezzi fisici di trasmissione.
Conoscere lo standard Ethernet.
Conoscere lo standard Wi-Fi.

abilità

Saper calcolare i tempi di attesa in caso di collisione tra frame su canali condivisi.
Saper distinguere i frame Ethernet.
Saper posizionare correttamente gli access point.

competenze

Saper affrontare le problematiche tipiche dei diversi standard di trasmissione.
Saper scegliere i dispositivi per lo switching e il PoE.
Saper scegliere la tecnologia trasmissiva adatta alle esigenze.
Saper usare gli applicativi Packet Tracer e Wireshark.

FLIPPED CLASSROOM

A casa

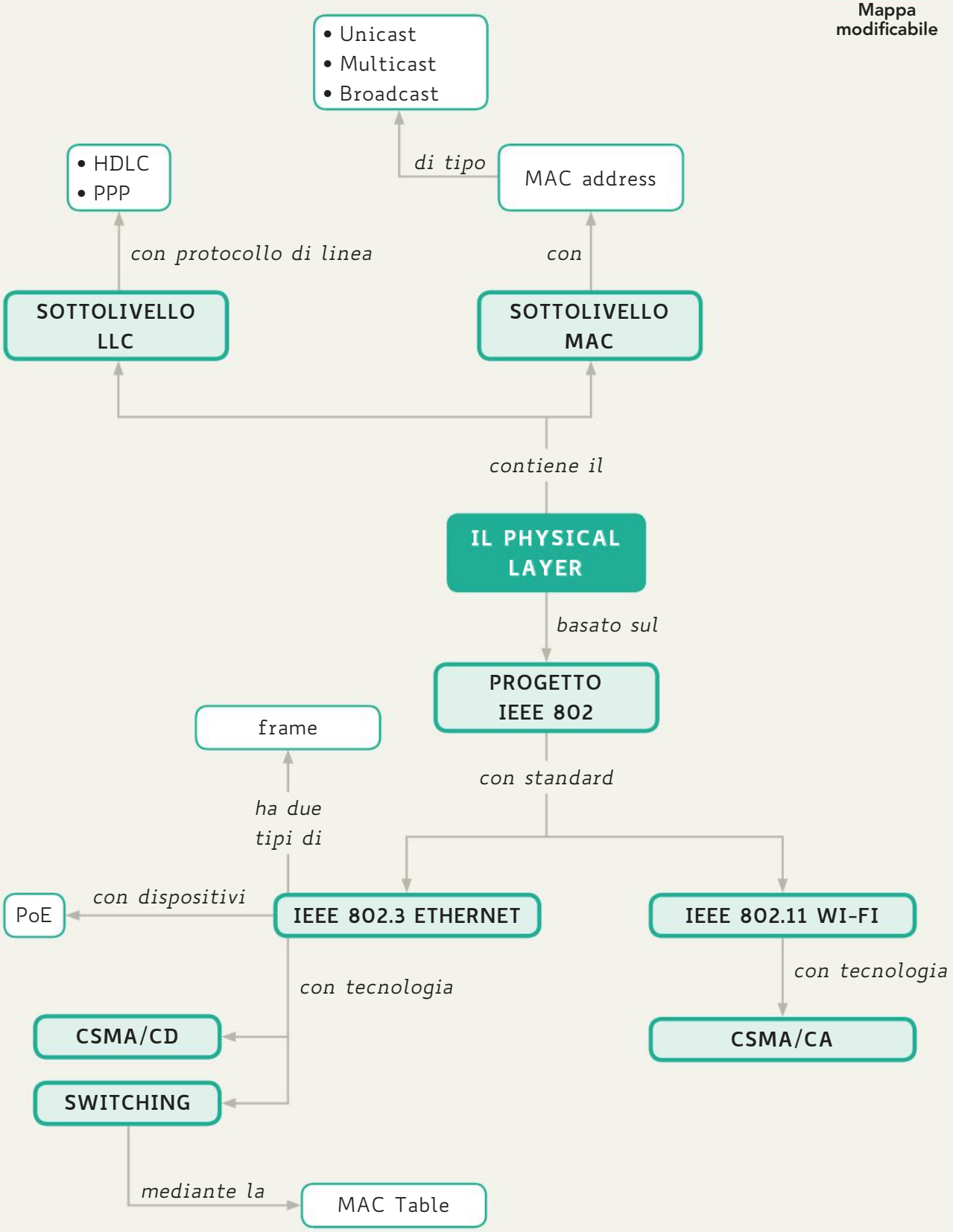
- Leggi la Lezione 7 di questa unità;
- installa un analizzatore Wi-Fi su un qualsiasi dispositivo mobile;
- visualizza le reti a 2,4 GHz e 5,2 GHz nella copertura del tuo dispositivo;
- raccogli i risultati in una presentazione (massimo 5 slide) riportando le reti Wi-Fi individuate e i canali a 2,4 GHz trovati più occupati.

In classe

- Confrontate le analisi fatte;
- valutate se tra queste vi sono differenze;
- discutete i motivi che spiegano le eventuali differenze.

MAPPA CONCETTUALE

Mappa modificabile



1 IL PROGETTO IEEE 802

IEEE, ISO e ANSI hanno sviluppato uno standard, noto come **Progetto IEEE 802**, così chiamato in virtù dell'anno (1980) e del mese (febbraio) in cui ebbe ufficialmente inizio, per stabilire come debbano essere realizzate le reti LAN ai livelli Physical e Data Link del modello ISO/OSI in termini di servizi disponibili e di protocolli per l'espletamento di questi servizi.
Gli standard introdotti dal Progetto 802 stabilirono 20 categorie con cui identificare **i diversi modi di accedere al canale di trasmissione** già commercializzate e quelle ancora in fase progettuale:

802 – Overview and Architecture
802.1 – Bridging and Management
802.2 – Logical Link Control
802.3 – CSMA/CD Access Method [Ethernet]
802.4 – Token-Passing Bus Access Method
802.5 – Token Ring Access Method
802.6 – DQDB Access Method
802.7 – Broadband (Technical Advisory Group)
802.8 – Fiber-Optic (Technical Advisory Group)
802.9 – Isochronous LAN
802.10 – Interoperable LAN/MAN Security
802.11 – Wireless LAN [Wi-Fi]
802.12 – Demand Priority Access Method [100VG AnyLAN]
802.13 – Cable-TV Based Broadband Networks
802.14 – Cable Modem LAN
802.15 – Wireless PAN [Bluetooth]
802.16 – Broadband Wireless MAN
802.17 – Resilient Packet Ring
802.18 – Radio Regulatory (Technical Advisory Group)
802.19 – Coexistence (Technical Advisory Group)
802.20 – Mobile Broadband Wireless Access

Il progetto 802 è in continuo aggiornamento e ha prodotto la standardizzazione delle principali tecnologie di rete oggi in uso e la definizione delle regole per passare da una tecnologia all'altra.
Sono per esempio in continuo sviluppo gli standard 802.3 (Ethernet) e 802.11 (Wi-Fi) i cui #frame approfondiremo in questa unità. Una parte della standardizzazione del livello Physical è delegata ai diversi enti che regolano le caratteristiche di cavi e connettori (EIA, TIA, ecc.) come abbiamo visto nel volume del terzo anno.

#techwords
Con **frame** si intende una particolare sequenza di byte suddivisa in campi secondo regole prestabilite.

> Il livello Physical deve preoccuparsi dell'accesso alla rete di comunicazione tenendo presente che le trasmissioni broadcast condividono un unico canale e che quindi è necessario verificare che il canale sia effettivamente libero prima di effettuare una trasmissione, risolvendo eventuali conflitti tra più stazioni che vogliano accedere contemporaneamente alla risorsa.

Occorre allora stabilire un algoritmo di accesso, cioè una tecnica che regoli il diritto a trasmettere sul canale condiviso.

Due sono le possibilità: la tecnica a contesa e la tecnica deterministica.

- La **tecnica a contesa** prevede l'accesso casuale al canale e se due o più stazioni cercano di trasmettere simultaneamente, il conflitto viene risolto secondo alcune regole di mediazione. Le prestazioni possono essere calcolate solo statisticamente, in relazione alla probabilità che all'inizio di una trasmissione non vi sia una con-

LEZIONE ONLINE
IEEE 802.5: TOKEN RING
È uno standard con topologia ad anello (ring) unidirezionale creato dall'IBM nel 1976. L'accesso al mezzo fisico è deterministico mediante una tecnica di token passing: il gettone (token) gira nell'anello, chi se ne impossessa trasmette e poi rilascia il token.

tesa tra stazioni per l'accesso al mezzo trasmissivo. Non è quindi possibile stabilire a priori l'intervallo di tempo necessario per portare a termine una trasmissione. La più nota tecnica a contesa è la **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection) basata sulla gestione delle collisioni. Le reti Ethernet (IEEE 802.3) con hub e velocità di 10 Mbps utilizzano questa tecnica che verrà approfondita nella Lezione 5.
Un'altra tecnica è la **CSMA/CA** (Carrier Sense Multiple Access with Collision Avoidance) basata sulla prevenzione delle collisioni. Le reti Wi-Fi (IEEE 802.11) utilizzano questa tecnica che verrà approfondita nella Lezione 7.

- La **tecnica deterministica** dà luogo a reti deterministiche, in cui ogni trasmissione avviene in un istante definito e sicuramente va a buon fine, dato che in quell'istante la stazione trasmittente è l'unica a possedere l'accesso al canale. Le prestazioni delle reti non a contesa possono essere determinate con precisione, rendendo le reti particolarmente adatte alle trasmissioni in real time. Le più note tecniche deterministiche sono quelle basate su un **token** (gettone) che autorizza a trasmettere chi se ne impossessa. Lo standard Token Ring (IEEE 802.5) utilizza questa tecnica che viene approfondita nella lezione online.

Le architetture di rete hanno dovuto tenere conto delle diverse tecnologie e dei metodi di accesso alternativi standardizzati dal Progetto 802.
In particolare il livello Data Link di ISO/OSI è stato suddiviso in due sottolivelli: **LLC** (Logical Link Control) e **MAC** (Media Access Control) che rappresentano il cuore del Progetto 802 (FIGURA 1).

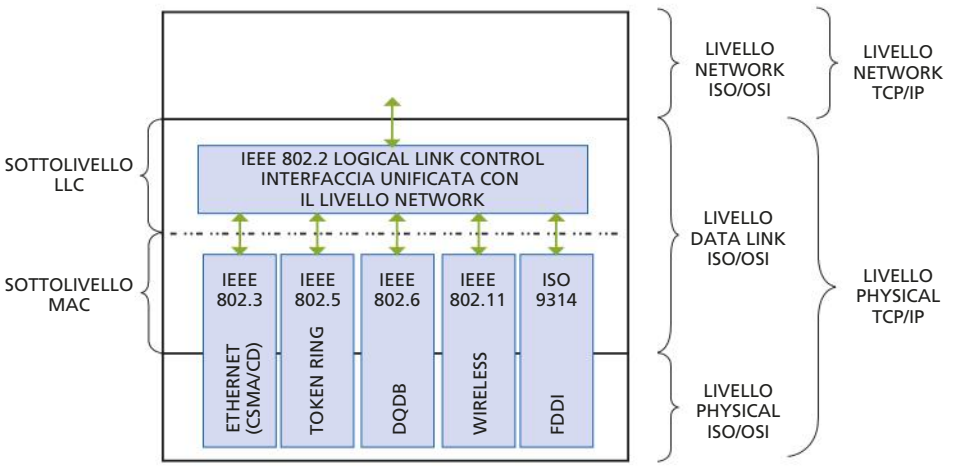


FIGURA 1
Sottolivelli LLC e MAC

LEZIONE ONLINE

IEEE 802.6: DQDB (Distributed Queue Dual Bus)
Si tratta di uno standard progettato agli inizi degli anni '90 per fungere da dorsale (backbone) di interconnessione di LAN diverse (Ethernet, Token Ring, ...) creando così reti metropolitane (MAN) o geografiche (WAN) generalmente in fibra ottica.

LEZIONE ONLINE

ISO 9314: FDDI (Fiber Distributed Data Interface)
È uno standard che rappresenta un'evoluzione della Token Ring sempre basato sul token passing ma con un doppio anello (dual ring) in fibra ottica. I due anelli sono unidirezionali e in opposizione, cioè le trasmissioni fluiscono in direzioni opposte.

FISSA LE CONOSCENZE

- In che cosa consiste il Progetto IEEE 802?
- Il livello Physical di TCP/IP quali livelli di ISO/OSI comprende?
- Che cosa si intende per tecnica a contesa?
- Che cosa si intende per tecnica deterministica?

2 I SOTTOLIVELLI LLC E MAC

2.1 Il sottolivello LLC

Il sottolivello superiore è l'**LLC** che ha il fondamentale compito di fornire un'interfaccia unificata verso il livello Network, pur a fronte di tecnologie trasmissive e mezzi fisici differenziati. Può inoltre occuparsi del controllo del flusso di trasferimento dei dati.

Poiché a livello di Network possono operare vari protocolli (il principale è IP), il sottolivello LLC deve individuare qual è il protocollo usato per la comunicazione. Proprio per questo scopo il frame di LLC (FIGURA 2) contiene due indirizzi, da un byte ciascuno, detti **DSAP** (Destination Service Access Point) e **SSAP** (Source Service Access Point), che rappresentano rispettivamente l'identificatore del protocollo di livello superiore, cui deve essere consegnato il frame (**PDU**) ricevuto, e l'identificatore del protocollo di livello superiore, da cui il packet è arrivato nel dialogo tra due peer entity.

FIGURA 2 Il frame LLC

#techwords

Il **piggybacking** è la tecnica per cui il destinatario può rimandare l'invio dell'ACK includendolo nel successivo messaggio inviato al mittente. In questo caso si può dire che il messaggio inviato "piggybacks" (porta con sé) l'ACK relativo all'ultima ricezione avvenuta.

Il campo **Control** può essere lungo 1 o 2 byte e avere 3 formati:

- il formato **information** è usato per le trame che trasportano i dati in modalità connessa e ha anche la possibilità di trasportare un acknowledge (ACK) per la trasmissione nella direzione inversa (tecnica detta di **#piggybacking**). Le trame di questo formato sono dette **I-frame**;
- il formato **supervisor** non prevede la presenza del campo information nella trama ed è usato per trasportare informazioni di controllo relative agli I-frame; per esempio, fornire un ACK in assenza di traffico nella direzione inversa oppure operare il controllo di flusso. Le trame di questo formato sono dette **S-frame**;
- il formato **unnumbered** è utilizzato per due scopi diversi: trasportare dati di utente in modalità non connessa e trasportare messaggi di controllo del collegamento (inizializzazione, diagnostica, ecc.). Le trame di questo formato sono dette **U-frame**.

Il campo **NETWORK PDU** può avere 0 o più byte (non è stabilito un limite massimo, ma PDU troppo grandi potrebbero essere frammentate dal sottolivello MAC) che contengono la PDU che il livello superiore (Network Layer) si attende di ricevere dal sottolivello MAC in ricezione o invia al sottolivello MAC in trasmissione.

Il sottolivello LLC prevede 3 modi di funzionamento:

- **Unacknowledged Connectionless Service:** costituito solo da primitive di trasferimento dati, è un servizio non affidabile e non orientato alla connessione.

È costituito da singoli datagrammi che vengono trasmessi in modo indipendente l'uno dall'altro, senza richiedere conferma sulla ricezione (ACK) e senza alcuna forma di correzione degli errori né di controllo di flusso. Qualora tali funzioni siano necessarie, devono essere fornite dai protocolli di livello superiore. Non sono richieste comunicazioni preliminari allo scambio dei dati;

- **Connection Oriented Service:** costituito da primitive di trasferimento e di apertura e chiusura di una connessione con le funzioni per il controllo di errore, di flusso e di conservazione della sequenza. È un servizio affidabile e orientato alla connessione quindi richiede la conferma della ricezione (ACK);
- **Semireliable Connectionless Service:** costituito anch'esso solo da primitive di trasferimento dati ma, pur essendo non orientato alla connessione, prevede una conferma di ricezione (ACK) per i datagrammi inviati e garantisce la consegna ordinata dei dati trasmessi. Anche in questo caso non sono richieste comunicazioni preliminari allo scambio dei dati.

2.2 Il sottolivello MAC

Il sottolivello inferiore è il **MAC** che risolve il problema dell'accesso al mezzo trasmissivo condiviso. Cioè il suo compito è arbitrare l'accesso all'unico mezzo trasmissivo comune tra tutti i sistemi che hanno necessità di trasmettere in una determinata rete.
Quindi mentre l'LLC è unico, si avrà invece uno standard MAC (condiviso tra sottolivello MAC e livello Physical) diverso per ogni tipo di rete e mezzo fisico di trasmissione.

Anche il frame del MAC (FIGURA 3) contiene due indirizzi di tipo **DSAP** e **SSAP**, detti proprio **indirizzi MAC** del sorgente e del destinatario, che hanno evidentemente lo scopo di identificare l'indirizzo fisico delle due peer entity che si stanno scambiando la PDU destinata al, o in arrivo dal, sottolivello LLC.

FIGURA 3 Il frame MAC

Il campo **LLC PDU** contiene il frame LLC (mostrato in Figura 2).
L'**FCS** (Frame Check Sequence) è la tecnica di correzione degli errori in trasmissione **CRC** (Cyclic Redundancy Check) affrontata nel terzo anno.

Ricordiamo anche che l'indirizzo MAC è costituito da 6 byte rappresentati da 12 cifre esadecimali raggruppate in sei coppie separate da un trattino.
Per esempio:

08-00-2B-C4-BE-F3

Gli indirizzi MAC possono essere di 3 tipi:

- **unicast**: individua una stazione singola;

#prendinota

La caratteristica fondamentale dell'indirizzo MAC è di essere univoco: non esistono due indirizzi MAC uguali, cioè non ci sono due schede di rete (quindi due host) con lo stesso MAC.

- **multicast**: individua un gruppo di stazioni. In questi casi si trasmette per primo il bit meno significativo (bit 0) del primo byte messo a 1. La rappresentazione formale è del tipo:

  FF-FF-FF-[0:F]X-XX-XX

  usata per indicare tutti gli host che hanno scheda di rete di qualsiasi produttore con la prima cifra del terzo byte compresa tra 0 e F;
- **broadcast**: **FF-FF-FF-FF-FF-FF** individua tutti gli host connessi alla rete.

Per sapere l'indirizzo MAC della scheda di rete del proprio computer in ambiente Windows è sufficiente aprire (per esempio con start-esegui-cmd) il prompt dei comandi e digitare **ipconfig/all**.
Tra le molte informazioni che compariranno ci sarà

Scheda Ethernet Connessione alla rete locale (LAN)

seguita dalla descrizione del produttore e del modello della scheda.
Subito sotto si potrà leggere l'indirizzo fisico, cioè il MAC address che stavamo cercando.

Anche le schede wireless hanno un indirizzo fisico e anch'esso è rilevabile col comando ipconfig/all.

Come in tutti i modelli a strati il passaggio di livello (e sottolivello in questo caso) produce l'operazione di **incapsulamento** del frame, come mostrato nella FIGURA 4.

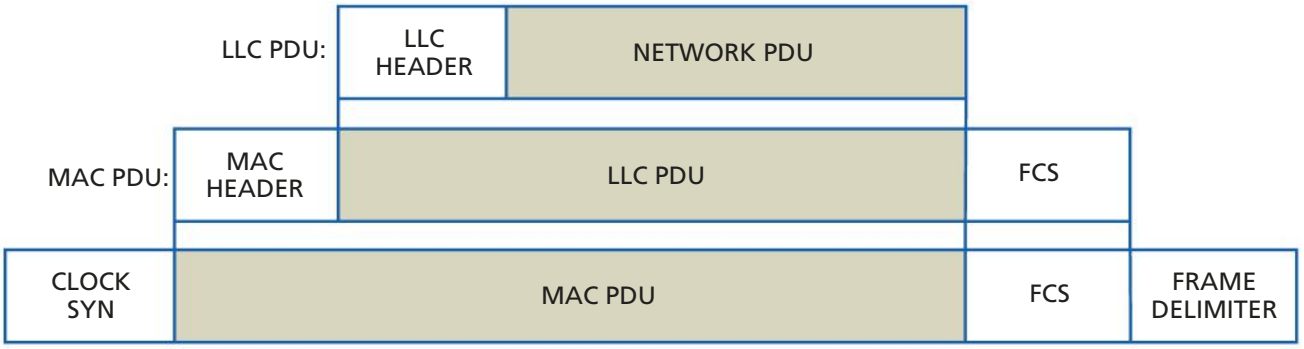


FIGURA 4 L'operazione di incapsulamento

Ricordiamo che nelle architetture a livelli, gli header incapsulati a un certo livello (o sottolivello) verranno letti solo dal corrispondente livello (o sottolivello) dell'host con cui si sta comunicando.
Il meccanismo dell'incapsulamento serve proprio a questo: a far sì che ogni livello (o sottolivello) inoltri la PDU incapsulata ricevuta, analizzandone solo l'header e disinteressandosi del resto.

FISSA LE CONOSCENZE

- Spiega a che cosa serve il sottolivello superiore LLC.
- Che cosa rappresentano i due indirizzi contenuti nel frame LLC?
- Spiega a che cosa serve il sottolivello MAC.
- Che cosa rappresentano i due indirizzi contenuti nel frame MAC?
- Gli indirizzi MAC si dividono in 3 tipi. Quali sono?

3 L'EVOLUZIONE DI LLC: HDLC E PPP

3.1 HDLC (High Level Data Link Control)

LLC oltre a essere un sottolivello del Physical Layer è un **protocollo di linea** (data link protocol) standardizzato nel Progetto 802 con la sigla IEEE 802.2.
I protocolli di linea sono i protocolli (di livello 2 ISO/OSI: Data Link) che vengono utilizzati sulle linee pubbliche per la trasmissione di dati, progettati per canali geografici di tipo punto-punto o multipunto.
Tali protocolli formano una famiglia i cui componenti più importanti sono LLC, HDLC e PPP.

Il protocollo adottato per LLC, il cui frame abbiamo visto nella Figura 2 della Lezione precedente, è una versione semplificata di HDLC.

Il **protocollo HDLC** è generalmente utilizzato su reti di grandi dimensioni. Può essere usato per connessioni multipunto, ma attualmente è usato quasi esclusivamente per collegamenti punto-punto. Lo standard OSI prevede esplicitamente l'adozione di HDLC.

Numerose sono state le specifiche emesse dall'ISO relative a questo protocollo, l'ultimo standard, pubblicato nel 2002, che rende obsoleti tutti i precedenti è ISO 13239.
In HDLC lo scambio delle informazioni avviene con frame di formato fisso.
Il frame HDLC è composto da 3 parti: un header (costituito dai campi address e control), un campo dati a lunghezza variabile e un trailer per il controllo degli errori (FCS). Il tutto racchiuso tra due sequenze di flag (FIGURA 5).

FIGURA 5 Il frame HDLC

flag	address	control	data	FCS	flag
01111110	8 bit	8 o 16 bit	lunghezza variabile, 0 o più bit a multipli di 8	16 o 32 bit	01111110
	|← header →|			|← trailer →|	

Vediamo nel dettaglio i singoli campi:

- **flag**: due particolari sequenze di 8 bit 01111110, che racchiudono ogni frame. Hanno il compito di stabilire la sincronizzazione, inoltre vengono trasmessi in modo continuativo quando non ci sono altre informazioni da trasmettere (la linea è idle). Accorgimenti particolari devono essere perciò usati nella trasmissione di sequenze di bit in cui figurino più di 5 bit a 1 consecutivi. In particolare, in trasmissione viene inserito un bit a 0 dopo 5 bit a 1 consecutivi (**bit stuffing**); in ricezione questo bit viene tolto in modo da ricostituire la sequenza originale;
- **address**: si utilizza solo per linee multipunto per identificare i diversi terminali, infatti il protocollo HDLC si usa di norma su link punto-punto e quindi non necessita di un indirizzo di destinazione;
- **control**: resta identico al campo control del frame LLC con i 3 formati: information (I-frame), supervisor (S-frame) e unnumbered (U-frame);

#prendinota

La tecnica del **bit stuffing** garantisce che solo il carattere flag contenga 6 bit a 1 consecutivi. Infatti il bit stuffing analizza la trama (flag esclusi) prima di trasmetterla e inserisce un bit a 0 dopo 5 bit a 1 consecutivi (indipendentemente dal valore del bit successivo). Il ricevitore, se riceve una sequenza di 5 bit a 1 e uno 0, elimina lo 0 che era stato inserito dal bit stuffing, se riceve 6 bit a 1 e uno 0 identifica il carattere flag.

#prendinota

La tecnica a finestra prevede di non inviare il riscontro a ogni messaggio ricevuto, ma solo dopo una finestra di *n* messaggi. La tecnica di **sliding window** è un'evoluzione della tecnica a finestra in cui non è più necessario terminare una finestra prima di aprirne un'altra.

- **data**: contiene i dati da trasmettere. Non esistono limiti di lunghezza per questo campo visto che sarà la sequenza flag a determinare la fine della trama (l'operazione di bit stuffing fa sì che non siano trasmesse sequenze consecutive di bit a 1 superiori a 5, evitando qualsiasi confusione con la sequenza di flag);
- **FCS** (Frame Check Sequence): è il codice di ridondanza ciclica (CRC) che viene utilizzato dal ricevitore per controllare la correttezza di quanto ricevuto.

Il protocollo HDLC sfrutta il sistema di trasmissione a finestre scorrevoli (**sliding window**) che permette un incremento della velocità generale del sistema.

I tempi di trasmissione possono essere relativamente lunghi e conviene pertanto spedire più frame di seguito, considerato che è assai più probabile che un frame risulti ricevuto correttamente piuttosto che distrutto o alterato.

3.2 PPP (Point to Point Protocol)

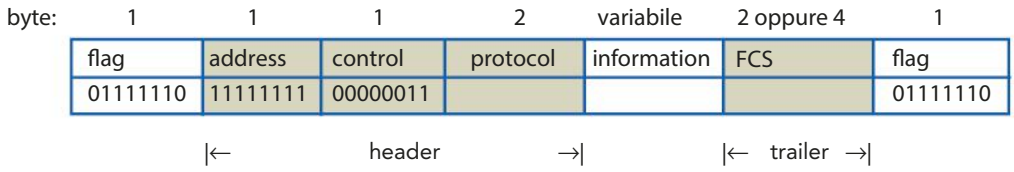
Il protocollo HDLC ha una grave carenza: non ha una modalità standard per trasmettere sullo stesso canale pacchetti generati da protocolli diversi, di livello superiore. Per questo motivo la comunità di Internet ha introdotto nel luglio 1990 una estensione di HDLC, basata sullo standard ISO 4335, detta **PPP**.

La differenza principale rispetto all'HDLC risiede nella presenza di un campo **protocol** lungo 2 byte. Tale campo contiene la codifica del protocollo di livello superiore la cui PDU è contenuta nel campo **information**, per esempio al protocollo IP corrisponde il codice esadecimale 0021h.

Il frame PPP è composto da 3 parti: un header (costituito dai campi address, control e protocol), un campo dati a lunghezza variabile e un trailer per il controllo degli errori (FCS).

Il tutto racchiuso tra due sequenze di flag (FIGURA 6).

FIGURA 6 Il frame PPP



Le specifiche di PPP definiscono anche il contenuto dei campi del frame, ponendo delle limitazioni:

- **address** deve sempre contenere la sequenza binaria 11111111 che corrisponde alla codifica broadcast. PPP non assegna indirizzi alle stazioni essendo un protocollo punto-punto;
- **control** deve sempre contenere la sequenza 00000011 a indicare che si tratta di un U-frame, cioè un frame senza numero di sequenza.
  La lunghezza del campo control è quindi sempre pari a 1 byte e la trasmissione è di tipo non connesso;
- **information** ha una lunghezza compresa tra 0 e 1500 byte (standard dei cosiddetti pacchetti Ethernet), anche se la lunghezza massima può essere cambiata su negoziazione tra i due host mittente e destinatario;

- **FCS** ha una lunghezza di 2 byte, ma può essere portato a 4 byte su negoziazione tra i due host, mittente e destinatario. È il codice di ridondanza ciclica (CRC) che viene utilizzato dal ricevitore per controllare la correttezza di quanto ricevuto.

PPP fornisce un metodo standard per trasmettere pacchetti provenienti da più protocolli diversi, sullo stesso collegamento seriale. Principalmente viene usato per la comunicazione punto-punto tra due router o nella comunicazione tra utente e provider (per esempio, tra Internet Service Provider e utente che accede tramite una connessione telefonica).

Per fare ciò utilizza:

- un protocollo di controllo **LCP** (Link Control Protocol) per creare, configurare e testare la linea; LCP stabilisce e termina la connessione PPP e negozia le opzioni di configurazione, per esempio la lunghezza dei campi protocol, information o FCS;
- una famiglia di protocolli **NCP** (Network Control Protocol) per configurare i diversi protocolli di rete; per esempio, nel caso del protocollo di rete IP, viene usato per negoziare l'attribuzione dell'indirizzo IP dinamico all'host (DHCP: Dynamic Host Configuration Protocol che vedremo nell'Unità 7).

I protocolli HDLC e PPP sono **protocolli sincroni**.

Nelle **trasmissioni sincrone** i dati da inviare sono raggruppati in frame di molti byte e ogni frame viene preceduto da alcuni byte per la sincronizzazione che permettono al ricevitore di sincronizzarsi con il trasmettitore. Il ricevitore ricava dai byte di sincronismo un segnale di clock locale che pilota la lettura dei restanti bit di dati.

La **trasmissione asincrona** invece permette di trasmettere e ricevere un solo byte per volta, delimitato da un bit di start e uno di stop. Inoltre non è definito il tempo che intercorre tra l'arrivo di un byte e il successivo.

Per questi motivi le trasmissioni sincrone sono più veloci di quelle asincrone, ma hanno lo svantaggio che un solo bit errato danneggia l'intero frame.

#prendinota

La trasmissione **asincrona** è così chiamata perché l'intervallo di tempo tra il bit finale di una trasmissione e il bit iniziale della successiva è indefinito.

FISSA LE CONOSCENZE

- Descrivi il frame del protocollo HDLC.
- In che cosa il protocollo PPP migliora il protocollo HDLC?
- In quali contesti si usa il protocollo PPP?
- Che differenza c'è tra una trasmissione sincrona e una asincrona?

4 IEEE 802.3: LA RETE ETHERNET

4.1 Evoluzione dello standard IEEE 802.3

Ethernet è il più diffuso tipo di rete locale che esista al mondo come abbiamo visto nell'Unità 8 del volume di terza. Nel 1985 Ethernet si evolve e diventa lo standard IEEE 802.3 (poi adottato da ISO con la sigla 8802) che col passare degli anni sviluppa tutta una serie di standard che si distinguono per mezzo fisico, velocità di trasmissione e topologia.
Si è passati dal cavo coassiale (non più utilizzato) al doppino e dal doppino alla fibra. Allo stesso modo le velocità sono salite dai 10 Mbps all'ordine delle decine di Gbps. Anche la topologia è cambiata abbandonando il bus per passare alla stella o stella estesa. La TABELLA 1 riassume i principali di tali standard.

TABELLA 1 L'evoluzione degli standard IEEE 802.3

802.3	**Ethernet su cavo coassiale** 10Base-5 (Thick Ethernet) 10Base-2 (Thin Ethernet)
802.3i	**Ethernet su doppino** 10Base-T (Twisted-pair)
802.3d	**Fibra ottica tra repeater per Thick e Thin Ethernet** FOIRL (Fiber Optic Inter-Repeater Link)
802.3j	**Ethernet su fibra ottica** 10Base-F (Fiber)
802.3y	**Fast Ethernet su doppino** 100Base-T (Twisted-pair)
802.3u	**Fast Ethernet su fibra ottica** 100Base-FX (Fiber)
802.3z	**Gigabit Ethernet su fibra ottica** 1000Base-SX (Short Wave) 1000Base-LX (Long Wave) **Gigabit Ethernet su doppino** 1000Base-CX (Shielded balanced copper cable)
802.3ab	**Gigabit Ethernet su doppino full-duplex** 1000Base-T (Twisted-pair)
802.3ae	**10 Gigabit Ethernet** 10GBase-LX (Long Wave)

La sempre maggiore richiesta di banda per un traffico in costante aumento ha comportato la necessità di avere ridondanza negli apparati e determinismo nei percorsi.

> La modalità half-duplex è stata sostituita dalla full-duplex. Questo, unito all'arrivo degli switch al posto degli hub, ha eliminato il problema delle collisioni sostituendo la tecnica CSMA/CD con lo switching.

Studieremo queste due tecniche nelle prossime Lezioni di questa unità.
Reti sempre più trafficate hanno portato a realizzare cavi full-duplex e a gestire i flussi di pacchetti in transito con tecniche come le sliding windows.

Reti locali sempre più complesse hanno portato all'implemetazione delle VLAN (Virtual LAN) di cui tratteremo al quinto anno. Dorsali di rete sempre più trafficate hanno poi suggerito di aggregare più link fisici in un singolo canale logico per avere più banda.
I principali miglioramenti apportati alla rete Ethernet sono riportati in TABELLA 2.

802.3x	Full-duplex e Controllo di flusso (sliding windows)
802.3ac	VLAN (Virtual LAN)
802.3ad	Aggregamento dei collegamenti
802.3af **802.3at**	PoE (Power over Ethernet) 15,4 W per porta PoE+ (Power over Ethernet) 30 W per porta

TABELLA 2 I principali miglioramenti della rete Ethernet

4.2 PoE (Power over Ethernet)

Quando all'inizio degli anni 2000 incominciò ad affermarsi la tecnologia VoIP (Voice over IP) nacque la necessità di telealimentare i telefoni per evitare di utilizzare alimentatori esterni e per rendere l'installazione più semplice e "pulita".
L'IEEE prese allora a sviluppare lo standard **802.3af** denominato **PoE**.

> PoE è una tecnologia che permette di alimentare gli apparati utilizzando lo stesso cavo che li collega alla rete Ethernet, a condizione che sia del tipo twisted-pair.

Per il momento queste tecniche sono utilizzate soprattutto nell'alimentazione di apparecchiature che richiedono poca potenza, nell'ordine di poche decine di watt, come telecamere industriali, telefoni VoIP, access point e webcam.
Risulta molto utile quando ci sono difficoltà nel reperimento di fonti elettriche in prossimità degli apparati di rete o anche per ridurre il numero di elementi e cavi.

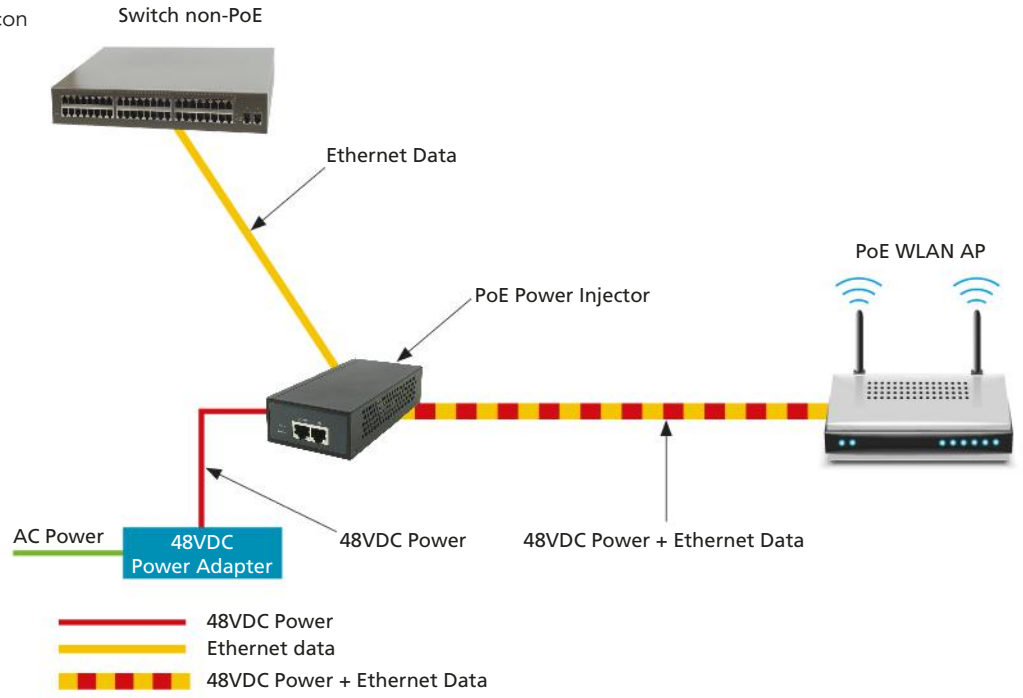
Nel dettaglio possiamo dire che PoE consente:
- di utilizzare un solo cavo sia per l'alimentazione che per la trasmissione dati; di fatto la tecnologia PoE comporta un risparmio di denaro sull'acquisto e utilizzo dei cavi per le apparecchiature di rete e i telefoni VoIP;
- di semplificare e rendere più conveniente l'installazione o l'espansione della rete negli edifici in cui sarebbe troppo dispendioso in termini di costi e sforzi installare nuove linee di alimentazione;
- di installare dispositivi in luoghi in cui sarebbe impossibile far arrivare l'alimentazione, come per esempio nei controsoffitti;
- di ridurre il numero di cavi e di prese elettriche richiesti in ambienti ad alta concentrazione di apparecchiature o all'interno di un rack (armadio di cablaggio).

Le specifiche PoE prevedono due tipi di dispositivi:
- **PSE** (Power Sourcing Equipment), sono le apparecchiature che forniscono l'alimentazione: switch PoE o power injector PoE destinati all'uso con switch non-PoE; il power injector viene interposto tra lo switch che non dispone di porte PoE e l'apparato da alimentare
- **PD** (Powered Device), sono i dispositivi che vengono alimentati: i telefoni VoIP, gli access point wireless e le webcam IP e in generale ogni apparato in tecnologia Ethernet.

Gli scenari che si presentano possono dunque essere di due tipi.

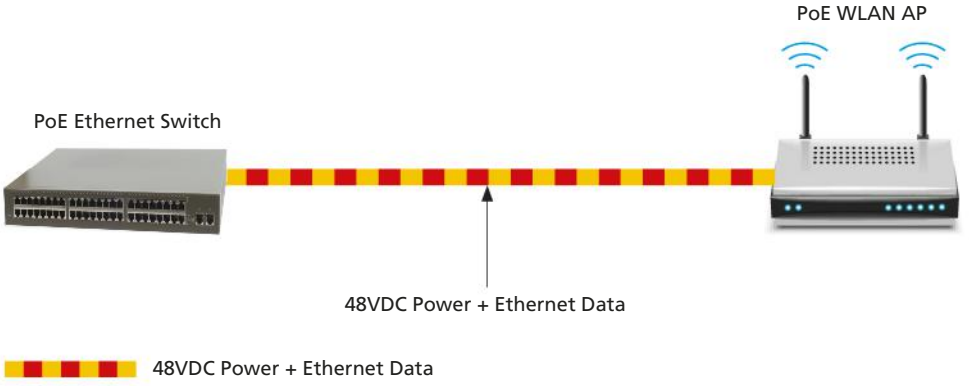
Il primo scenario, mostrato in FIGURA 7, in presenza di uno switch non-PoE, necessita di un **PoE Power Injector** per portare i dati e l'alimentazione all'access point che fa da PD.

FIGURA 7 Switch non-PoE con PoE Power Injector



Il secondo scenario invece, mostrato in FIGURA 8, mediante uno switch PoE porta direttamente i dati e l'alimentazione all'access point che fa da PD.

FIGURA 8 Switch PoE



#techwords

Nel linguaggio delle telecomunicazioni lo **splitter** (dall'inglese *to split*, separare) è un dispositivo che ha la funzionalità di ripartire la potenza di un segnale in ingresso tra due o più collegamenti in uscita.

Entrambi gli scenari, a fronte della presenza di molti dispositivi da collegare e alimentare, possono prevedere l'utilizzo di uno **#splitter PoE**. Lo splitter viene collegato allo switch PoE o al power injector PoE consentendo di raggiungere più dispositivi (anziché uno solo come nelle Figure 7 e 8).

La versione più recente della tecnologia PoE è lo standard **IEEE 802.3at**, ovvero la tecnologia **PoE+**.

La principale differenza tra lo standard 802.3af (PoE) e 802.3at (PoE+) risiede nel fatto

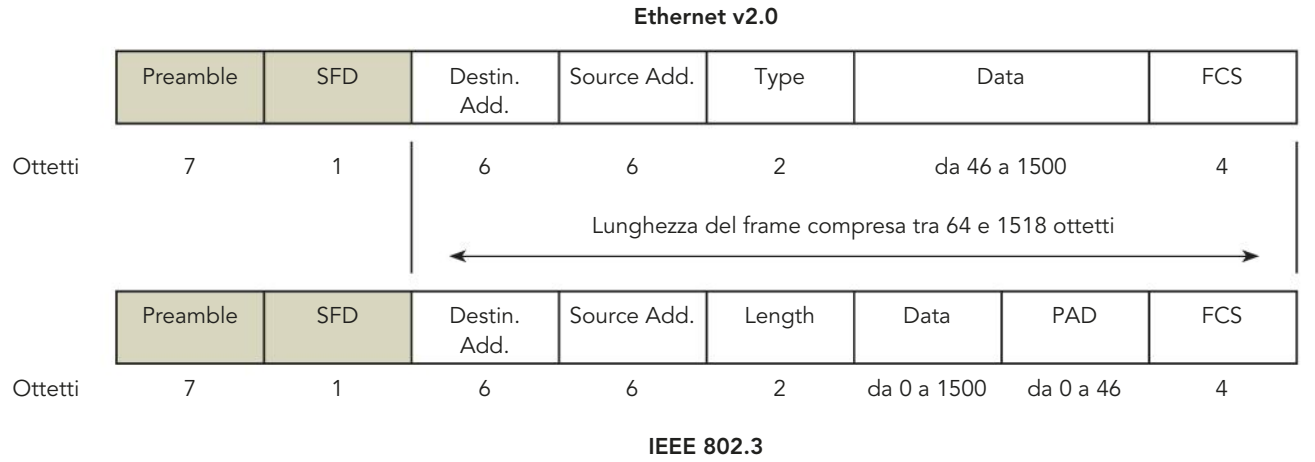
che le apparecchiature PSE di tipo PoE+ sono in grado di fornire quasi il doppio dell'alimentazione utilizzando un solo cavo Ethernet.
Le apparecchiature PSE PoE+ sono in grado di fornire alimentazione sia a dispositivi PD di tipo PoE, sia a quelli di tipo PoE+, mentre le apparecchiature PSE di tipo PoE possono fornire alimentazione solo ai dispositivi PD di tipo PoE. I dispositivi PD PoE+ richiedono infatti più alimentazione rispetto a quella che sono in grado di fornire le apparecchiature PSE PoE.

4.3 I frame Ethernet

Il frame Ethernet ha una lunghezza variabile compresa tra 64 e 1518 ottetti (byte), preceduti da un preambolo e da un byte di start.
Esistono due formati del frame che attualmente convivono sulle reti Ethernet (FIGURA 9):

- Ethernet v2.0;
- IEEE 802.3.

FIGURA 9 Formati del frame Ethernet v2.0 e IEEE 802.3



I campi del frame Ethernet v2.0 sono:

- **Preamble**: è un preambolo costituito da 7 byte tutti uguali con valori 10101010 allo scopo di permettere al destinatario di sincronizzarsi;
- **SFD** (Start Frame Delimiter): è un byte uguale a 11010101 che indica l'inizio del frame;
- **Destination Address**: contiene l'indirizzo fisico del destinatario (il MAC);
- **Source Address**: contiene l'indirizzo fisico del sorgente (il MAC);
- **Type**: contiene il codice associato al protocollo di livello superiore che ha generato la PDU (Protocol Data Unit) contenuta nel campo Data;
- **Data**: contiene i dati da trasmettere, può anche essere vuoto (frame di controllo);
- **FCS** (Frame Check Sequence): contiene i bit di chek del CRC per la rilevazione degli errori di trasmissione.

I campi del frame IEEE 802.3 hanno, come uniche differenze rispetto al frame v2.0, la presenza del campo Length al posto del campo Type e l'introduzione di un campo PAD.

I nuovi campi contengono:

- **Length**: contiene la lunghezza in byte del successivo campo Data;
- **PAD**: contiene una sequenza riempitiva che garantisce che la lunghezza minima del frame sia di 64 byte (campo dati vuoto) al fine di rendere possibile distinguere un frame da un frammento di frame a seguito di una collisione.

Gli insiemi di valori ammissibili nei due casi sono disgiunti: in 802.3 il campo Length può assumere valori nell'intervallo 0-1500, mentre le codifiche di Type in Ethernet sono tutte maggiori o uguali a 1536.

#prendinota

L'**inter-frame spacing** è noto anche come **IPG** (Inter-Packet Gap).

In entrambi i formati non esiste un segnalatore di fine frame. Tale ruolo è assunto dall'**inter-frame spacing**, che ha lo scopo di definire lo spazio temporale minimo tra due frame consecutivi. Il valore minimo standardizzato per l'inter-frame spacing delle reti Ethernet è di **96 bit time**.

Lo spazio temporale minimo tra due frame consecutivi dipende dalla velocità trasmissiva, quindi 96 bit time corrispondono a 9,6 µs a 10 Mbps, 960 ns a 100 Mbps, 96 ns a 1 Gbps, 9,6 ns a 10 Gbps e così via per le altre velocità. Notare come la distanza minima in termini temporali tra due frame consecutivi si riduca molto all'aumentare della velocità trasmissiva.

Un dispositivo in rete si dice che trasmette a **wire speed** (massima efficienza trasmissiva) quando l'elettronica è così efficiente da introdurre l'inter-frame spacing minimo tra i pacchetti di lunghezza minima. L'efficienza trasmissiva viene espressa in **pps** (packets per second) e viene valutata con i pacchetti di lunghezza minima di 64 ottetti. La TABELLA 3 mostra i diversi valori di pps con diverse lunghezze di pacchetti e velocità trasmissive.

TABELLA 3 Valori di pps con divere lunghezze di pacchetti e velocità trasmissive

Velocità Ethernet	64 ottetti - pps	75 ottetti - pps	1518 ottetti - pps
10 Mb/s	14.880	1.623	812
100 Mb/s	148.800	16.234	8.120
1 Gb/s	1.488.000	162.338	81.200
10 Gb/s	14.880.000	1.623.337	812.000
25 Gb/s	37.200.000	4.058.442	2.030.000
40 Gb/s	59.520.000	6.493.506	3.248.000
50 Gb/s	74.400.000	8.116.883	4.060.000
100 Gb/s	148.800.000	16.233.766	8.120.000

È molto comune trovare delle reti miste e, in particolare, è oggi molto diffusa la situazione in cui l'hardware è conforme allo standard IEEE 802.3 e continua a evolvere in termini di velocità trasmissive, ma il formato dei frame continua a essere quello di Ethernet v2.0. Questo non crea alcun problema alle schede di rete poiché in fase di ricezione sono comunque in grado di distinguere e trattare entrambi i tipi di frame.

FISSA LE CONOSCENZE

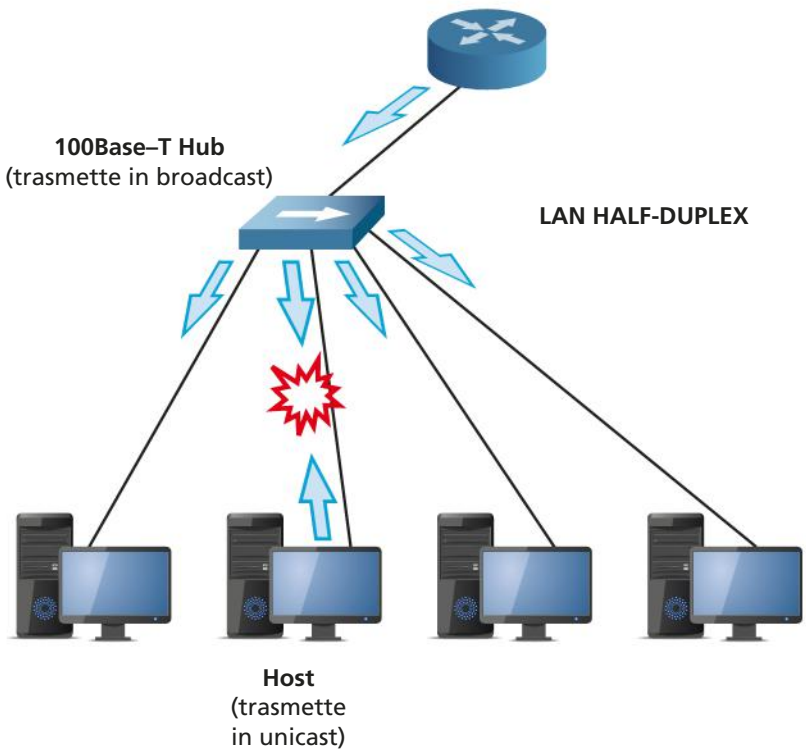
- Elenca i principali standard Ethernet.
- A che cosa servono i campi *Preamble* e *Start Frame Delimiter* nei frame Ethernet?
- Che differenze ci sono tra il frame Ethernet v2.0 e il frame IEEE 802.3?
- Che cos'è la tecnologia PoE?
- Che differenza c'è tra la PoE e la PoE+?

5 LA TECNICA A CONTESA CSMA/CD

5.1 La CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

Le reti Ethernet di tipo half-duplex (trasmissione bidirezionale ma uno per volta), solitamente dotate dei più economici hub rispetto ai più costosi switch, trasmettono utilizzando una tecnica a contesa che non prevede prenotazioni e relativo uso esclusivo del canale. Questo implica che occorre rilevare le eventuali collisioni (FIGURA 10).
Il dominio di broadcast viene a coincidere col dominio di collisione che include l'intera rete come abbiamo visto nell'Unità 8 del volume di terza.

FIGURA 10 Collisione



La CSMA/CD per gestire le collisioni prevede che l'host si metta in ascolto del canale di accesso condiviso (accesso multiplo: Multiple Access) e se lo trova inattivo (**idle**), cioè non rileva altri frame trasmessi da altre stazioni (rilevamento della portante: Carrier Sense), attende un tempo di 96 bit time (l'inter-frame spacing delle reti Ethernet pari a 9,6 μs per reti di questo tipo con velocità di 10 Mbps) e poi procede alla trasmissione; se invece è occupato (**busy**) attende che il canale torni libero prima di ritrasmettere.
Se però l'hub durante l'inter-frame spacing dell'host si è messo anch'esso in ascolto e non rilevando ancora nessuna trasmissione inizia a trasmettere, avviene una collisione.
In seguito all'avvenuto rilevamento della collisione (Collision Detection) si mette in moto il seguente procedimento:

- 1° passo: un qualsiasi host su quel segmento di rete, che si accorge per primo della collisione ricevendo frame incompleti (frammenti), interrompe la trasmissione;
- 2° passo: lo stesso host che per primo si è accorto della collisione immette sulla rete un frame speciale, diverso da tutti gli altri, noto come **sequenza di jamming** della lunghezza di 48 bit (dallo standard 802.3 in poi, prima era di 32 bit);
- 3° passo: gli host in ascolto, riconosciuto il jamming, interrompono anch'essi le trasmissioni e scartano i frammenti ricevuti;
- 4° passo: prima di ricominciare a trasmettere, ogni host attende un tempo pseudocasuale dato dall'**algoritmo di backoff esponenziale binario**.

Il tempo di attesa minimo (**Slot Time**) è pari alla lunghezza minima del frame Ethernet diviso la velocità del canale.
La lunghezza minima del frame è 64 byte.

#prendinota

Il protocollo implementa la direttiva: "Ascolta prima di trasmettere e mentre trasmetti. Se mentre trasmetti rilevi collisioni, fermati, segnala a tutte le altre stazioni la collisione e riprova più tardi secondo modalità di ritrasmissione stabilite".

La si ha nel caso di un frame di controllo con campo dati vuoto (0 byte):
6 (DSAP) + 6 (SSAP) + 2 (Data Length) + 0 (Data) + 46 (PAD) + 4 (CRC) = 64 B
Si noti che non vanno computati i campi di sincronizzazione Preamble e Start Frame Delimiter che non fanno parte del frame vero e proprio.

Se supponiamo il canale a velocità 10 Mbps, abbiamo che il tempo minimo necessario per inviare i 64 byte è:

$$\text{Slot Time} = \frac{64\ \text{B}}{10\ \text{Mbps}} = 51{,}2\ \mu\text{s}$$

L'algoritmo di backoff stabilisce che il tempo di attesa effettivo sia un multiplo r dello Slot Time, calcolato nel seguente modo:

- **Tempo di attesa effettivo = r · Slot Time**
- con $0 < r < 2^k - 1$
- con $k = \min(n, 10)$ dove k è il **limite di backoff** e n è il **numero di collisioni consecutive**.

L'algoritmo tende a **premiare gli host col minor numero di collisioni consecutive n**, poiché esse minimizzano k che a sua volta riduce il range in cui viene estratto random (in questo consiste la pseudocasualità) il fattore moltiplicativo r.
Notare che k può avere valore massimo pari a 10 perché così r può essere al massimo 1.023 ($2^{10} - 1$). Questo valore (1.023) è considerato la giusta via di mezzo tra il rischio da parte di un host di attendere troppo (si vuole avere un basso ritardo quando poche stazioni collidono) e il rischio, quando la collisione coinvolge tante stazioni, che molte di loro estraggano lo stesso intervallo di attesa e quindi si ripeta la collisione. L'algoritmo tende insomma ad adattarsi dinamicamente al numero di host che tentano di trasmettere e al numero di collisioni verificatesi consecutivamente.
Se si verificano 16 collisioni consecutive scatta una segnalazione d'errore al livello Network.
La TABELLA 4 riassume le caratteristiche della CSMA/CD per reti Ethernet half-duplex a 10 Mbps.

Max frame size	1518 byte
Min frame size	64 byte
Address size (MAC)	6 byte
Jam size	32-48 bit
Inter frame spacing	9,6 μs
Slot Time	51,2 μs
Attempt limit	16
Backoff limit	10

TABELLA 4 Caratteristiche della CSMA/CD

FISSA LE CONOSCENZE

- Quali reti Ethernet funzionano con la tecnica CSMA/CD?
- Descrivi i passi dell'algoritmo di backoff esponenziale binario.
- Qual è la lunghezza minima di un frame Ethernet? Perché?
- In quali condizioni lo Slot Time è pari a 51,2 μs?

6 LO SWITCHING

6.1 Reti con switching

Gli switch hanno il vantaggio che il dominio di collisione non coincide più col dominio di broadcast come invece avviene per gli hub. Questo riduce il traffico sulla rete ma non elimina le collisioni in canali half-duplex.

> A partire dallo standard 802.3x viene introdotto il **full-duplex** nelle reti Ethernet. L'arrivo degli switch full-duplex rende superata la tecnica CSMA/CD non avendo più il rischio delle collisioni.

Naturalmente l'uso dello stesso cavo in modo bidirezionale dimezza la banda a disposizione per ogni trasmissione.

Lo switch è un dispositivo di livello Data Link che assume un ruolo attivo nel bufferizzare e inoltrare i frame Ethernet. Il suo compito è esaminare l'indirizzo MAC del frame in arrivo e inoltrarlo selettivamente a uno o più link (porte Ethernet) in uscita. Il suo ruolo è trasparente essendo l'host ignaro della presenza dello switch ed è plug-and-play quindi gli switch (a differenza dei router) non devono essere configurati. La FIGURA 11 mostra che se l'host A trasmette dei frame all'host B e contemporaneamente l'host B trasmette dei frame all'host A, non ci sono collisioni. Lo switch full-duplex consentirà la trasmissione bidirezionale simultanea. Durante questa trasmissione tra A→B e tra B→A non sarà invece consentito all'host C di trasmettere simultaneamente verso A o verso B. Se C trasmettesse comunque (ignaro del fatto che il canale è occupato) i suoi frame verrebbero ritardati (mediante bufferizzazione) o addirittura scartati dallo switch.

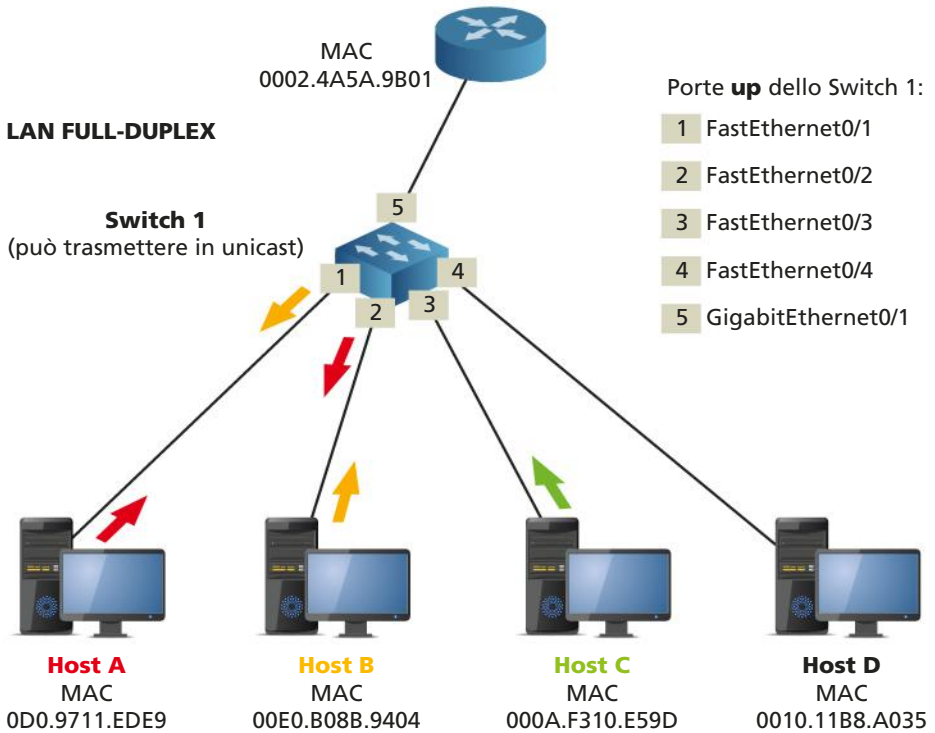


FIGURA 11 Switching

#prendinota

Oltre alla **MAC Table** usata dagli switch per gli indirizzi MAC in una LAN, ci sono altre 3 tabelle fondamentali che incontreremo nel nostro percorso:

- **ARP Table** (Address Resolution Protocol Table) usata da switch e router per gli indirizzi logici IP in una LAN;
- **Routing Table** usata dai router per gli indirizzi logici IP in una WAN;
- **NAT Table** (Network Address Translation Table) usata dai router per mascherare gli indirizzi logici IP in una WAN.

Come fa lo switch a sapere su quale linea (porta) inoltrare un frame avente un certo indirizzo MAC come destinatario?
Serve uno strumento che metta in corrispondenza il MAC address con la porta. Questo strumento è la **MAC Table** (anche detta Switch Table). In questa tabella ogni entry contiene la corrispondenza tra il MAC address dell'host collegato su una linea e la porta dello switch dedicata a tale linea.

Supponiamo che lo Switch 1 in Figura 11 abbia 24 porte FastEthernet e 2 porte GigabitEthernet. Collegando le prime 4 porte FastEthernet agli host e la prima porta GigabitEthernet al router, avremo, dopo lo scambio di alcuni frame tra gli host o tra gli host e il router, la MAC Table completa della rete mostrata in FIGURA 12.

MAC Table for Switch 1

VLAN	Mac Address	Port
1	0002.4A5A.9B01	GigabitEthernet0/1
1	000A.F310.E59D	FastEthernet0/3
1	0010.11B8.A035	FastEthernet0/4
1	00D0.9711.EDE9	FastEthernet0/1
1	00E0.B08B.9404	FastEthernet0/2

FIGURA 12 MAC Table dello Switch 1

Poiché in una rete LAN vi sono normalmente più switch, ognuno di essi compilerà una propria MAC Table, arricchendola di una nuova entry quando arriva un frame contenente un MAC address non ancora mappato su una propria porta.

6.2 Le tecniche di switching

L'uso degli switch è fondamentale per segmentare le reti e rendere compatibili le velocità in reti da 10/100/1.000 Mbps fino a 10 Gbps e oltre. Lo switch esegue tutte le proprie elaborazioni via hardware e non via software, perciò non rallenta il fluire del traffico tra i segmenti. Si dice che la connessione è **wire speed**, cioè lascia transitare i pacchetti alla velocità massima consentita dal tipo di conduttore usato per il cablaggio. Nella realtà, un rallentamento esiste sempre, anche se marginale, e la sua entità dipende dal modo in cui lo switch funziona. La prima tecnica di switching si chiama **store-and-forward**. Ogni frame che arriva su una delle porte dello switch viene salvato in un buffer e quindi inoltrato o scartato a seconda che l'indirizzo di destinazione (MAC address) sia corretto oppure no.

L'operazione è velocissima, ma comporta in ogni caso un certo rallentamento perché il frame deve arrivare per intero nel buffer dello switch prima di cominciare a essere ritrasmesso su un'altra porta a cui corrisponde un altro segmento di rete. È la tecnica di inoltro più affidabile, poiché prima di rispedire il pacchetto lo switch si accerta di averlo per intero e ne verifica la correttezza attraverso il calcolo del CRC. Inoltre è l'unica tecnica utilizzabile quando si collegano segmenti funzionanti a velocità diverse, come Ethernet e FastEthernet, per esempio. Tuttavia su impianti molto veloci, come nel caso di una dorsale GigabitEthernet, il numero di frame in circolazione è molto elevato e il ritardo che si accumula per la registrazione di ciascuno si fa sentire.

L'alternativa ideata per eliminare quest'ultimo inconveniente è lo switching **cut-through**. La parola significa "prendere una scorciatoia" e in effetti è proprio quello che accade. Non appena lo switch comincia a ricevere un frame su una qualsiasi delle sue porte, ne legge l'indirizzo di destinazione e, se questo corrisponde a un segmento collegato a un'altra porta, inizia immediatamente a trasmettere il frame senza aspettare che questo sia arrivato per intero. In questo modo, dopo aver letto l'indirizzo, la trasmissione in uscita avviene quasi in contemporanea con la ricezione e il ritardo è minimo (fino a 20 volte inferiore a quello della tecnica store-and-forward). Benché molto efficace sotto il profilo della velocità, questa tecnica presenta lo svantaggio di far passare anche i frame difettosi o frammentati. Lo switch si limita a controllare l'indirizzo e quindi fa passare tutto quel che segue senza controllo alcuno.

Dal confronto di questi due approcci, ne è stato ideato un terzo, intermedio, che si chiama **fragment-free**, il quale applica lo store-and-forward ma solo per i primi 64 byte allo scopo di verificare eventuali errori che statisticamente cadono più frequentemente nei primi byte del frame.

6.3 La vulnerabilità degli switch

Per proteggere una LAN è utile attivare la **protezione delle porte** negli switch (o nei bridge). I frame il cui indirizzo MAC non è specificato per una determinata porta dello switch, non potranno accedere allo switch attraverso quella porta e di conseguenza non potranno accedere alle reti alle quali lo switch è connesso.
Ogni porta dello switch, di regola, può fornire protezione a 1.024 indirizzi MAC (anche se alcuni switch arrivano solo a 64 indirizzi MAC) più uno predefinito per ogni porta. Il numero massimo di indirizzi MAC per ciascuna porta dipende dalla configurazione della LAN, l'unica limitazione consiste nel fatto che il numero totale di indirizzi MAC non può superare il 1.024 più uno per porta.
Uno switch a 8 porte potrà quindi fornire protezione a 1.032 indirizzi MAC: 1.024 (suddivisi tra le 8 porte) + 8 (uno per porta).

Un esempio valido di allocazione per uno switch a 8 porte potrebbe essere quello mostrato nella FIGURA 13.

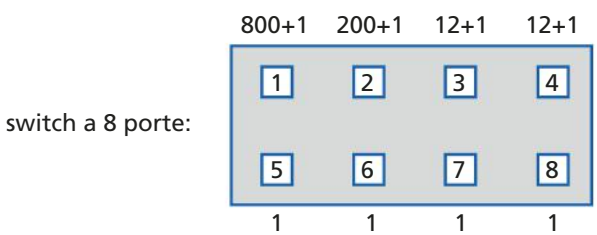


FIGURA 13 Esempio di protezione in uno switch a 8 porte

Si hanno:

- 801 (800 + 1) indirizzi MAC sulla porta 1;
- 201 (200 + 1) indirizzi MAC sulla porta 2;
- 13 (12 + 1) indirizzi MAC sulla porta 3;
- 13 (12 + 1) indirizzi MAC sulla porta 4;
- 4 indirizzi MAC, uno per ciascuna delle restanti 4 porte (5-6-7-8).

Dopo aver allocato il numero massimo di indirizzi MAC su una porta, è possibile:

- specificare manualmente tutti gli indirizzi;
- consentire alla porta di configurare automaticamente tutti gli indirizzi;
- specificare manualmente alcuni indirizzi e lasciare che gli altri siano configurati automaticamente.

L'elenco degli indirizzi MAC mappati per ciascuna porta, vengono memorizzati nella NVRAM (Nonvolatile Rapid-Access Memory), che è un tipo di RAM in grado di non perdere i dati caricati, anche in caso di un'interruzione della corrente elettrica, grazie alla presenza di una batteria o di un accumulatore.
Di default tutti gli indirizzi specificati per ogni porta sono protetti permanentemente. È però possibile specificare un **intervallo di durata** per ogni porta, allo scadere del quale gli indirizzi MAC della porta perdono la protezione.
Quando una porta protetta dello switch riceve un frame, l'indirizzo MAC del mittente (presente nel frame) viene confrontato con l'elenco di indirizzi MAC, memorizzati nella NVRAM, accettati da quella porta. Se l'indirizzo non risulta in elenco, la porta passa in modalità **disattivazione**. La porta può rimanere in questo stato in modo permanente oppure per un certo intervallo di tempo preconfigurato. Nella modalità disattivazione nessun frame può passare dalla porta.
In presenza di un vero e proprio tentativo di violazione della protezione, la porta può passare o in modalità disattivazione o in modalità restrittiva. Quest'ultima consente alla porta di restare attiva, permettendo il passaggio dei frame con indirizzo MAC riconosciuto e bloccando gli altri.

FISSA LE CONOSCENZE

- Quali reti Ethernet funzionano con la tecnica di switching?
- Che differenza c'è fra half-duplex e full-duplex?
- Che cosa contiene la MAC Table?
- Descrivi la tecnica di switching store-and-forward.
- Dopo aver allocato il numero massimo di indirizzi MAC su una porta cosa è possibile fare?
- Qual è di norma il numero massimo di indirizzi MAC configurabile su una porta?
- Che cos'è e a che cosa serve una NVRAM?

7 IEEE 802.11: LA RETE WI-FI

7.1 Gli standard wireless

Lo standard wireless 802.11 nacque nel 1997 ma praticamente rimase solo sulla carta per via delle insufficienti prestazioni che consentiva (tra cui velocità solo fino a 1 o 2 Mbps).

Nel 1999 la IEEE emise due nuovi standard:

- **802.11a** che, sfruttando una delle più versatili tecniche di modulazione (QAM-64), poteva raggiungere i 54 Mbps a 5,2 GHz;
- **802.11b** con due nuove velocità: 5,5 Mbps e 11 Mbps a 2,4 GHz.

Lo standard che ebbe più successo fu l'802.11b, perché molti governi (tra cui quello italiano) hanno mantenuto libere alcune bande di frequenze tra cui quella a 2,4 GHz, nota come banda ISM (Industrial, Scientific and Medical). Tale frequenza può essere usata liberamente da chiunque, senza dover richiedere licenze, a patto di rispettare precisi limiti di potenza e di utilizzare tecniche di spread spectrum che consistono nel distribuire il segnale su una banda molto più larga del necessario, in modo che esso appaia come rumore ai dispositivi non interessati, con lo scopo di limitare le interferenze fra i diversi dispositivi.

Inoltre all'aumentare della frequenza aumentano gli effetti di riflessione e di assorbimento delle onde elettromagnetiche e di conseguenza diminuiscono le distanze raggiungibili. In particolare, a 2,4 GHz è possibile coprire una distanza 4 volte superiore rispetto a 5 GHz (circa 80 m a 2,4 GHz contro 20 m a 5 GHz, in assenza di ostacoli). Per questi motivi, la maggioranza degli standard utilizza la banda ISM a 2,4 GHz.

Molti apparecchi sfruttano le bande ISM che possono interferire con il normale funzionamento delle reti wireless domestiche o aziendali:

- telefoni cordless;
- forni a microonde;
- radiocomandi per cancelli automatici, sistemi d'allarme e giocattoli;
- apparati radar;
- Bluetooth.

Lo standard 802.11b a 11 Mbps è anche noto come marchio **Wi-Fi** (Wireless Fidelity) creato dalla **Wi-Fi Alliance**.

L'unico svantaggio della 802.11b rispetto alla 802.11a è la velocità (11 Mbps contro 54 Mbps).
Nel 2003 l'IEEE propose una sua variante all'802.11b, l'**802.11g**, in grado di raggiungere i 54 Mbps nella banda ISM tradizionale a 2,4 GHz, mantenendo inoltre la compatibilità verso il basso con i dispositivi 802.11b.
Gli standard 802.11b (Wi-Fi) e 802.11g (compatibile Wi-Fi) dividono lo spettro in 14 sottocanali (di cui 13 utilizzabili in Europa) con larghezza di banda pari a 22 MHz ciascuno nell'intervallo 2.412-2.484 MHz (FIGURA 14).
I canali sono parzialmente sovrapposti tra loro in frequenza, quindi tra due canali consecutivi esiste una forte interferenza.

#prendinota

La **Wi-Fi Alliance** è stata formata nel 1999 per guidare l'adozione di un unico standard per la banda larga senza fili nel mondo. Wi-Fi Alliance è inoltre proprietaria del trademark Wi-Fi che certifica l'interoperabilità di un dispositivo con lo standard wireless IEEE 802.11.

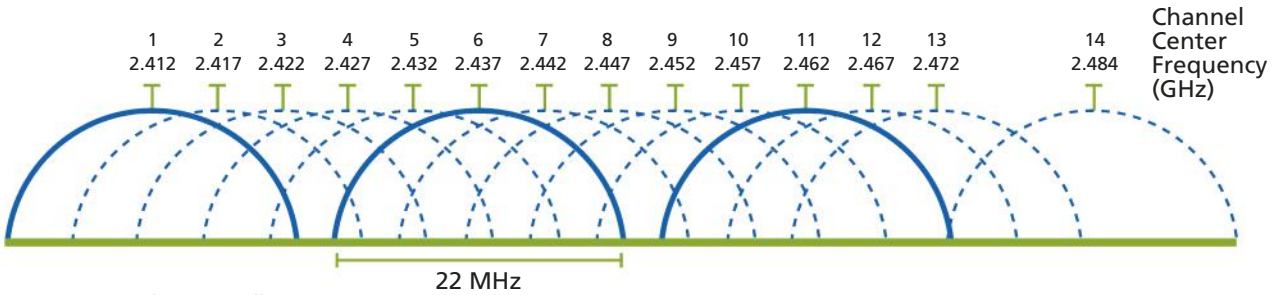


FIGURA 14 Canali Wi-Fi nella banda 2,4 GHz

In caso di presenza di più reti wireless, per evitare sovrapposizioni, si usa la **regola del 5**. Si usano cioè i 2 gruppi di canali distanti 5: **1-6-11 e 2-7-12** che non si sovrappongono **(FIGURA 15)**.

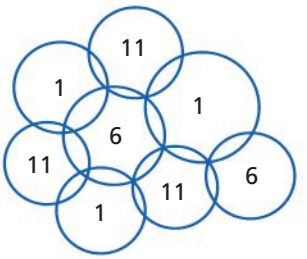


FIGURA 15 Regola del 5

Molte varianti si sono succedute negli anni contraddistinte da lettere diverse. La **TABELLA 5** riassume i principali standard attualmente approvati dall'IEEE e commercializzati dalla **Wi-Fi Alliance** con sigle a partire da Wi-Fi 4:

TABELLA 5 I principali standard IEEE 802.11

STANDARD	BANDA DI LAVORO	VELOCITÀ
802.11a	5,2 GHz	54 Mbps
802.11b	2,4 GHz	11 Mbps
802.11g	2,4 GHz	54 Mbps
802.11n (Wi-Fi 4)	2,4 GHz e 5,2 GHz (Dual Band)	300 Mbps (poi 450 Mbps)
802.11ac (Wi-Fi 5)	5,2 GHz	1,3 Gbps
802.11ax (Wi-Fi 6)	7 GHz (compatibile 2,4 GHz e 5,2 GHz)	10 Gbps

Per il 2024 è in fase di progettazione il nuovo standard **802.11be**, evoluzione dell'802.11ax, a 30 Gbps e con frequenze a 2,4 GHz, 5,2 GHz e 6GHz.

7.2 LAN wireless e wired

I dispositivi che costituiscono le reti wireless sono due:

- i **Wireless Terminal** (WT): sono dispositivi mobili (notebook, netbook, tablet, smartphone, ...) dotati di interfaccia 802.11 integrata o su schede PCMCIA o USB, oppure fissi (Personal Computer, stampanti di rete) con schede PCI o adattatori USB;
- gli **Access Point** (AP): hanno un doppio scopo, da un lato sono dei bridge che collegano la parte cablata (wired) con la parte wireless, dall'altro consentono ai WT di collegarsi alla rete wireless (agiscono quindi da gateway). È possibile anche usare dei computer dotati di apposito software per funzionare da AP.

È possibile inoltre collegare più AP alla rete cablata o collegare tra loro più AP e poi collegarli alla rete cablata creando una rete LAN mista, con parte wired e wireless **(FIGURA 16)**.

Le reti LAN attuali sono tutte realizzate con gli standard Ethernet e Wi-Fi.

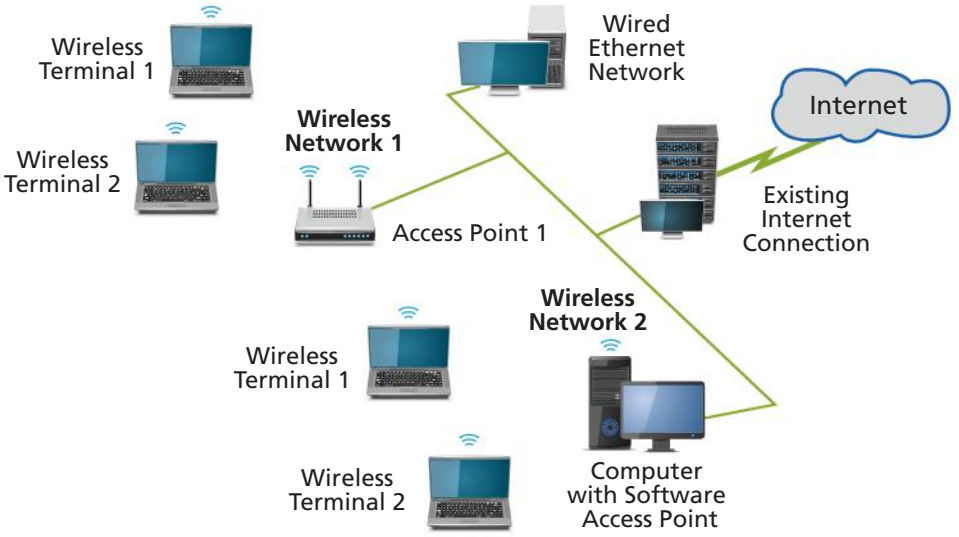


FIGURA 16 LAN con parte wired e parte wireless

7.3 Problematiche delle LAN wireless

Nelle trasmissioni wireless non è possibile rilevare le collisioni (Collision Detection). Tecniche come la CSMA/CD delle reti Ethernet non sono applicabili a un mezzo fisico tipicamente half-duplex come sono le trasmissioni via etere.
Le collisioni vanno dunque evitate a priori. Il modo più semplice è quello di costringere la stazione trasmittente ad ascoltare il canale e verificare che sia libero prima di iniziare la trasmissione. In alcuni casi, però, questo semplice accorgimento non basta. Due scenari, quello della stazione esposta e quello della stazione nascosta, lo dimostrano.

a) Problema della stazione esposta (FIGURA 17):
supponiamo di avere quattro stazioni, A, B, C e D, con i raggi d'azione di B e C raffigurati e che B stia trasmettendo ad A mentre C voglia trasmettere a D.
Ascoltando il canale, C sentirà la trasmissione di B e concluderà erroneamente di non poter trasmettere; invece, essendo D fuori della portata di B, e A fuori della portata di C, le due trasmissioni potrebbero avvenire parallelamente senza interferenze.

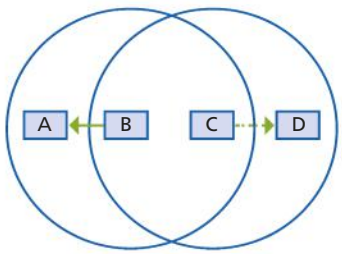


FIGURA 17 Stazione esposta

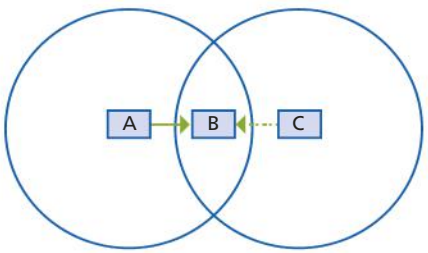
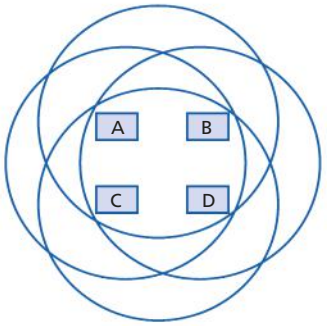


FIGURA 18 Stazione nascosta

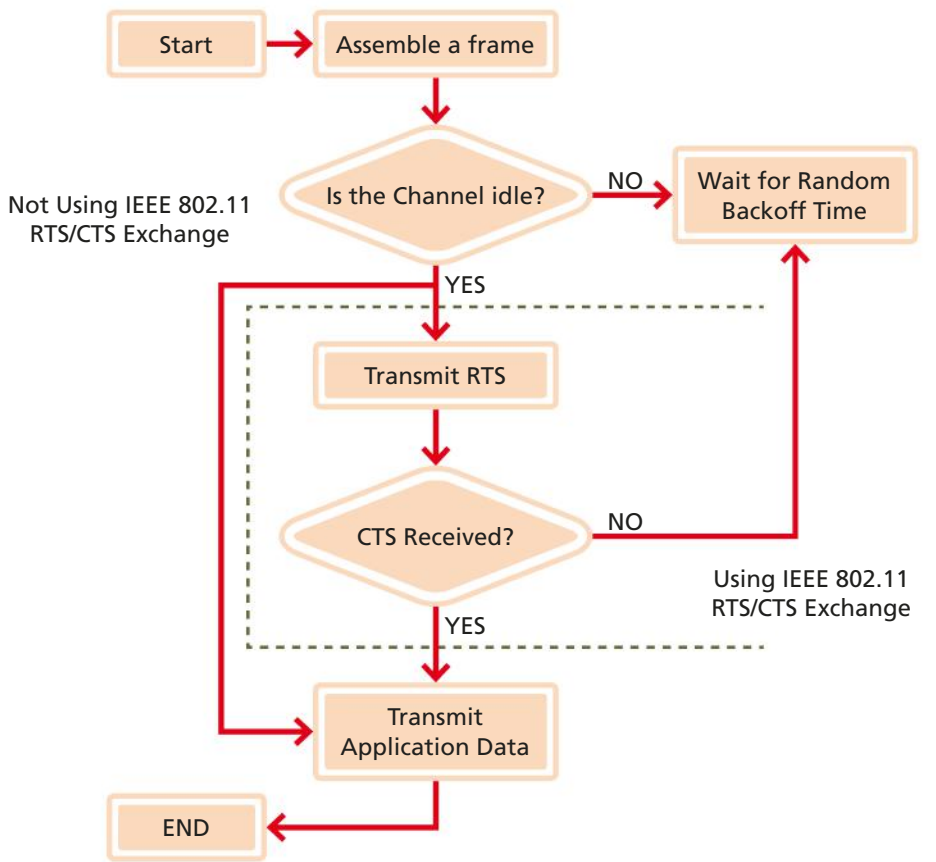
b) Problema della stazione nascosta (FIGURA 18):
supponiamo di avere tre stazioni, A, B e C, con i raggi d'azione di A e C raffigurati, e che A stia trasmettendo a B. Se ora C ascolta il canale, lo troverà libero e sarà convinta di poter trasmettere a B; cominciando a trasmettere disturberà la trasmissione di A, impedendo a B di riceverla; sia A che C saranno costrette a ritrasmettere.

FIGURA 19 Stazioni tutte nei rispettivi raggi d'azione



Il primo problema (stazione esposta) può essere risolto solo da un'accurata progettazione fisica della rete, sistemando le stazioni tutte nei rispettivi raggi d'azione (situazione ideale ma comunque assai frequente) (FIGURA 19).
Il secondo problema (stazione nascosta) è invece risolvibile anche mediante tecniche di **Carrier Sensing virtuale**. Questa tecnica di ascolto consiste innanzitutto nell'invio, da parte del mittente, di un frame **RTS** (Request To Send) al destinatario contenente l'informazione sulla durata della trasmissione che intende effettuare. Il destinatario risponde con un frame **CTS** (Clear To Send) in cui copia il valore relativo alla durata. Entrambi i frame RTS e CTS contengono quindi l'informazione della durata della trasmissione del mittente. Alla ricezione del frame CTS, il mittente può cominciare a trasmettere. Ogni altra stazione nel raggio d'azione delle due stazioni riceverà uno o entrambi i frame, sarà quindi al corrente della durata della trasmissione ed eviterà di trasmettere per quella durata per non creare interferenze.
Il Carrier Sensing virtuale ha però il problema di non garantire la mutua esclusione nell'uso del canale e le conseguenti collisioni (interferenze) qualora vi siano tentativi di acquisizione del canale **contemporanei**. Basti pensare che lo scambio dei frame RTS e CTS non può avvenire in un tempo infinitesimo e dunque c'è il rischio che una stazione terza avvii anch'essa lo scambio dei due frame e proceda contestualmente alla trasmissione.
La soluzione più efficace è stata messa a punto con una tecnica di tipo CSMA che riduce, pur non eliminando del tutto, le collisioni: la **CSMA/CA** (Carrier Sense Multiple Access with Collision Avoidance). Nella forma più semplice la CSMA/CA esegue la sequenza di passi mostrata in FIGURA 20 con lo scambio di pacchetti RTS/CTS.

FIGURA 20 Algoritmo CSMA/CA semplificato



Questa tecnica introduce un intervallo di tempo chiamato **AIFS** (Arbitration Inter-Frame Space) durante il quale il trasmettitore attende al fine di accertarsi che non vi siano altri frame RTS o CTS sul canale (lo standard definisce 4 intervalli di tempo AIFS, i quali forniscono diversi livelli di priorità ai vari protocolli). Se non ve ne sono procede alla trasmissione dei suoi frame. Se però un'altra stazione contestualmente tenta di trasmettere provocando una collisione (il mittente se ne accorge perché non riceve il CTS di risposta dal destinatario), allora viene avviato un algoritmo di **backoff esponenziale binario** concettualmente simile a quello precedentemente descritto per la tecnica CSMA/CD dell'Ethernet. In pratica si tratta di attendere un tempo random prima di ritentare l'invio di un frame RTS.
Questo algoritmo di backoff si basa su una finestra di contesa (**slotted backoff window**), cioè una finestra dotata di un certo numero di Slot Time che ne rappresentano la larghezza, indicata con **CW** (Contention Window). L'algoritmo di backoff sceglierà random uno di questi Slot Time entro il limite rappresentato da CW e la stazione mittente ritenterà la trasmissione durante quello Slot Time. Se allo scadere dell'intervallo di tempo predefinito AIFS il mittente non riceve nuovamente il CTS dal destinatario, vuol dire, molto probabilmente, che l'RTS del mittente ha colliso ancora con un altro frame; spesso ciò significa che due stazioni hanno scelto lo stesso Slot Time nella finestra di contesa. Per questo motivo, prima di ritentare la trasmissione, il mittente raddoppia la dimensione della finestra di contesa (CW) e poi ripete l'algoritmo di backoff. Lo scopo di tale raddoppio è quello di adattare la dimensione della finestra al numero di contendenti, in considerazione del fatto che le collisioni sono indice di affollamento.

La finestra di contesa (CW) è generalmente inizializzata a 7 (CW minimo); le ritrasmissioni, come detto, implicano un raddoppio del numero di Slot Time, portando CW a 15, 31, 63, 127, 255. Arrivata a 255 (CW massimo), la dimensione non cresce più. Al corretto completamento di una trasmissione, CW viene riportato a 7.
In questo modo si riduce la probabilità che più stazioni, in attesa che il canale si liberi, tentino di acquisire contemporaneamente il canale quando questo viene rilasciato. È infatti poco probabile che due mittenti peschino random lo stesso numero (Slot Time) se il range, rappresentato da CW, viene ogni volta raddoppiato.

FISSA LE CONOSCENZE

- Quali sono le principali bande di frequenza a cui trasmettono i dispositivi Wi-Fi?
- Come si chiamano i due dispositivi che costituiscono la rete wireless?
- Descrivi il problema della stazione nascosta.
- A che cosa servono i frame RTS e CTS?
- Descrivi il problema della stazione esposta.
- Su che cosa si basa l'algoritmo di backoff esponenziale binario usato dalla tecnica CSMA/CA?

8 WIRESHARK: IL PROTOCOLLO ETHERNET

Utilizziamo Wireshark per analizzare il frame del protocollo Ethernet e verificare quanto studiato nella Lezione 4.

esercizio

➜ PROBLEMA

Dopo aver catturato dei pacchetti relativi alla richiesta di una pagina web, visualizzarne il frame Ethernet individuando il contenuto di ciascun campo.

➜ SVOLGIMENTO

Per poter rispondere alle richieste dell'esercizio dobbiamo prima procedere alla cattura dei pacchetti e all'individuazione di quello relativo alla richiesta (GET) di una pagina web (HTTP), per esempio *www.wireshark.org*.

Bisogna quindi prima eseguire le seguenti azioni:

- svuotare la cache del browser (cancellarne la cronologia);
- mediante il menu **Cattura** → **Opzioni** di Wireshark aprire la scheda **Interfacce di cattura** dove selezionare l'interfaccia di rete sulla quale avviare la cattura dei pacchetti;
- impostare il filtraggio dei pacchetti destinati a *www.wireshark.org* mediante la procedura spiegata nella Lezione 6 dell'Unità 1;
- avviare la cattura;
- accedere al sito *www.wireshark.org*;
- interrompere la cattura;
- trovare la riga, nella finestra di visualizzazione, relativa alla GET con protocollo HTTP.

Nella FIGURA 21 è stato evidenziato il messaggio HTTP da analizzare e nella sezione sotto si è espanso Ethernet II, così da avere in evidenza i dati contenuti nel frame Ethernet.

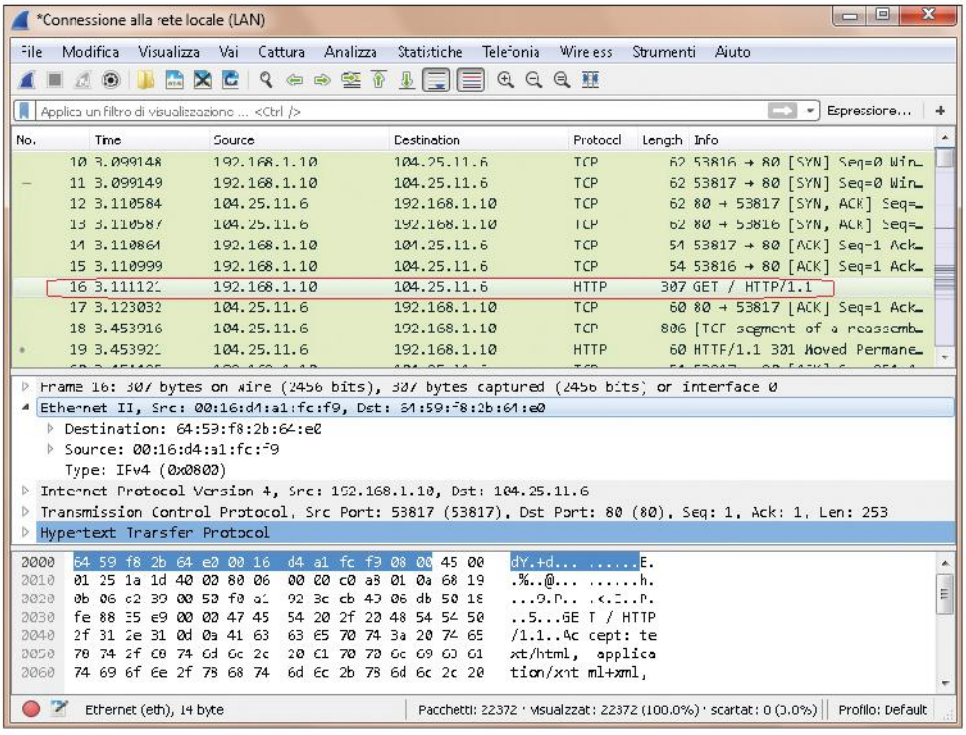


FIGURA 21 Finestra di Wireshark con i pacchetti catturati

Ora che abbiamo individuato i pacchetti da analizzare, cerchiamo di trarne le informazioni richieste sul frame Ethernet rispondendo alle seguenti domande:

D1. Nel pacchetto che contiene il messaggio HTTP GET, qual è l'indirizzo MAC del mittente?

R. L'indirizzo MAC mittente è: **00:16:d4:a1:fc:f9**

D2. Qual è l'indirizzo MAC del destinatario? È l'indirizzo MAC del web server *www.wireshark.org*?

R. L'indirizzo MAC destinatario è: **64:59:f8:2b:64:e0**
Non è l'indirizzo del web server, ma è l'indirizzo del router della rete locale o del default gateway.

D3. Qual è il valore esadecimale del campo Type nell'header del frame Ethernet? A quale protocollo di livello superiore corrisponde?

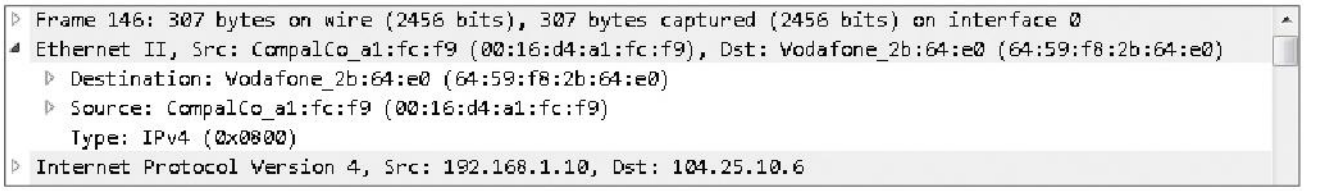
R. Campo Type = **0x0800**, tale valore indica che il protocollo di livello superiore è IPv4.

D4. Wireshark visualizza anche il nome del produttore della scheda di rete del mittente e del destinatario, da dove ricava tale informazione?

R. L'indirizzo MAC contiene il campo OUI (Organizationally Unique Identifier) che identifica univocamente il produttore. Wireshark ricerca nel OUI database (*www.wireshark.org/tools/oui-lookup.html*) quello contenuto nel MAC address del pacchetto. Nell'indirizzo mittente 00:16:d4 identifica il produttore Compal Communications, Inc.

La **FIGURA 22** mostra lo stesso frame Ethernet visualizzato in Figura 21, ma questa volta si è scelto di abilitare l'opzione "Risolvi gli indirizzi MAC".

FIGURA 22 Analisi del frame Ethernet con l'opzione "Risolvi gli indirizzi MAC"



D5. Selezionare il primo pacchetto catturato. Che tipo di pacchetto è? Qual è l'indirizzo MAC di destinazione? Chi realmente riceve questo pacchetto?

R. Il tipo di frame è **0x0806**, tale valore indica che il protocollo di livello superiore è ARP.
L'indirizzo MAC di destinazione è: **FF:FF:FF:FF:FF:FF**
Tutti i computer sulla rete locale riceveranno questo pacchetto in quanto l'indirizzo di destinazione è di broadcast.

FISSA LE CONOSCENZE

- Come potremmo ricavare l'indirizzo MAC del mittente se avessimo a disposizione solo l'area che visualizza il contenuto in esadecimale del pacchetto selezionato (area più in basso)?
- Quale altro campo del frame Ethernet v2.0 visto nella Lezione 4 puoi individuare nella stessa area?

9 PACKET TRACER: RETE ETHERNET E WI-FI

CONFIGURIAMO UNA LAN

In questa esercitazione realizzeremo con il simulatore Packet Tracer quanto studiato nelle Lezioni 4 e 7. Ogni rete LAN deve oggi avere una componente cablata che collega tutti i PC desk e gli apparati di rete e una componente senza fili per consentire il collegamento dei dispositivi mobili.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

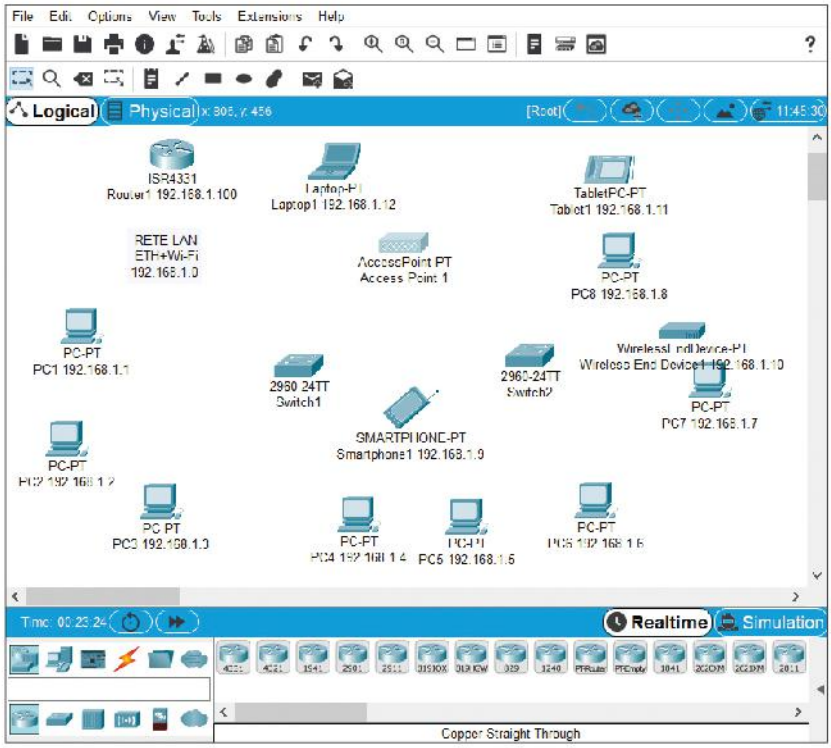
Realizzare una piccola rete LAN che ha sia la parte wired con standard 802.3 (Ethernet) e realizzata mediante 2 switch, sia la parte wireless con standard 802.11 (Wi-Fi) e realizzata mediante un Access Point (AP) con crittografia WEP. Dopo aver inviato alcuni pacchetti tra gli end device, verificare il contenuto delle MAC Table dei due switch.

➜ ANALISI DEL PROBLEMA

#prendinota

Il router è l'unico dispositivo che necessita di metter ON le interfacce collegate.

Per realizzare la LAN richiesta servono 1 router, 2 switch, un AP e una serie di host, fissi e mobili. Il router per collegare l'intera rete LAN alla WAN (per esempio a Internet), i due switch per segmentare fisicamente la rete in varie zone e l'AP per i dispositivi mobili, in grado di coprire l'intera LAN trasformandola di fatto in una WLAN (Wireless LAN).
Su ogni host configureremo un indirizzo IP appartenente alla rete 192.168.1.0 e la subnet mask corrispondente 255.255.255.0.
Sull'AP e sui dispositivi mobili dovremo configurare l'SSID e la crittografia WEP. I cavi saranno tutti di tipo doppino, crossover o straight-through a seconda dei casi, collegati a interfacce FastEthernet o GigabitEthernet.



➜ SVOLGIMENTO

Nella **FIGURA 23** è stato disegnato un possibile scenario LAN, per ora senza i collegamenti, con i requisiti richiesti dall'esercizio. Sotto ogni dispositivo c'è l'indirizzo IP statico da caricare nell'apposita scheda (*IP Configuration*).

Tutti gli end system hanno una sola interfaccia Ethernet (o Wi-Fi) cui assegnare l'indirizzo IP, tranne il router che ne ha 3. Per convenzione scegliamo la prima delle 3 porte disponibili e gli assegniamo un IP non in sequenza.
Lo Switch1 rappresenta lo switch principale (centro stella) a cui saranno collegati lo Switch2 e l'AP per creare una LAN con topologia a stella estesa.

FIGURA 23 Scenario LAN wired e wireless: gli indirizzi IP

Per quanto riguarda la scelta dei dispositivi, non avendo esigenze particolari, prendiamo sempre il primo proposto dall'apposito menu.
Quindi avremo:

- 1 router 4331 dotato di tre interfacce GigabitEthernet di cui una sola utilizzata con indirizzo 192.168.1.100;
- 2 switch 2960 dotati di 24 interfacce FastEthernet e 2 interfacce GigabitEthernet ciascuno;
- 1 Access Point AP-PT dotato di una interfaccia FastEthernet e una interfaccia wireless;
- 8 host fissi PC-PT dotati di una interfaccia FastEthernet ciascuno con indirizzo in sequenza da 192.168.1.1 a 192.168.1.8;
- 4 dispositivi mobili dotati di interfaccia wireless: Smartphone, WirelessEndDevice, Tablet e Laptop.

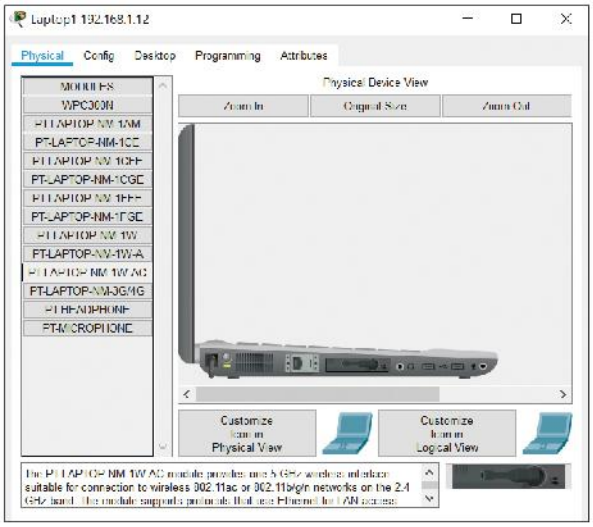


FIGURA 24 Sostituzione modulo sul Laptop

Attenzione: il **Laptop** non è fornito di default di un'interfaccia wireless ma ha un'interfaccia FastEthernet. Bisogna quindi sostituire il modulo FastEthernet con un modulo wireless tra quelli a disposizione, come il PT-LAPTOP-NM-1W-AC **(FIGURA 24)**.

Per sostituire il modulo seguire i seguenti passi: cliccare sul Laptop → selezionare la scheda Physical → spegnere il Laptop → trascinare col mouse l'interfaccia FastEthernet e metterla con i MODULES a sinistra → selezionare il modulo wireless PT-LAPTOP-NM-1W-AC da MODULES → trascinarlo col mouse nello slot dove c'era il modulo FastEthernet → riaccendere il Laptop.
Mettendo i cavi è possibile scegliere l'interfaccia Ethernet del dispositivo tra quelle a disposizione, in particolare sugli switch che hanno molte porte FastEthernet e alcune GigabitEthernet.
Si è scelto di usare le interfacce GigabitEthernet per collegare il router agli switch e le FastEthernet per collegare i PC e l'AP agli switch.
Tutti i cavi sono di tipo straight-through tranne il cavo crossover che collega i due switch. Mediante il menu Options → Preferences, vistando *Always Show Port Labels In Logical Workspace*, è possibile visualizzare le interfacce Ethernet assegnate **(FIGURA 25)**.
L'AP e i dispositivi wireless collegati utilizzano lo standard Wi-Fi 802.11 anziché l'Ethernet 802.3.

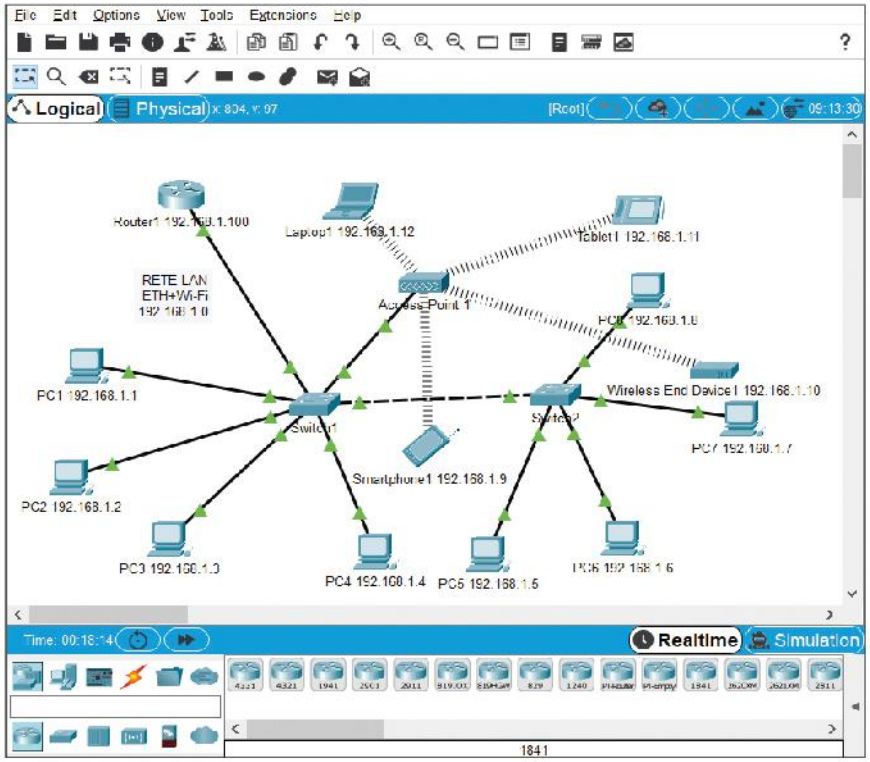


FIGURA 25 Scenario LAN wired e wireless: le interfacce

Il Wi-Fi prevede che venga garantita una sicurezza nelle trasmissioni almeno pari alle trasmissioni wired. Per fare ciò impostiamo la crittografia WEP (Wired Equivalent Privacy) e scegliamo come chiave di crittografia a 10 cifre esadecimali WEP Key = ABCDEF1234 di 40 bit. Affronteremo i vari tipi di crittografia in quinta. Come nome della rete scegliamo SSID = ReteETH-WIFI, mentre lasciamo il canale 6 di default. Notare come i dispositivi mobili risultino collegati da qualunque zona della rete.

La **FIGURA 26** mostra la scheda Config → Port 1 dell'AP dopo l'inserimento dei parametri di configurazione.

FIGURA 26 Configurazione dell'AP

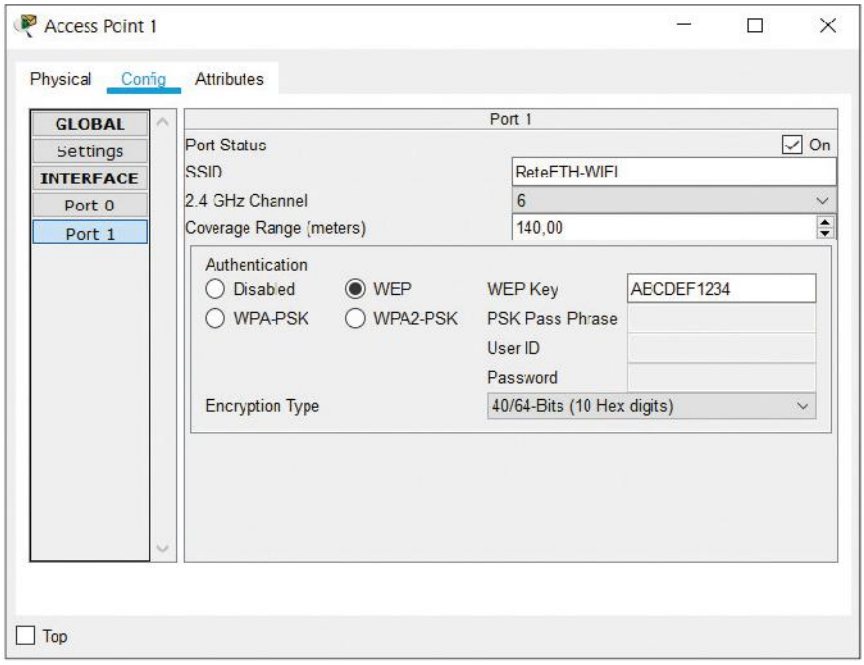
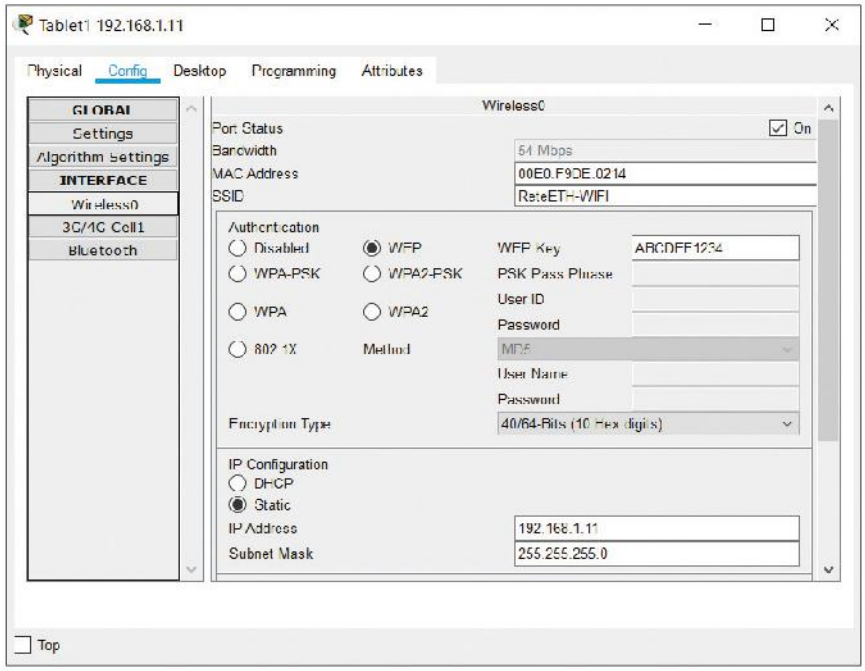


FIGURA 27 Configurazione del tablet



Affinché i 4 dispositivi mobili possano dialogare con l'AP e quindi entrare nella LAN, devono essere configurati con gli stessi parametri di SSID e di WEP Key impostati sull'AP.

La **FIGURA 27** mostra la configurazione dell'interfaccia Wireless0 impostata sul Tablet.

Analizziamo ora il funzionamento della rete mediante l'osservazione delle MAC Table dei due switch.

Al termine della realizzazione dei collegamenti, cioè quando tutti i triangolini sono diventati verdi, alcune PDU saranno trasmesse in background tra gli switch (e anche tra il router e lo switch collegato) che quindi inizieranno a compilare la propria MAC Table. Utilizzando la lente per cliccare sugli switch, è possibile selezionare e visualizzare la MAC Table **(FIGURE 28A** e **28B)** di uno switch, dove nella colonna MAC Address vi è il MAC di destinazione e nella colonna Port l'interfaccia di uscita dello switch stesso da usare per i frame con quella destinazione.

Si può notare come nelle Table ci siano solo gli indirizzi MAC delle interfacce GigabitEthernet del router e dello Switch2 per la MAC Table dello Switch1 (Figura 28A), e solo dello Switch1 per la MAC Table dello Switch2 (Figura 28B).

Nel nostro caso avremo che nella MAC Table dello Switch1 (Figura 28A), nella colonna MAC Address dei

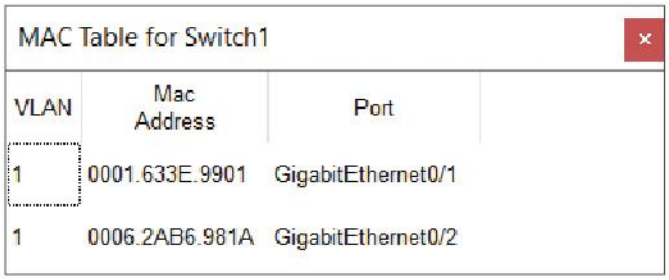


FIGURE 28A E 28B MAC Table Switch1 e Switch2 dopo i collegamenti

destinatari, ci sarà l'indirizzo MAC dell'interfaccia dello Switch2 (0006.2ABS.981A) collegata allo Switch1.

Simmetricamente, nella MAC Table dello Switch2 (Figura 28B), nella colonna MAC Address dei destinatari, ci sarà l'indirizzo MAC dell'interfaccia dello Switch1 (00E0.F717.431A) collegata allo Switch2.

Se inviamo una semplice PDU (la bustina nel menu in alto consente il ping con un pacchetto ICMP) tra due host collegati allo stesso switch (per esempio PC1 e PC4 per lo Switch1), vediamo che la sua MAC Table si modifica con l'aggiunta di due entry: quella del MAC e dell'interfaccia degli host mittente e destinatario. Lo switch è in grado di aggiornare la MAC Table grazie al fatto che nella PDU (un frame Ethernet in questo caso) vi è l'indirizzo MAC del mittente e del destinatario.
Selezioniamo quindi la busta PDU e clicchiamo prima su PC1 e poi su PC4 per indicare la direzione della trasmissione. Visualizziamo il contenuto dei campi del frame mediante la lente e cliccando sulla bustina pronta all'invio su PC1. Selezionando la scheda Output PDU Details si può vedere il frame Ethernet col MAC address del mittente e del destinatario **(FIGURA 29)**.

#prendinota

Per vedere gli indirizzi MAC delle interfacce degli switch è sufficiente fermarsi col mouse sopra lo switch.
Gli indirizzi MAC **degli switch** che compaiono nel campo MAC Address di una MAC Table sono solo quelli delle interfacce degli altri switch collegati. Gli altri indirizzi MAC presenti nella MAC Table sono tutti del router o degli end system e non degli switch.

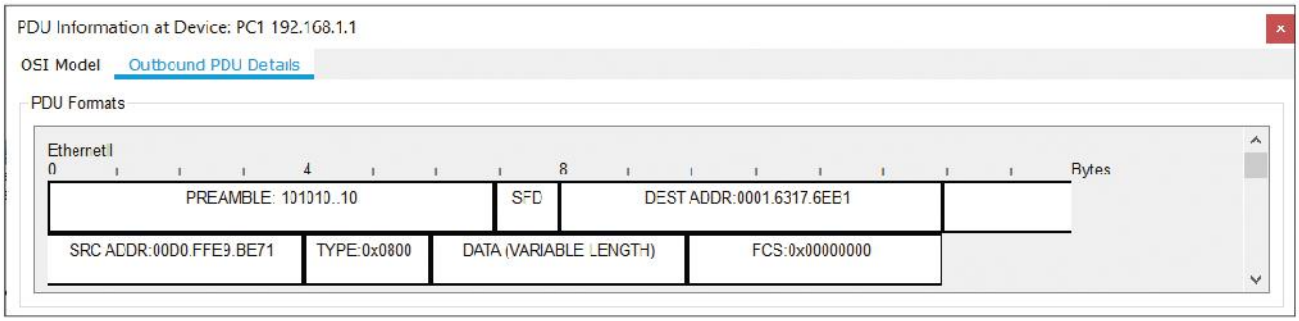


FIGURA 29 Il frame Ethernet inviato da PC1 a PC4

A questo punto partiranno in background una serie di pacchetti ARP in broadcast per tutta la rete per trovare l'IP corrispondente al MAC del destinatario. Terminata la ricerca lo Switch1 potrà "annotarsi" nella propria MAC Table quale interfaccia utilizzare per raggiungere quel MAC. Questa operazione viene fatta dagli switch sia per il destinatario che per il mittente.

La **FIGURA 30** mostra la MAC Table dello Switch1 dopo il ping tra PC1 e PC4 a esso direttamente collegati.

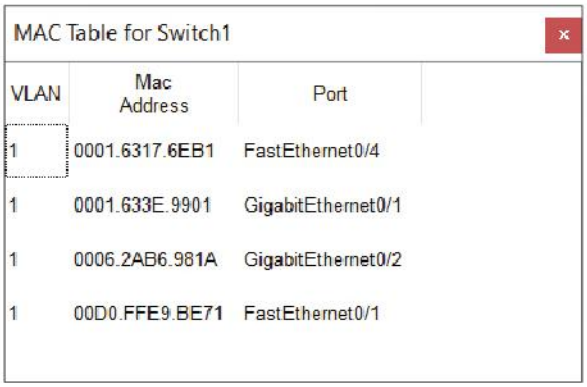


FIGURA 30 MAC Table Switch1 dopo il ping tra PC1 e PC4

Se ripetessimo ora nuovamente lo stesso ping tra PC1 e PC2, i pacchetti ARP non partirebbero più questa volta. Partirebbe subito il pacchetto ICMP diretto al destinatario essendo questo già noto allo Switch1.
La MAC Table dello Switch2 risulterà inalterata non essendo coinvolto in questa trasmissione nessun host collegato a tale switch. Affinché la MAC Table dello Switch2 si modifichi, inviamo una PDU tra PC1 e PC5.
Nuovamente partiranno i pacchetti ARP fino a trovare l'IP corrispondente al MAC del PC5. Al termine, le due tabelle risulteranno modificate come mostrato nelle FIGURE 31A e 31B.

MAC Table for Switch1

VLAN	Mac Address	Port
1	0001.6317.6EB1	FastEthernet0/4
1	0001.633E.9901	GigabitEthernet0/1
1	0006.2AB6.981A	GigabitEthernet0/2
1	0010.1188.57ED	GigabitEthernet0/2

MAC Table for Switch2

VLAN	Mac Address	Port
1	0010.1188.57ED	FastEthernet0/1
1	00D0.FFE9.BE71	GigabitEthernet0/2
1	00E0.F717.431A	GigabitEthernet0/2

FIGURE 31A E 31B MAC Table Switch1 e Switch2 dopo il ping tra PC1 e PC5

Per lo Switch1 si può notare l'aggiunta dell'entry (riga 4 Figura 31A) col MAC del PC5 e la Port GigabitEthernet0/2 dello Switch1 stesso che serve a raggiungere PC5 passando dallo Switch2.

#prendinota

Tutti i MAC dei PC collegati allo Switch2 saranno mappati nella MAC Table dello Switch1 con la porta GigabitEthernet0/2 dello Switch1 stesso e viceversa.

Simmetricamente, per lo Switch2 si può notare l'aggiunta dell'entry (riga 2 Figura 31B) col MAC del PC1 e la Port GigabitEthernet0/2 dello Switch2 stesso che serve a raggiungere PC1 passando dallo Switch1.

Lo Switch 2 aggiunge anche un'altra entry (riga 1 Figura 31B) col MAC del PC5 e la Port FastEthernet0/1 dello Switch2 stesso che serve a raggiungere direttamente PC5.

Infine, se proviamo a fare un ping tra il PC5 collegato allo Switch2 e il Laptop collegato all'AP, otterremo le MAC Table mostrate nelle FIGURE 32A e 32B.

Si può notare come in entrambe le tabelle si siano aggiunte 4 entry relative ai 4 dispositivi mobili presenti sulla rete, ognuno col suo MAC. Tali dispositivi, nella MAC Table dello Switch1, sono mappati con la porta FastEthernet0/6 che porta direttamente all'AP, mentre nella MAC Table dello Switch2 sono mappati con la porta GigabitEthernet0/1 che porta allo Switch1.

L'AP si comporta diversamente dagli switch. Un ping a uno dei dispositivi mobili a esso collegato, produce tante entry nella MAC Table dello switch quanti sono i dispositivi mobili connessi all'AP. Questo perché tutti i dispositivi mobili rispondono alle richieste in broadcast dell'AP comunicando il proprio MAC, che l'AP inoltra direttamente allo switch e che quindi questi può mappare.
Invece, come abbiamo visto, un ping a un end system collegato a uno switch non produce la mappatura di tutti gli end system collegati allo switch stesso, ma solo dello specifico destinatario.

MAC Table for Switch1

VLAN	Mac Address	Port
1	0001.6317.6EB1	FastEthernet0/4
1	0001.633E.9901	GigabitEthernet0/1
1	0003.E4B9.532B	FastEthernet0/6
1	0006.2AB6.981A	GigabitEthernet0/2
1	0010.1188.57ED	GigabitEthernet0/2
1	0060.703D.9C4B	FastEthernet0/6
1	00D0.FF53.81C4	FastEthernet0/6
1	00D0.FFE9.BE71	FastEthernet0/1
1	00E0.F9DE.0214	FastEthernet0/6

MAC Table for Switch2

VLAN	Mac Address	Port
1	0003.E4B9.532B	GigabitEthernet0/2
1	0010.1188.57ED	FastEthernet0/1
1	0060.703D.9C4B	GigabitEthernet0/2
1	00D0.FF53.81C4	GigabitEthernet0/2
1	00D0.FFE9.BE71	GigabitEthernet0/2
1	00E0.F717.431A	GigabitEthernet0/2
1	00E0.F9DE.0214	GigabitEthernet0/2

FIGURE 32A E 32B MAC Table Switch1 e Switch2 dopo il ping tra PC5 e Laptop

Questo perché tutti i dispositivi fissi scartano la richiesta in broadcast se non ne sono i destinatari (cioè se il MAC cercato non è il loro).

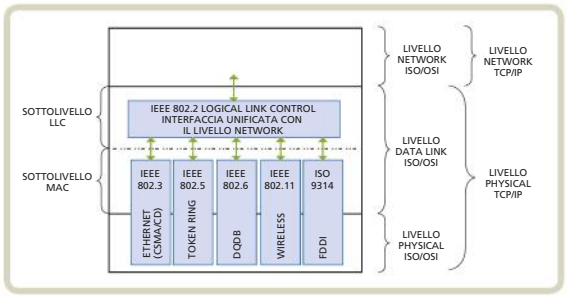
Inviando le PDU tra tutti i dispositivi presenti nella rete, si arriverà ad avere le due MAC Table complete sui due switch. A quel punto la tecnica di switching raggiungerà la sua massima performance riuscendo immediatamente a indirizzare tutte le comunicazioni interne alla rete direttamente verso il destinatario, senza generare traffico inutile.

FISSA LE CONOSCENZE

- Come si fa a selezionare una determinata interfaccia dello switch per collegare un cavo?
- Come si sostituisce un modulo su un dispositivo?
- Quali parametri vanno configurati sull'AP?
- Come si fa a vedere la MAC Table di uno switch?
- Grazie a che cosa lo switch riesce a compilare la propria MAC Table?

1 Il Progetto IEEE 802

IEEE, ISO e ANSI hanno sviluppato uno standard, noto come Progetto IEEE 802, per stabilire come debbano essere realizzate le reti LAN ai livelli Physical e Data Link del modello ISO/OSI.
Gli standard introdotti dal Progetto 802 stabilirono 20 categorie con cui identificare i diversi modi di accedere al canale di trasmissione. Stabilito il modo occorre stabilire una tecnica che regoli il diritto a trasmettere sul canale condiviso.
Due sono le possibilità: la tecnica a contesa e la tecnica deterministica.



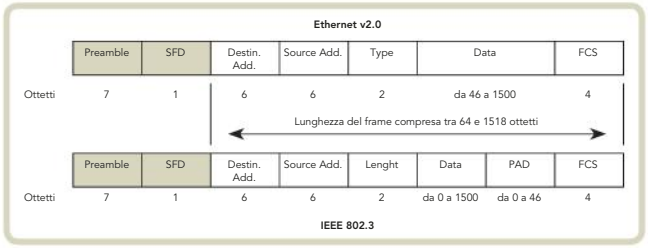
2 I sottolivelli LLC e MAC

Il livello Data Link di ISO/OSI è stato suddiviso in due sottolivelli: LLC (Logical Link Control) e MAC (Media Access Control) che rappresentano il cuore del Progetto 802.
Il sottolivello superiore è l'LLC che ha il fondamentale compito di fornire un'interfaccia unificata verso il livello Network, pur a fronte di tecnologie trasmissive e mezzi fisici differenziati. Il sottolivello inferiore è il MAC che risolve il problema dell'accesso al mezzo trasmissivo condiviso.
Come in tutti i modelli a strati il passaggio di livello (e sottolivello in questo caso) produce l'operazione di incapsulamento del frame.

3 L'evoluzione di LLC: HDLC e PPP

Il protocollo HDLC è generalmente utilizzato su reti di grandi dimensioni. Può essere usato per connessioni multipunto, ma attualmente è usato quasi esclusivamente per collegamenti punto-punto. Lo standard OSI prevede esplicitamente l'adozione di HDLC.
Il protocollo HDLC ha una grave carenza: non ha una modalità standard per trasmettere sullo stesso canale pacchetti generati da protocolli diversi di livello superiore. Per questo motivo la comunità di Internet ha introdotto un'estensione di HDLC, detta PPP (Point to Point Protocol). Principalmente viene usato per la comunicazione punto-punto tra due router o nella comunicazione tra utente e provider.

4 IEEE 802.3: la rete Ethernet

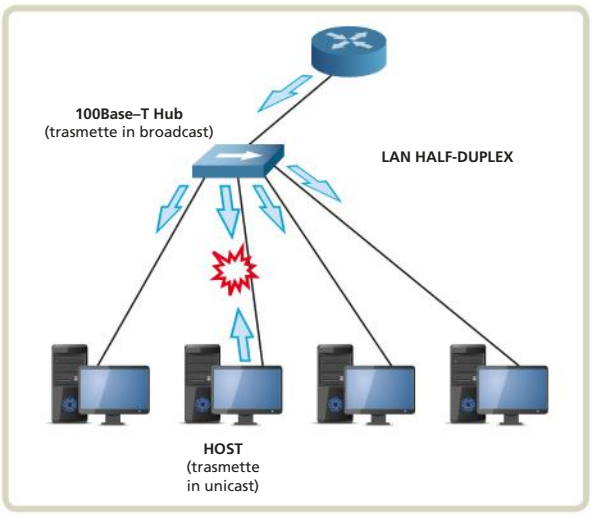


Ethernet è il più diffuso tipo di rete locale che esista al mondo. Nel 1985 Ethernet si evolve e diventa lo standard IEEE 802.3 che col passare degli anni sviluppa tutta una serie di standard che si distinguono per mezzo fisico, velocità di trasmissione e topologia. Si è passati dal cavo coassiale (non più utilizzato) al doppino e dal doppino alla fibra. Allo stesso modo le velocità sono salite dai 10 Mbps all'ordine

delle decine di Gbps. Anche la topologia è cambiata abbandonando il bus per passare alla stella o stella estesa.
Il frame Ethernet ha una lunghezza variabile compresa tra 64 e 1518 ottetti (byte), preceduti da un preambolo e da un byte di start. Esistono due formati del frame che attualmente convivono sulle reti Ethernet: Ethernet v2.0 e IEEE 802.3.

5 La tecnica a contesa CSMA/CD

La CSMA/CD si applica nelle reti Ethernet di tipo half-duplex, solitamente dotate dei più economici hub rispetto ai più costosi switch. Questo implica che occorre rilevare le eventuali collisioni quando due host trasmettono contemporaneamente. In seguito al rilevamento della collisione una sequenza di jamming avverte tutti dell'avvenuta collisione e un algoritmo di backoff esponenziale binario decide dopo quanto tempo un host può ritentare la trasmissione.



6 Lo switching

A partire dallo standard 802.3x viene introdotto il full-duplex nelle reti Ethernet. L'arrivo degli switch full-duplex rende superata la tecnica CSMA/CD non avendo più il rischio delle collisioni.
Serve uno strumento che metta in corrispondenza il MAC address con la porta dello switch. Questo strumento è la MAC Table (anche detta Switch Table).
In questa tabella ogni entry contiene la corrispondenza tra il MAC address dell'host collegato su una linea e la porta dello switch dedicata a tale linea.

MAC Table for Switch 1

VLAN	Mac Address	Port
1	0002.4A5A.9B01	GigabitEthernet0/1
1	000A.F310.E59D	FastEthernet0/3
1	0010.11B8.A035	FastEthernet0/4
1	00D0.9711.EDE9	FastEthernet0/1
1	00E0.B08B.9404	FastEthernet0/2

7 IEEE 802.11: la rete Wi-Fi

Lo standard wireless 802.11 nacque nel 1997 ma praticamente rimase solo sulla carta per via delle insufficienti prestazioni che consentiva (tra cui velocità solo fino a 1 o 2 Mbps). Nel 1999 la IEEE emise due nuovi standard: 802.11a che poteva raggiungere i 54 Mbps a 5,2 GHz e 802.11b con due nuove velocità: 5,5 Mbps e 11 Mbps a 2,4 GHz. Lo standard 802.11b a 11 Mbps è anche noto come marchio Wi-Fi (Wireless Fidelity) creato dalla Wi-Fi Alliance.
Le reti LAN attuali sono tutte realizzate con gli standard Ethernet e Wi-Fi.

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Il Progetto 802 prevede 10 categorie. V F
2. Con MAC si può intendere sia un sottolivello che un indirizzo. V F
3. Con 802.3 si intende la Token Ring. V F
4. Con 802.5 si intende l'Ethernet. V F
5. Con 802.11 si intende il Wi-Fi. V F
6. Il passaggio di livello o di sottolivello produce l'operazione di incapsulamento del frame. V F
7. L'Unacknowledged Connectionless Service è un servizio non affidabile e non orientato alla connessione. V F
8. Il Connection Oriented Service è un servizio affidabile ma non orientato alla connessione. V F
9. Il sottolivello superiore è l'LLC e ha il compito di fornire un'interfaccia unificata verso il livello Transport. V F
10. Il protocollo HDLC è generalmente utilizzato su reti di piccole dimensioni. V F
11. PPP ha una modalità standard per trasmettere sullo stesso canale pacchetti generati da protocolli diversi di livello superiore. V F
12. La fragment-free è una tecnica di switching. V F
13. Le tecniche a contesa non possono prevedere a priori i tempi di trasmissione. V F
14. Le tecniche deterministiche sono adatte alle trasmissioni real time. V F
15. La principale tecnica di switching è la store-and-forward. V F
16. L'indirizzo MAC è scritto nella notazione decimale puntata. V F
17. La MAC Table mette in corrispondenza il MAC address con la porta dello switch. V F
18. Per sapere l'indirizzo MAC della scheda di rete del proprio computer in ambiente Windows si può usare ipconfig/all. V F
19. Per proteggere una LAN è utile attivare la protezione delle porte negli switch. V F
20. Uno switch a 8 porte può fornire protezione a 1032 indirizzi MAC. V F
21. La sequenza di jamming nelle reti Ethernet segnala che la trasmissione è andata a buon fine. V F
22. L'algoritmo di backoff esponenziale binario tende a premiare la stazione che ha poche collisioni consecutive. V F
23. PoE è una tecnologia che permette di alimentare gli apparati utilizzando lo stesso cavo che li collega alla rete Ethernet, a condizione che sia del tipo twisted-pair. V F
24. A fronte dalla presenza di molti dispositivi da collegare e alimentare, si può prevedere l'utilizzo di uno splitter PoE. V F
25. Le specifiche PoE prevedono due tipi di dispositivi: PSE (Power Sourcing Equipment) e PD (Powered Device). V F
26. La banda ISM (Industrial, Scientific and Medical) è intorno ai 10 GHz. V F
27. Per evitare sovrapposizioni di frequenze, i canali wireless devono usare la regola del 5. V F
28. Le reti wireless utilizzano una tecnica di Collision Detection come le reti Ethernet. V F
29. I dispositivi che costituiscono le reti wireless sono i wireless terminal e gli switch. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **LLC e MAC sono:**
- A livelli di TCP/IP
- B livelli di ISO/OSI
- C sottolivelli del livello Data Link
- D sottolivelli del livello Network

2. **L'algoritmo di backoff esponenziale binario serve a:**
- A calcolare un tempo di attesa
- B suddividere un canale in sottocanali
- C rilevare un errore di trasmissione
- D codificare in binario un messaggio

3. **Lo standard wireless 802.11g raggiunge i:**
- A 54 Mbps a 5,2 GHz
- B 11 Mbps a 2,4 GHz
- C 54 Mbps a 2,4 GHz
- D 11 Mbps a 5,2 GHz

4. **Lo standard denominato PoE è:**
- A 802.3ac
- B 802.3ad
- C 802.3af
- D 802.3at

5. **Quale tra le seguenti non è una tecnica di switching?**
- A Store-and-forward
- B Cut-through
- C Fragment-free
- D Inter-frame spacing

6. **Il frame IEEE 802.3 ha, rispetto al frame v2.0, il campo Length al posto del campo:**
- A Preamble
- B Type
- C Data
- D FCS

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Spiega la differenza tra la tecnica a contesa e la tecnica deterministica.**

2. LEZIONE 2 **Qual è il compito del Sottolivello LLC (Logical Link Control) e qual è l'informazione principale che deve contenere il suo frame?**

3. LEZIONE 2 **Qual è il compito del Sottolivello MAC (Media Access Control) e qual è l'informazione principale che deve contenere il suo frame?**

4. LEZIONE 2 **In che cosa consiste l'incapsulamento nel passaggio tra livelli o sottolivelli?**

5. LEZIONE 3 **Quali sono i principali protocolli di linea?**

6. LEZIONE 3 **La tecnica del bit stuffing come funziona?**

7. LEZIONI 2, 3, 4 **Cosa contiene il campo FCS (Frame Check Sequence) comune a molti tipi di frame?**

8. LEZIONE 4 **Quali miglioramenti comporta la modalità full-duplex nelle LAN con switch?**

9. LEZIONE 4 **Cosa contiene il campo Type del frame Ethernet v2.0?**

10. LEZIONE 5 **Spiega la tecnica CSMA/CD (Carrier Sense Multiple Access with Collision Detection) della rete Ethernet.**

11. LEZIONE 6 **Descrivi la tecnica di switching cut-through.**

12. LEZIONE 6 **Descrivi la tecnica di switching fragment-free.**

13. LEZIONE 6 **Dopo aver allocato il numero massimo di indirizzi MAC su una porta dello switch, quali alternative si hanno?**

14. LEZIONE 7 **Che cosa sono i Wireless Terminal e che compito hanno?**

15. LEZIONE 7 **Che cosa sono gli access point e che compito hanno?**

16. LEZIONE 7 **Descrivi il problema della stazione esposta.**

17. LEZIONE 7 **Spiega la tecnica CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) della rete wireless.**

18. LEZIONE 7 **Elenca almeno 5 apparecchi che utilizzano la banda ISM (Industrial, Scientific and Medical).**

IN ENGLISH, PLEASE

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

The Physical Layer of TCP/IP Architecture

The Physical Layer is given the task of defining the rules for access to it. This Layer consists of two sub-Layers: the sub-Layer LLC (Logical Link Control) with the basic task of providing a unified interface to the Network Layer, even when faced with different transmission technologies and physical medium, and the sub-Layer MAC (Medium o Media Access Control) which resolves the problem of access to the shared means of transmission.
IEEE 802 family of standards specifies Physical Layer protocols and services for local and metropolitan area networks. The most widely used standards are for Ethernet, Token Ring, Wireless networks. Each device in a network has a different MAC address.

EXERCISES

Use the appropriate number to match words and meanings.

...	Binary exponential backoff	1	Different physical address for each network card
...	MAC address	2	Filler with dummy data
...	Algorithm	3	Switching technique
...	Store-and-forward	4	Protocol analyzer
...	Wi-Fi	5	Pseudo-random algorithm
...	Wireshark	6	Finite sequence of well-defined computer-implementable instructions
...	Padding	7	Simultaneous bidirectional transmission
...	Full-duplex	8	Wireless standard

Choose the correct answer.

1. The FastEthernet has a speed of:
- A 10 Mbps
- B 100 Mbps
- C 1 Gbps
- D 10 Gbps

2. The technique for delayed sending of the ACK is:
- A bit stuffing
- B collision detection
- C piggybacking
- D fragment-free

GLOSSARY

Access Point: devices which connect the wired and wireless parts of a network and enable Wireless Terminals to connect to the network.
Bit stuffing: technique that consists in inserting a 0 bit after five consecutive 1 bits.
Inter-frame spacing: has the purpose of defining the minimum time space between two consecutive frames.
MAC Table (also called Switch Table): in this table each entry contains the correspondence between the MAC address of the host connected on a line and the switch port dedicated to that line.
PD (Powered Device): devices that are powered by PoE technology.
Piggybacking: technique of temporarily delaying the acknowledgement (ACK) so that it can be hooked with next outgoing data frame.
PoE (Power over Ethernet): it is a technology that allows to power devices using the same twisted-pair cable that connects them to the Ethernet network.
PSE (Power Sourcing Equipment): these are the devices that supply power in PoE technology.
Splitter: a device that has the function of distributing the power of an input signal between two or more output connections.
Switching: the process that forwards packets coming in from one port to a port leading towards the destinations.
Wireless Terminal: moveable devices in wireless networks.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper collegare dispositivi in rete scegliendo i giusti cavi e le opportune interfacce.
- Saper verificare il funzionamento della rete.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Sapere usare un simulatore di reti.
- Esporre i risultati della ricerca alla classe.

tempi

- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Simulatore di reti Packet Tracer.
- Dispositivo connesso a Internet.
- Carta e penna.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Realizzare una porzione di rete LAN costituita da una serie di switch in cascata. Verificare il funzionamento dello switching attraverso l'osservazione delle MAC Table appena realizzata la connessione fisica (triangolino verde) e dopo aver inviato una PDU tra due PC posti alle due estremità della rete.

File sorgenti
Scarica il file

SVOLGIMENTO

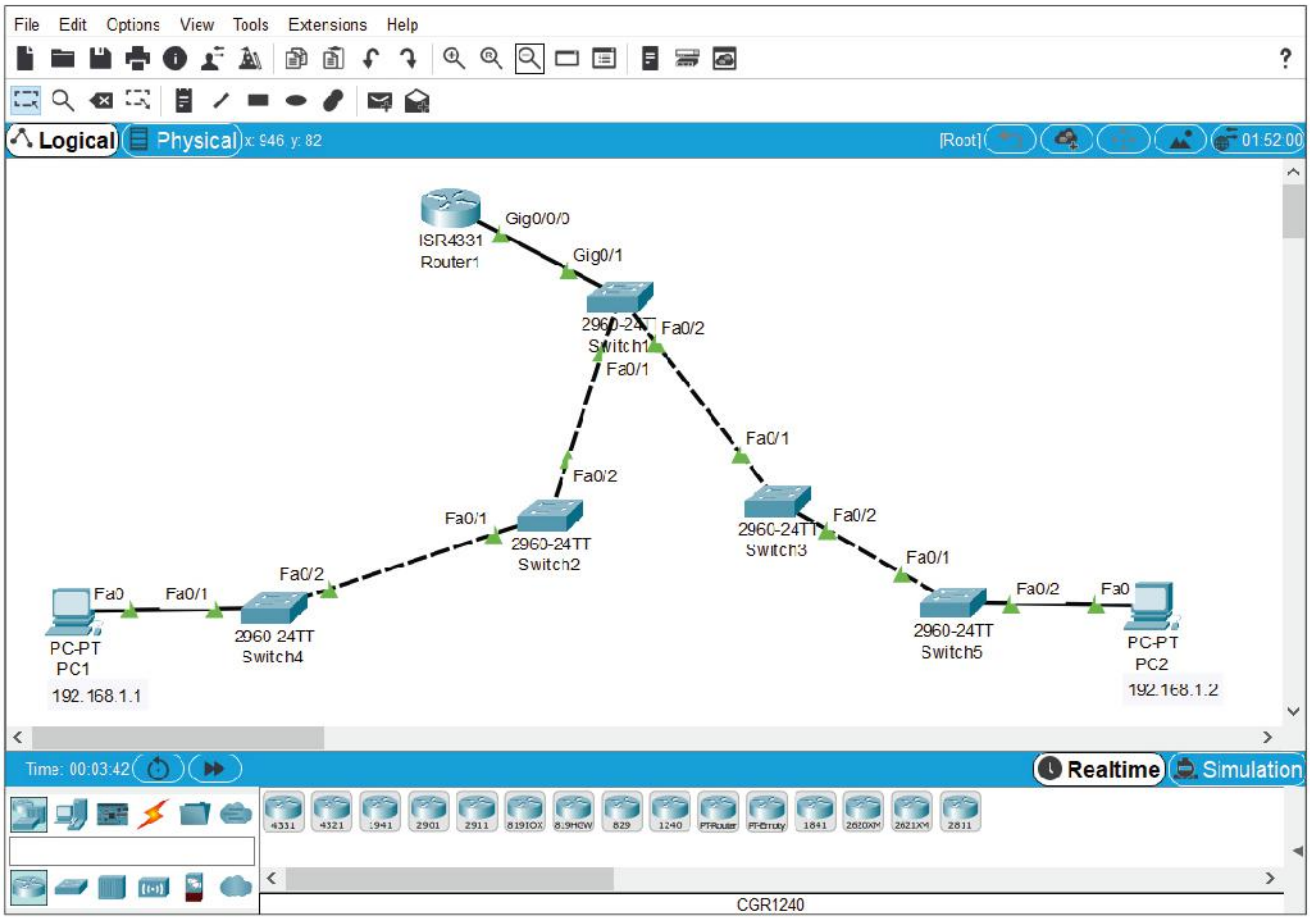
Progettiamo la nostra rete con uno switch principale collegato da un lato al router e dall'altro a due segmenti di rete che hanno ciascuno altri due switch in cascata.

Ai due ultimi switch colleghiamo due PC. La figura nella pagina seguente mostra lo scenario progettato, dove sono stati configurati gli indirizzi IP privati e le subnet mask per l'interfaccia del router e dei PC, col seguente schema:

- Router 1: 192.168.1.100/24
- PC1: 192.168.1.1/24
- PC2: 192.168.1.2/24

A questo punto siamo pronti a rispondere alle richieste del tema proposto che prevedono tre azioni in sequenza:

- lettura delle MAC Table;
- invio di un messaggio (simple PDU);
- rilettura delle MAC Table.



Prima di inviare qualsiasi PDU andiamo a vedere il contenuto delle MAC Table dei 5 switch:
Lo switch al livello più alto ha 3 entry: 1 GigabitEthernet verso il router e 2 FastEthernet verso i 2 switch al livello intermedio.

MAC Table for Switch1

VLAN	Mac Address	Port
1	0001.43AB.4001	GigabitEthernet0/1
1	0001.967C.5401	FastEthernet0/2
1	0004.9A47.2C02	FastEthernet0/1

Gli switch al livello intermedio hanno 2 entry FastEthernet: una verso lo switch di livello superiore e una verso lo switch di livello inferiore.

MAC Table for Switch2

VLAN	Mac Address	Port
1	000C.8596.D201	FastEthernet0/2
1	0050.0F99.2702	FastEthernet0/1

MAC Table for Switch3

VLAN	Mac Address	Port
1	000C.8596.D202	FastEthernet0/1
1	00D0.9740.4701	FastEthernet0/2

Infine gli switch di livello inferiore hanno una sola entry FastEthernet verso lo switch di livello intermedio.

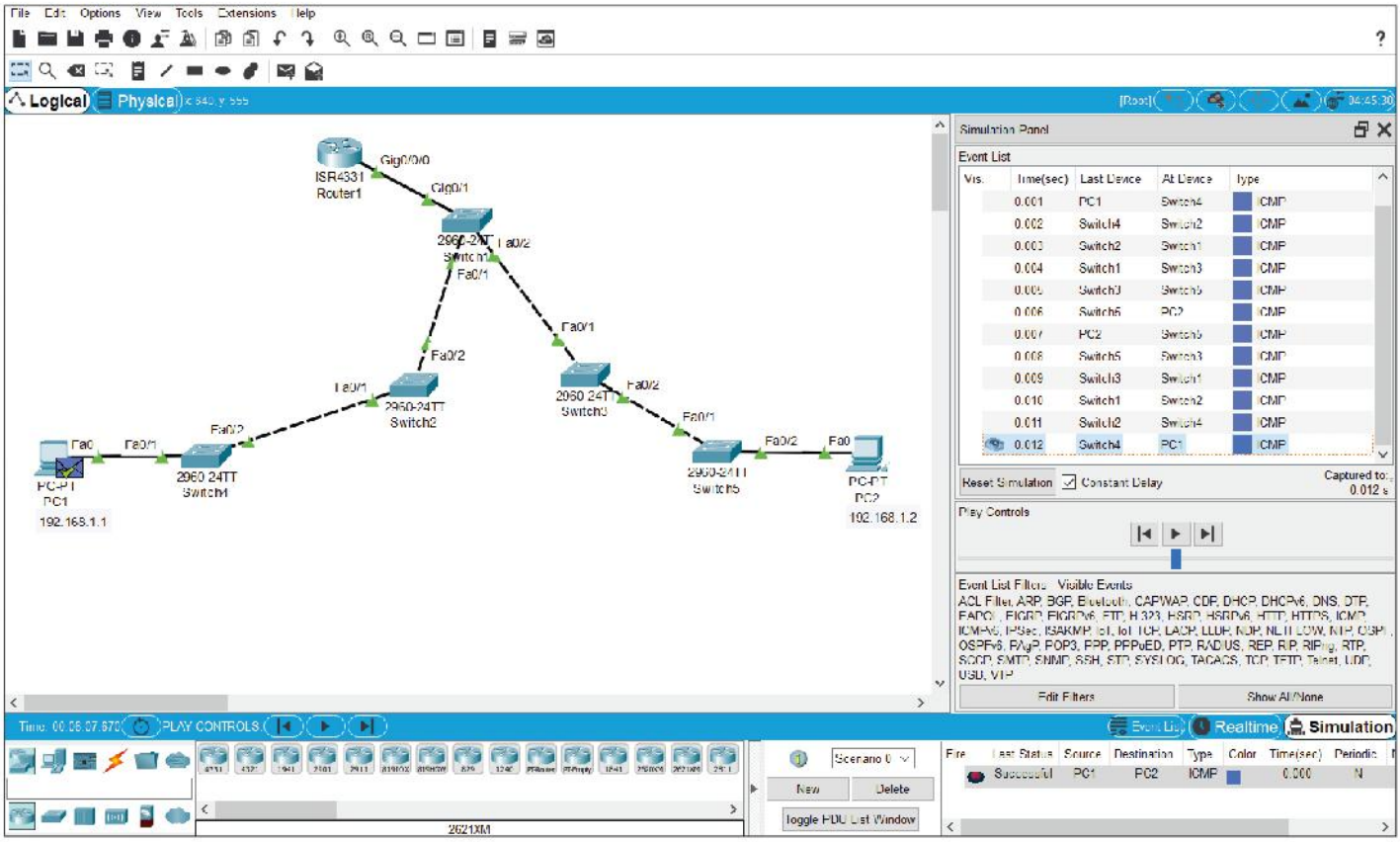
MAC Table for Switch4

VLAN	Mac Address	Port
1	0004.9A47.2C01	FastEthernet0/2

MAC Table for Switch5

VLAN	Mac Address	Port
1	0001.967C.5402	FastEthernet0/1

A questo punto non ci resta che inviare un PDU da PC1 a PC2 e verificare se le MAC Table sono state aggiornate tutte.
Selezioniamo la bustina in alto per il semplice PDU e clicchiamo prima su PC1 e poi su PC2 per effettuare il ping.
Procedendo passo passo con la simulazione vediamo i pacchetti ICMP (comando ping) passare di switch in switch portando con sé il frame che contiene i MAC address dei due PC.
Sulla destra della figura, nella Event List, si può vedere il percorso del pacchetto ICMP attraverso i vari switch per raggiungere PC2 e il ritorno, sempre attraverso tutti gli switch, a PC1.



Questo farà sì che al termine della trasmissione (Successful) le MAC Table siano tutte aggiornate **con due nuove righe** contenenti gli indirizzi MAC dei due PC e l'interfaccia da usare per indirizzarvi i pacchetti in transito.

Nel nostro caso, in ogni tabella sono comparse le righe:

0001.9A71.C5C1 FastEthernet0/1 (terza riga)

0010.11E7.84E5 FastEthernet0/2 (quinta riga)

MAC Table for Switch1

VLAN	Mac Address	Port
1	0001.43AB.4001	GigabitEthernet0/1
1	0001.967C.5401	FastEthernet0/2
1	0001.97A1.C5C1	FastEthernet0/1
1	0004.9A47.2C02	FastEthernet0/1
1	0010.11E7.84E5	FastEthernet0/2

MAC Table for Switch2

VLAN	Mac Address	Port
1	0001.97A1.C5C1	FastEthernet0/1
1	000C.8596.D201	FastEthernet0/2
1	0010.11E7.84E5	FastEthernet0/2
1	0050.0F99.2702	FastEthernet0/1

MAC Table for Switch3

VLAN	Mac Address	Port
1	0001.97A1.C5C1	FastEthernet0/1
1	000C.8596.D202	FastEthernet0/1
1	0010.11E7.84E5	FastEthernet0/2
1	00D0.9740.4701	FastEthernet0/2

MAC Table for Switch4

VLAN	Mac Address	Port
1	0001.97A1.C5C1	FastEthernet0/1
1	0004.9A47.2C01	FastEthernet0/2
1	0010.11E7.84E5	FastEthernet0/2

MAC Table for Switch5

VLAN	Mac Address	Port
1	0001.967C.5402	FastEthernet0/1
1	0001.97A1.C5C1	FastEthernet0/1
1	0010.11E7.84E5	FastEthernet0/2

A CASA

- Ipotizza una tua soluzione al tema proposto.
- Leggi la proposta di SVOLGIMENTO per verificare se le tue ipotesi si adattano al caso preso in esame e se la trattazione proposta risulta completa.
- Modifica la rete collegando un PC a uno switch a piacere e ricontrolla le MAC Table.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i casi che sono stati presentati.
- Stabilite quale caso rappresenta l'esempio migliore per completezza e che meglio si adatta alla soluzione del tema proposto.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho simulato il funzionamento della rete senza difficoltà?	Non sempre sono riuscito a selezionare i dispositivi e i cavi adatti. ☐	Con la guida del docente e la collaborazione dei compagni ho simulato il funzionamento della rete. ☐	Ho simulato il funzionamento della rete autonomamente. ☐	Ho simulato il funzionamento della rete anche con un PC in più. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni ma un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono sempre riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ 3

IL NETWORK LAYER DEL TCP/IP

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

conoscenze	abilità	competenze
Conoscere i servizi offerti dal livello Network. Conoscere il protocollo IP. Conoscere la struttura degli indirizzi IP e delle subnet mask. Conoscere la differenza tra indirizzo privato e indirizzo pubblico. Conoscere i 4 livelli operativi (mode) della CLI con cui operare su un router Cisco.	Saper segmentare una rete locale. Saper usare la tecnica del supernetting. Saper definire subnet mask di lunghezza variabile. Saper configurare le interfacce di un router. Saper usare la Command Line Interface (CLI) di un router. Saper usare la porta console del router.	Realizzare il piano d'indirizzamento di una LAN. Riprodurre il funzionamento di una rete reale tramite la simulazione. Interfacciarsi con il Sistema Operativo di un router per attività di configurazione e diagnostica.

DDI+ Didattica Digitale Integrata Plus

FLIPPED CLASSROOM

A casa

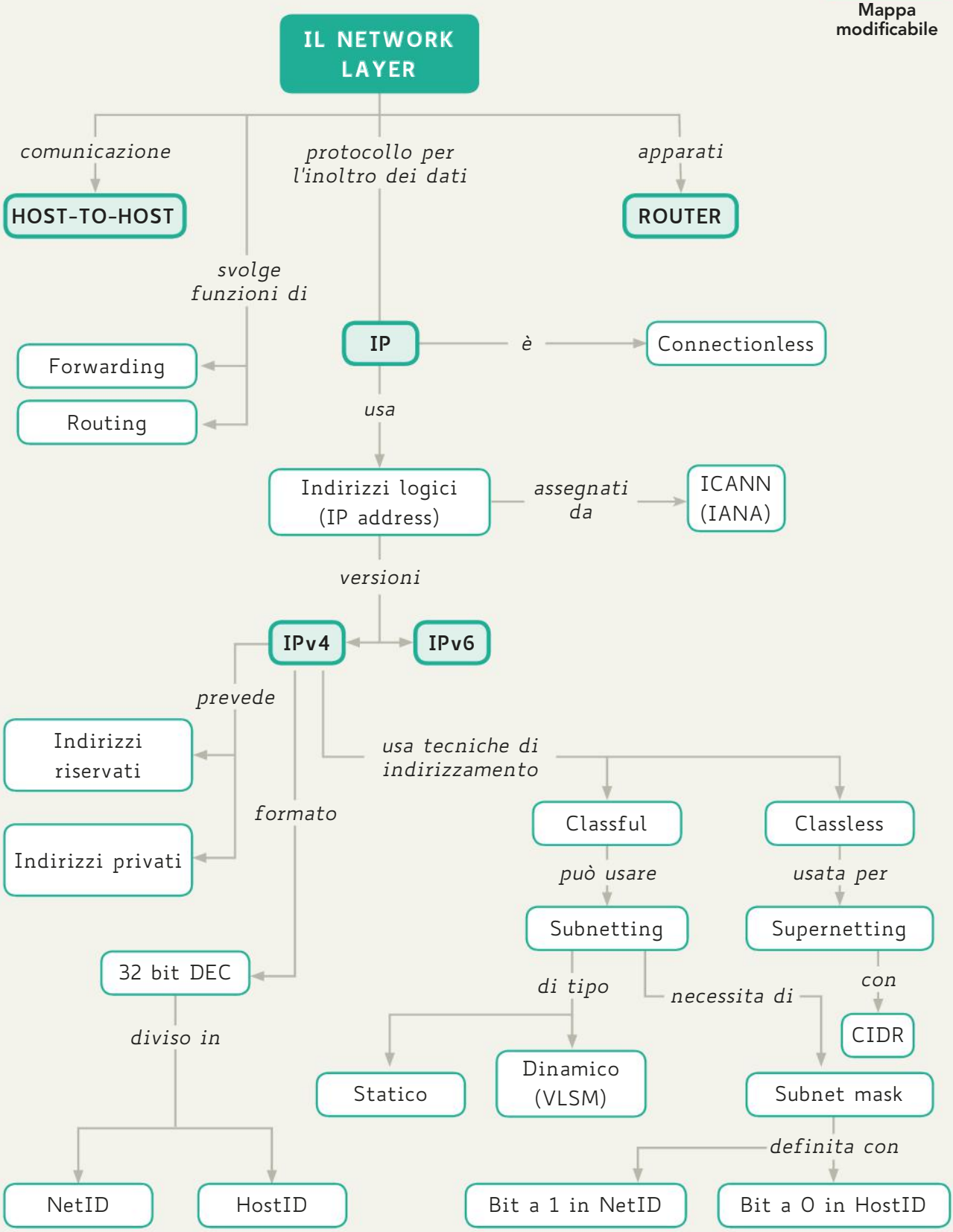
- Il 29 ottobre 1969 fu effettuata la trasmissione di un primo pacchetto di dati tra due computer: uno posto all'università di Los Angeles, l'altro al Research Institute di Stanford. La rete si chiamava Arpanet ed è considerata l'antenata di Internet;
- approfondisci Arpanet e le origini di Internet;
- sintetizza i risultati della tua ricerca in una mappa concettuale.

In classe

- Confrontate le mappe prodotte;
- dopo un momento di discussione sulle eventuali differenze, create una mappa che raccolga i contributi di tutti.

MAPPA CONCETTUALE

Mappa modificabile



1 IL LIVELLO NETWORK E IL PROTOCOLLO IP

1.1 Le funzioni svolte a livello Network

Nello stack TCP/IP il Network Layer ha il ruolo di trasferire i pacchetti da un host mittente a un host destinatario, usufruendo dei servizi offerti dal sottostante livello Physical. Per realizzare questa funzione, svolge due compiti fondamentali:

- **forwarding**: è l'attività di inoltro dei dati nella rete, svolta all'interno del router quando riceve un pacchetto da una linea (*input link*) e deve selezionare la linea su cui trasmetterlo (*output link*) per inoltrarlo al prossimo router;
- **routing**: il livello Network determina il percorso (*route*) che seguono i pacchetti nella rete, passando da un router all'altro, usando alcuni algoritmi detti **algoritmi di routing**, descritti nell'Unità 5; il **routing** permette anche la localizzazione degli instradamenti alternativi da usare in caso di guasti nella rete.

Spesso quando si parla di reti si fa riferimento al caso reale di un veicolo che parte da un punto, percorre più strade e arriva a destinazione. Il processo di forwarding si attiva quando, per esempio, il veicolo arriva a uno svincolo autostradale e viene instradato su un'altra strada che gli consente di continuare il viaggio verso la destinazione.
Prima di partire, l'automobilista pianifica il tragitto, consulta le mappe stradali e sceglie il percorso da seguire, che risulterà formato da un insieme di segmenti di strada, connessi tramite gli svincoli. Il routing è simile a questo processo di pianificazione del viaggio.

Per meglio affrontare la realizzazione di questi due importanti compiti, nelle moderne reti **SDN (Software-Defined Network)** il Network Layer è strutturato in due sottolivelli:

- **Data Plane** (il piano dei dati): qui si trovano le funzioni di **#forwarding**, svolte dal singolo router;
- **Control Plane** (il piano del controllo): implementa le funzioni di **#routing**, stabilendo il percorso che deve seguire un pacchetto nella rete. Ha quindi un ruolo di coordinamento delle azioni di routing attuate nella rete tramite gli algoritmi e i protocolli di routing.

Il Network Layer può offrire sia servizi connessi (**connection-oriented**) sia servizi non connessi (**connectionless**). I primi sono solitamente implementati tramite circuiti virtuali e si ritrovano nelle reti di estrazione telefonica, quali X.25, Frame Relay e ATM.
I secondi sono offerti dalle reti TCP/IP, come Internet, che sono attualmente le reti più diffuse.
Il Network Layer è il primo strato dello stack TCP/IP in grado di garantire una connettività a livello WAN; deve quindi poter identificare univocamente ogni host della rete mediante un identificativo apposito.
Un protocollo di livello Network deve conoscere la topologia della rete, scegliere di volta in volta il cammino migliore, gestendo le problematiche derivanti dalla presenza di più reti realizzate con tecnologie diverse a livello Physical.

#techwords

Forwarding vs Routing
Questi due termini sono spesso usati come sinonimi, ma in realtà individuano due funzioni ben distinte del router: **forwarding** si riferisce a un'azione locale al router, spesso svolta a livello hardware, mentre **routing** è un processo che coinvolge gli altri router della rete per determinare un percorso e viene implementato a livello software.

Il principale protocollo del livello Network nelle reti TCP/IP è **Internet Protocol (IP)**, usato per trasferire i dati nella rete WAN. Sempre in questo livello sono stati specificati anche alcuni **#protocolli di controllo** come ARP, RARP, ICMP, i protocolli di routing e altri ancora.

1.2 Il protocollo IP

Il protocollo IP, nelle versioni 4 e 6 si occupa dell'indirizzamento, della suddivisione in pacchetti e del trasferimento dei dati che arrivano dal Transport Layer.

IP è connectionless, dunque consente a due host di scambiarsi PDU, denominate **IP #datagram**, senza stabilire una connessione. La consegna non è garantita a questo livello, ma, se richiesta dall'applicazione, se ne occupa il protocollo TCP a livello Transport.

Il protocollo IP è stato specificato nel 1981 e pubblicato in **RFC 791**, in seguito aggiornato dagli RFC 1349, 2474, 6864.

IN ENGLISH PLEASE

RFC: 791

INTERNET PROTOCOL
DARPA INTERNET PROGRAM
PROTOCOL SPECIFICATION
September 1981

[…]

1.1. Motivation

The Internet Protocol is designed for use in interconnected systems of packet-switched computer communication networks. Such a system has been called a "catenet" [1]. The Internet Protocol provides for transmitting blocks of data called datagrams from sources to destinations, where sources and destinations are hosts identified by fixed length addresses. The internet protocol also provides for fragmentation and reassembly of long datagrams, if necessary, for transmission through "small packet" networks.

Il protocollo IP aggiunge ai dati (payload) un **header**, della lunghezza minima di 20 byte, per formare il pacchetto (massimo 65.535 byte) da inoltrare al livello Physical. I campi più importanti dell'header sono rappresentati dagli indirizzi IP del mittente e del destinatario.

Vediamone nel dettaglio la composizione in FIGURA 1.

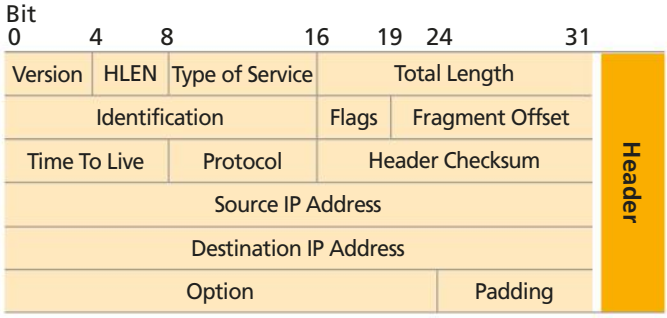


FIGURA 1 Formato dell'header IPv4

#techwords

Protocolli di controllo
Si tratta di protocolli che, a differenza di IP, non sono usati per trasferire i dati delle applicazioni nella rete. Infatti, essi sono usati dai router per comunicare informazioni di tipo gestionale e di controllo quali, per esempio, cambiamenti nella topologia della rete.

#techwords

Datagram
Il termine datagram è usato per indicare la PDU di un protocollo **connectionless**, come IP o UDP (protocollo di livello Transport).

Il contenuto dei singoli campi è il seguente:

- **Version**: 4 bit che rappresentano la versione del protocollo IP (0100 = versione 4); se l'host destinatario non è in grado di gestire la versione del protocollo IP specificata, il pacchetto verrà scartato.
- **HLEN** (Header LENgth): 4 bit che indicano la lunghezza dell'header IP espressa in word (gruppi di 32 bit). Tutti i campi dell'header sono di lunghezza fissa tranne Options e Padding, se questi non sono presenti, l'header è lungo **20 byte**.
- **TOS** (Type Of Service): 8 bit che servono a far capire all'IP come gestire il pacchetto. È costituito da 6 sottocampi:

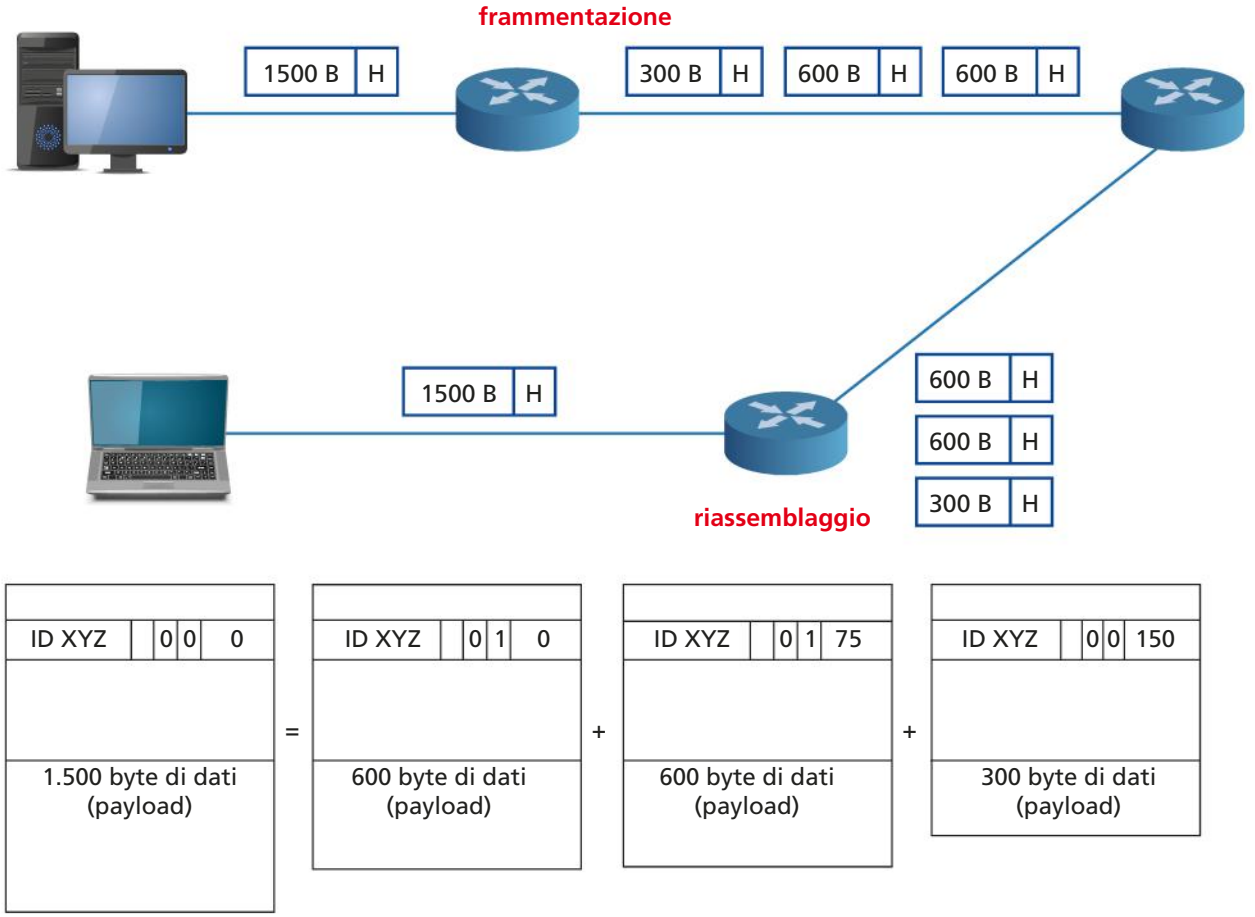
Precedence (3 bit)	Delay (1 bit)	Throughput (1 bit)	Reliability (1 bit)	Monetary Cost (1 bit)	Unused (1 bit)

  - **Precedence**: 3 bit che indicano la priorità del pacchetto, da 0 (normale) a 7 (controllo di rete); più il numero è alto più il pacchetto è importante.
  - **Delay**: un bit che se impostato a 1 indica che si vuole un ritardo minimo.
  - **Throughput**: un bit che se impostato a 1 indica che si vuole un throughput massimo.
  - **Reliability**: un bit che se impostato a 1 indica che si vuole massima affidabilità.
  - **Monetary Cost**: un bit che se impostato a 1 indica che si vuole il percorso dal costo minimo.
  - **Unused**: 1 bit inutilizzato. Nella versione 4 di IP è fissato a 0.
- **Total Length**: 16 bit che contengono la lunghezza totale del datagram espressa in byte (header + payload) che potrà quindi essere al massimo $2^{16} - 1 = 65.535$ byte; di norma, però, un pacchetto IP è lungo 1.500 byte, così da poter essere trasportato interamente come payload nel frame Ethernet.
- **Identification (ID), Flags e Fragment Offset** sono campi relativi alla funzione di frammentazione: a volte può essere necessario per un router dividere un datagram in pacchetti più piccoli, per consentirgli di attraversare una rete con caratteristiche diverse da quella di provenienza (Ethernet, Token Ring, Wi-Fi, ecc.).
  - **ID:** 16 bit che identificano univocamente i frammenti di un medesimo datagram; tutti i frammenti in cui è suddiviso un datagram avranno lo stesso ID.
  - **Flags**: 3 bit per il controllo della frammentazione:
    - il primo bit è attualmente inutilizzato;
    - il secondo bit è detto DF (Don't Fragment): se impostato a 1 indica che il datagram non può essere frammentato;
    - il terzo bit è detto MF (More Fragment): se impostato a 1 indica che il frammento è seguito da altri frammenti; solo l'ultimo frammento avrà quindi MF = 0.
  - **Fragment Offset**: 13 bit che indicano l'offset del frammento rispetto all'inizio del datagram, il valore indicato raggruppa 8 byte alla volta. L'offset è fissato dal router che esegue la frammentazione.

Nella FIGURA 2 viene esemplificato il contenuto dei campi ID, Flags e Fragment Offset nel caso di un datagram con payload di 1.500 byte suddiviso in 3 frammenti di dimensione 600 byte, 600 byte e 300 byte, rispettivamente.

Si noti l'offset indicato: il primo frammento ha dimensione 600 byte, quindi contiene i byte 0 - 599, il secondo i successivi byte 600 - 1.199 e il terzo 1.200 - 1.499. L'offset indicato nel secondo frammento è 600 : 8 = 75, in quanto i byte sono raggruppati 8 alla volta. Analogamente, nell'ultimo frammento l'offset è 1.200 : 8 = 150.

FIGURA 2 Frammentazione di un IP datagram



- **TTL** (Time To Live): 8 bit inizializzati al numero massimo di passaggi da un router al successivo (hop) che il datagram può effettuare. Il suo scopo è di evitare che un pacchetto continui a circolare nella rete all'infinito. Il TTL è usato come un contatore, il cui valore viene decrementato di 1 ogni volta che il datagram attraversa un router e quando arriva a 0 viene scartato. Il TTL raccomandato per IPv4 è 64 (RFC 791 e RFC 1122).
- **Protocol**: 8 bit usati per indicare quale protocollo di livello superiore è stato utilizzato per creare il payload. Ogni protocollo è identificato da un numero, detto **Assigned Internet Protocol Numbers**, per esempio: 1 = ICMP, 2 = IGMP, 6 = TCP, 17 = UDP (l'elenco completo è reperibile sul sito IANA *www.iana.org/assignments/protocol-numbers*). È noto anche come SAP (Service Address Point). Questo valore è usato solo nell'host di destinazione.
- **Header Checksum**: 16 bit usati per il calcolo della checksum relativa al solo header, per rilevare eventuali bit errati nell'IP datagram ricevuto dal router. In ogni router attraversato viene ricalcolata essendosi modificato l'header per via del decremento del valore TTL.

- **Source IP Address**: 32 bit dell'indirizzo IP del mittente.
- **Destination IP Address**: 32 bit dell'indirizzo IP del destinatario.
- **Options**: ogni opzione è lunga 8 bit e un datagram può contenere più opzioni. Gli 8 bit di un'opzione sono suddivisi in 3 campi:
  - **Copy Flag**: 1 bit che impostato a 0 indica che, in caso di frammentazione, l'opzione va copiata solo sul primo frammento; se impostato a 1 invece va copiata su tutti i frammenti.
  - **Option Class**: 2 bit che impostati al valore 0 indicano che l'opzione è di controllo del datagram o della rete; impostati al valore 2 indicano che l'opzione serve per debug o misurazioni; i valori 1 e 3 sono riservati a usi futuri.
  - **Option Number**: 5 bit che identificano l'opzione nell'ambito della Option Class di appartenenza.
    Per esempio:
    - Option Class = 0 e Option Number = 0: fine delle opzioni;
    - Option Class = 0 e Option Number = 1: riempitivo in caso di nessuna opzione impostata;
    - Option Class = 0 e Option Number = 7: costringe i router ad aggiungere il proprio IP all'elenco delle opzioni; è usato quando si vuole tenere traccia dei percorsi (record route);
    - Option Class = 2 e Option Number = 4: costringe i router ad aggiungere un segnatempo (time stamp) in millisecondi per ricostruire il tempo impiegato da un pacchetto lungo una strada.

    Le opzioni sono registrate sul sito IANA, dove si trova l'elenco completo e aggiornato: "IP Option Numbers" *www.iana.org/assignments/ip-parameters*.
- **Padding**: riempitivo con dati fittizi la cui dimensione dipende dal numero di opzioni presenti. È usato per rendere l'header di lunghezza multipla di 32 bit.

FISSA LE CONOSCENZE

- Qual è la lunghezza minima dell'header IP?
- Perché nell'header IP sono presenti due campi relativi alla lunghezza (HLEN e Total Length)?
- Descrivi il meccanismo della frammentazione attraverso gli opportuni campi dell'header IP.
- Spiega come viene usato il campo TTL.
- Il campo Protocol contiene un numero compreso tra 0 e 255; come si può conoscere a quale protocollo si riferisce?
- Descrivi il formato del campo Options.
- A cosa serve il campo Padding?

2 LA STRUTTURA DEGLI INDIRIZZI IP

2.1 L'indirizzo IP

Il protocollo IP fornisce l'indirizzo logico degli host di una rete TCP/IP. A ciascun host viene assegnato un indirizzo IP univoco rispetto alla rete su cui sta lavorando. Quindi l'indirizzo IP assegnato a un host non solo rappresenta l'host, ma indica, unitamente alla subnet mask, anche su quale sottorete logica si trovi, consentendo l'inoltro dei pacchetti da parte dei router solo quando è necessario (i concetti di sottorete e subnet mask saranno spiegati nella lezione successiva).
In realtà un indirizzo IP non identifica un host, ma una sua interfaccia di rete. Nel caso in cui un nodo abbia più interfacce verso la rete, ogni interfaccia avrà un indirizzo diverso. Per esempio, un computer può avere due interfacce di rete: una wired (802.3) e una wireless (802.11), ciascuna con un suo indirizzo IP.
La FIGURA 3 mostra un router che collega 3 reti e che quindi ha almeno 3 distinti indirizzi IP, uno per ogni interfaccia di rete.

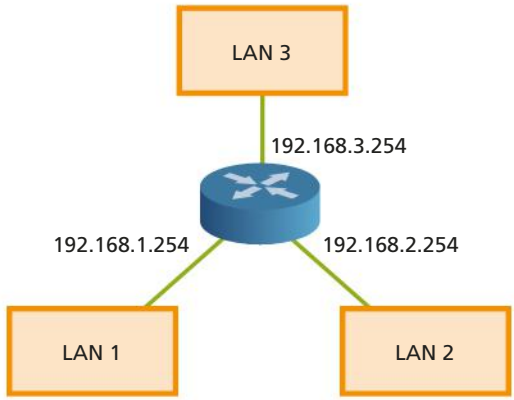


FIGURA 3 Un router che collega 3 reti usando 3 indirizzi IP diversi

Gli indirizzi IPv4 sono numeri di 32 bit suddivisi in 4 byte (anche detti **ottetti**). Vengono solitamente espressi nella notazione **decimale puntata** costituita da 4 numeri decimali compresi tra 0 e 255, separati da un punto.

Per esempio:

192.168.1.254

Più raramente si trovano espressi nella notazione binaria:

11000000.10101000.00000001.11111110

L'indirizzo IPv4 usa 32 bit, quindi il numero massimo, teorico, di indirizzi a disposizione è $\mathbf{2^{32} = 4.294.967.296}$.
Il numero è teorico, in quanto non tutti gli indirizzi possono essere usati e assegnati a un'interfaccia di rete, come si vedrà in seguito. Quando fu implementato l'IP si trattava di un numero elevato di indirizzi, ma, in realtà, era destinato a esaurirsi con la crescente diffusione che ebbe la rete Internet alla fine degli anni Ottanta e negli anni Novanta. All'inizio furono implementate delle tecniche per ottimizzare l'uso degli indirizzi IP: subnetting, CIDR, VLSM, che vedremo nelle lezioni successive. Nel contempo si studiò un nuovo protocollo di rete, IPv6, destinato a sostituire la precedente versione 4, che risolse il problema della carenza di indirizzi e introdusse nuove funzionalità, per rispondere alle mutate esigenze dovute all'enorme diffusione di Internet. Attualmente IPv4 e IPv6 continuano a coesistere e le schede di rete degli host usano entrambi gli indirizzi. Nell'Unità 4 approfondiremo il protocollo IPv6.

Gli indirizzi IP pubblici sono assegnati da **ICANN** (**I**nternet **C**orporation for **A**ssigned **N**ames and **N**umbers), che ha assunto la funzione di **IANA** (**I**nternet **A**ssigned **N**umbers **A**uthority).

ICANN è un'autorità che opera a livello mondiale, non assegna singoli indirizzi, ma blocchi di indirizzi agli Internet Service Provider e ad altre organizzazioni.
ICANN delega la gestione di blocchi di indirizzi IP a enti locali denominati **RIR (Regional Internet Registry)** che tutti insieme formano la ASO (Address Supporting Organization) di ICANN.
Ogni RIR assegna gli indirizzi per una specifica zona del mondo. Al momento esistono 5 RIR nel mondo, ciascuno con la sua area di competenza (FIGURA 4).

FIGURA 4 Regional Internet Registry

2.2 Le classi

L'indirizzo IP di un host **è diviso in due parti**:

- **NetID** (o NetworkID): è l'indirizzo della rete in cui si trova l'host;
- **HostID**: è l'indirizzo dell'host all'interno della rete NetID.

Se si assegna un numero di bit elevato al NetID, avremo molte piccole reti, ciascuna con pochi host. Al contrario, se si usano pochi bit per il NetID, si avranno poche grandi reti, ciascuna con un elevato numero di host.

In base al valore dei bit più significativi, gli indirizzi IP sono suddivisi in 5 classi: A, B, C, D, E, ma solo le prime 3: A, B e C, possono essere utilizzate per assegnare indirizzi agli host. Questo sistema di indirizzamento è detto **classful** (FIGURA 5).

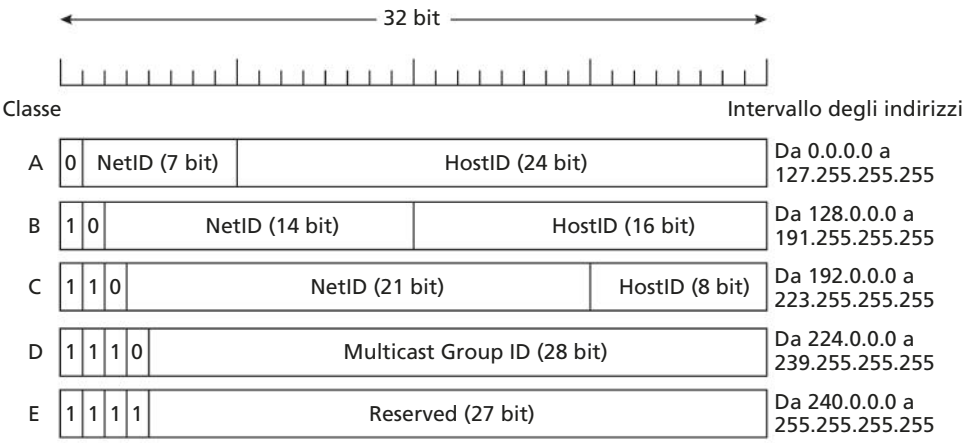


FIGURA 5 Le classi degli indirizzi IP

Classe A
- Ha il primo bit del primo ottetto fisso al valore 0;
- dedica il primo ottetto alla rete (N) e gli altri 3 agli host (H), è quindi nella forma NHHH;
- ha un range di indirizzi da 0.0.0.0 a 127.255.255.255;
- ha 7 bit dedicati alla rete ma può indirizzare solo $2^7 - 2 = 126$ reti perché i valori 0 (this Network) e 127 (loopback net) non possono essere assegnati in quanto indirizzi riservati;
- ha 24 bit dedicati agli host e può indirizzare $2^{24} - 2 = 16.777.214$ host per ogni rete perché i valori 0.0.0 (this host) e 255.255.255 (broadcast) non possono essere assegnati in quanto indirizzi riservati;
- gli indirizzi in classe A sono adatti a network di grandi dimensioni.

Classe B
- Ha i primi 2 bit del primo ottetto fissi al valore 10;
- dedica i primi 2 ottetti alla rete e gli altri 2 agli host (NNHH);
- ha un range di indirizzi da 128.0.0.0 a 191.255.255.255;
- ha 14 bit dedicati alla rete e può indirizzare $2^{14} = 16.384$ reti;
- ha 16 bit dedicati agli host e può indirizzare $2^{16} - 2 = 65.534$ host per ogni rete perché i valori 0.0 (this host) e 255.255 (broadcast) non possono essere assegnati in quanto indirizzi riservati;
- gli indirizzi in classe B sono adatti a network di medie dimensioni.

Classe C
- Ha i primi 3 bit del primo ottetto fissi al valore 110;
- dedica i primi 3 ottetti alla rete e l'ultimo agli host (NNNH);
- ha un range di indirizzi da 192.0.0.0 a 223.255.255.255;
- ha 21 bit dedicati alla rete e può indirizzare $2^{21} = 2.097.152$ reti;
- ha 8 bit dedicati agli host e può indirizzare $2^8 - 2 = 254$ host per ogni rete perché i valori 0 (this host) e 255 (broadcast) non possono essere assegnati in quanto indirizzi riservati;
- gli indirizzi in classe C sono adatti a network di piccole dimensioni.

Classe D
- Ha i primi 4 bit del primo ottetto fissi al valore 1110;
- ha un range da 224.0.0.0 a 239.255.255.255;
- non sono indirizzi assegnabili ai singoli host;
- servono per il multicasting cioè a indirizzare gruppi di host.

Classe E
- Ha i primi 4 bit del primo ottetto fissi al valore 1111;
- ha un range da 240.0.0.0 a 255.255.255.255, ma l'indirizzo con tutti 1 è escluso in quanto indirizzo riservato;
- non sono indirizzi assegnabili ai singoli host;
- sono indirizzi riservati per usi futuri o per scopi sperimentali.

esercizio

➜ PROBLEMA

Dato l'indirizzo IP **130.10.67.160** determinare:

1) la classe di appartenenza
2) l'indirizzo di rete
3) l'indirizzo di broadcast
4) l'HostID

➜ SVOLGIMENTO

Per prima cosa convertiamo l'indirizzo in binario:

1° ottetto	2° ottetto	3° ottetto	4° ottetto
130	10	67	160
10000010	**00001010**	**01000011**	**10100000**

Ricordiamo che:
- per individuare la classe dell'indirizzo è necessario esaminare i bit più significativi (Figura 5);
- un indirizzo IP di rete ha tutti i bit dell'HostID uguali a 0;
- un indirizzo IP di broadcast ha tutti i bit dell'HostID uguali a 1.

Risposte:
1) primo bit = **1**, secondo bit = **0** , l'indirizzo è di **classe B**
2) indirizzo di rete = **130.10.0.0**
3) indirizzo di broadcast = **130.10.255.255**
4) HostID = **67.160**

2.3 Indirizzi riservati o speciali

Esistono degli indirizzi che non possono essere assegnati agli host di una rete.

- **Indirizzi di rete**: sono quegli indirizzi che hanno tutti 0 nella parte dedicata agli host:
  classe A: X.0.0.0
  classe B: X.Y.0.0
  classe C: X.Y.Z.0
  Individuano la rete corrente.
- **Indirizzi di host sulla rete corrente**: sono quegli indirizzi che hanno tutti 0 nella parte dedicata alla rete:
  classe A: 0.X.Y.Z
  classe B: 0.0.X.Y
  classe C: 0.0.0.X
- **Indirizzi di broadcast**: sono quegli indirizzi che hanno tutti 1 nella parte dedicata agli host:
  classe A: X.255.255.255
  classe B: X.Y.255.255
  classe C: X.Y.Z.255
  Sono indirizzi utilizzati per mandare pacchetti a tutti gli host di quella rete. Sono indirizzi di tipo **broadcast limited** essendo riferiti solo alla rete locale specificata.
- **Indirizzo di rete di default**: è l'indirizzo con tutti 0 (0.0.0.0), si usa per il routing o per identificare l'host corrente in fase di bootstrap.
- **Indirizzo di broadcast di default**: è l'indirizzo con tutti 1 (255.255.255.255) ed è usato per inviare pacchetti a tutta la rete corrente. È anch'esso un indirizzo di tipo broadcast limited essendo riferito solo alla rete locale corrente.
- **Indirizzo di loopback**: è l'indirizzo 127.0.0.1 e serve per verificare se l'host è correttamente configurato rispetto al protocollo TCP/IP, quando ancora non gli è stato assegnato un indirizzo IP. Rappresenta il **localhost**, cioè l'indirizzo IP dell'host stesso.

- **Indirizzi APIPA** (**A**utomatic **P**rivate **IP A**ddressing)**:** il range di indirizzi 169.254.0.0 - 169.254.255.255 è riservato per l'autoconfigurazione degli host. Un tipico impiego è nelle reti che usano l'assegnazione dinamica degli indirizzi IP con DHCP (il protocollo DHCP è descritto nell'Unità 7): quando un host non riceve l'indirizzo IP dal server DHCP si assegna in automatico un indirizzo 169.254.X.Y.

2.4 Indirizzi pubblici/privati e statici/dinamici

Gli indirizzi che si affacciano sulla rete Internet sono detti **pubblici** e sono univoci in tutto il pianeta. Poiché il numero di indirizzi IPv4 non è sufficiente per indirizzare tutti gli host esistenti (2^{32} = 4.294.967.292 indirizzi diversi possibili contando anche gli indirizzi speciali) sono stati riservati dei range di indirizzi **privati** per ogni classe.

> Gli indirizzi privati non possono essere utilizzati per affacciarsi direttamente alla rete pubblica Internet, ma servono per indirizzare gli host di reti private.

#prendinota

Esistono moltissimi host appartenenti a reti locali diverse aventi lo stesso indirizzo IP (privato), che però si affacciano a Internet mediante un indirizzo IP univoco (pubblico).

Gli host con indirizzi privati si connettono a Internet mediante un indirizzo pubblico gestito da un **proxy server** o mediante un router **NAT** (Network Address Translation).

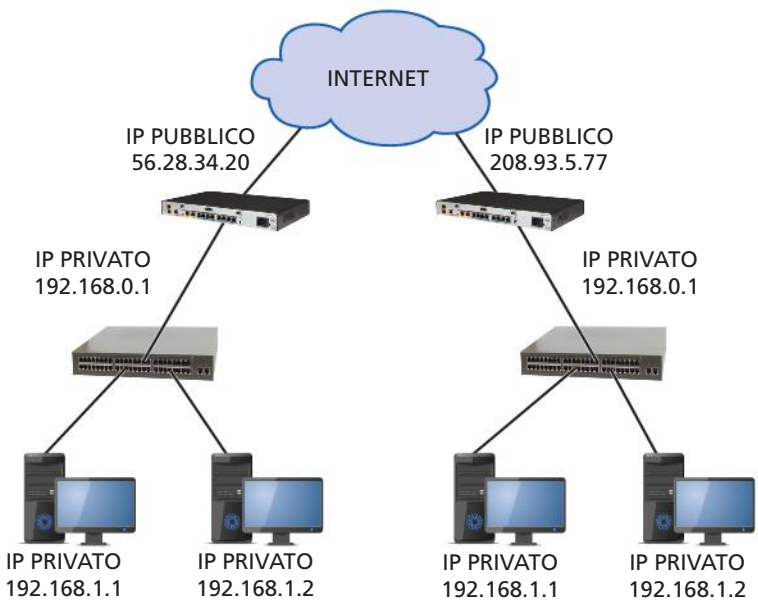
I range di indirizzi privati sono definiti in **RFC 1918** "Address Allocation for Private Internets" e valgono:

- per la classe A: **da 10.0.0.0 a 10.255.255.255** (blocco di 2^{24} indirizzi);
- per la classe B: **da 172.16.0.0 a 172.31.255.255** (blocco di 2^{20} indirizzi);
- per la classe C: **da 192.168.0.0 a 192.168.255.255** (blocco di 2^{16} indirizzi, utilizzabile come se fosse una classe B avendo gli ultimi 2 ottetti a 0).

Un'altra tecnica utilizzata per sopperire allo scarso numero di indirizzi IP a disposizione è quella di assegnare, in particolare agli utenti privati, degli indirizzi **dinamici**, cioè degli indirizzi che cambiano ogni volta che ci si collega a Internet.
In questo modo gli ISP (Internet Service Provider) possono utilizzare uno stesso indirizzo IP pubblico per più utenti in momenti diversi, sfruttando il fatto che difficilmente un utente privato resta collegato 24 ore su 24 a Internet.
Alle aziende vengono invece solitamente assegnati degli indirizzi **statici**, cioè fissati una volta per tutte, che tali aziende utilizzeranno per collegarsi a Internet, quindi usati come indirizzi pubblici per connettere tutti gli host dell'azienda che hanno indirizzi privati.

Come già sottolineato in precedenza, un altro modo per sopperire allo scarso numero di indirizzi IP è usare il **CIDR** (Classless InterDomain Routing) di cui tratteremo nella Lezione 5.
L'arrivo dell'IPv6 porrà fine a queste carenze.
La FIGURA 6 mostra due LAN che si affacciano a Internet con due indirizzi IP pubblici (dunque univoci in tutto il pianeta) e organizzate

FIGURA 6 Host di LAN diverse con stessi IP privati



internamente con gli stessi indirizzi privati che, pur duplicati, non creano conflitti o fraintendimenti. È come avere a Roma e a Milano due "via Garibaldi 22": l'ufficio postale saprà comunque indirizzare correttamente la posta, trattandosi di due città diverse. Allo stesso modo faranno i provider trattandosi di due LAN diverse.

esercizio

➜ PROBLEMA

Non tutti gli indirizzi IP possono essere assegnati a un host. Indicare per ciascuno dei seguenti indirizzi se può essere assegnato o meno a un host, giustificando la risposta.

150.100.255.255 175.100.255.18 195.234.253.0 100.0.0.23 188.258.221.176
127.34.25.189 224.156.217.73 255.255.255.255 210.3.80.43 141.230.0.0

➜ SVOLGIMENTO

Indirizzo IP	È un indirizzo IP valido per un host?	Spiegazione della risposta
150.100.255.255	No	È l'indirizzo di broadcast della rete 150.100.0.0, classe B (tutti bit a 1 negli ultimi 2 ottetti)
175.100.255.18	Sì	È un indirizzo di host valido appartenente alla rete 175.100.0.0, classe B
195.234.253.0	No	È un indirizzo di rete di classe C (tutti i bit a 0 nell'ultimo ottetto)
100.0.0.23	Sì	È un indirizzo di host valido appartenente alla rete 100.0.0.0, classe A
188.258.221.176	No	L'indirizzo è errato: secondo ottetto = 258 (i valori possibili sono nel range 0 - 255)
127.34.25.189	No	127 è un indirizzo di classe A non assegnabile a una rete, è usato per il loopback
224.156.217.73	No	È un indirizzo di classe D quindi riservato per il multicasting
210.3.80.43	Sì	È un indirizzo di host valido appartenente alla rete 210.3.80.0, classe C
255.255.255.255	No	È l'indirizzo di broadcast limitato
141.230.0.0	No	È un indirizzo di rete, classe B (tutti i bit a 0 negli ultimi 2 ottetti)

FISSA LE CONOSCENZE

- Descrivi la classe A degli indirizzi IP in termini di range degli indirizzi, numero massimo di reti e numero massimo di host per rete, giustificando le risposte.
- Descrivi la classe B degli indirizzi IP in termini di range degli indirizzi, numero massimo di reti e numero massimo di host per rete, giustificando le risposte.
- Descrivi la classe C degli indirizzi IP in termini di range degli indirizzi, numero massimo di reti e numero massimo di host per rete, giustificando le risposte.
- Descrivi alcuni indirizzi IP riservati.
- Che differenza c'è tra indirizzi IP pubblici e privati?
- Che differenza c'è tra indirizzi IP statici e dinamici?

3 PIANIFICAZIONE DI RETI IP: IL SUBNETTING

3.1 Subnetting: dividere la rete in sottoreti

Per ottimizzare il traffico in una rete risulta particolarmente utile suddividerla in una serie di sottoreti logiche, collegate tra loro da router interni alla rete stessa. Questa operazione di segmentazione della rete in sottoreti prende il nome di **subnetting** ed è stata specificata da IETF nel 1985 in **RFC 950** "Internet Standard Subnetting Procedure".

Il subnetting si realizza "sacrificando" alcuni dei bit che le classi A, B e C dedicano agli host per definire un indirizzo di sottorete, SubnetID, come mostrato nella FIGURA 7.

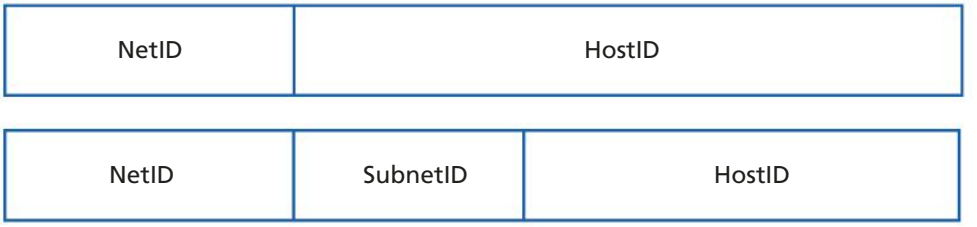


FIGURA 7 Suddivisione dei 32 bit dell'indirizzo IP

Per segmentare una rete occorre, in fase di progettazione, stabilire **quante subnet** servono e, di conseguenza, quanti bit occorrono per indirizzarle univocamente. Nel calcolo è necessario anche valutare il **numero massimo di host previsto per ogni subnet** e in base a ciò calibrare il quantitativo di bit da suddividere rispettivamente tra subnet e host. Il risultato di questa attività è il **#piano di indirizzamento** della rete.

#techwords

Piano di indirizzamento
È il risultato della pianificazione dell'assegnazione degli indirizzi IP agli host. È realizzato dall'amministratore di rete utilizzando gli indirizzi IP di rete a disposizione e applicando, se necessario, il subnetting. Un piano di indirizzamento deve anche considerare possibili sviluppi futuri della rete.

È necessario evitare che nella parte subnet e in quella host vi siano **contemporaneamente** tutti 0 o tutti 1 perché diventerebbero indirizzi speciali, riservati rispettivamente alla rete e al broadcast. Questo problema si risolve eliminando dal piano di indirizzamento gli HostID con tutti i bit = 0 e con tutti i bit = 1.

Inoltre, l'amministratore di rete deve disporre di un indirizzo IP appartenente a una classe opportuna rispetto al numero di subnet da realizzare e al numero di host che devono essere inseriti in ciascuna subnet.
Vediamo con un esempio come individuare il numero di bit necessari per creare le subnet e a quale classe deve appartenere l'indirizzo IP di rete.

esempio

Supponiamo di dover realizzare 50 subnet: dobbiamo usare una SubnetID di **6 bit**, essendo $2^6 = 64$ maggiore di 50, mentre 5 bit non sarebbero bastati ($2^5 = 32$ minore di 50). Se usassimo un indirizzo di **classe C** resterebbero solo **2 bit** per gli indirizzi di host, infatti:

classe C: 8 bit per HostID, ne usiamo 6 per SubnetID, restano 8 – 6 = 2 bit per HostID

$2^2 - 2 =$ **2 indirizzi di host per ciascuna subnet**

(si deve sottrarre 2 in quanto per gli host non si possono usare i due indirizzi riservati "tutti 0" e "tutti 1").

Quindi, un indirizzo IP in classe C, a parte situazioni molto particolari come il collegamento tra due router che vedremo nelle prossime Lezioni, non è adeguato in quanto il numero di host per subnet è troppo limitato e non offrirebbe la possibilità di inserire nuovi host in futuro.
Se, invece, scegliamo un indirizzo di **classe B** avremmo **10 bit** per gli indirizzi di host:

classe B: 16 bit per HostID, ne usiamo 6 per SubnetID, restano 16 – 6 = 10 bit per HostID

$2^{10} - 2$ = **1.022 indirizzi di host per ciascuna subnet**

#prendinota

"All one" / "All zero" subnet
L'utilizzo di SubnetID con i bit tutti a 0 (la prima subnet) e tutti a 1 (l'ultima subnet) non era consentito nelle prime specifiche, per evitare di incorrere in errori di progettazione. L'RFC 1878 del 1995 annullò questo vincolo, pur continuando a essere rispettato da molti amministratori di rete. Nei router Cisco le subnet *all one* e *all zero* sono abilitate di default, lasciando però la possibilità di disabilitarle con il comando *no ip subnet-zero* (o *one*).

È buona regola abbondare anche nel calcolo dei bit da dedicare alle subnet per garantire future creazioni di nuove subnet senza dover ripianificare tutti gli indirizzi. Se, per esempio, il progetto prevede 64 subnet è consigliabile usare 7 bit anziché i 6 bit che basterebbero senza resti.

Nota: negli esercizi che svolgeremo, per semplicità, non applicheremo queste considerazioni, ma useremo sempre il numero minimo di bit necessari per creare le subnet richieste.

In generale, posto N il numero di subnet richieste, si avrà che il numero X di bit necessari a indirizzarle univocamente è dato da:

$$X = \lfloor \log_2 N \rfloor + 1$$

cioè parte intera del logaritmo in base 2 di N, più 1.

3.2 La subnet mask

Abbiamo visto nella lezione precedente che per capire qual è l'indirizzo IP della rete a cui appartiene un host è sufficiente considerare i bit più significativi. Per esempio:

- **130.10.67.160** è un indirizzo di classe B, dal momento che i primi 2 bit sono "1" e "0" (130_{10} = **10**000010_2);
- sappiamo che un indirizzo di rete deve sempre avere HostID=0;
- quindi l'host con indirizzo 130.10.67.160 appartiene alla rete **130.10.0.0**.

Con l'introduzione del subnetting non è più così semplice, infatti lo stesso indirizzo IP 130.10.67.160 potrebbe appartenere alla seconda subnet di una rete che è stata suddivisa in 4 subnet. In questo caso l'indirizzo di rete dell'host è **130.10.64.0** !

Riprendiamo l'esercizio presentato nel Paragrafo 2.2 della Lezione precedente e introduciamo il subnetting.

esercizio

➜ PROBLEMA

Dato l'indirizzo IP **130.10.67.160** determinare qual è il suo indirizzo di rete, sapendo che la rete è stata suddivisa in **4 subnet**.

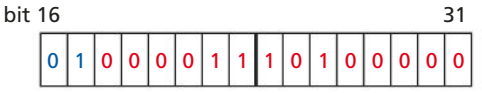
➜ SVOLGIMENTO

L'indirizzo è di classe B, la rete di appartenenza è 130.10.0.0, senza subnetting.
Per creare 4 sottoreti, dobbiamo usare alcuni bit dell'HostID: quanti? $4 = 2^2 \rightarrow 2$ bit.

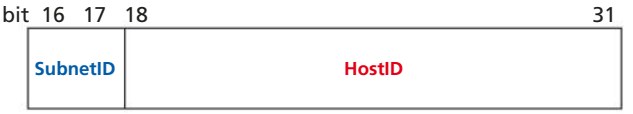
Per calcolare gli indirizzi delle subnet, riprendiamo la tabella con la conversione in binario:

1° ottetto	2° ottetto	3° ottetto	4° ottetto
130	10	67	160
10000010	**00001010**	**01000011**	**10100000**

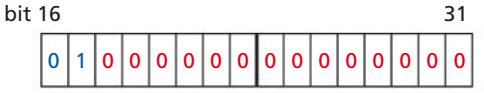
e focalizziamoci sul terzo e quarto ottetto che rappresentano l'HostID:



I 2 bit che si usano per le subnet sono quelli più significativi, da sinistra evidenziati in blu, mentre i restanti, evidenziati in rosso, sono quelli per l'indirizzo di host:



Applichiamo il concetto che un indirizzo IP di rete ha tutti i bit dell'HostID uguali a 0:



e riconvertiamo in decimale i 2 ottetti: $01000000_2 = 64_{10}$ $00000000_2 = 0_{10}$

La risposta al problema è: **130.10.64.0** (indirizzo di rete dell'host 130.10.67.160).

Nell'esercizio è stato semplice calcolare l'indirizzo di rete perché avevamo un dato fondamentale: le **4** subnet.
È quindi necessario inserire questo dato nella configurazione IP di un host, così lo potrà usare anche un router quando inoltra un pacchetto verso la rete di destinazione (il router non inoltra usando l'HostID, come vedremo nell'Unità 5 sul routing).

Si è quindi definita una nuova stringa da 32 bit (esattamente come gli indirizzi IP) che prende il nome di **subnet mask** e ha lo scopo di indicare quanti bit dell'indirizzo appartengono alla rete e quanti all'host.
Nella subnet mask hanno valore 1 i bit in corrispondenza ai bit di rete e sottorete, mentre hanno valore 0 i bit dell'HostID.

Riprendiamo l'esercizio precedente e calcoliamo la subnet mask.

esercizio

➜ PROBLEMA

Dato l'indirizzo IP **130.10.67.160**, determinare la **subnet mask** sapendo che la rete è stata suddivisa in **4 subnet**.

➜ SVOLGIMENTO

Dai calcoli fatti in precedenza, è emerso che per realizzare le 4 subnet si devono usare 2 bit dell'HostID. Questi bit si aggiungono ai 16 bit di rete (Classe B): in totale ci sono 18 bit per la rete (NetID + SubnetID) e 14 bit per l'HostID.
L'indirizzo di rete a cui appartiene l'host è 130.10.64.0.
Per calcolare la subnet mask occorre porre a 1 i bit in corrispondenza ai bit di rete dell'indirizzo e a 0 i bit corrispondenti all'HostID. Successivamente, si divide la stringa in 4 ottetti e si convertono in decimale.
Risultato: la subnet mask è **255.255.192.0**, come mostrato in FIGURA 8.

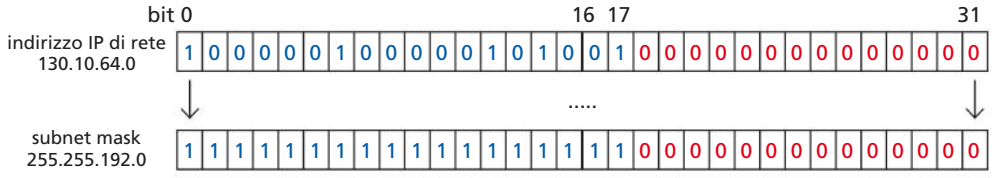


FIGURA 8 Calcolo della subnet mask

Di norma la subnet mask è la stessa per tutte le subnet della rete. In alcuni casi ci possono essere esigenze particolari che richiedono l'uso di subnet mask di lunghezza variabile (un esempio è descritto nella Lezione 5).

Le **subnet mask di default** sono:
- classe A: **255.0.0.0** (11111111.00000000.00000000.00000000)
- classe B: **255.255.0.0** (11111111.11111111.00000000.00000000)
- classe C: **255.255.255.0** (11111111.11111111.11111111.00000000)

Il numero di valori che può assumere un ottetto della subnet mask è ben definito, dal momento che i bit a 1 devono essere consecutivi, come mostrato nella TABELLA 1. Per esempio, un host con indirizzo 192.168.1.1 non potrà mai avere una subnet mask 255.255.255.150!

TABELLA 1 Valori di un ottetto della subnet mask

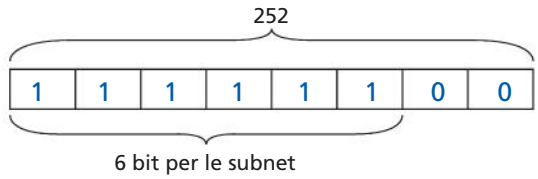
Ottetto in binario	Valore decimale
00000000	0
10000000	128
11000000	192
11100000	224
11110000	240
11111000	248
11111100	252
11111110	254
11111111	255

Quando non si possono dedicare ottetti interi al subnetting come nelle subnet mask di default, ma solo una parte di essi, bisogna saper suddividere l'ottetto in binario e poi ricomporlo nella notazione decimale.
Come abbiamo visto nell'esercizio precedente, può capitare che un numero decimale "nasconda" una parte di subnet e una parte di host.

esempio

Riprendiamo l'esempio descritto nel Paragrafo 3.1: una rete è costituita da 50 subnet, che richiedono 6 bit per essere indirizzate ($2^6 = 64$).
La subnet mask da usare cambia a seconda della classe di appartenenza della rete:

- 255.**252**.0.0 per la classe A (6 bit SubnetID, 18 bit HostID);
- 255.255.**252**.0 per la classe B (6 bit SubnetID, 10 bit HostID);
- 255.255.255.**252** per la classe C (6 bit SubnetID, 2 bit HostID).



I bit a 1 vanno scritti da sinistra verso destra (cioè dal bit più significativo al bit meno significativo) nel primo ottetto che non è dedicato alla rete.

L'ottimizzazione del traffico di rete, evitando le congestioni dei canali, passa attraverso una corretta pianificazione degli indirizzi IP (piano di indirizzamento) e dunque un'appropriata scelta delle maschere di sottorete.
Inoltre, per evitare di dover riconfigurare tutte le macchine, occorre saper prevedere l'eventuale espansione della rete sia in termini di numero di subnet sia in termini di numero di host per ciascuna di esse (**#scalabilità**).

#techwords

Scalabilità
Indica la capacità della rete di aumentare (o diminuire) di scala, in funzione delle necessità e delle disponibilità. Si realizza soprattutto con un'opportuna dislocazione degli apparati di rete, ma anche con un'oculata pianificazione degli indirizzi IP e, di conseguenza, delle subnet mask.

3.3 Il processo di "messa in AND"

Il subnetting e le subnet mask svolgono un ruolo fondamentale nell'ottimizzare il traffico della rete evitando, per esempio, che datagram IP inviati da un host a un altro host, residente nella stessa sottorete, escano e rientrino per giungere a destinazione.
Si individuano, dunque, due tecniche di inoltro dei pacchetti nella rete:

- **forwarding diretto**: host mittente e destinatario hanno lo stesso NetID (incluso il SubnetID), quindi la comunicazione avviene all'interno della stessa rete senza coinvolgere il router;
- **forwarding indiretto**: host mittente e destinatario hanno un diverso NetID (incluso il SubnetID), quindi sono su reti differenti, il pacchetto viene inviato al router che lo inoltrerà a un altro router e così via fino a destinazione.

Il meccanismo che permette di verificare se due host appartengono alla stessa rete (e sottorete) è detto **processo di messa in AND** (ANDing process) bitwise, cioè bit a bit.

Questo processo consiste nel fare:

1. un'operazione di AND bit a bit tra l'indirizzo IP del **mittente** e la subnet mask del **mittente** ottenendo il NetID e il SubnetID del mittente e azzerando l'HostID;
2. un'operazione di AND bit a bit tra l'indirizzo IP del **destinatario** e la subnet mask del **mittente** ottenendo il NetID e il SubnetID e azzerando l'HostID;
3. il confronto tra i due risultati ottenuti:
   - se sono uguali mittente e destinatario sono nella stessa subnet (comunicazione diretta);
   - se sono diversi mittente e destinatario non sono nella stessa subnet (comunicazione attraverso il router).

#prendinota

L'host mittente non conosce la subnet mask del destinatario, quindi può svolgere l'AND solo con la sua subnet mask, per scoprire se l'host di destinazione si trova nella sua stessa rete.

La funzione di messa in AND ha le seguenti proprietà:

- se i due valori confrontati sono entrambi 1, il risultato è 1;
- se uno dei due valori confrontati è 0, il risultato è 0.

La messa in AND rappresenta il prodotto logico in cui basta uno 0 per ottenere risultato 0.

esempio

Esempio di messa in AND

Supponiamo di avere 2 host in classe B e di aver utilizzato 8 bit per mascherare la subnet dell'host mittente:

- host A (mittente): IP_A = 150.169.3.8
- subnet mask (SM): 255.255.255.0
- host B (destinatario): IP_B = 150.169.5.2

1) IP_A AND SM_A:

10010110.10101001.00000011.00001000
11111111.11111111.11111111.00000000
10010110.10101001.00000011.00000000

2) IP_B AND SM_A:

10010110.10101001.00000101.00000010
11111111.11111111.11111111.00000000
10010110.10101001.00000101.00000000

3) I risultati delle due messe in AND sono diversi, quindi i due host non si trovano nella stessa subnet. Dunque il pacchetto andrà inoltrato al di fuori della sottorete.

3.4 Slash notation

È possibile riassumere la coppia indirizzo IP e subnet mask mediante la **slash notation** in cui all'indirizzo IP viene fatto seguire il **prefix length**, un numero decimale che indica il numero di bit a 1 della maschera.

Per esempio:

150.169.3.2/24

indica che la maschera ha 24 bit a 1, e cioè vale 255.255.255.0.
Questa notazione è stata introdotta con l'indirizzamento **classless**, che non prevede l'uso delle classi A-B-C, introdotto con la tecnica CIDR descritta nella Lezione 5.

Case study
Progettare una rete e assegnare indirizzi IP

FISSA LE CONOSCENZE

- Che cos'è il subnetting e come si realizza?
- Spiega come si calcola il numero di bit necessari per indirizzare N subnet.
- Quanto valgono le subnet mask di default per le classi A, B e C?
- Spiega a che cosa serve e come funziona il processo di messa in AND.

4 ESEMPI DI PIANI DI INDIRIZZAMENTO IP

4.1 Piano di indirizzamento in classe B

esercizio

➜ PROBLEMA

Si supponga di voler pianificare l'indirizzamento di una rete progettata per essere suddivisa in almeno 20 sottoreti in grado di contenere ciascuna almeno 80 host.

➜ ANALISI DEL PROBLEMA

A prescindere dall'indirizzo IP pubblico acquistato, scegliamo di utilizzare un indirizzo IP privato in classe B e avere così 2 ottetti a disposizione per il subnetting, più che sufficienti per indirizzare le subnet e gli host previsti.

Avremo dunque:

- indirizzo di rete = 172.16.0.0;
- indirizzo di broadcast per la rete = 172.16.255.255;
- subnet mask di default = 255.255.0.0;
- ci serviranno 5 bit del terzo ottetto per indirizzare le subnet essendo $|\log_2 20| + 1 = 5$, infatti $2^5 = 32$, quindi restano ben 12 indirizzi di sottorete liberi per future espansioni della rete;
- la subnet mask varrà quindi 255.255.248.0 essendo il terzo ottetto uguale a 11111000;
- per gli host restano a disposizione 3 bit del terzo ottetto più tutto il quarto ottetto quindi 11 bit, da cui ben $2^{11} - 2 = 2.046$ indirizzi di host per subnet.

Dal momento che il 1° host della prima subnet (subnet 0) ha indirizzo 172.16.0.1, il 1° host della quinta subnet (subnet 4) ha indirizzo 172.16.32.1 mentre l'ultimo (il 2.046°) ha indirizzo 172.16.39.254. Il 12° host della subnet 7 ha indirizzo 172.16.56.12.

Quando le richieste di progettazione lo consentono (e spesso accade con la classe B) si può assegnare un intero ottetto alle subnet e uno agli host, rendendo la pianificazione di immediata comprensione per l'uomo; nulla ovviamente cambia per la macchina che attua sempre la messa in AND (lo vedremo meglio nel prossimo esercizio).

Se lo avessimo fatto nel nostro esercizio, il 12° host della subnet 7 avrebbe avuto indirizzo 172.16.7.12 con subnet mask 255.255.255.0.

➜ SVOLGIMENTO

Il piano d'indirizzamento risultante è riassunto nella tabella seguente:

n. subnet	Indirizzo di subnet (terzo ottetto)	Indirizzo di broadcast (terzo ottetto)	Range per gli indirizzi di host
0	172.16.0.0 (**00000** 000)	172.16.7.255 (00000 111)	da 172.16.0.1 a 172.16.7.254
1	172.16.8.0 (**00001** 000)	172.16.15.255 (00001 111)	da 172.16.8.1 a 172.16.15.254
2	172.16.16.0 (**00010** 000)	172.16.23.255 (00010 111)	da 172.16.16.1 a 172.16.23.254
3	172.16.24.0 (**00011** 000)	172.16.31.255 (00011 111)	da 172.16.24.1 a 172.16.31.254
4	172.16.32.0 (**00100** 000)	172.16.39.255 (00100 111)	da 172.16.32.1 a 172.16.39.254
5	172.16.40.0 (**00101** 000)	172.16.47.255 (00101 111)	da 172.16.40.1 a 172.16.47.254

n. subnet	Indirizzo di subnet (terzo ottetto)	Indirizzo di broadcast (terzo ottetto)	Range per gli indirizzi di host
6	172.16.48.0 (**00110** 000)	172.16.55.255 (00110 111)	da 172.16.48.1 a 172.16.55.254
7	172.16.56.0 (**00111** 000)	172.16.63.255 (00111 111)	da 172.16.56.1 a 172.16.63.254
8	172.16.64.0 (**01000** 000)	172.16.71.255 (01000 111)	da 172.16.64.1 a 172.16.71.254
9	172.16.72.0 (**01001** 000)	172.16.79.255 (01001 111)	da 172.16.72.1 a 172.16.79.254
10	172.16.80.0 (**01010** 000)	172.16.87.255 (01010 111)	da 172.16.80.1 a 172.16.87.254
11	172.16.88.0 (**01011** 000)	172.16.95.255 (01011 111)	da 172.16.88.1 a 172.16.95.254
12	172.16.96.0 (**01100** 000)	172.16.103.255 (01100 111)	da 172.16.96.1 a 172.16.103.254
13	172.16.104.0 (**01101** 000)	172.16.111.255 (01101 111)	da 172.16.104.1 a 172.16.111.254
14	172.16.112.0 (**01110** 000)	172.16.119.255 (01110 111)	da 172.16.112.1 a 172.16.119.254
15	172.16.120.0 (**01111** 000)	172.16.127.255 (01111 111)	da 172.16.120.1 a 172.16.127.254
16	172.16.128.0 (**10000** 000)	172.16.135.255 (10000 111)	da 172.16.128.1 a 172.16.135.254
17	172.16.136.0 (**10001** 000)	172.16.143.255 (10001 111)	da 172.16.136.1 a 172.16.143.254
18	172.16.144.0 (**10010** 000)	172.16.151.255 (10010 111)	da 172.16.144.1 a 172.16.151.254
19	172.16.152.0 (**10011** 000)	172.16.159.255 (10011 111)	da 172.16.152.1 a 172.16.159.254
20	172.16.160.0 (**10100** 000	172.16.167.255 (10100 111)	da 172.16.160.1 a 172.16.167.254

4.2 Progettare una rete con subnet

esercizio

➜ PROBLEMA

Progettare una rete suddivisa in 3 sottoreti locali con le seguenti caratteristiche:

- 15 PC client;
- 3 computer server;
- 1 server e 5 client per ciascuna sottorete;
- 1 router per collegare tra loro le 3 sottoreti.

Evidenziare gli apparati di rete e i mezzi trasmissivi utilizzati, assegnare tutti gli indirizzi IP necessari sapendo di avere a disposizione una rete in classe B con indirizzo **158.110.0.0** e di scegliere per comodità come subnet mask **255.255.255.0**.

#prendinota

Per convenzione, gli amministratori di rete solitamente assegnano all'interfaccia del router l'indirizzo più alto disponibile nella sottorete e ai server quello più basso.

➜ ANALISI DEL PROBLEMA

Considerato lo scenario di rete proposto, oltre al router è necessario prevedere 3 switch per ciascuna sottorete a cui collegare i computer e l'interfaccia del router, secondo una topologia a stella estesa o gerarchica.
Ogni switch è dotato di porte Ethernet a 10/100 Mbps a cui collegare i PC client e a 1 Gbps a cui collegare il server e il router. I collegamenti usano cavi UTP cat. 5e.
Il router ha 3 porte Ethernet a 1 Gbps, le rispettive interfacce sono configurate con un indirizzo IP diverso relativo a ciascuna delle 3 subnet create.

Alcune osservazioni per una buona pianificazione degli indirizzi IP:

- il terzo ottetto è stato interamente dedicato al subnetting per numerare le 3 reti con i valori 1, 2 e 3;

- alle 3 interfacce di rete del router (che funge da gateway per ciascuna rete) è stato assegnato l'ultimo indirizzo disponibile del range (quarto ottetto = 254, mentre il 255 è riservato al broadcast);
- a ogni server è stato assegnato il primo indirizzo disponibile del range (quarto ottetto = 1);
- a ogni host è stato assegnato un IP statico crescente (quarto ottetto con valori da 2 a 6).

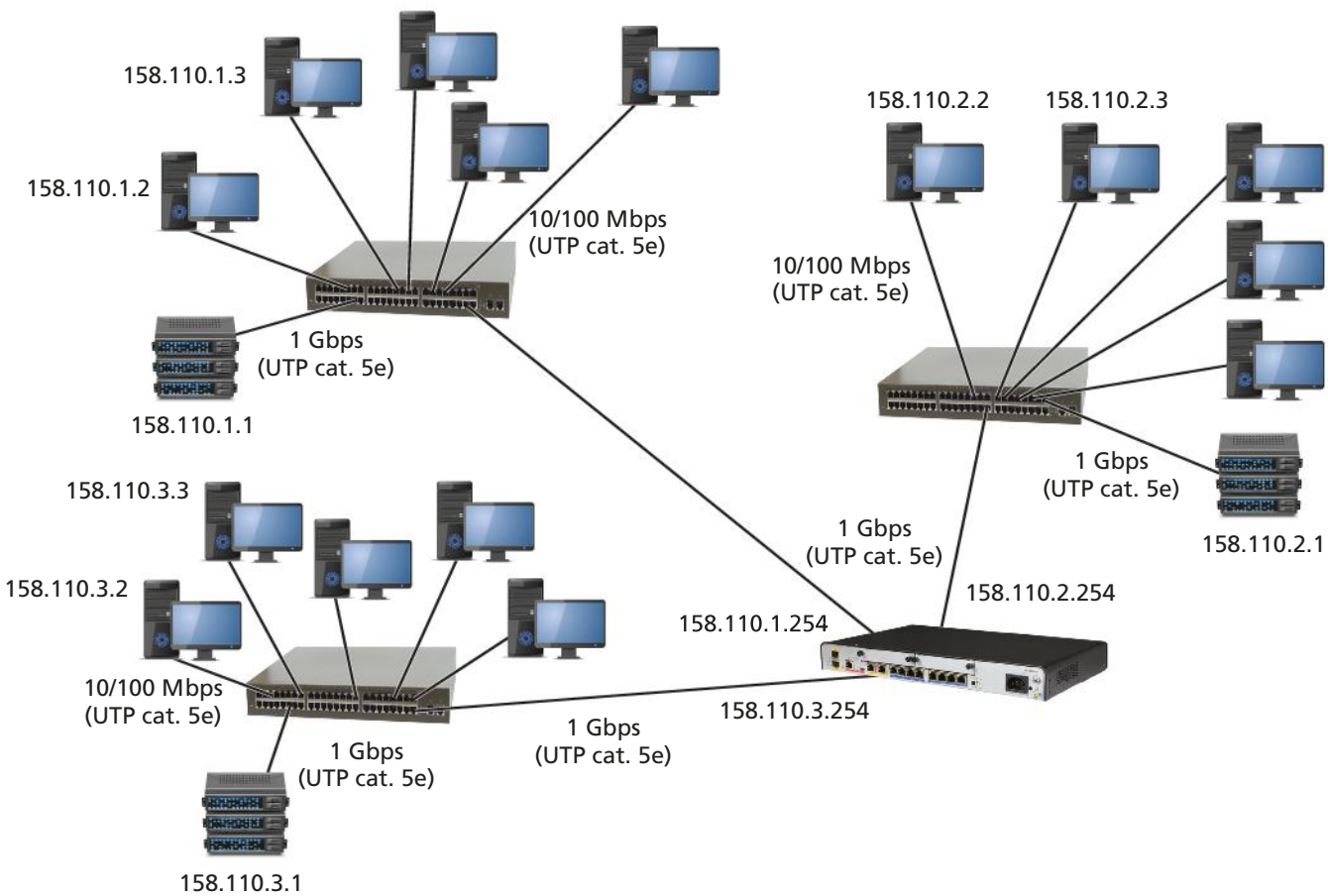
➜ SVOLGIMENTO

Il piano di indirizzamento della rete è mostrato nella seguente tabella:

Rete	Subnet mask	Router	Server	Host	Broadcast
158.110.1.0	255.255.255.0	158.110.1.254	158.110.1.1	da 158.110.1.2 a 158.110.1.6	158.110.1.255
158.110.2.0	255.255.255.0	158.110.2.254	158.110.2.1	da 158.110.2.2 a 158.110.2.6	158.110.2.255
158.110.3.0	255.255.255.0	158.110.3.254	158.110.3.1	da 158.110.3.2 a 158.110.3.6	158.110.3.255

La FIGURA 9 mostra la progettazione della rete con apparati, computer e mezzi trasmissivi, più gli indirizzi IP assegnati alle varie schede di rete sulla base del piano di indirizzamento definito (per non appesantire il disegno non sono stati scritti gli indirizzi di tutti i PC).

FIGURA 9 Schema della rete



4.3 Calcolare il numero di subnet e di host

esercizio

➜ PROBLEMA

Dato l'IP address **150.185.3.25** e la subnet mask **255.255.224.0**, quante sottoreti sono state create e quanti indirizzi di host sono disponibili per ciascuna sottorete?

➜ ANALISI DEL PROBLEMA

L'indirizzo IP 150.185.3.25 è di classe B, quindi per creare le sottoreti si usano alcuni bit del 3° e 4° ottetto (HostID).
È la subnet mask che rivela "quanti" di questi 16 bit sono stati usati per creare delle subnet: 255.255.224.0. Dal momento che il 4° ottetto è uguale a 0, esaminiamo il 3° ottetto:

$$224_{10} = \mathbf{111}00000_2 \rightarrow \text{sono stati usati 3 bit}$$

➜ SVOLGIMENTO

Con 3 bit si possono avere $2^3 = 8$ combinazioni di 1 e 0 : 000, 001, 010, 011, 100, 101, 110, 111, quindi 8 subnet.
La classe B ha l'HostID di 16 bit, sottraendo i 3 bit usati per creare le subnet restano 13 bit per gli indirizzi di host.
Ora abbiamo tutti i dati utili per rispondere alle domande del problema:
– sono state create **8 subnet**;
– ciascuna subnet ha $2^{13} - 2 =$ **8190 indirizzi di host** a disposizione.

esercizio

➜ PROBLEMA

Dato l'IP address **150.185.3.25** e la subnet mask **255.255.224.0**, fornire:
1) l'indirizzo IP di un host appartenente alla stessa sottorete dell'host 150.185.3.25;
2) l'indirizzo IP di un host appartenente alla stessa rete locale dell'host 150.185.3.25 ma non alla stessa sottorete.

➜ ANALISI DEL PROBLEMA

Nel precedente esercizio, abbiamo stabilito che nella rete 150.185.0.0 con subnet mask 255.255.224.0 sono state create 8 subnet. Nella tabella seguente sono indicati i rispettivi indirizzi:

n. subnet	Indirizzo di subnet (terzo ottetto)
0	150.185.0.0 (**000** 00000)
1	150.185.32.0 (**001** 00000)
2	150.185.64.0 (**010** 00000)
3	150.185.96.0 (**011** 00000)
4	150.185.128.0 (**100** 00000)
5	150.185.160.0 (**101** 00000)
6	150.185.192.0 (**110** 00000)
7	150.185.224.0 (**111** 00000)

➜ SVOLGIMENTO

Convertiamo in binario il 3° ottetto dell'indirizzo 150.185.3.25 per individuare a quale subnet appartiene:

$$3_{10} = \mathbf{000}00011_2$$

I bit in rosso indicano che appartiene alla subnet 0.
Ora abbiamo tutti i dati utili per rispondere alle domande del problema:

1) **150.185.10.100** è un indirizzo IP di un host sulla stessa subnet 0;

2) **150.185.165.30** è un indirizzo IP di un host appartenente a una differente subnet, la subnet 5.

4.4 Individuare la subnet di appartenenza

esercizio

➜ PROBLEMA

Dato l'indirizzo IP **192.198.1.75** e la subnet mask **255.255.255.240**, indicare a quale host appartiene e in quale subnet si trova.

➜ ANALISI DEL PROBLEMA

È un indirizzo di classe C, l'HostID è limitato agli 8 bit del 4° ottetto. Convertiamo dunque in binario il 4° ottetto della subnet mask:

$$240_{10} = \mathbf{1111}0000_2 \rightarrow 2^4 = 16 \text{ subnet}$$

➜ SVOLGIMENTO

Convertiamo in binario il 4° ottetto dell'indirizzo: $75_{10} = 01001011_2$.
Applichiamo la maschera (4 bit per subnet, 4 bit per host): **1111**0000.

Otteniamo:

Subnet: **0100** $\rightarrow 4_{10} \rightarrow$ subnet 4

Host: **1011** $\rightarrow 11_{10} \rightarrow$ 11° host

L'indirizzo 192.198.1.75 è l'**11° host della subnet 4**.

FISSA LE CONOSCENZE

- Descrivi i passi da compiere quando si richiede di realizzare il piano di indirizzamento di una rete, della quale si conosce il numero di dispositivi minimo che sarà presente in ogni subnet.
- Perché, nel caso di un indirizzo di classe B, è più comodo assegnare un intero ottetto alle subnet e un intero ottetto agli host? Quali sono gli ottetti coinvolti?
- Che operazione deve essere svolta sull'indirizzo IP di un host, quando si scrive "applichiamo la maschera"?
- All'interno del range di indirizzi disponibili per una rete o sottorete, quali sono solitamente assegnati ai router e quali ai server?

Esercizio commentato
Il subnetting

5 PIANIFICAZIONE DI RETI IP: CIDR E VLSM

5.1 CIDR e la tecnica del supernetting

La suddivisione degli indirizzi in classi può creare un grosso spreco di indirizzi, basti pensare all'esercizio del Paragrafo 4.1 della Lezione precedente, in cui erano richiesti 80 host per subnet e la pianificazione ne consentiva invece 2.046.
D'altro canto un indirizzo in classe A avrebbe comportato un numero di indirizzi inutilizzati ancora maggiore, mentre la classe C non avrebbe consentito la pianificazione. Infatti, un indirizzo di classe C ha solo 8 bit per subnet e host, mentre per soddisfare le richieste dell'esercizio servivano 5 bit per le 20 subnet e 7 bit per gli 80 host in ogni subnet.

Per cercare di porre rimedio a sprechi e carenze, in attesa dell'IPv6, nel 1993 è stato introdotto un nuovo schema di indirizzamento, la **CIDR** (**Classless Inter Domain Routing**, pronunciata "*saider*"), anche nota come **supernetting** perché crea una super rete composta da più reti. In pratica la CIDR non applica subnetting ed elimina il concetto di classe di indirizzi (classless).

Il suo utilizzo è particolarmente indicato per gli indirizzi di rete in classe C (che è la meno costosa) visto il ridotto numero di bit con cui tale classe permette di giostrare. L'idea di partenza è quella di assegnare a una rete una sequenza contigua di indirizzi in classe C senza fare subnetting, anziché un unico e più costoso indirizzo in classe B (ce ne sono meno) con cui fare subnetting.
La subnet mask andrà però calcolata correttamente, e dovrà essere una maschera di classe B, se si vuole far figurare gli indirizzi in classe C come sottoreti appartenenti a un'unica rete. Vediamo con un esempio pratico come comportarsi.

esempio

Supponiamo di dover indirizzare 1.500 host e che quindi un indirizzo in classe C non basti ($2^8 - 2 = 254$) e uno in classe B sia eccessivo ($2^{16} - 2 = 65.534$).

Prendiamo allora 6 indirizzi in classe C consecutivi:

1° 200.45. 8.0 = 11001000.00101101.**00001**000.00000000
2° 200.45. 9.0 = 11001000.00101101.**00001**001.00000000
3° 200.45.10.0 = 11001000.00101101.**00001**010.00000000
4° 200.45.11.0 = 11001000.00101101.**00001**011.00000000
5° 200.45.12.0 = 11001000.00101101.**00001**100.00000000
6° 200.45.13.0 = 11001000.00101101.**00001**101.00000000

Ciascuno di loro consentirà di indirizzare 254 host avendo l'ultimo ottetto interamente libero per gli host, per un totale di 254 x 6 = 1.524 host, quindi misuratamente oltre le richieste di pianificazione.
Concentriamo ora la nostra attenzione sul terzo ottetto e notiamo che i primi 5 bit più significativi sono uguali per tutti e 6 gli indirizzi: **00001**, mentre gli ultimi 3 bit cambiano. Di conseguenza, affinché la messa in AND dia lo stesso risultato di NetID, indicando così che tutti appartengono alla stessa rete, occorrerà che la maschera sia settata al valore:

255.255.248.0 = 11111111.11111111.**11111**000.00000000

#prendinota

In questo contesto nella *slash notation* il prefix length rappresenta il numero di bit comuni a tutte le reti; nell'esempio sono 21: 200.45.8.0/21.

La messa in AND per qualunque host darà sempre come risultato il seguente NetID:

200.45.8.0

cioè l'indirizzo della **prima rete**, che prende il nome di **summary route**, intendendo che questo singolo indirizzo di rete rappresenta un gruppo di reti contiguo.

Con la CIDR non è corretto parlare di subnet mask, in quanto non esistono sottoreti e non viene fatto il subnetting. È invece corretto far riferimento a una maschera per l'intera rete costituita dal blocco di indirizzi IP di rete consecutivi. Tale maschera prende il nome di **netmask**.

Dunque con questa tecnica un singolo indirizzo IP può rappresentare un gruppo di indirizzi IP, snellendo le tabelle di routing, quindi semplificando e velocizzando il lavoro dei router e ottimizzando l'uso degli indirizzi IP. Ovviamente il router deve supportare le specifiche CIDR. IETF ha emesso le prime specifiche nel 1993 con RFC 1519, aggiornato nel 2006 con RFC 4632.

5.2 VLSM e la tecnica della subnet mask variabile

Tutte le subnet di una stessa rete tipicamente usano la stessa subnet mask (**subnetting statico**). Tuttavia questa strategia, pur essendo semplice da implementare e facile da gestire, in alcuni casi spreca spazio di indirizzamento. Alcune subnet possono avere molti host e altre soltanto pochi, ma tutte utilizzano (consumano) l'intero spazio di indirizzi assegnato, con evidente spreco.
Inoltre, il subnetting statico costringe a fissare sia il numero massimo di subnet nella rete sia il numero massimo di host per subnet al momento della progettazione della rete e della conseguente pianificazione degli indirizzi. Qualsiasi eventuale evoluzione della rete stessa (come l'esigenza di più subnet rispetto al previsto o la necessità di superare il numero massimo di host preventivati in una subnet) potrebbe costringere al completo rifacimento del piano di indirizzamento.
In definitiva il subnetting statico non facilita la flessibilità e la scalabilità di una rete.

Per procedere alla corretta pianificazione di una rete di una certa complessità ci si deve sempre porre 4 domande fondamentali:

1. Quante subnet in totale occorrono oggi?
2. Quante subnet in totale occorreranno in futuro?
3. Quanti host ci sono oggi nella più grande sottorete?
4. Quanti host ci saranno in futuro nella più grande sottorete?

Quanto più si riesce a rispondere con precisione a tali domande, tanto più si riuscirà a progettare un'architettura di rete efficiente e duratura.
Nonostante tutti gli accorgimenti, il subnetting statico è intrinsecamente non ottimizzabile: vi saranno sempre subnet con molti più indirizzi di quanti ne servirebbero e senza la possibilità di utilizzare tali indirizzi in eccesso per altre subnet.

Una tecnica che permette ai provider di utilizzare in modo più efficiente lo spazio di indirizzi è detta **VLSM** (**Variable Length Subnet Mask**). Un provider può usare una subnet mask lunga sulle subnet con pochi host e una subnet mask breve sulle subnet con molti host.

#prendinota

I protocolli di routing RIPv2 e OSPF, descritti nell'Unità 5, supportano VLSM.

La complessità aumenta e il concetto di suddivisione degli indirizzi in classi non viene considerato. Si ha quindi un indirizzamento di tipo **classless**.

esercizio

➜ PROBLEMA

Supponiamo di avere una rete con indirizzo privato in classe C, per esempio 192.168.1.0/24 e di avere la necessità di dividerla in 3 subnet, con 100 host nella prima subnet (N1), 50 host nella seconda (N2) e 50 host nella terza (N3).

➜ ANALISI DEL PROBLEMA

Senza utilizzare la tecnica VLSM sarebbe impossibile pianificare gli indirizzi partendo dall'indirizzo di rete assegnato. Infatti per N1 servirebbero 7 bit al fine di coprire i 100 host previsti. Ma questo implicherebbe l'avere a disposizione un solo bit per il subnetting e quindi la possibilità di creare 2 sole sottoreti (subnet 0 e subnet 1) mentre la richiesta è di 3.
Dovremmo ricorrere a un indirizzo di rete in classe B con un enorme spreco di indirizzi.

Usando VLSM si può invece mantenere l'indirizzo di rete in classe C previsto e utilizzare 2 maschere di sottorete di lunghezza diversa **partendo da quella col prefisso più corto** cioè dalla subnet che prevede **più host**.

➜ SVOLGIMENTO

Iniziamo con N1 che avendo 100 host necessita di 7 bit per indirizzarli (7 bit = 128 host indirizzabili) e consente 2 sole subnet (avendo un solo bit a disposizione per il subnetting) i cui possibili indirizzi sono:

- 192.168.1.0/25 (quarto ottetto **0** 0000000)
- 192.168.1.128/25 (quarto ottetto **1** 0000000)

Di conseguenza, per queste 2 subnet, la subnet mask varrà 255.255.255.128 (quarto ottetto **1** 0000000).

Tra i 2 possibili indirizzi scegliamo, per esempio, il primo.
Dunque N1 avrà indirizzo **192.168.1.0/25** e subnet mask **255.255.255.128**.

Per N2 e N3 invece, avendo 50 host ciascuna, servono 6 bit (6 bit = 64 host indirizzabili) e, avendo 2 bit a disposizione per il subnetting, avremo 4 possibili indirizzi di rete:

- 192.168.1.0/26 (quarto ottetto 00 000000)
- 192.168.1.64/26 (quarto ottetto 01 000000)
- 192.168.1.128/26 (quarto ottetto 10 000000)
- 192.168.1.192/26 (quarto ottetto 11 000000)

Di conseguenza, per queste 4 subnet, la subnet mask varrà 255.255.255.192 (quarto ottetto **11** 000000).

Poiché **non ci possono essere nell'intera rete 2 host con lo stesso indirizzo** e poiché per N1 abbiamo scelto l'indirizzo che inizia per 0, siamo costretti a escludere per N2 e N3 i primi 2 possibili indirizzi cioè quelli che cominciano per 0.
Dunque N2 avrà indirizzo **192.168.1.128/26** e subnet mask **255.255.255.192**.
Mentre N3 avrà indirizzo **192.168.1.192/26** e subnet mask sempre **255.255.255.192**.

La rete 192.168.1.0 risulterà così divisa in 3 subnet con 2 subnet mask:

Nome	Indirizzo rete	Range indirizzi	Subnet mask	Broadcast
N1	192.168.1.0/25	192.168.1.1 - 192.168.1.126	255.255.255.128	192.168.1.127
N2	192.168.1.128/26	192.168.1.129 - 192.168.1.190	255.255.255.192	192.168.1.191
N3	192.168.1.192/26	192.168.1.193 - 192.168.1.254	255.255.255.192	192.168.1.255

In particolare **il quarto ottetto** in binario risulta essere:

Nome	Indirizzo rete	Range indirizzi	Subnet mask	Broadcast
N1	0 0000000	0 0000001 - 0 1111110	1 0000000	0 1111111
N2	10 000000	10 000001 - 10 111110	11 000000	10 111111
N3	11 000000	11 000001 - 11 111110	11 000000	11 111111

La FIGURA 10 mostra lo schema logico della rete. Se per esempio arriva un pacchetto con IP di destinazione 192.168.1.200, il router deve capire a quale delle sottoreti va indirizzato.
Il router esegue la messa in AND bit a bit per decidere verso quale subnet inoltrare il pacchetto.
Ricordiamo che la messa in AND va fatta tra IP destinatario e mask delle varie subnet: il risultato dà l'indirizzo della sottorete di appartenenza del destinatario del pacchetto.

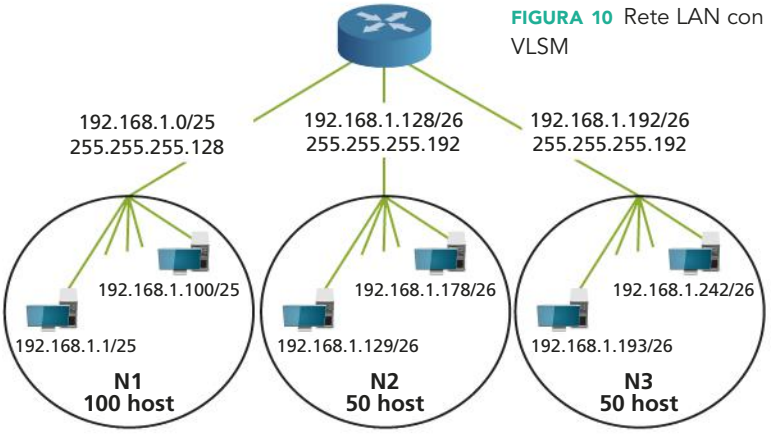


FIGURA 10 Rete LAN con VLSM

- Per N1 (192.168.1.0/25) avremo:
  192.168.1.200 AND 255.255.255.128 = 192.168.1.128 → NO: indirizzo di rete diverso da quello di N1.
- Per N2 (192.168.1.128/26) avremo:
  192.168.1.200 AND 255.255.255.192 = 192.168.1.192 → NO: indirizzo di rete diverso da quello di N2.
- Per N3 (**192.168.1.192/26**) avremo:
  192.168.1.200 AND 255.255.255.192 = **192.168.1.192** → SÌ: indirizzo di rete uguale a quello di N3.

esercizio

➜ PROBLEMA

Supponiamo di avere una rete sempre con indirizzo privato in classe C, 192.168.1.0/24, che però, a differenza dell'esercizio precedente, sia costituita invece che da una LAN con 3 subnet (cioè un solo router), da 3 LAN fisicamente separate (3 router) ma che costituiscono un'unica rete logica su un territorio più ampio (una MAN).
In questo caso occorre pianificare gli indirizzi anche per i link tra i router.
Le 3 LAN hanno rispettivamente 50 host la prima (N1), 40 la seconda (N2) e 20 la terza (N3).

➜ SVOLGIMENTO

Per indirizzare gli host delle reti N1 e N2 serviranno 6 bit, mentre ne basteranno 5 per N3. Di conseguenza, indirizzi di rete e subnet mask potrebbero essere:

- N1 → 192.168.1.0/26 (quarto ottetto **00** 000000); subnet mask: 255.255.255.192 (quarto ottetto **11** 000000);
- N2 → 192.168.1.64/26 (quarto ottetto **01** 000000); subnet mask: 255.255.255.192 (quarto ottetto **11** 000000);
- N3 → 192.168.1.128/27 (quarto ottetto **100** 00000); subnet mask: 255.255.255.224 (quarto ottetto **111** 00000).

Essendo 3 i link si dovranno assegnare 3 indirizzi di rete per identificarli. A ogni router occorrono 2 indirizzi a disposizione delle sue porte per interfacciare il resto della rete, quindi basterà prevedere sottoreti con 2 bit (gli ultimi 2) per l'indirizzamento. Occorre sempre fare molta attenzione che non ci siano porte di host o router con lo stesso indirizzo.
Una possibile soluzione potrebbe essere:

- L1 → 192.168.1.240/30 (quarto ottetto **111100** 00); subnet mask: 255.255.255.252 (quarto ottetto: **111111** 00);
- L2 → 192.168.1.248/30 (quarto ottetto **111110** 00); subnet mask: 255.255.255.252 (quarto ottetto: **111111** 00);
- L3 → 192.168.1.252/30 (quarto ottetto **111111** 00); subnet mask: 255.255.255.252 (quarto ottetto: **111111** 00).

La **FIGURA 11** mostra lo schema logico della rete.

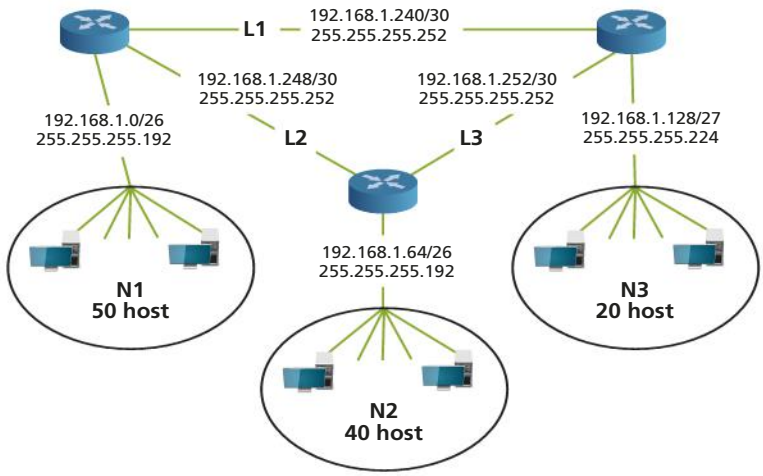


FIGURA 11 Rete MAN con VLSM

FISSA LE CONOSCENZE

- Che cosa caratterizza la tecnica di indirizzamento CIDR?
- Che cosa si intende per *supernetting*?
- Che cos'è il *summary route*?
- Che cosa rappresenta il *prefix length* nella CIDR?
- Quali sono i difetti del subnetting statico?
- Qual è la caratteristica principale della tecnica VLSM?
- La tecnica VLSM si applica a partire dalla subnet con meno host o da quella con più host?
- Ci possono essere 2 host di 2 subnet diverse, appartenenti però alla stessa rete logica, con lo stesso IP?

6 PACKET TRACER: LAVORARE CON I ROUTER

6.1 IOS: il Sistema Operativo dei router Cisco

Nell'Unità 6 del primo volume abbiamo visto come il router sia un computer a tutti gli effetti, con CPU, RAM, memoria non volatile ecc., con la particolarità di essere dedicato a svolgere funzioni di rete come routing e forwarding. Inoltre un router ha delle interfacce di rete (NIC) specializzate per interconnettere i dispositivi alle reti. Come tutti i computer, anche il router necessita di un Sistema Operativo. Spesso si tratta di un sistema proprietario, cioè specifico per gli apparati di rete di un certo produttore; esempi sono Cisco IOS per gli apparati di Cisco e Junos OS per quelli di Juniper Networks. Esistono anche sistemi operativi di rete che si basano su kernel Linux.

Il Sistema Operativo **Cisco IOS (Internetwork Operating System)** è utilizzato su tutti i router Cisco su molti switch. È un software che necessita di una licenza d'uso, gli aggiornamenti sono scaricabili dal sito Cisco. Esistono più versioni di IOS (Cisco le chiama "train") in base al tipo e alla fascia di apparati da gestire, per esempio: **IOS XR** è usato nei router di fascia alta della serie 1200 e **IOS XE** in quelli di fascia media della serie 4000 (questi ultimi li useremo nelle esercitazioni con Packet Tracer).

#prendinota

Cisco Content Hub
Informazioni e manualistica sui prodotti Cisco per il networking sono disponibili all'indirizzo: *https://content.cisco.com/welcome.html*.

ACCESSO ALLA CLI DI CISCO IOS

È possibile accedere alla CLI di IOS in due modi distinti.

- **Porta console**: è una porta speciale dei router a cui collegarsi con un qualsiasi computer dotato di porta seriale (RS232) e di un programma di comunicazione su seriale (per esempio in Linux si può usare *Minicom* e in Windows si può usare *Putty*). Per il collegamento si usa un cavo particolare, detto **rollover** di tipo seriale. La porta console è spesso usata per configurare un router nuovo, che non ha ancora un IP assegnato e non è raggiungibile tramite la rete.
- **Accesso via rete**: è possibile accedere alla CLI anche attraverso i protocolli di rete Telnet e SSH. Telnet è un'applicazione client/server che emula il terminale e permette di accedere a IOS, dopo essersi autenticati, da un qualsiasi computer remoto con un client Telnet. SSH (Secure Shell) è simile a Telnet, ma aggiunge il servizio di crittografia, rendendo così più sicura la comunicazione.

#prendinota

Negli apparati più recenti la console è una porta USB, in quanto è difficile trovare una porta seriale sugli attuali computer.

#prendinota

In realtà esiste anche un terzo modo, tramite la **porta AUX** alla quale si può collegare un modem e raggiungere così il router tramite una chiamata telefonica. È solitamente usata quando non funziona la porta console.

STARTUP E RUNNING CONFIGURATION

Gli apparati Cisco memorizzano i comandi di configurazione in due file:

- **startup configuration**: è la configurazione "stabile" dell'apparato, memorizzata in una memoria non volatile;
- **running configuration**: qui sono memorizzati i comandi eseguiti in Global Configuration Mode, il file è memorizzato nella RAM, quindi se l'apparato viene spento, queste configurazioni andranno perse. Per mantenerle è necessario memorizzarle nella startup configuration, eseguendo il comando:

```
router#copy running-config startup-config
```

LE MODALITÀ OPERATIVE DI IOS

L'accesso a IOS avviene tramite un'interfaccia a **linea di comando** (CLI, Command Line Interface), che prevede 4 livelli operativi distinti (FIGURA 12):

- **User Exec Mode**, permette di ottenere informazioni di sistema ed eseguire alcuni comandi base come ping e traceroute;
- **Privileged Exec Mode**, permette di ottenere informazioni di sistema più dettagliate, attivare/disattivare la modalità di debug, salvare/ripristinare la configurazione di sistema, ecc.;
- **Global Configuration Mode**, permette di configurare le impostazioni globali dell'apparato (hostname, data e ora, password di accesso, ecc.);
- **Specific Configuration Mode**, permette di configurare in modo specifico una interfaccia/servizio sul router (per esempio: interface mode, subinterface mode, controller mode, ecc.).

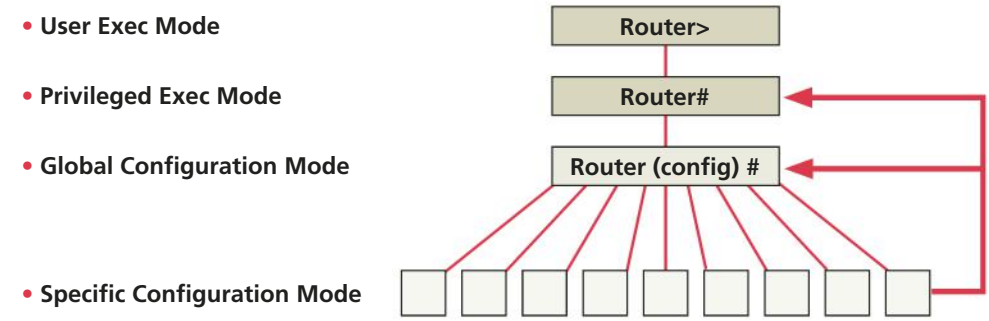


FIGURA 12 Le modalità operative di IOS

Ogni modalità operativa prevede un prompt diverso per aiutare l'utente nella scelta dei comandi da inserire. La TABELLA 2 mostra il prompt e i comandi da dare per entrare/uscire da una modalità.

TABELLA 2 Prompt e comandi per alcune modalità operative

Modalità	Prompt	Comando per entrare	Comando per uscire
User Exec	Router>	È la modalità di default dopo il boot, richiede il login con la password, se configurata.	**exit**
Privileged Exec	Router#	**>enable** da *User Exec Mode*	**exit** o **end**
Global Configuration	Router(config)#	**#configure terminal** da *Privileged Exec Mode*	**exit**
Interface Configuration	Router(config-if)#	**#interface *type number*** da *Global Configuration Mode*	**exit** per tornare in *Global Configuration Mode*
Sub-Interface Configuration	Router(config-subif)#	**#interface *type sub interface number*** da *Global Configuration Mode* o da *Interface Configuration Mode*.	– **exit** per tornare nella modalità precedente; – **end** per tornare in *Privileged Exec Mode*.
Setup	Parameter[Parameter value]:	Dopo il boot, IOS automaticamente inizia a lavorare in questa modalità se non trova una "running configuration".	**CTRL+C** per annullare. Alla fine del setup rispondere: – **Yes** per salvare la configurazione; – **No** per uscire senza salvare.
ROMMON	ROMMON>	**CTRL+C** durante i primi secondi del processo di boot (analogamente al BIOS)	**exit**

6.2 I comandi di IOS

La CLI permette di inviare comandi al router e offre alcune funzionalità utili.

- **Ottenere un elenco dei possibili comandi e delle opzioni**:
  con il comando speciale **?** è possibile ottenere tutti i comandi eseguibili in modo contestuale alla modalità operativa in cui ci si trova. Il comando **?** permette, più in generale, di ottenere tutti i possibili completamenti della linea di comando che si sta scrivendo.
- **Segnalazioni di errore**:
  viene posizionato il simbolo ' (apice) in corrispondenza del primo carattere immesso errato.
- **Abbreviazione dei comandi**:
  Per specificare ogni parola chiave è sufficiente inserire i primi caratteri che rendono univoco il comando. Per esempio il comando configure terminal può essere solitamente abbreviato con il comando conf t.
  La CLI di IOS offre anche una funzione di autocompletamento dei comandi, eseguibile premendo il tasto **TAB**.

Vediamo ora i comandi per accedere alle varie modalità operative e quali comandi possiamo dare.

Quando si accede alla CLI, si inizia in modalità **User Exec Mode**.
In questa modalità è possibile ottenere informazioni sul sistema in esecuzione grazie al comando **show**.

```
Router>show ?
  cdp           CDP information
  clock         Display the system clock
  controllers   Interface controllers status
  frame-relay   Frame-Relay information
  history       Display the session command history
  interfaces    Interface status and configuration
  ip            IP information
  version       System hardware and software
```

Dalla modalità **User Exec Mode** si passa a quella **Privileged Exec Mode** scrivendo **enable** (**disable**).
Anche in **Privileged Exec Mode** è possibile ottenere informazioni sul sistema in esecuzione grazie al comando **show**; esse saranno, però, più dettagliate:

```
Router#show ?
  access-lists    List access lists
  arp             Arp table
  cdp             CDP information
  clock           Display the system clock
  controllers     Interface controllers status
  frame-relay     Frame-Relay information
  history         Display the session command history
  interfaces      Interface status and configuration
  ip              IP information
  running-config  Current operating configuration
  startup-config  Contents of startup configuration
  version         System hardware and software status
```

Dalla modalità **Privileged Exec Mode** si passa a **Global Configuration Mode** scrivendo **configure terminal** (**end**).

In modalità **Global Configuration Mode** si possono configurare le impostazioni globali del router oppure entrare in modalità di configurazione specifica, **Specific Configuration Mode**, per le singole componenti, per esempio:

- modalità interfaccia
- modalità router
- modalità line

Per configurare un'interfaccia di rete si usa il comando **interface**:

```
Router (config) #interface tipo porta
```

oppure:

```
Router (config) #interface tipo slot/porta
```

Per attivare/disattivare un'interfaccia si usa il comando **shutdown** e la sua negazione **no shutdown**:

```
Router (config-if) #shutdown
```

oppure:

```
Router (config-if) #no shutdown
```

Per assegnare un indirizzo IP a una interfaccia si usa il comando **ip address**:

```
Router (config-if) #ip address indirizzo_ip netmask_rete
```

esempio

Configurazione di un'interfaccia di rete FastEthernet

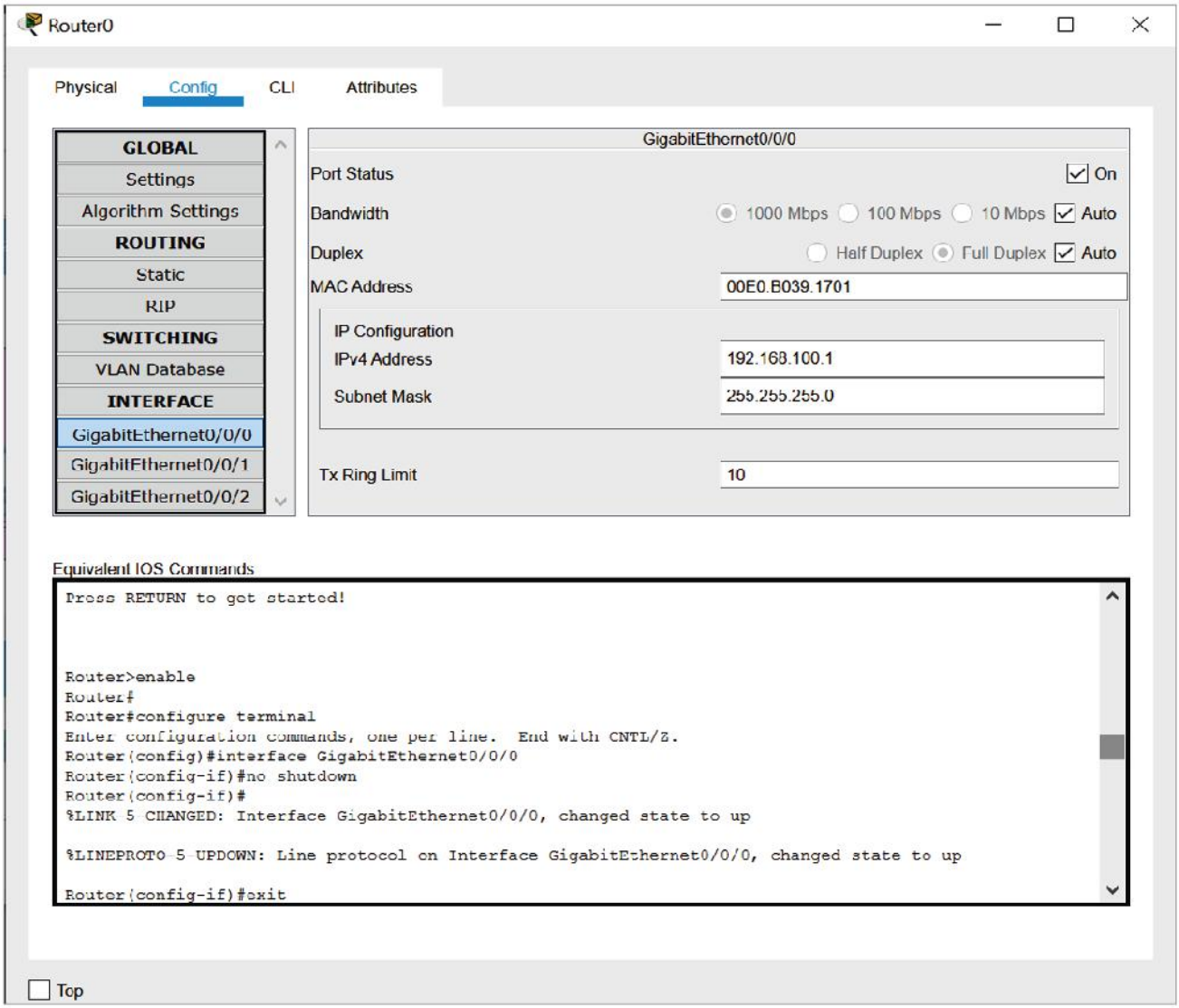
```
Router (config) #
Router (config) #interface FastEthernet 0/0
Router (config-if) #ip address 15.0.0.1   255.0.0.0
Router (config-if) #no shutdown
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Router (config-if) #end
Router#show ip interface FastEthernet 0/0
FastEthernet0/0 is up, line protocol is up (connected)
    Internet address is 15.0.0.1/8
    Broadcast address is 255.255.255.255
    ...
Router#show interface FastEthernet 0/0
FastEthernet0/0 is up, line protocol is up (connected)
    Hardware is Lance, address is 0001.420e.35e2 (bia 0001.420e.35e2)
    Internet address is 15.0.0.1/8
    ...
```

6.3 Accesso a Cisco IOS da Packet Tracer

In Packet Tracer è possibile accedere al Sistema Operativo IOS di un apparato in due modalità: tramite interfaccia grafica (FIGURA 13) e da terminale (FIGURA 14).

Nella finestra di configurazione del router, la scheda Config consente di configurare il router e le sue interfacce, senza conoscere i comandi di IOS necessari. Infatti, cliccando sulle schede del menu a sinistra, si aprono delle finestre in cui inserire i dati richiesti o semplicemente spuntare una casella.
Nella Figura 13 è visualizzata la schermata per la configurazione dell'interfaccia GigabitEthernet 0/0/0: alcuni dati sono già presenti, per esempio il MAC Address, altri devono essere impostati, per esempio l'IP address e la subnet mask.
Anche lo stato dell'interfaccia può essere impostato dall'utente, infatti, appena acceso il router, di default, l'interfaccia è spenta (down) e se deve essere usata per la connessione con un altro dispositivo è necessario spuntare la casella Port Status.
Da notare che nella scheda Config è presente la sezione "Equivalent IOS Commands" in cui ritroviamo gli stessi comandi presenti nella scheda CLI.

FIGURA 13 PT: accesso a IOS da interfaccia grafica (Config)



La Figura 14 mostra la parte finale dell'avvio del router (boot), in cui sono visualizzati alcuni dati sulla sua configurazione hardware. In basso si leggono i comandi usati per attivare l'interfaccia GigabitEthernet: **no shutdown**, può essere dato solo nella modalità specifica "Interface Configuration", a cui si giunge dopo aver dato i comandi **enable** e **configure terminal**.

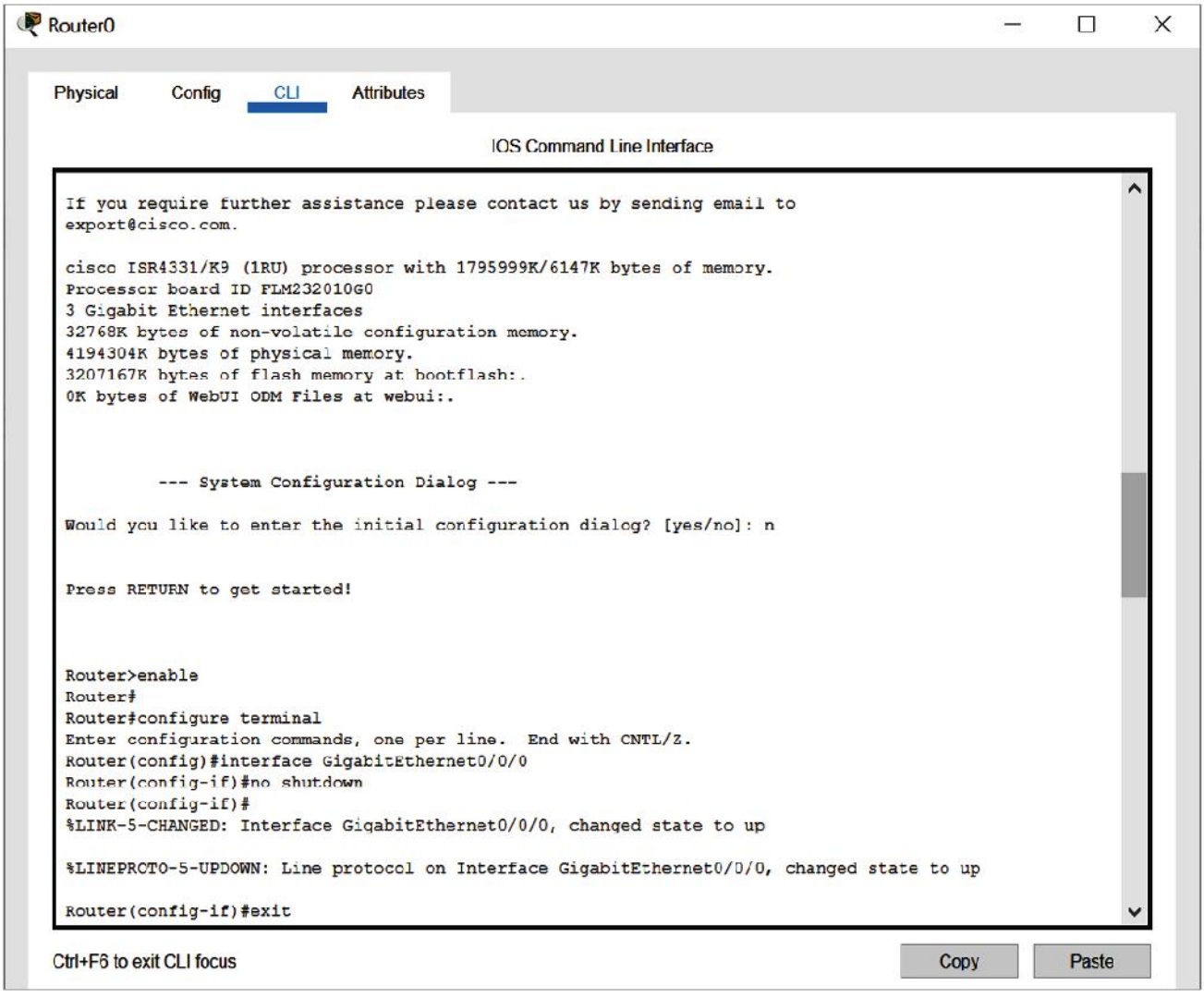


FIGURA 14 PT: accesso a IOS da terminale (CLI)

In Packet Tracer è possibile gestire i file di configurazione dall'interfaccia grafica della scheda Config del router, alla voce GLOBAL Settings **(FIGURA 15)**. Per esempio, in **NVRAM**, con un click sul pulsante **Save,** la configurazione impostata viene salvata come backup nella memoria non volatile del router (memoria NVRAM) ed equivale al comando copy running-config startup-config;

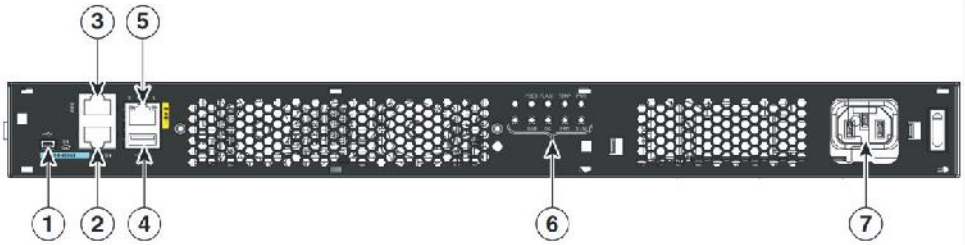
FIGURA 15 La gestione dei file di configurazione del router in Packet Tracer

6.4 La configurazione hardware del router

I router, come anche gli switch, possono avere una configurazione hardware fissa oppure modulare. Il vantaggio di avere uno **#chassis** a moduli è di poter scegliere di installare solo le **#interfacce** di interesse. Infatti, tipicamente, nel router ma anche nello switch, sono presenti alcune porte e interfacce fisse: per esempio la porta console e alcuni slot vuoti da riempire in base alle proprie esigenze, attuali e future. Non è infatti necessario inserire da subito tutte le schede negli slot vuoti.

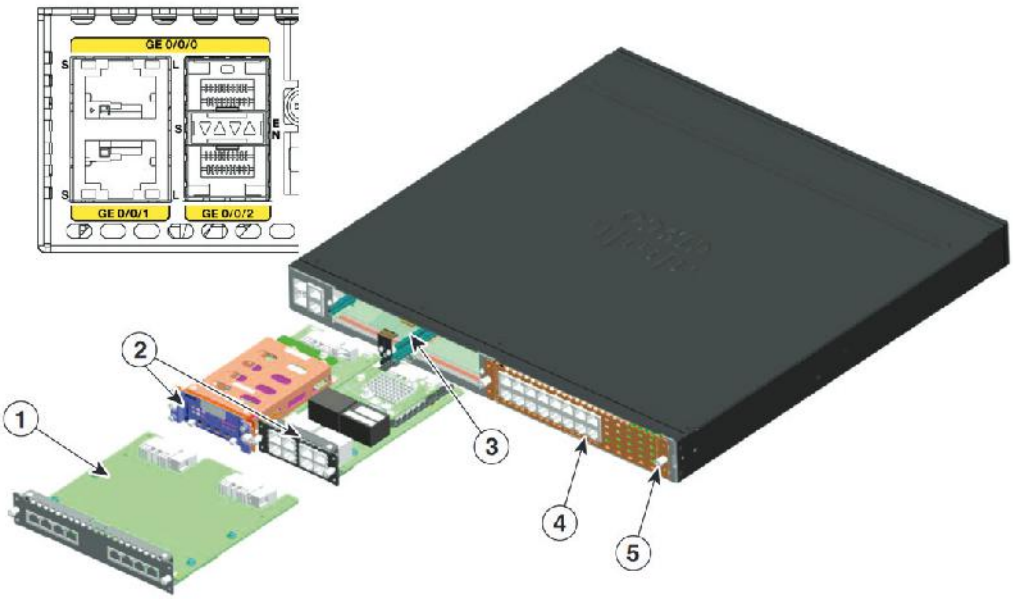
CISCO ROUTER ISR 4331

Negli esercizi di laboratorio svolti con Packet Tracer, visti nell'Unità 1 e in questa Unità, usiamo il router **#Cisco ISR 4331** (FIGURE 16, 17). Si tratta di un router di fascia media usato per connettività LAN e WAN. È modulare, infatti sono disponibili degli slot vuoti sullo chassis che possono ospitare vari moduli di interfaccia di rete: **SM-X**, Enhanced Service Module, e **NIM**, Network Interface Module.



Legenda
1) USB Type B mini port
2) Serial console port
3) AUX port
4) USB Type A port
5) Management port
6) LEDs
7) AC Power

FIGURA 16 Chassis del router ISR 4331



Legenda
1) Double-wide NIM
2) NIMs
3) Removable module slot divider
4) SM-X slot
5) Ground connection

FIGURA 17 Retro dello chassis del router ISR 4331

#techwords

Chassis
È il telaio del router o switch; tipicamente in metallo, ospita le interfacce di rete, i led indicatori di stato, gli slot vuoti.

Interfaccia
Un connettore fisico il cui scopo è quello di ricevere e inoltrare i pacchetti sulla corrispondente rete fisica. I router sono strutturati in modo tale da supportare interfacce su tecnologie di rete diverse: possono ricevere pacchetti da una rete Ethernet e inoltrarli, per esempio, verso Internet su una connessione ADSL.

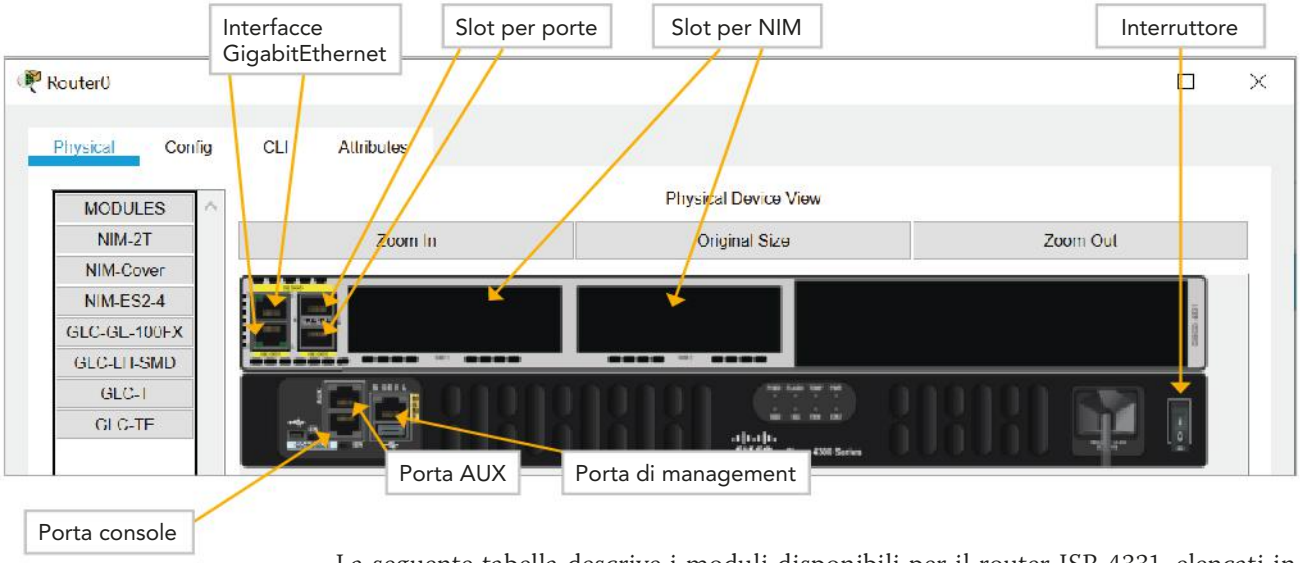
Cisco ISR
Integrated Services Router
Sono router che integrano in un unico apparato diversi servizi di rete, oltre al routing, come switching, wireless (access point), sicurezza (firewall), ecc. Sono i router delle serie 800, 1900, 2900, 3900 e 4000.

LA CONFIGURAZIONE FISICA DEL ROUTER CON PACKET TRACER

In Packet Tracer ritroviamo lo chassis del router, mostrato nelle Figure 16 e 17, riprodotto nella scheda **Physical** della finestra di configurazione (FIGURA 18). Nel menu laterale MODULES c'è l'elenco dei moduli di espansione (NIM) da inserire nei due slot vuoti e delle singole porte per cavi UTP e in fibra ottica.
L'inserimento avviene selezionando con il mouse il modulo di interesse e trascinandolo nello slot o porta libera. Al rilascio del mouse il modulo sarà inserito e farà parte della configurazione del nostro router. Se, al contrario, si vuole eliminare un modulo precedentemente inserito, sarà sufficiente selezionarlo e trascinarlo nell'elenco a sinistra con gli altri moduli.
Attenzione: i moduli possono essere installati e rimossi **solo a router spento**.

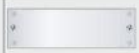
Da sottolineare il supporto delle velocità **10/100/1000** in autonegoziazione: ciò significa che è possibile collegare un PC con porta FastEthernet (100 Mbps) alla porta GigabitEthernet (1000 Mbps) del router.

FIGURA 18 La scheda Physical del router ISR 4331



La seguente tabella descrive i moduli disponibili per il router ISR 4331, elencati in MODULES nella scheda Physical.

IN ENGLISH PLEASE

Module Name	*Thumbnails*	*Description*
NIM-2T		The NIM-2T is a 2 port multi-protocol Synchronous Serial NIM.
NIM-Cover		The NIM cover plate provides protection for the internal electronic components. It also helps maintain adequate cooling by normalizing airflow.
NIM-ES2-4		The NIM-ES2-4 provides four switching ports.
GLC-GE-100FX		100BASE-FX SFP module for Gigabit Ethernet ports, 1310 nm wavelength, 2 km over MMF.
GLC-LH-SMD		The 1000BASE-LX/LH SFP operates in Gigabit Ethernet ports of Cisco Industrial Ethernet and SmartGrid switches and routers.
GLC-T		The 1000BASE-T SFP operates on standard Category 5 unshielded twisted-pair copper cabling of link lengths up to 100 m (328 ft). Cisco 1000BASE-T SFP modules support 10/100/1000 auto negotiation and Auto MDI/MDIX.
GLC-TE		The 1000BASE-T SFP operates on standard Category 5 unshielded twisted-pair copper cabling of link lengths up to 100 m (328 ft). Cisco 1000BASE-TE SFP modules support 10/100/1000 auto negotiation and Auto MDI/MDIX. Cisco 1000BASE-TE SFP modules can work in a wide range of temperature environment without any damage.

6.5 La configurazione del router con Packet Tracer

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare una rete formata da due reti locali connesse tramite un **router**. In LAN1 è presente un PC client che deve essere messo in comunicazione con un server presente nella LAN2. Il PC è connesso al router attraverso uno switch, mentre il server è collegato direttamente al router. La configurazione del router prevede anche di assegnare un **nome** e le **password di accesso**.
Verificare la connettività tra gli host delle reti locali e il default gateway.

➜ ANALISI DEL PROBLEMA

Il **router** è l'apparato di rete che opera a livello 3 del modello ISO/OSI:

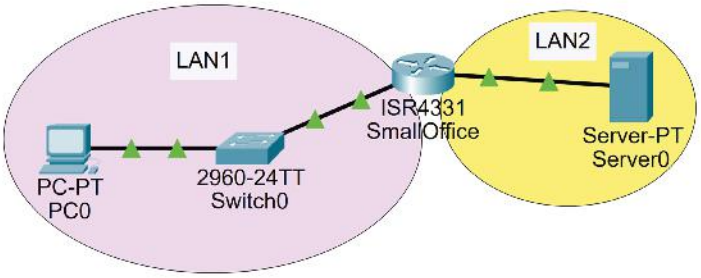
- dal punto di vista dell'indirizzamento IP, ogni singola **interfaccia** fisica del router deve appartenere a una rete logica diversa dalle altre, quindi deve avere assegnato un indirizzo IP all'interno dell'intervallo di indirizzi di host (*host range*) della rete logica su cui si affaccia;
- nel software di gestione del router, ogni interfaccia è rappresentata con un nome simbolico (per esempio *FastEthernet0/0 o GigabitEthernet0/0/0*) per permettere l'assegnazione di un indirizzo IP di host all'interfaccia stessa.

Ogni operazione di configurazione del router può essere fatta dalla scheda Config o da CLI, come visto nelle precedenti Figure 13 e 14.
Alle reti locali assegniamo indirizzi di rete privati di classe C, senza subnetting, quindi la subnet mask sarà per tutti: 255.255.255.0. Il piano di indirizzamento sarà il seguente:

	Network address	Broadcast address	Host address range
LAN1	192.168.100.0/24	192.168.100.255	192.168.100.1 ... 192.168.100.254
LAN2	192.168.200.0/24	192.168.200.255	192.168.200.1 ... 192.168.200.254

➜ SVOLGIMENTO

Nel paragrafo 7.5 dell'Unità 1 sono state fornite le indicazioni di base su come configurare i dispositivi e collegarli tra loro.
Applichiamo ora questi concetti allo scenario richiesto dall'esercizio.



Assegnazione di un nome al router

- Nella toolbar inferiore selezionare la tipologia di dispositivi Network Devices e, sotto, Routers. Tra i router che compaiono a destra selezionare **4331** e trascinarlo nel workspace;
- facendo un singolo clic sull'icona del router nel workspace, si aprirà la finestra di dialogo per la configurazione del router (Figura 13, Paragrafo 6.3), al termine del boot del router, selezionare la scheda **Config**;
- alla voce **Global Settings**, impostare nel box **hostname** il seguente nome del router: **SmallOffice**; se, invece, volessimo usare i comandi da terminale, ci spostiamo nella scheda **CLI** e scriviamo i seguenti comandi:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Router (config) #hostname SmallOffice
SmallOffice (config) #
SmallOffice#
```

- impostare nella casella **Display Name** il nome del router, SmallOffice, affinché venga visualizzato nel workspace.

Assegnazione delle password al router

Nel paragrafo 6.1 abbiamo visto le modalità di accesso al router tramite porta console e via Telnet. È possibile proteggere il router dagli accessi indesiderati, inserendo una password:

- assegnazione di una password per l'accesso da **console**: il comando **login** obbliga l'utente ad autenticarsi, mentre il comando **password** imposta la parola segreta di accesso:

```
SmallOffice#conf t
SmallOffice (config) #line console 0
SmallOffice (config-line) #password mypsw
SmallOffice (config-line) #login
SmallOffice (config-line) #exit
SmallOffice (config) #
```

- assegnazione di una password per l'accesso tramite **Telnet**:

```
SmallOffice (config) #line vty 0 4
Il settaggio fa riferimento agli accessi via telnet da 0 a 4
SmallOffice (config-line) #password mypsw
SmallOffice (config-line) #login
SmallOffice (config-line) #exit
SmallOffice#
```

Un'altra categoria di password è quella della **password di enable**, che permette di entrare nella modalità operativa **Privileged Exec Mode**. Questa password viene impostata tramite il comando **enable password**; così facendo, però, la password risulta visibile nel file di configurazione con i comandi: show running-config o show startup-config.

Quindi, è consigliato l'utilizzo di password crittografate tramite il comando **enable secret**:

```
Router #configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router (config) #enable secret mypsw
la password è così presente crittografata nel file di configurazione
```

Configurazione delle interfacce del router

- Nell'elenco a sinistra della scheda Config selezionare la voce **INTERFACE** e l'interfaccia **GigabitEthernet0/0/0** per aprire la finestra di dialogo (Figura 13) e configurare l'interfaccia corrispondente (*osservare sempre nel box in basso i corrispondenti comandi Cisco IOS*);
- spuntare **Port Status** per attivare l'**interfaccia**; infatti, le interfacce dei router Cisco sono di default spente (**down**), per renderle operative (**up**), oltre ad assegnare un indirizzo IP nell'host range della rete corrispondente, occorre fisicamente attivarle con il comando **no-shutdown** da CLI oppure in Packet Tracer selezionando il flag **On** di Port Status;
- lasciare spuntate le impostazioni automatiche per Bandwidth e Duplex;

- assegnare l'**indirizzo IP** con la corrispondente subnet mask, sulla rete LAN1:
  – indirizzo di interfaccia: 192.168.100.1
  – subnet mask: 255.255.255.0
- procedere a configurare anche la seconda interfaccia **GigabitEthernet0/0/1** su LAN2, mettendo Port Status a On e assegnando l'indirizzo IP:
  – indirizzo di interfaccia: 192.168.200.1
  – subnet mask: 255.255.255.0
- alla voce Global Settings, **NVRAM**, con un click sul pulsante **Save** la configurazione impostata viene salvata come backup nella memoria non volatile del router (memoria NVRAM); stesso risultato lo otteniamo scrivendo i seguenti comandi da CLI:

```
SmallOffice>enable
SmallOffice#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
SmallOffice#
```

Per verificare la configurazione attuale del router e delle sue interfacce, è sufficiente posizionare il mouse sull'icona nel workspace: comparirà un box con le interfacce di rete **(FIGURA 19)**.

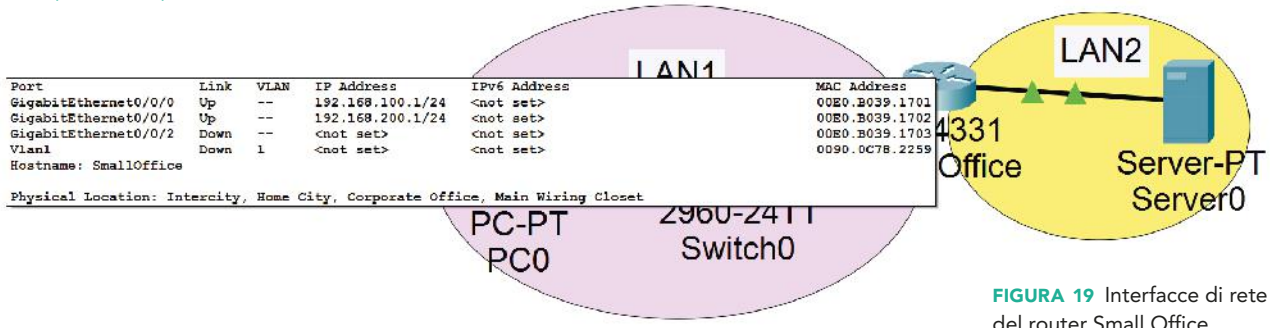


FIGURA 19 Interfacce di rete del router Small Office

Inserimento degli altri dispositivi

- Nella toolbar inferiore, tra i Network Devices, selezionare lo **switch 2960** e trascinarlo nel workspace;
- tra gli End Devices selezionare un **PC** e trascinarlo nel workspace, quindi configurarlo con le seguenti informazioni:
  – in Global Settings, Static, inserire l'indirizzo del Default Gateway: 192.168.100.1
  – in INTERFACE, selezionare FastEthernet0, Static, e inserire:
    • IPv4 Address: 192.168.100.20
    • Subnet Mask: 255.255.255.0
- tra gli End Devices selezionare un **server** e trascinarlo nel workspace, quindi configurarlo con le seguenti informazioni:
  – in Global Settings, Static, inserire l'indirizzo del Default Gateway: 192.168.200.1
  – in INTERFACE, selezionare FastEthernet0, Static, e inserire:
    • IPv4 Address: 192.168.200.20
    • Subnet Mask: 255.255.255.0

Creazione dei collegamenti

Nella toolbar inferiore selezionare Connections, quindi scegliere il cavo Copper Straight-Through e collegare le interfacce di rete configurate:

- PC FastEthernet0 con Switch FastEthernet0/1

- Switch GigabitEthernet0/1 con Router GigabitEthernet0/0/0
- Server FastEthernet0 con Router GigabitEthernet0/0/1

Test della connessione di rete

- All'interno della finestra di dialogo per la configurazione del PC, selezionare il tab **Desktop**. Con un clic sull'icona Command Prompt si apre la simulazione della finestra per la linea di comando di Windows (cmd).
  Da linea di comando, si digiti **ping 192.168.100.1**; se tutto funziona, il default gateway dovrebbe rispondere al comando ping e sul terminale dovrebbe comparire per 4 volte un messaggio simile a questo:

```
Reply from 192.168.100.1: bytes = 32 time = 1 ms TTL = 255
```

  seguito da alcuni dati di tipo statistico;

- proseguire a testare le altre connessioni:
  – da server su LAN2 a interfaccia GigabitEthernet0/0/1 sul router;
  – da PC su LAN1 a interfaccia GigabitEthernet0/0/1 sul router;
  – da server su LAN2 a interfaccia GigabitEthernet0/0/0 sul router;
  – da PC su LAN1 a server su LAN2;
  – da server su LAN2 a PC su LAN1.

Nella FIGURA 20 si mostra lo scenario realizzato e il risultato dei ping effettuati (in basso a destra), usando la busta **Add Simple PDU**, come spiegato nell'Unità 1.

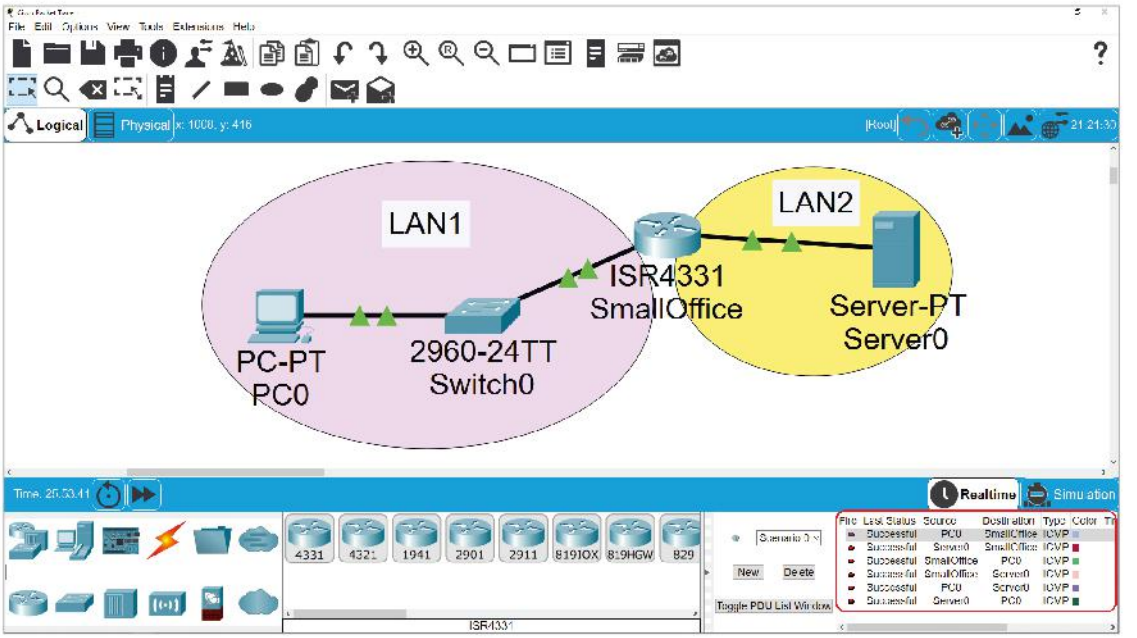


FIGURA 20
Scenario di rete e test

FISSA LE CONOSCENZE

- Descrivi le caratteristiche del Sistema Operativo di un router.
- Quali modalità operative sono previste per Cisco IOS?
- Che cosa si intende con interfaccia di un router?
- Quali password possono essere configurate sul router?
- Che cosa significa la frase: "Lo stato di un'interfaccia è *shutdown*"?

7 PACKET TRACER: IL COLLEGAMENTO TRA ROUTER

7.1 Il collegamento di due router attraverso l'interfaccia GigabitEthernet

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Si vogliono collegare due router tramite le loro interfacce **GigabitEthernet**, utilizzando quindi un cavo UTP. Al termine verificare con il comando ping la connettività tra i due router.

➜ ANALISI DEL PROBLEMA

Per svolgere l'esercizio utilizziamo nuovamente il **router 4331**, esaminato nella precedente lezione. Questo modello di router ha già installate due interfacce GigabitEthernet, pertanto non è necessario agire sulla sua configurazione fisica.
Il collegamento tra i due router sarà fatto utilizzando un cavo cross, in quanto si tratta di due apparati che operano allo stesso livello della pila OSI (Network Layer).
Per la configurazione dell'indirizzo IP usiamo l'indirizzo di rete 192.168.100.0, con subnet mask 255.255.255.0:

RouterA: Gig0/0/0 192.168.100.1 RouterB: Gig0/0/0 192.168.100.2

➜ SVOLGIMENTO

In Packet Tracer realizziamo un semplice scenario di rete, trascinando nel workspace i due router 4331 e collegandoli tramite un cavo UTP di tipo cross (Copper Cross-Over).

Apriamo su ciascun router la finestra di configurazione, cambiamo il loro nome in RouterA / RouterB e assegniamo l'indirizzo IP all'interfaccia GigabitEthernet0/0/0. Queste operazioni possono essere svolte sia in modalità grafica nella scheda Config sia da linea di comando nella scheda CLI, come visto nella Lezione precedente.
Possiamo verificare la connettività con il comando ping eseguito nella CLI oppure graficamente usando la busta Add Simple PDU. Se non sono stati commessi errori nella configurazione, il ping tra i due router darà esito positivo.

7.2 Il collegamento di due router attraverso la linea seriale

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Si vogliono collegare due router tramite linea seriale, utilizzando quindi un'interfaccia seriale presente su ciascuno dei due router. Al termine verificare con il comando ping la connettività tra i due router.

➜ ANALISI DEL PROBLEMA

Lo svolgimento di questo esercizio richiede una **linea seriale**, cioè una linea solitamente usata per i collegamenti WAN, per esempio per collegare un dispositivo alla

rete di un provider e poter così accedere a Internet. La connessione su linea seriale necessita di un segnale di temporizzazione (**clock**) che verrà fornito dall'apparato che assumerà il ruolo di DCE:

- Data Communication Equipment (DCE) è il dispositivo convertitore che fornisce la connessione all'ISP; fornisce anche il segnale di temporizzazione per la linea seriale;
- Data Terminal Equipment (DTE) è il dispositivo che riceve la connessione seriale e il segnale di temporizzazione dal DCE, tipicamente è il router del cliente.

Nelle esercitazioni di laboratorio, il collegamento WAN viene simulato utilizzando una connessione diretta fra due router, dove uno svolge il ruolo di DCE. L'interfaccia seriale del router DCE dovrà essere impostata con un valore di clock per il segnale di temporizzazione (parametro **clock rate**).

Per la configurazione dell'indirizzo IP dell'interfaccia seriale del router, questa volta usiamo un indirizzo di rete pubblico 197.85.8.0/24:

RouterA: Serial0/1/0 197.85.8.2 RouterB: Serial0/1/0 197.85.8.1

➜ SVOLGIMENTO

In Packet Tracer realizziamo lo scenario di rete analizzato, trascinando nel workspace due router 4331 e rinominandoli RouterA e RouterB. Il RouterA svolgerà il ruolo di DTE, mentre il RouterB di DCE.

Configurazione di RouterA

- Cliccare sull'icona del router nel workspace, nella finestra che compare selezionare la prima scheda **Physical**; il router 4331 non dispone, nella sua configurazione di base, di interfacce seriali, è quindi necessario selezionare un modulo che le contenga e inserirlo in uno degli slot liberi:

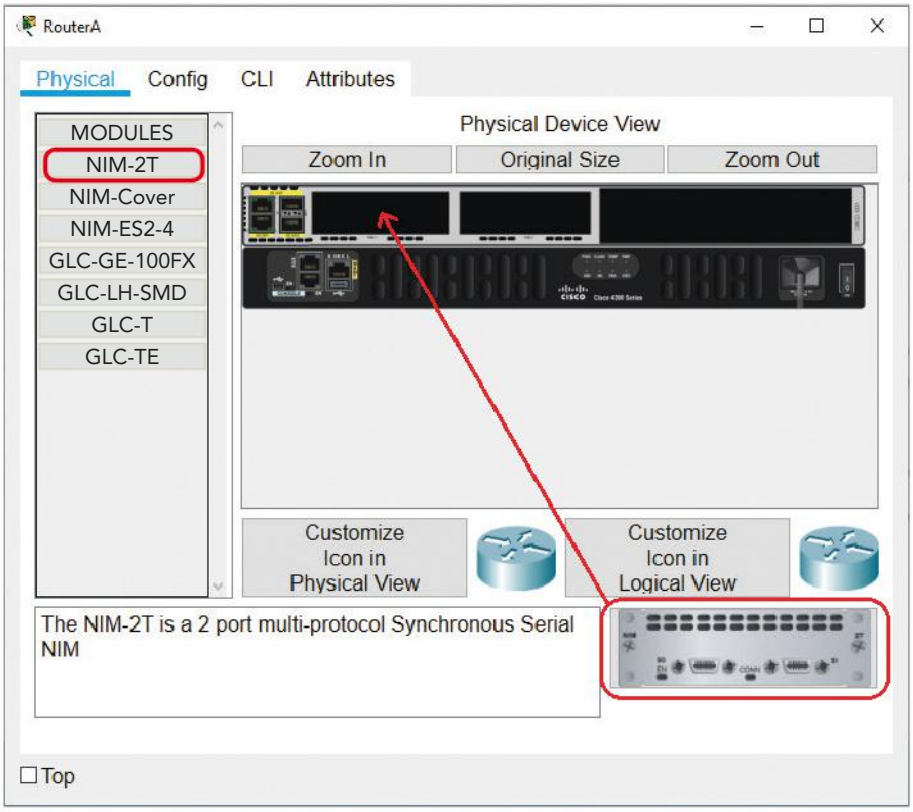


FIGURA 21 Inserimento nel router del modulo con le interfacce seriali

- prima di inserire una scheda, occorre **spegnere il router** facendo clic sul disegno dell'interruttore di accensione;
- nel menu a sinistra selezionare il modulo **NIM-2T** che dispone di due interfacce seriali (FIGURA 21); in basso a destra comparirà l'icona della scheda: trascinare (drag-and-drop) l'icona fino allo slot libero nel router;
- riavviare il router dal pulsante di accensione;

- al termine del boot, selezionare la scheda **Config**, nella sezione INTERFACE compaiono ora le due nuove interfacce seriali; selezionare **Serial0/1/0**, quindi:
  – spuntare Port Status per attivare l'interfaccia;
  – non si deve impostare il Clock Rate in quanto RouterA svolge il ruolo di DTE;
  – assegnare l'indirizzo IP 197.85.8.2 e subnet mask 255.255.255.0;
- alla voce Global Settings, NVRAM, selezionare Save per salvare la configurazione come backup nella memoria non volatile del router.

Configurazione di RouterB

- Procedere all'inserimento del modulo NIM-2T come per il RouterA;
- al termine del boot, selezionare la scheda **Config**, nella sezione INTERFACE selezionare **Serial0/1/0**, quindi:
  – spuntare Port Status per attivare l'interfaccia;
  – per il Clock Rate selezionare dal menu a tendina il valore 64000 (questa impostazione viene fatta in quanto il RouterB svolge il ruolo di DCE nel collegamento seriale);
  – assegnare l'indirizzo IP 197.85.8.1 e subnet mask 255.255.255.0;
- alla voce Global Settings, NVRAM, selezionare Save.

Collegamento con cavo seriale fra i due router

- Nella categoria Connections selezionare il cavo Serial DCE;
- con un clic su RouterB, comparirà l'elenco delle interfacce disponibili: selezionare l'interfaccia Serial0/1/0 precedentemente configurata e collegarla alla medesima interfaccia su RouterA;
- passando il mouse sopra l'estremità della linea, oltre al nome dell'interfaccia, per RouterB compare anche il simbolo dell'orologio, a indicare che svolge il ruolo di DCE e fornisce la temporizzazione a RouterA (DTE).

Test della connessione di rete

Possiamo verificare la connettività con il comando ping eseguito nella CLI di entrambi i router oppure, graficamente, usando la busta Add Simple PDU. Se non sono stati commessi errori nella configurazione, il ping tra i due router darà esito positivo.

7.3 La configurazione del router da console

esercizio

➜ PROBLEMA

Inviare comandi IOS al router usando un laptop come console.

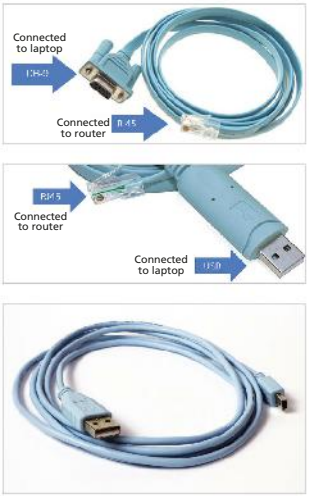
File sorgenti
Scarica il file

➜ ANALISI DEL PROBLEMA

La porta console del router è usata per accedere alla CLI del Sistema Operativo del router e svolgere così attività di configurazione, diagnostica, ecc. Anche per questo esercizio usiamo il router ISR 4331 con il sistema IOS-XE e un laptop su cui è installata un'applicazione di emulazione di terminale. Router e laptop devono essere connessi tramite un cavo "console".

La **porta console** del router è un'interfaccia seriale asincrona (EIA/TIA-232) con connettore **RJ45**, posizionato nel pannello frontale dello chassis (vedi Figura 16 della Lezione 6). I router come ISR 4331 hanno anche una porta con connettore **miniUSB**

FIGURA 22 Esempi di cavi console

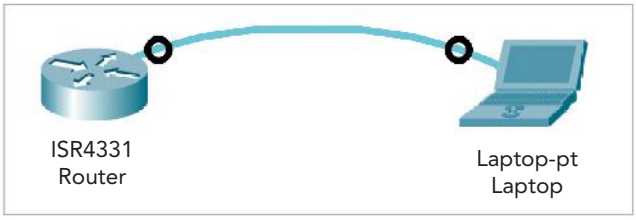


da usare come console. Questa permette agli amministratori di accedere al router in console usando un cavo USB.
A seconda delle porte a disposizione sul computer e sul router si usano cavi console con connettori differenti; alcuni esempi sono presentati nella **FIGURA 22**.

➜ SVOLGIMENTO

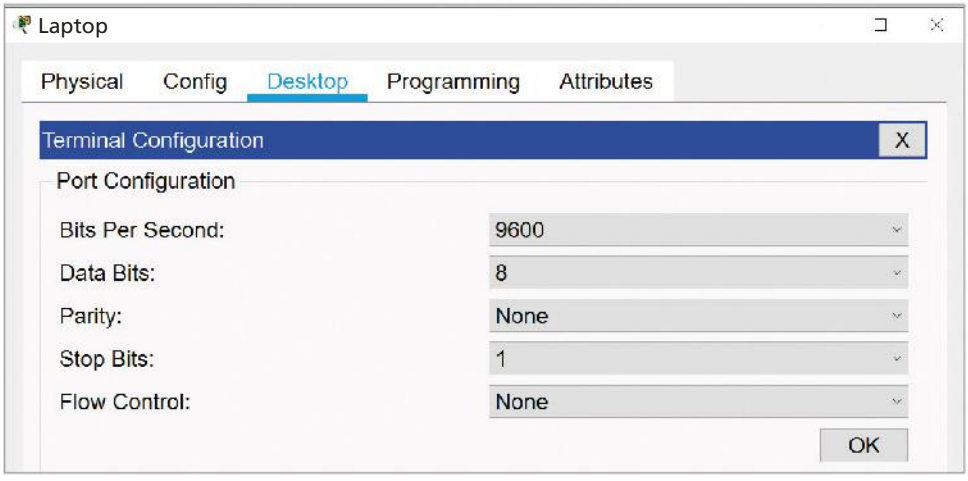
Accesso alla CLI usando una connessione diretta alla console

1) In questa modalità di connessione si usa un cavo console (selezionare il cavo azzurro Console nell'elenco delle connessioni) e si collega sul router alla **porta console** e sul pc laptop alla **porta RS 232** (è la porta seriale).
   In questo esercizio usiamo un PC portatile in quanto è quello tipicamente usato dal tecnico che deve intervenire "sul posto" per connettersi al router.



2) Nella connessione con console è necessario usare sul PC l'applicazione che emula il terminale del router:
   – clic sull'icona del laptop e nella scheda Desktop selezionare **Terminal**;
   – prima di visualizzare la finestra del terminale, viene chiesto di configurare la comunicazione seriale impostando i parametri della porta; lasciamo quelli di default, come mostrato nella **FIGURA 23**.

FIGURA 23 I parametri di configurazione della porta seriale



3) Una volta dato l'OK alla configurazione della porta seriale, viene visualizzata la finestra del terminale che riporta i messaggi inviati dal router nella fase di boot, come mostrato nella **FIGURA 24**, dove si possono leggere informazioni di sistema come la versione 16.7 di IOS del router, ecc.

Rispondere "no" alla richiesta "Would you like to enter the initial configuration dialog?" e dare Return per ottenere il prompt e iniziare così a scrivere i comandi IOS. Nella **FIGURA 25** si mostra la sequenza di comandi per accendere (up) l'interfaccia GigabitEthernet0/0/0 (comando no shutdown).

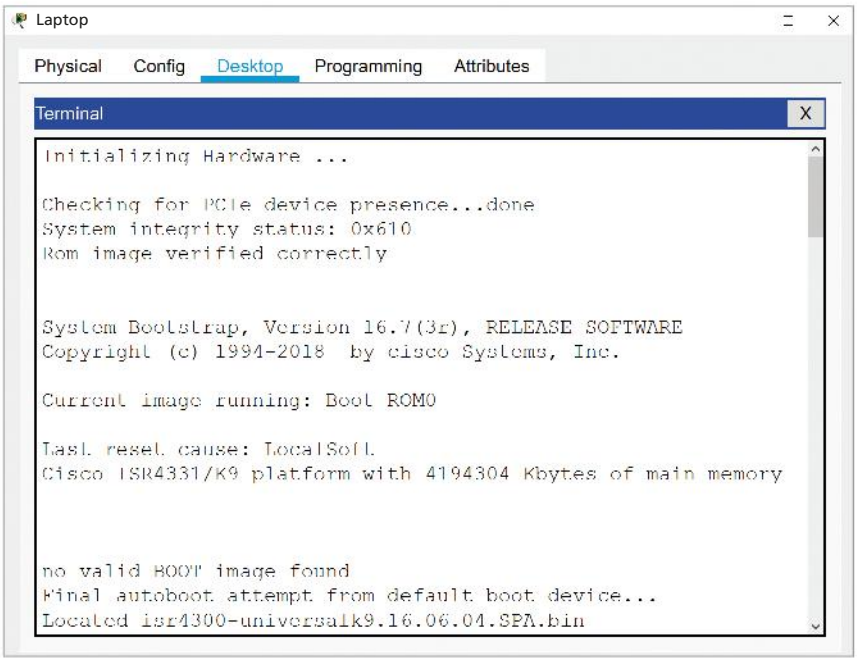


FIGURA 24 I messaggi di boot del router nel terminale

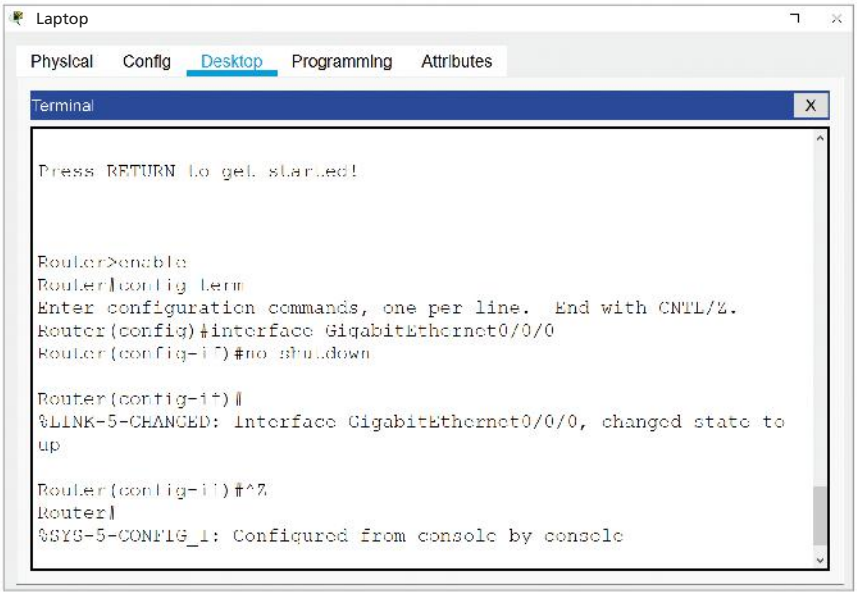


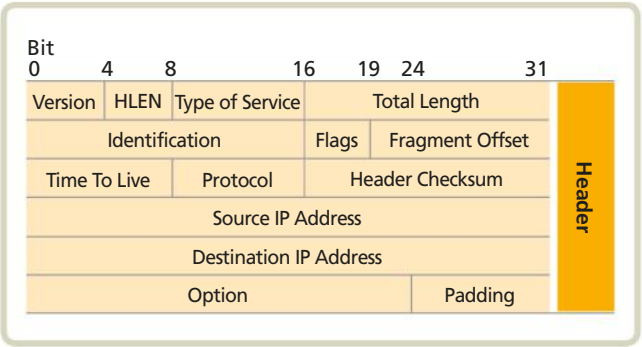
FIGURA 25 L'accensione dell'interfaccia GigabitEthernet0/0/0 del router da terminale

FISSA LE CONOSCENZE

- Come si possono connettere due router tra loro?
- Che cosa sono le linee seriali?
- Come si deve operare sul router se non dispone, di default, di un'interfaccia seriale?
- Perché nel collegamento seriale, Packet Tracer mostra il simbolo di un orologio vicino a un'estremità della connessione?
- Perché quando si seleziona la scheda CLI compare il prompt?
- Che utilità ha la porta console del router?
- Quali tipi di cavi console possiamo usare per connettere un PC a un router?

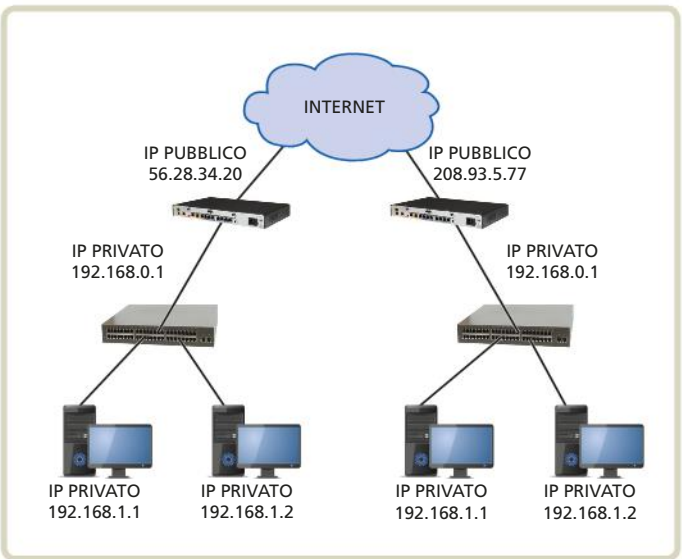
1 Il livello Network e il protocollo IP

Il livello Network ha i seguenti compiti fondamentali: **forwarding**, che consiste nell'instradare i messaggi su una rete utilizzando un indirizzamento univoco, e **routing**, che determina il percorso che segue un pacchetto in rete, localizzando anche eventuali instradamenti alternativi in caso di guasti. Queste due funzioni sono la peculiarità dei due sottolivelli Data Plane, il forwarding, e Control Plane, il routing. Il principale protocollo del livello Network nelle reti TCP/IP è **Internet Protocol (IP)**, usato per trasferire i dati nella rete WAN. Il protocollo IP è connectionless, dunque consente a due host di scambiarsi pacchetti (IP datagram) senza stabilire una sessione. La consegna non è garantita a questo livello, ma se ne occupa il protocollo TCP a livello Transport. L'header del protocollo IP, oltre ai campi per gli indirizzi IP del mittente e del destinatario del pacchetto, contiene alcune informazioni utili per la gestione del pacchetto nella rete: i campi per la frammentazione, il contatore degli hop (TTL), la checksum e alcuni campi opzionali.

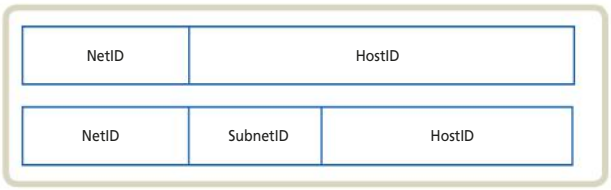


2 La struttura degli indirizzi IP

Il Network Layer è il primo strato dello stack TCP/IP in grado di garantire una connettività a livello WAN; deve quindi poter identificare univocamente ogni host della rete mediante l'indirizzo logico. Nel protocollo IP questo indirizzo prende il nome di IP address; in particolare, nella versione 4 esso è un numero di 32 bit suddivisi in 4 byte (anche detti ottetti). L'**IP address v4** è espresso nella notazione decimale puntata costituita da 4 numeri decimali compresi tra 0 e 255 separati da un punto. Gli indirizzi IP sono suddivisi in 5 **classi**: A, B, C, D ed E, ma solo le prime 3 (A, B, C) possono essere utilizzate per assegnare indirizzi agli host. Oltre agli indirizzi pubblici da usare su Internet, nelle classi A, B e C sono stati riservati blocchi di indirizzi privati da usare nella LAN privata.



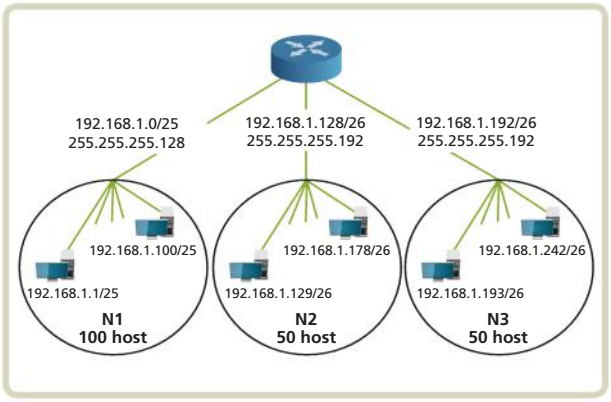
3 Pianificazione di reti IP: il subnetting



Per ottimizzare il traffico in una rete risulta particolarmente utile suddividerla in una serie di sottoreti logiche, collegate tra loro da router interni alla rete stessa. Questa operazione di segmentazione della rete in sottoreti prende il nome di **subnetting** ed è realizzata "sacrificando" alcuni dei bit che le classi A, B e C dedicano agli host per definire un indirizzo di sottorete. Il meccanismo che permette di verificare se due host appartengono alla stessa sottorete, e quindi alla stessa rete, è detto **processo di messa in AND**.

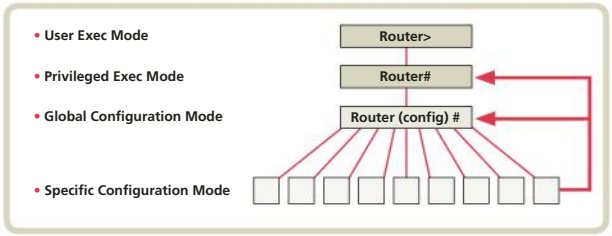
5 Pianificazione di reti IP: CIDR e VLSM

Per cercare di porre rimedio a sprechi e carenze, in attesa dell'IPv6, nel 1993 è stato introdotto un nuovo schema di indirizzamento, la tecnologia **CIDR** (pronunciata "*saider*"), anche nota come supernetting perché crea una super rete composta da più reti. In pratica la CIDR non applica subnetting ed elimina il concetto di classe di indirizzi. In questo caso, il piano di indirizzamento è definito **classless** e la subnet mask prende il nome di **netmask**.



In alternativa alla CIDR è possibile utilizzare le **VLSM**: maschere a lunghezza variabile. Questa tecnica permette di utilizzare in modo più efficiente lo spazio di indirizzi: un provider può usare una netmask lunga sulle subnet con pochi host e una netmask breve sulle subnet con molti host. Anche in questo caso l'indirizzamento è classless.

6 Packet Tracer: lavorare con i router



I router, come i computer, hanno un proprio Sistema Operativo. Quello dei router Cisco è denominato **IOS** (**Internetwork Operating System**) ed è usato anche su alcuni modelli di switch. L'accesso a IOS avviene tramite un'interfaccia a linea di comando (CLI), usando la porta console o attraverso la rete, con protocolli come Telnet e SSH. L'accesso a IOS prevede 4 livelli operativi distinti, man mano più specifici: User Exec Mode, Privileged Exec Mode, Global Configuration Mode e Specific Configuration Mode. Tra le modalità Specific una delle più importanti e usate è la configurazione delle interfacce (*config-if*).

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Il protocollo IP è connection-oriented. V F
2. La lunghezza minima dell'header IP è 20 byte. V F
3. Il campo HLEN dell'header IP indica la lunghezza dell'header in byte. V F
4. Per la gestione della qualità del servizio (QoS) si usa il campo TOS dell'header. V F
5. Gli indirizzi IP si scrivono nella notazione decimale puntata. V F
6. Gli indirizzi IPv4 sono costituiti da 16 byte (128 bit). V F
7. Le classi di indirizzi IP sono 5 ma solo 3 utilizzabili per il subnetting. V F
8. Per il subnetting si utilizzano bit dedicati alla net. V F
9. La tecnica CIDR è di tipo classless. V F
10. La tecnica del supernetting si usa soprattutto con gli indirizzi di classe A. V F
11. Con la tecnica VLSM tutte le subnetmask della rete sono uguali. V F
12. La tecnica VLSM è usata per aggregare gli indirizzi IP. V F
13. Con il termine IOS si identifica il Sistema Operativo dei router Cisco. V F
14. La porta console è usata per collegarsi da remoto, via rete, al router. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **La lunghezza minima dell'header IP è:**
   - A 16 byte
   - B 20 byte
   - C 24 byte
   - D 32 byte

2. **Nel caso in cui la lunghezza del campo Options dell'header IP non risulti multipla di 32 bit:**
   - A si scarta il pacchetto
   - B si segnala l'errore al mittente
   - C si modifica la checksum
   - D si aggiunge un padding

3. **Se il protocollo IP riceve dal protocollo di livello Transport una quantità di dati superiore a quella che può essere trasferita nella rete:**
   - A divide il messaggio ricevuto in più pacchetti
   - B segnala al protocollo di livello Transport l'impossibilità a trasmettere i dati nella rete
   - C trasmette ugualmente il messaggio in rete aggiungendo una opportuna informazione nell'header

4. **Quale tra i seguenti indirizzi IP è un indirizzo di rete (classful)?**
   - A 10.10.0.0
   - B 150.1.25.0
   - C 170.10.0.0
   - D 200.1.35.10

5. **Data la subnet mask 255.255.240.0 a quale/i classe/i può riferirsi?**
   - A Solo classe A
   - B Classi A o B
   - C Solo classe B
   - D Classi B o C

6. **Quale tra questi indirizzi IP in classe C rappresenta il 5° host della subnet 3?**
   - A 200.10.20.61/28
   - B 200.10.20.41/28
   - C 200.10.20.103/28
   - D 200.10.20.53/28

7. **Data una rete in classe B con 21 bit di prefix (/21), quante subnet posso realizzare e con quanti host al massimo per ciascuna?**
   - A 62 subnet con 1.022 host ciascuna
   - B 14 subnet con 4.094 host ciascuna
   - C 32 subnet con 2.046 host ciascuna
   - D 126 subnet con 510 host ciascuna

8. **Se un indirizzo IP ha 12 bit di prefix, quale subnet mask occorre usare?**
   - A 255.224.0.0
   - B 255.255.240.0
   - C 255.255.224.0
   - D 255.240.0.0

9. **Dato l'indirizzo IP di rete 198.128.10.0 con subnet mask 255.255.255.240, qual è l'indirizzo del 3° host della subnet 6?**

A 198.128.10.99
B 198.128.10.86
C 198.128.10.42
D 198.128.10.68

10. **Dato l'indirizzo IP 50.121.132.52 con subnet mask 255.255.192.0, a quale subnet appartiene?**

A 396
B 412
C 482
D 486

11. **I due host appartengono alla stessa sottorete? (applica la messa in AND)**

- HOST1 indirizzo IP 120.249.101.1 subnet mask 255.255.224.0
- HOST2 indirizzo IP 120.253.101.1 subnet mask 255.255.224.0

A Sì
B No

12. **Quando l'indirizzamento IP non segue la suddivisione in classi (A, B, C, ...), si usa il termine:**

A classless
B classful
C multi-class
D classany

13. **Con la tecnica del supernetting (o CIDR) gli indirizzi IP vengono:**

A aggregati
B suddivisi
C utilizzati a gruppi di 6
D utilizzati a gruppi di 5

14. **Le tecniche CIDR e VLSM sono state pensate per:**

A migliorare le prestazioni delle reti IP
B evitare lo spreco di indirizzi IP
C aumentare la velocità di trasmissione dei dati
D semplificare l'instradamento dei pacchetti

15. **Qual è il simbolo usato come prompt nel User Exec Mode di IOS?**

A >
B $
C #
D -

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Quali sono le funzioni svolte dal Network Layer del TCP/IP?**

2. LEZIONE 1 **Descrivi il formato della PDU del protocollo IP.**

3. LEZIONE 2 **Descrivi la struttura degli indirizzi IP.**

4. LEZIONE 2 **Chi assegna gli indirizzi IP a livello mondiale?**

5. LEZIONE 3 **Spiega a che cosa serve il subnetting.**

6. LEZIONE 3 **Spiega qual è il ruolo della subnet mask e come si definisce.**

7. LEZIONE 3 **Descrivi il meccanismo utilizzato dal router per verificare se l'host mittente e l'host destinatario appartengono alla stessa rete (o sottorete).**

8. LEZIONE 4 **Perché nelle reti locali spesso l'amministratore di rete utilizza un indirizzo di classe B per il subnetting?**

9. LEZIONE 5 **Prova a spiegare in che cosa consiste la tecnica CIDR.**

10. LEZIONE 5 **Prova a spiegare in che cosa consiste la tecnica VLSM.**

IN ENGLISH, PLEASE

CLIL

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

The Network Layer of TCP/IP

The Network Layer provides an host-to-host communication service to the Transport Layer. There is a piece of the Network Layer in each and every host and router in the network, specifically in their NICs. The two most important funcions of the Network Layer are: forwarding and routing. Forwarding involves the transfer of a packet from an incoming link to an outgoing link within a single router. Routing involves all of a network's routers, whose collective interactions via routing protocols determine the paths that packets take on their trips from source to destination node.
The Internet has many protocols related to the Network Layer. The Internet Protocol (IP) is a datagram protocol that provides a connectionless service. When a node wants to send a packet (IP datagram), it stamps the packet with the destination address. It is important to note that an IP address refers to a network interface, for example a router has many interfaces and thus many IP addresses.
IP uses unique addresses; in IPv4 they are 32-bit long and expressed in dot-decimal notation. An IP address consists of two parts: a NetID and a HostID. IPv4 uses the subnetting technique, that is characterized by a subnet mask. This can be ANDed with the IP address to extract only the network portion (NetID), so a router can forward packets based on only the network portion of the address.
The traditional class division are substitute by the classless technique with CIDR; VLSM uses a classless addressing, too.

EXERCISES

Use the appropriate number to match words and meanings.

...	Private IP addresses	1	It is applied by a bitwise AND operation
...	IP prefix	2	Making changes to device parameters' value
...	Serial port	3	It is used to connect the IOS when you are in the same location with the router
...	Subnet mask	4	Router line for incoming/outgoing packets
...	Connectivity	5	It is not reachable from the Internet
...	Configuration	6	An interface through which information transfers one bit at a time
...	Console port	7	It uses the slash character
...	Interface	8	It is required to transfer data in a network

GLOSSARY

Datagram: the unit of data exchanged between a pair of IP entity (includes the header).
Default Gateway: a host in the local area network, usually a router, that forwards IP packets when no other route is known for the destination address.
Dot-decimal notation: a string of decimal numbers, using the full stop (dot) as a separation character.
Interface: router line for incoming/outgoing packets, it provides network connectivity to the router.
Internet Protocol: the fundamental protocol that is used to transmit information over the Internet.
Loopback: a communication channel with only one endpoint: any message transmitted through such a channel is received by the sender.
Subnetting: the operation of splitting a network into smaller networks.
Regional Internet Registry (RIR): a registry responsible for allocation of IP address resources within a particular region.

LAVORARE PER COMPETENZE

COMPETENZE IN GIOCO

Competenze disciplinari

- Realizzare scenari di rete di tipo LAN/WAN usando un simulatore di rete e verificandone la correttezza tramite test.
- Saper operare sui router usando la CLI per inviare comandi di configurazione e gestione sia del router sia delle sue interfacce.
- Saper lavorare sul router tramite console.
- Utilizzare la terminologia tecnica anche in lingua inglese.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Comunicare.
- Risolvere problemi.
- Competenza digitale.

obiettivi formativi

- Imparare a usare sia la modalità grafica sia quella da terminale (CLI) per configurare e gestire gli apparati di rete.
- Esporre i risultati del proprio lavoro alla classe.

tempi

- Personale risoluzione del tema proposto: 2 ore.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Applicazione Cisco Packet Tracer.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Progettare una semplice rete aziendale e il piano di indirizzamento IP.

Un'azienda deve realizzare la rete di comunicazione per la sua nuova sede. Il progetto di rete prevede di sviluppare 3 reti locali distinte, una per ogni piano dell'edificio, collegate a un router centrale che svolge la funzione di internetworking fra le LAN e di gateway verso Internet. L'amministratore di rete ha a disposizione un indirizzo IP pubblico, 199.205.50.3, mentre nelle reti locali si useranno indirizzi privati.

Realizzare il piano di indirizzamento IP della rete aziendale, tenendo conto anche di possibili espansioni future.

Sviluppare il progetto di rete con un simulatore, verificando che ogni dispositivo possa comunicare con gli altri presenti nella rete e con il gateway.

File sorgenti
Scarica il file

SVOLGIMENTO

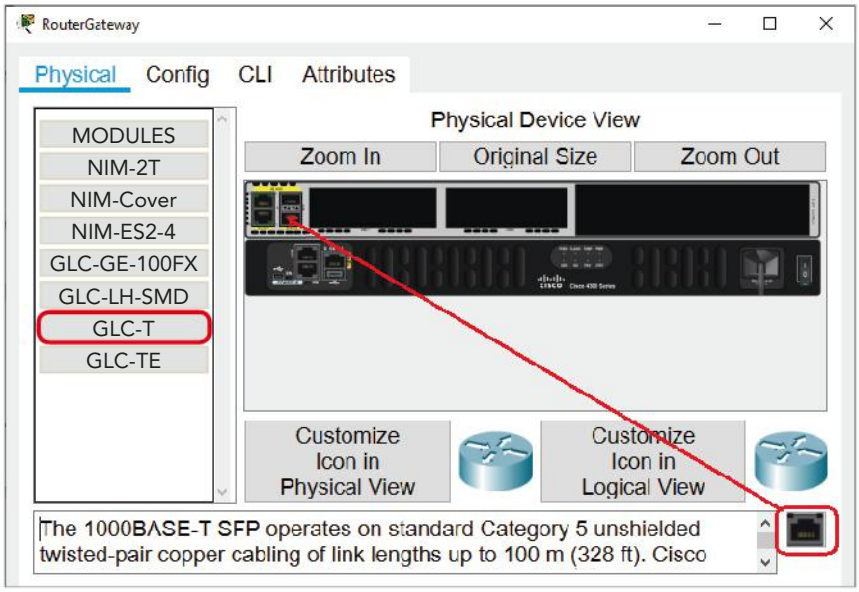
Prima di realizzare lo scenario di rete aziendale con il simulatore Packet Tracer, è opportuno definire il piano di indirizzamento IP della rete. L'azienda ha acquistato un solo indirizzo pubblico 199.205.50.3 che sarà assegnato a router gateway per accedere a Internet. Le reti locali avranno quindi un piano di indirizzamento privato.

Per soddisfare l'esigenza di scalabilità espressa dall'azienda e far fronte a espansioni future senza dover rifare l'intero piano di indirizzamento, l'amministratore di rete sceglie di utilizzare l'indirizzo IP privato, di classe B, **172.16.0.0** con subnet mask **255.255.255.0**, realizzando la seguente pianificazione:

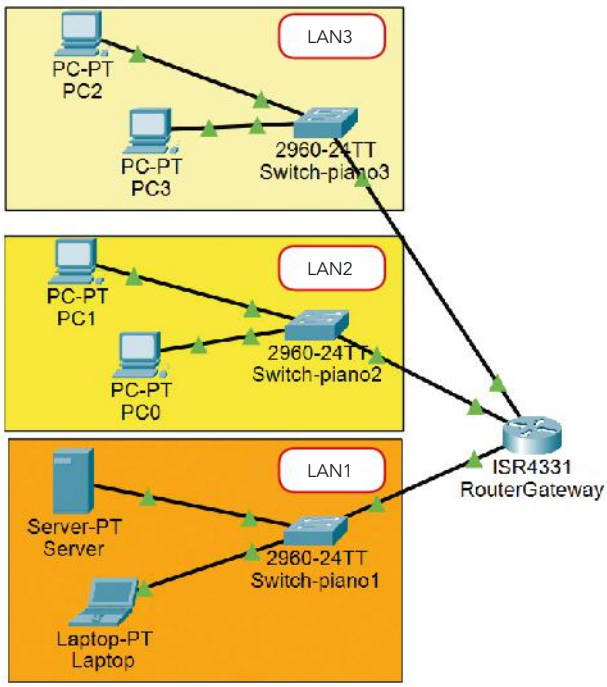
	Ind. di rete	Subnet mask	Range indirizzi di host	Ind. di broadcast
Piano 1 (LAN1)	172.16.1.0	255.255.255.0	Da 172.16.1.1 a 172.16.1.254	172.16.1.255
Piano 2 (LAN2)	172.16.2.0	255.255.255.0	Da 172.16.2.1 a 172.16.2.254	172.16.2.255
Piano 3 (LAN3)	172.16.3.0	255.255.255.0	Da 172.16.3.1 a 172.16.3.254	172.16.3.255

Creiamo ora lo scenario con Packet Tracer: inseriamo due computer connessi a uno switch, per ciascuna LAN. Gli switch saranno collegati a un'interfaccia del router, scegliendo quelle a più alta velocità GigabitEthernet.

Scegliamo di usare nuovamente il router Cisco ISR 4331, dovendo questa volta disporre di 3 interfacce GigabitEthernet. Solo 2 sono presenti di default nello chassis, dobbiamo quindi inserire una nuova interfaccia GigabitEthernet nella porta di espansione libera (come visto nella Lezione 6 per il modulo NIM).



Una volta effettuata questa configurazione hardware, la salviamo nella NVRAM (scheda Config del router, Global Settings) e passiamo ad accenderne una per una le 3 interfacce GigabitEthernet del router (scheda Config, Interface, flag Port Status).
A questo punto possiamo creare i collegamenti tra i vari dispositivi presenti nel workspace usando i cavi Copper Straight-Through: la colorazione verde dei triangolini presenti alle estremità dei cavi segnala il corretto funzionamento della connessione fisica.



I collegamenti dei PC, server, laptop con i rispettivi switch sono fatti usando le interfacce FastEthernet, mentre quelli del router con gli switch seguono il seguente schema:

Interfaccia Router Gateway	Indirizzo IP	Switch a cui è connesso	Interfaccia dello switch
GigabitEthernet0/0/0	172.16.1.254	Switch-piano1	GigabitEthernet0/1
GigabitEthernet0/0/1	172.16.2.254	Switch-piano2	GigabitEthernet0/1
GigabitEthernet0/0/2	172.16.3.254	Switch-piano3	GigabitEthernet0/1

Come indicato in precedenza, la subnet mask è uguale per tutti gli indirizzi: 255.255.255.0.
Gli indirizzi IP assegnati alle interfacce dei dispositivi sono i seguenti:

Laptop	Server	PC0	PC1	PC2	PC3
172.16.1.10	172.16.1.1	172.16.2.10	172.16.2.11	172.16.3.10	172.16.3.11

Effettuare un ping per verificare la raggiungibilità dei vari dispositivi. Si noterà che i PC raggiungono il router, ma non i PC o server nelle altre LAN. Questo succede perché non è stato configurato il Default Gateway. In ogni computer, nella scheda Config-Global Settings, procedere a inserire come Default Gateway l'indirizzo dell'interfaccia del router a cui sono collegati. Ripetere il ping, ora dovrebbe dare esito positivo.

A CASA

- Ipotizza una tua soluzione al tema proposto, utilizzando un altro piano di indirizzamento IP, con subnet mask 255.255.240.0. Leggi lo svolgimento per verificare se le tue ipotesi si adattano al caso preso in esame.
- Supponi che l'azienda acquisti un nuovo locale in un edificio adiacente, in cui collocare alcuni PC che saranno connessi a uno switch che a sua volta sarà collegato a un router. Questo nuovo router dovrà essere collegato al Router Gateway della prima sede tramite un cavo seriale.
  - Realizza con Packet Tracer la nuova rete LAN e il collegamento seriale tra i due router, come visto nella Lezione 6.
  - Esegui i test di raggiungibilità degli apparati della nuova LAN con il nuovo router e tra di loro: se la configurazione IP è corretta daranno esito positivo.
  - Verifica la raggiungibilità del RouterGateway della prima sede da parte di un PC che si trova nel nuovo locale; qual è il suo esito? Prova a ipotizzarne il motivo. (Nella prossima Unità vedremo come risolvere il problema).
- Raccogli i tuoi risultati in una presentazione (massimo 3 slide per il punto 1 e 6 slide per il punto 2).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i piani di indirizzamento presentati (punto 1) e come sono stati realizzati i progetti della nuova LAN (punto 2). Trovate una spiegazione ai differenti risultati del comando ping.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste, aiutato dal docente. ☐	Aiutato da docente e compagni ho compreso le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho creato il nuovo piano di indirizzamento con subnetting senza difficoltà?	Ho avuto difficoltà nel capire la subnet mask e come usarla. ☐	Aiutato da docente e compagni ho creato il nuovo piano di indirizzamento. ☐	Ho creato il nuovo piano di indirizzamento autonomamente. ☐	Ho creato il piano di indirizzamento senza difficoltà e ragionato sul numero di subnet e di host. ☐
Ho creato la nuova rete locale e collegato i due router con la linea seriale, raccogliendo i risultati dei vari test svolti?	Ho avuto difficoltà nel selezionare i dispositivi e i cavi corretti. Non ho saputo svolgere i test. ☐	Aiutato da docente e compagni ho realizzato la nuova LAN, definito gli indirizzi e svolto in parte i test. ☐	Ho creato la nuova LAN e ho connesso i due router, verificandone il funzionamento in modo autonomo. ☐	Ho realizzato la nuova LAN, connesso i due router, svolto i test e raccolto i risultati ottenuti. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione con poche informazioni. ☐	Ho preparato una presentazione non chiara e non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione strutturata, ma non mi sono spiegato bene. ☐	Ho preparato una presentazione ben strutturata, tutti mi hanno capito. ☐

UNITÀ

4 L'EVOLUZIONE DI IP E IL MONITORING DELLA RETE

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

DDI+ Didattica Digitale Integrata Plus

conoscenze	abilità	competenze
Conoscere le nuove funzionalità di IPv6. Conoscere la struttura degli indirizzi IPv6. Conoscere i vari tipi di messaggi ICMP. Conoscere il funzionamento del protocollo ARP.	Essere in grado di lavorare con gli indirizzi IPv6 e verificare la configurazione dell'interfaccia di un dispositivo da CLI. Saper definire un indirizzo IPv6 locale (Link local) a partire dal MAC address. Saper usare i comandi ping e traceroute.	Realizzare il piano d'indirizzamento di una LAN. Testare la raggiungibilità di un host con il comando ping. Verificare con il comando traceroute il percorso seguito da un pacchetto per arrivare a destinazione. Riprodurre il funzionamento di una rete IPv6 tramite la simulazione.

FLIPPED CLASSROOM

A casa

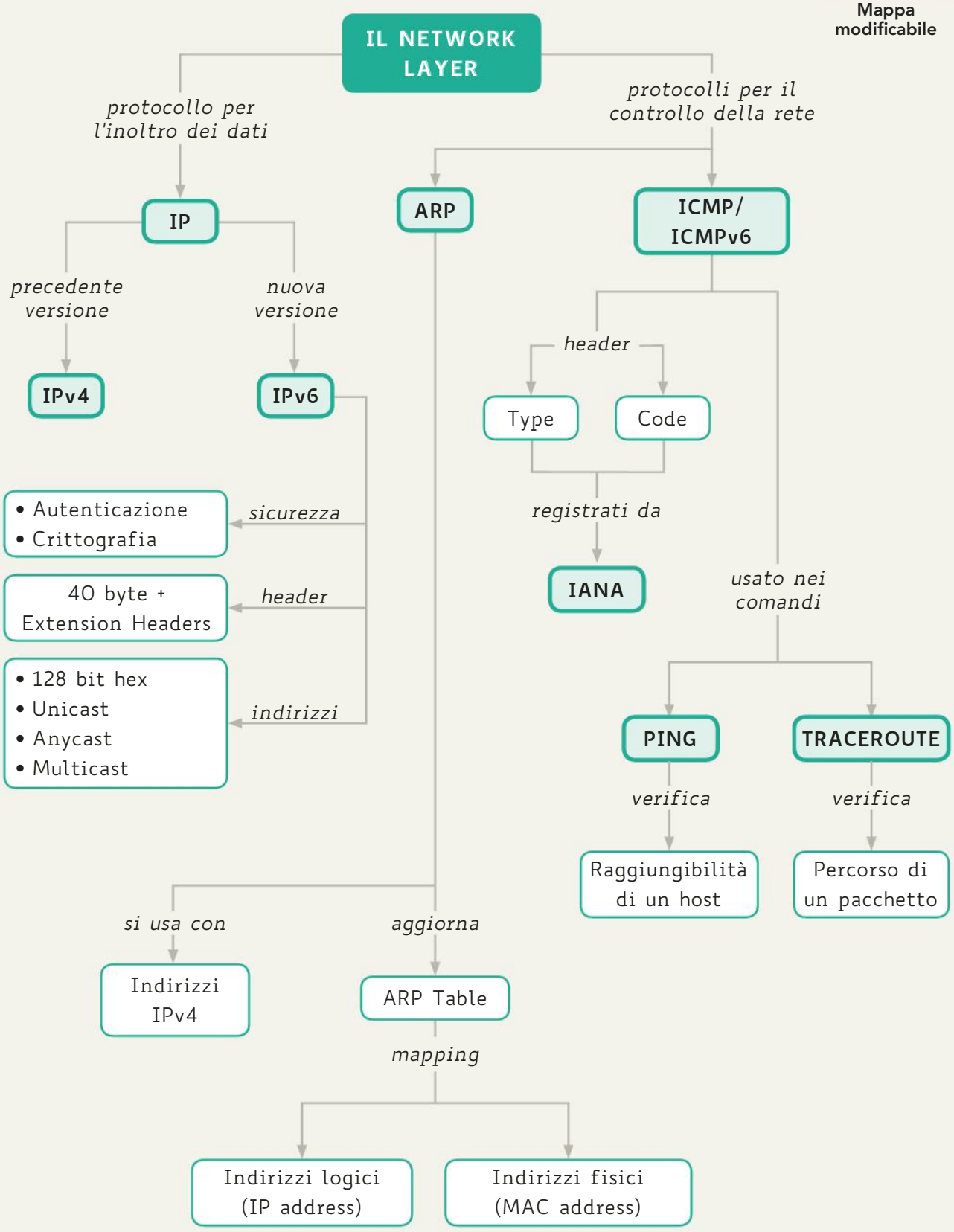
- Leggi la Lezione 5 di questa unità;
- prova i comandi ping e tracert sul tuo PC, usando come destinazione host raggiungibili e non;
- analizza le risposte ottenute: tempi, statistiche, nel caso di tracert anche quali sono i nodi attraversati;
- raccogli i risultati della tua analisi in una breve presentazione.

In classe

- Confrontate i risultati delle varie presentazioni;
- discutete i motivi che spiegano le eventuali differenze, al fine di comprendere meglio il funzionamento dei due comandi.

MAPPA CONCETTUALE

Mappa modificabile



1 L'EVOLUZIONE DEL PROTOCOLLO IP: IPv6

1.1 I cambiamenti introdotti da IPv6

Nonostante la serie di tecniche messe in campo per sopperire al limitato spazio degli indirizzi offerto da IPv4 (indirizzi dinamici, indirizzi privati, la CIDR, le VLSM), la grande diffusione della rete Internet spinse, fin dagli anni Novanta, a progettare una nuova versione di IP che garantisse un numero di indirizzi sufficiente a soddisfare tutte le richieste.

La nuova versione è stata definita dalle RFC 1883 e 1887 con il nome di **IPv6**, la cui caratteristica fondamentale è quella di quadruplicare lo spazio degli indirizzi portandolo da 4 a 16 byte (128 bit).

Questo consente di avere 2^{128} indirizzi possibili: circa 340 miliardi di miliardi di miliardi di miliardi di indirizzi contro i circa 4 miliardi di indirizzi consentiti da IPv4! È quindi possibile collegare in rete non solo qualsiasi host ne faccia richiesta, ma anche ogni tipo di dispositivo intelligente (cellulari, GPS, elettrodomestici).
Gli altri principali cambiamenti introdotti da IPv6 sono:

- semplificazione dell'header IP con una riduzione dei campi e una conseguente più rapida elaborazione delle informazioni in esso contenute;
- flessibilità dell'header IP per quanto riguarda i campi opzionali per garantire l'inserimento in futuro di nuove opzioni senza che si debba riprogettare l'intero formato dell'header;
- miglior controllo del flusso inserendo nell'header la possibilità di richiedere una migliore qualità del servizio o un minimo di larghezza di banda garantita a disposizione o una trasmissione in tempo reale;
- maggiore sicurezza attraverso estensioni dell'header per supportare l'autenticazione del mittente e quella del destinatario o richiedere la crittografia dei dati da trasmettere;
- maggiore efficienza eliminando dall'header il campo checksum il cui ricalcolo (dovuto alla modifica del campo TTL a ogni hop) costringeva ogni volta i router a elaborare i pacchetti più lentamente.

#prendinota

IPv6 è compatibile con tutta la suite di protocolli TCP/IP, ma non con IPv4.

Pur non essendo retro compatibili (backward compatible) i due protocolli IPv4 e IPv6 possono coesistere grazie alla creazione di un tunnel.

#techwords

Tunneling
La tecnica di tunneling è usata nelle telecomunicazioni quando è necessario far transitare i pacchetti di un protocollo che gli apparati di rete non sono in grado di gestire oppure quando si vuol "nascondere" il contenuto di una PDU, non solo i dati utente che trasporta, ma anche i campi dell'header (per esempio se è necessario crittografare gli indirizzi mittente/destinatario).

La FIGURA 1 mostra la tecnica di **#tunneling**, descritta in RFC 2473 e in RFC 4213. Un tunnel serve a trasmettere i pacchetti IPv6 inserendoli nel campo dati all'interno di un pacchetto IPv4. L'indirizzo di destinazione del pacchetto IPv4 è l'indirizzo di un sistema su cui sono attivi entrambi i protocolli. Il destinatario del pacchetto IPv4 estrae il pacchetto IPv6 e lo inoltra verso la destinazione IPv6 finale (che può essere lui stesso). Inoltre:

- il traffico IPv4 in una rete non subisce alcun inconveniente all'attivazione di IPv6;
- tutti gli attuali Sistemi Operativi consentono l'utilizzo contemporaneo dei due protocolli;

- molte applicazioni, quando si trovano su un computer dotato di indirizzo IPv6, scelgono il trasporto IPv6 quando anche il destinatario possiede un indirizzo IPv6: può dunque capitare di essere collegati alla rete Internet IPv6 senza saperlo.

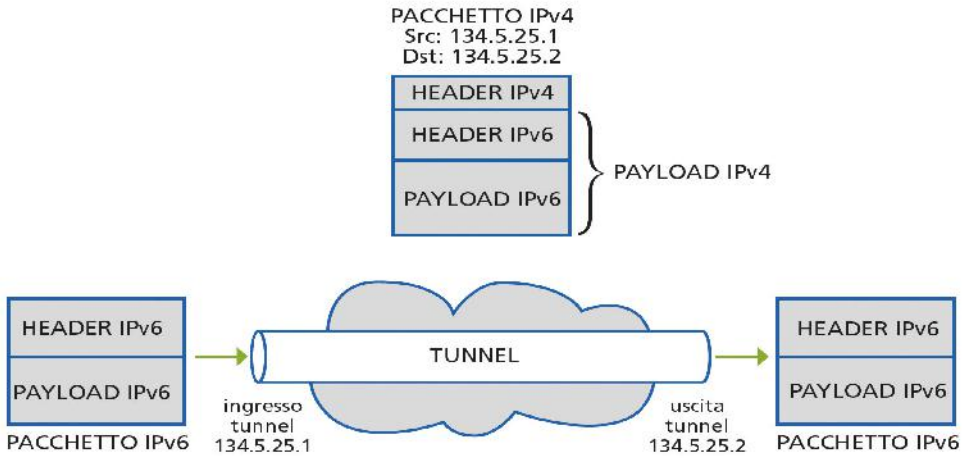


FIGURA 1 IPv6/IPv4 tunneling

1.2 L'header IPv6

Le specifiche di IPv6, emesse da IETF, hanno subito evoluzioni dall'iniziale RFC 2460 del 1998 all'attuale RFC 8200 del 2017, al quale si fa riferimento nella descrizione seguente del formato dei pacchetti IPv6 (FIGURA 2).

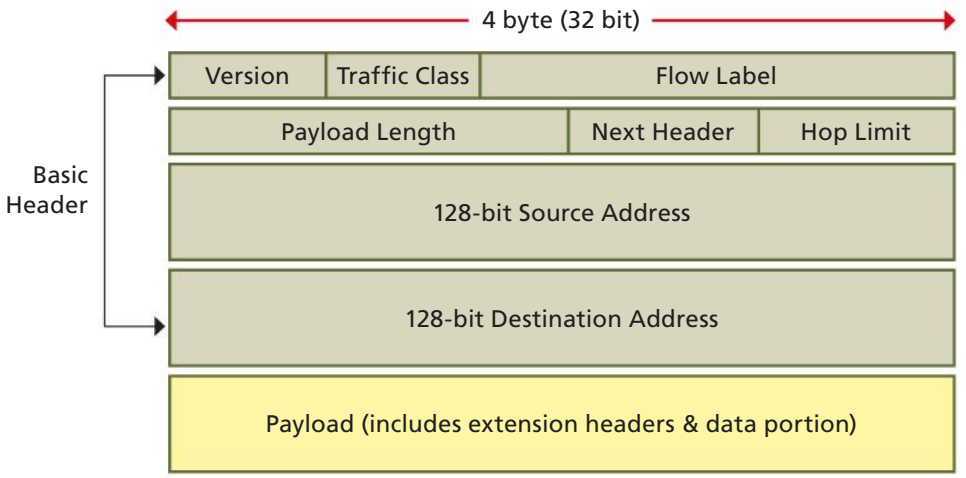


FIGURA 2 La PDU di IPv6

Il formato dell'header IPv6 (opzioni escluse) è costituito da 40 byte:

- **Version**: 4 bit, contiene il numero della versione IP, è impostato su 6 (0110);
- **Traffic Class** (o **Priority**): 8 bit, indica la priorità del pacchetto IPv6 (è simile al campo Type of Service di IPv4). Permette ai router di gestire il traffico in base alla priorità del pacchetto. Se si verifica una congestione sul router, i pacchetti con la priorità minore verranno scartati. Le priorità maggiori sono quelle da 8 a 15 e sono utilizzate per il traffico real time che non deve subire rallentamenti. Le priorità da 0 a 7 indicano invece i tipi di traffico che in caso di congestione possono anche subire rallentamenti;

- **Flow Label**: 20 bit, viene utilizzato dalla sorgente per etichettare i pacchetti appartenenti allo stesso flusso al fine di richiedere una gestione speciale da parte dei router IPv6 intermedi, come il servizio in tempo reale. Se un host o un router non supportano il servizio richiesto, il campo viene impostato a 0, mentre viene ignorato se è il destinatario a non supportare il servizio;
- **Payload Length**: 16 bit, indicano la lunghezza del pacchetto dati, in ottetti, senza l'header;
- **Next Header**: 8 bit, identificano il tipo di header che si trova immediatamente dopo l'header IPv6 di base (vedi descrizione sugli extension header);
- **Hop Limit**: 8 bit, rappresentano il numero massimo di hop consentiti al pacchetto per giungere a destinazione senza esser scartato. Equivale al campo TTL (Time To Live) di IPv4;
- **Source Address** e **Destination Address**: 128 bit ciascuno, rappresentano l'indirizzo IPv6 del mittente e del destinatario.

#prendinota

Nel caso in cui la lunghezza del payload sia maggiore di 65.535 byte (il payload fino a 65.535 byte può essere indicato con 16 bit), il campo **Payload Length** è impostato a 0 e l'opzione di **payload jumbo** viene utilizzata nell'extension Hop-by-Hop Option header.

GLI EXTENSION HEADER DI IPv6

Come già detto, una grossa innovazione di IPv6 è la gestione delle opzioni.
Il campo Options di IPv4 è stato eliminato a causa delle complicazioni che introduce: con le opzioni il tempo di elaborazione dei pacchetti IPv4 aumenta, in quanto ogni router attraversato deve processarle tutte prima di inoltrare il pacchetto.
Per soddisfare l'esigenza di poter specificare comunque delle opzioni, in modo più efficiente rispetto al passato, si è pensato al meccanismo degli **extension header**.

L'elenco aggiornato degli extension header è mantenuto da IANA:

www.iana.org/assignments/ipv6-parameters

Una implementazione completa di IPv6 include i 6 header di seguito elencati; gli altri presenti su IANA sono per contesti applicativi specifici, per esempio il **Mobility header** si usa per la gestione della mobilità di un host. Vediamoli nel dettaglio.

1. Hop-by-Hop Options header

L'header Hop-by-Hop Options contiene informazioni che possono essere elaborate da ogni router della rete attraversata dal pacchetto. Le opzioni che possono interessare tutti i router riguardano di solito funzioni di gestione o di debugging.
Questa opzione può anche indicare un cosiddetto pacchetto **jumbo** cioè un pacchetto superiore ai 65.536 byte (2^{16}).

2. Routing header

Il Routing header è usato da una sorgente per elencare uno o più router che un pacchetto deve attraversare prima di giungere a destinazione.

3. Fragment header

Una novità introdotta da IPv6 è l'eliminazione della frammentazione del pacchetto da parte dei router. Col nuovo protocollo la frammentazione è gestita dal mittente e non più dai router intermedi.
Se le dimensioni del pacchetto superano la dimensione massima consentita su un canale sul quale tale pacchetto deve transitare, il router lo scarta e invia al mittente un messaggio ICMP di errore. A quel punto il mittente frammenta e rispedisce il pacchetto tenendo conto che i router IPv6 garantiscono almeno 1280 byte per pacchetto.

4. Authentication header
IPv6, attraverso questo header, fornisce un servizio che assicura l'autenticità (cioè garantisce l'identità del mittente) e l'integrità del pacchetto (cioè che non sia stato modificato nel tragitto mittente-destinatario).

5. Encapsulating Security Payload header
L'Encapsulating Security header, invece, fornisce un meccanismo di crittografia per trattare i dati in modo che possano essere letti solo al termine del percorso dal destinatario che possiede la chiave appropriata per decrittografare i dati.

6. Destination Options header
È l'unico extension header che **può comparire due volte nello stesso pacchetto**, anche se in due posizioni diverse. Se è posto tra l'Hop-by-Hop e il Routing header potrà essere letto da ogni nodo intermedio. Altrimenti è visibile solo dal destinatario. Il Destination Option header può contenere anche un campo padding, al fine di rendere i pacchetti di lunghezza pari a multipli di 64 byte.
Gli extension header sono inseriti subito dopo l'header IPv6; in questo modo, per i router che non li devono processare, essi fanno parte del payload del pacchetto e non sono analizzati. Per questo motivo si ha un miglioramento delle prestazioni di forwarding.

Ogni pacchetto può contenere più di un extension header. In ognuno di essi c'è un campo Next Header in cui viene specificato il tipo del prossimo header, formando quella che viene chiamata **catena di header** (FIGURA 3).

#prendinota

Encapsulating Security Payload e Authentication header fanno parte anche della suite di protocolli IPSec che sarà descritta nel volume del quinto anno, applicata alle Virtual Private Network (VPN).

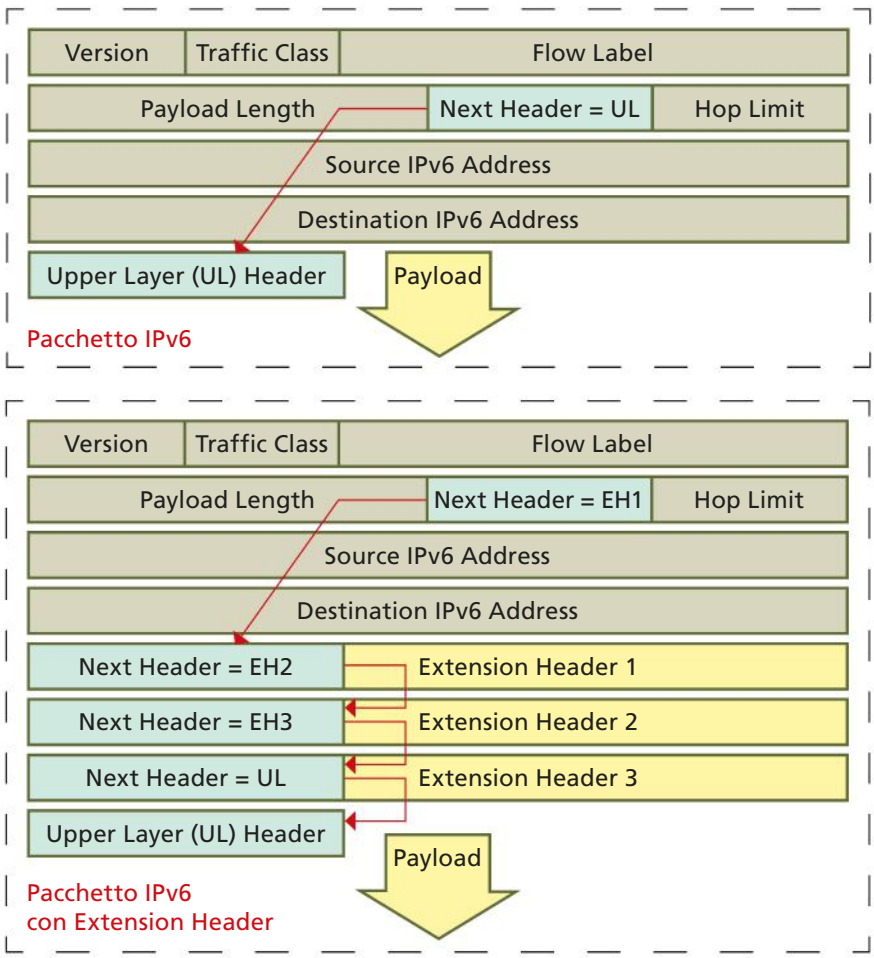


FIGURA 3 La catena degli extension header

Nel formare tale catena occorre rispettare un ordine ben preciso, in cui l'ultimo Next Header (Upper Layer) indica il protocollo di livello superiore trasportato nel payload. Questo protocollo è indicato con un numero, che è lo stesso per IPv4 e IPv6 e si trova nel registro di IANA (*www.iana.org/assignments/protocol-numbers*).

Nell'RFC 8200 è anche indicato l'ordine in cui devono comparire nel pacchetto IPv6 gli extension header, nel caso ne siano presenti più di uno:

1. IPv6 header;
2. Hop-by-Hop Options header;
3. Destination Options header (con opzione di instradamento);
4. Routing header;
5. Fragment header;
6. Authentication header;
7. Encapsulating Security Payload header;
8. Destination Options header;
9. Upper Layer header.

L'ordinamento corretto e la non ripetizione degli extension header è fondamentale: si sono registrati casi di nodi di rete andati in crash nell'elaborare pacchetti IPv6 formattati in modo errato.
In origine gli extension header, tranne l'Hop-by-Hop, erano stati definiti per essere trasparenti ai router intermedi ed essere processati solo dall'host di destinazione. Nelle reti attuali esistono apparati intermedi, come i firewall, che esaminano anche questi header, rallentando così il trasferimento del pacchetto, o addirittura lo scartano se non riconoscono un extension header.
La specifica di IPv6 raccomanda di non definire nuovi extension header, a meno che proprio nessuno di quelli già presenti possa essere usato per specificare la nuova opzione. Eventuali nuove informazioni che devono essere esaminate solo dal destinatario, devono essere trasportate usando il Destination Options header, in quanto fornisce una gestione ottimale e garantisce la compatibilità con le release precedenti (#backward compatibility).

#techwords

Backward compatibility
Con questo termine si indica la compatibilità con le versioni precedenti di un software o con dispositivi più datati. Se una nuova versione di un protocollo di rete è retrocompatibile significa che può comunicare con apparati che usano le versioni precedenti di quel protocollo.

FISSA LE CONOSCENZE

- Tra IPv4 e IPv6 chi consente il maggior numero di indirizzamenti? Perché?
- IPv6 è compatibile con tutta la suite di protocolli TCP/IP?
- Come si riesce a far coesistere IPv4 e IPv6?
- Quale campo dell'header IPv6 ha sostituito il TTL di IPv4? Qual è la differenza?
- Come viene usato il campo Next Header?
- Quali sono gli extension header definiti per IPv6?
- Quale extension header può (non "deve") essere analizzato dai router intermedi?
- Perché è importante mantenere l'ordine di sequenza con cui sono scritti gli extension header nel pacchetto?

2 GLI INDIRIZZI IPv6

2.1 Il formato degli indirizzi IPv6

L'**architettura degli indirizzi IPv6** è specificata nell'RFC 4291. Ha poi subito successive modifiche riportate in vari RFC, tutti elencati nel 4291 come "Updated by".
Il sistema introdotto da IPv6 richiede di distinguere gli indirizzi in 3 categorie fondamentali: **unicast**, **anycast** e **multicast**.

- **Unicast**: è un indirizzo che riguarda un'interfaccia di rete singola; in altri termini, un indirizzo unicast serve per raggiungere un'interfaccia di rete in modo univoco.
- **Anycast**: è un indirizzo che ha le stesse caratteristiche sintattiche di quello unicast, attribuito però a diverse interfacce di altrettanti nodi, con lo scopo di poter raggiungere **quello che risponde prima** (quello più vicino in base al protocollo di instradamento). I pacchetti inviati a un indirizzo anycast raggiungono un'unica interfaccia di rete, la prima che risponde. Gli indirizzi anycast possono essere usati solo dai router.
- **Multicast**: è attribuito a più interfacce di rete distinte; i pacchetti inviati a un indirizzo multicast raggiungono **tutte** le interfacce di rete cui questo indirizzo è stato attribuito.

#prendinota

Gli **indirizzi IPv4 broadcast** non esistono più in IPv6, ma si esprimono con indirizzi multicast.

La notazione prevede che i 128 bit vengano suddivisi in **8 gruppi di 16 bit** (detti **hextet**) e i 16 bit di ciascun gruppo siano rappresentati con **4 cifre esadecimali**. Gli 8 gruppi vengono poi separati dal carattere " **:** ".

Un esempio di indirizzo IPv6 è:

3401:0db6:0000:0000:00a9:0000:0000:000c

È consentito omettere gli 0 iniziali (*leading zero*) di ciascun gruppo e sostituire con un solo 0 un hextet composto da tutti 0 (0000):

3401:db6:0:0:a9:0:0:c

Se l'indirizzo contiene una o più sequenze, contigue, di gruppi di valore 0, una di queste sequenze può essere sostituita dalla notazione " :: ", come segue:

3401:db6::a9:0:0:c oppure 3401:db6:0:0:a9::c

Si noti che **solo una** delle sequenze 0000 può essere sostituita con ::, in caso contrario l'indirizzo IPv6 non potrebbe più essere ricostruito in modo univoco.

#prendinota

IETF nell'RFC 5952 ha definito delle **linee guida** per la rappresentazione compatta degli indirizzi IPv6:
- eliminare i leading zero
- rappresentare un campo da 0000 sempre con 0 e mai con ::
- abbreviare il più possibile
- abbreviare sempre il blocco più grande di zeri
- se due blocchi di zeri sono di pari lunghezza abbreviare quello più a sinistra
- usare lettere minuscole per a, b, c, d, e, f

LA STRUTTURA DEGLI INDIRIZZI IPv6

Vediamo ora come è strutturato un generico indirizzo **unicast IPv6** (FIGURA 4):

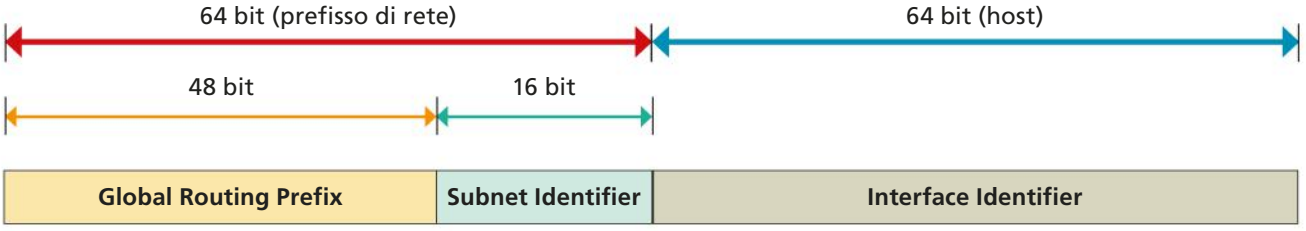


FIGURA 4 L'indirizzo IPv6

- **Global Routing Prefix** (o **Prefisso**): sono i primi 3 hextet e sono assegnati dagli ISP ai client IPv6; i primi 3 bit più significativi sono sempre impostati a 001;

- **Subnet ID**: è il quarto hextet e identifica una subnet presente nella rete. Si possono creare fino a 65.536 subnet diverse. Se non viene usato, può essere lasciato a 0 o a 1 o a qualunque altro valore;
- **Interface ID (IID)**: è rappresentato negli ultimi 4 hextet e corrisponde al campo HostID di IPv4.

LE TIPOLOGIE DI INDIRIZZI IPv6

Come abbiamo visto in precedenza gli indirizzi IPv6 si distinguono in unicast, anycast e multicast. La TABELLA 1 mostra alcuni tipi di indirizzi IPv6 unicast e multicast.

TABELLA 1 Indirizzi IPv6 unicast e multicast

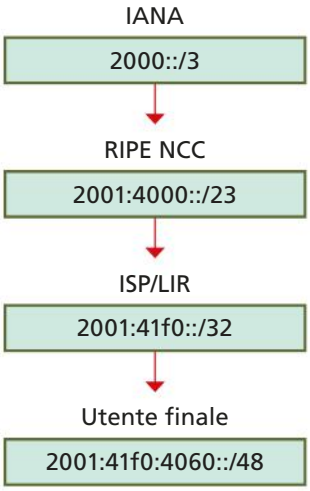
Indirizzo	Inizia con …	Descrizione
Global Unicast Address (GUA)	**2000::/3** (prefisso 001)	Indirizzi unicast pubblici e instradabili su Internet. Un'interfaccia di rete può averne assegnati anche più di uno.
Link Local Address (LLA)	**fe80::/10** (i 54 bit successivi al prefisso sono tutti 0)	Indirizzi unicast locali, ogni interfaccia deve averne almeno uno. Si usano solo per le comunicazioni nella rete locale e non su Internet.
Unique Local Unicast Address (ULA)	**fd00::/8** fc00::/7	Indirizzi unicast locali univoci (simili agli indirizzi privati IPv4). Come gli indirizzi Link local si usano solo all'interno di una LAN, ma, a differenza di questi, gli indirizzi ULA sono unici a livello globale (RFC 4193).
Loopback	**::1/128**	Indirizzo di loopback: 0:0:0:0:0:0:0:1
Unspecified	**::/128**	Indirizzo non specificato: 0:0:0:0:0:0:0:0, è utilizzato in fase di bootstrap come indirizzo sorgente quando l'host non conosce alcun altro suo indirizzo. Non può essere usato come indirizzo di destinazione.
Multicast	**ff::/8**	L'elenco degli indirizzi multicast predefiniti si trova nell'RFC 4291. Per esempio: **ff02::1** indirizza tutti i nodi, mentre **ff02::2** tutti i router.

#prendinota

Come **default gateway IPv6** sugli host, di solito, si usa il Link local address dell'interfaccia LAN del router, ma si può usare anche il Global Unicast.

L'indirizzo Link local è generato automaticamente allo startup per consentire la configurazione dell'host. Nel laboratorio presentato nella Lezione 6 si vedrà il procedimento per creare un indirizzo IPv6 di tipo Link local a partire dal MAC address dell'interfaccia di rete. Il MAC address è usato per creare la parte host dell'indirizzo; l'IID così creato prende il nome di "**EUI-64 Interface ID**" (EUI = Extended Unique Identifier).

2.2 L'assegnazione degli indirizzi IPv6



Come gli indirizzi IPv4, anche quelli IPv6 sono gestiti da IANA e, a livello regionale dai RIR (vedi la Lezione 2 dell'Unità 3). L'elenco aggiornato degli indirizzi Global Unicast IPv6 allocati ai RIR si trova al link: *www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml*
I RIR, come **RIPE NCC** per l'Europa, **allocano** gli indirizzi IPv6 agli Internet Registry locali all'area geografica in cui operano, detti Local Internet Registry (**LIR**), che spesso coincidono con gli Internet Service Provider (**ISP**), che a loro volta **assegnano** gli indirizzi agli utenti delle reti a cui forniscono il servizio (*provide*).

FISSA LE CONOSCENZE

- Da quanti bit è formato un indirizzo IPv6?
- Quali regole sono state definite da IETF per la rappresentazione compatta degli indirizzi IPv6?
- Spiega il Link local address e come viene usato sui router.

3 IL MONITORING DELLA RETE CON IL PROTOCOLLO ICMP

3.1 Internet Control Message Protocol (ICMP)

L'ICMP (Internet Control Message Protocol), **RFC 792**, fornisce un meccanismo di monitoraggio della rete, utilizzato prevalentemente dai router o dagli host destinatari per segnalare agli host mittenti eventuali insuccessi nell'instradamento dei pacchetti.

IN ENGLISH PLEASE

Network Working Group — J. Postel
Request For Comments: 792 — ISI
September 1981

Updates: RFCs 777, 760
Updates: IENs 109, 128

Internet Control Message Protocol

DARPA INTERNET PROGRAM
PROTOCOL SPECIFICATION

Introduction

The Internet Protocol (IP) [1] is used for host-to-host datagram service in a system of interconnected networks called the Catenet [2]. The network connecting devices are called Gateways.

These gateways communicate between themselves for control purposes via a Gateway to Gateway Protocol (GGP) [3,4]. Occasionally a gateway or destination host will communicate with a source host, for example, to report an error in datagram processing. For such purposes this protocol, the Internet Control Message Protocol (ICMP), is used. ICMP, uses the basic support of IP as if it were a higher level protocol, however, ICMP is actually an integral part of IP, and must be implemented by every IP module.

ICMP messages are sent in several situations: for example, when a datagram cannot reach its destination, when the gateway does not have the buffering capacity to forward a datagram, and when the gateway can direct the host to send traffic on a shorter route.

The Internet Protocol is not designed to be absolutely reliable. The purpose of these control messages is to provide feedback about problems in the communication environment, not to make IP reliable.

#prendinota

ICMP è spesso considerato parte di IP, in realtà, nell'architettura TCP/IP, è posizionato *sopra* IP. Infatti i messaggi ICMP sono trasportati all'interno del datagram IP, come payload.

Il pacchetto ICMP viene incapsulato nel pacchetto IP ed è caratterizzato da 4 campi, come mostrato nella **FIGURA 5**.

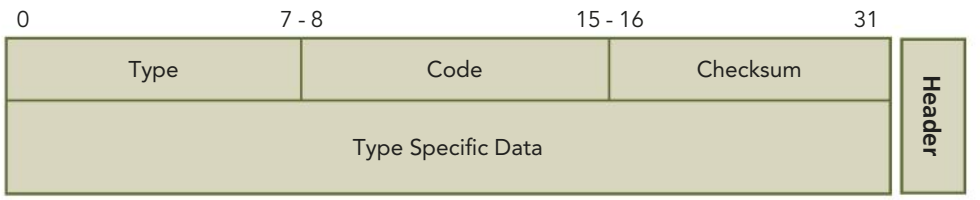


FIGURA 5 Il pacchetto ICMP

Vediamo in dettaglio i vari campi:

- **Type** è il più significativo, 8 bit che indicano il tipo di pacchetto ICMP trasmesso:

0 Echo Reply	**10** Router Solicitation	**20-29** Riservati (*per test di robustezza*)
1 *Non assegnato*	**11** Time Exceeded	**30** Traceroute (D)
2 *Non assegnato*	**12** Parameter Problem	**31** Datagram Conversion (D)
3 Destination Unreachable	**13** Timestamp Request	**32** Redirect su host mobile (D)
4 Source Quench (D)	**14** Timestamp Reply	**33** IPv6 Where-Are-You (D)
5 Routing Redirect	**15** Information Request (D)	**34** IPv6 I-Am-Here (D)
6 Alternate Host Address (D)	**16** Information Reply (D)	**35** Mobile Registration Request (D)
7 *Non assegnato*	**17** Address Mask Request (D)	**36** Mobile Registration Reply (D)
8 Echo Request	**18** Address Mask Reply (D)	**37** Domain Name Request (D)
9 Router Advertisement	**19** Riservato (*per sicurezza*)	**38** Domain Name Reply (D)

(D) = Deprecated, IETF ne sconsiglia ormai l'uso

#prendinota

L'elenco aggiornato dei valori definiti per Type e Code si trova sul sito di IANA: *www.iana.org/assignments/icmp-parameters*

Per IPv6: *www.iana.org/assignments/icmpv6-parameters*

- **Code** fornisce indicazioni aggiuntive non comprese nel campo Type;
- **Checksum** contiene i bit per il controllo degli errori di trasmissione;
- **Type Specific Data** contiene informazioni che dipendono dal tipo di servizio che l'ICMP sta offrendo. Per esempio le più comuni Echo Request/Echo Reply comprendono un identificatore e un numero sequenziale che servono a identificare ciascuna richiesta di eco e ciascuna risposta.

3.2 Le funzioni svolte da ICMP

Le principali funzioni che il protocollo ICMP può svolgere sono:

- fornire messaggi di eco per verificare la corretta configurazione di host sulla rete e che quindi una qualsiasi destinazione sia raggiungibile: Echo Request (Type 8) del mittente, Echo Reply (Type 0) del destinatario. Si realizza con il comando **ping**;
- segnalare una destinazione non raggiungibile perché sconosciuta o perché un pacchetto è troppo grande ma non è consentito frammentarlo: Destination Unreachable (Type 3);
- avvertire il mittente di rallentare l'invio dei pacchetti per problemi di congestione: Source Quench (Type 4);
- reindirizzare il traffico per fornire un instradamento efficiente in caso di router congestionato da traffico eccessivo: Routing Redirect (Type 5);
- avvertire il mittente che il tempo di vita di un suo pacchetto è scaduto (TTL = 0) e che quindi il pacchetto viene scartato: Time Exceeded (Type 11);
- valutare le prestazioni di una rete misurando il tempo di attraversamento: Timestamp Request (Type 13) del mittente, Timestamp Reply (Type 14) del destinatario;

- rilevare la lista dei nodi (router) attraversati da un pacchetto per giungere a destinazione: Traceroute (Type 30). Si realizza con il comando **tracert**.

ICMP consente ai router di scambiarsi informazioni di servizio (**messaggi router-to-router**) e di tenere sotto controllo le modalità con cui gli host generano pacchetti, inviando loro messaggi per rallentare o dirottare altrove un flusso di pacchetti (**messaggi router-to-host**).
Per quanto riguarda gli host invece, ICMP consente loro di scambiarsi informazioni di servizio (**messaggi host-to-host**) e di richiedere ai router informazioni utili sul funzionamento e la topologia della rete (**messaggi host-to-router**).

Nel laboratorio presentato nella Lezione 5, vedremo l'impiego dei messaggi ICMP nei comandi ping e traceroute (o tracert).

3.3 ICMPv6

Una nuova versione di ICMP è stata definita per lavorare con la versione 6 di IP, descritta nella Lezione 1. Infatti, lo sviluppo di IPv6 ha reso necessaria una riorganizzazione dei tipi e dei codici esistenti in ICMP e la definizione di nuovi. Il formato del pacchetto ICMPv6 è, però, rimasto lo stesso di ICMPv4. **ICMPv6** è specificato in **RFC 4443** e successivi aggiornamenti.

La versione ICMPv6 è stata potenziata rispetto alla ICMPv4, aggiungendo nuove funzionalità e incorporandone altre derivanti da protocolli IPv4 come IGMP e ARP. I numeri dei messaggi e dei tipi sono diversi da quelli ICMPv4, rendendo così incompatibili i due protocolli.

In ICMPv6 si distinguono due categorie di messaggi:

- **Error message**: riportano errori relativi all'inoltro di pacchetti IPv6, generati dal destinatario o dai nodi intermedi della rete; questi messaggi hanno il campo Type con valori da 0 a 127 (bit più significativo = 0), per esempio
  - Type 1, Destination Unreachable
  - Type 2, Packet Too Big (in IPv6 non è più prevista la frammentazione del pacchetto)
  - Type 3, Time Exceeded
- **Informational message**: forniscono informazioni di tipo diagnostico e sugli host della rete; questi messaggi hanno il campo Type con valori da 128 a 255 (bit più significativo = 1), per esempio
  - Type 128, Echo Request
  - Type 129, Echo Reply

FISSA LE CONOSCENZE

- Qual è lo scopo del protocollo ICMP?
- Descrivi il formato del pacchetto ICMP.
- Quali sono le principali funzioni che ICMP può svolgere?

4 INDIRIZZI FISICI E INDIRIZZI IP: IL PROTOCOLLO ARP

4.1 Address Resolution Protocol (ARP)

#prendinota

Nelle architetture di rete a strati, per realizzare l'indipendenza di un livello dall'altro, è necessario che i diversi strati implementino un proprio schema di indirizzamento. In TCP/IP troviamo il **MAC address** nel Physical Layer, l'**IP address** nel Network Layer e l'**host name** nell'Application Layer.

Come abbiamo visto nelle Lezioni precedenti, a ogni host di una rete TCP/IP viene assegnato un indirizzo logico IP che lo identifica univocamente. In realtà, abbiamo parlato di "interfaccia di rete", ossia l'indirizzo IP è assegnato alla NIC; se un host ha più schede di rete, potrà essere identificato con più indirizzi IP.
Affinché due host possano comunicare tra loro questo però non basta. Bisogna che le rispettive schede di rete siano capaci di localizzarsi reciprocamente. Occorre cioè che l'indirizzo fisico del destinatario sia noto al mittente, infatti il frame del livello Physical, per esempio il frame Ethernet, richiede sia il MAC address del mittente sia il MAC address del destinatario.

In ambito IETF è stato sviluppato il protocollo **ARP** (Address Resolution Protocol), **RFC 826**, che definisce le modalità di comunicazione tra gli host di una rete locale per trovare il MAC address di una scheda di rete della quale si conosce solo l'indirizzo IP. Questa operazione è detta **risoluzione dell'indirizzo IP**.
ARP è usato solamente per indirizzi IPv4, non funziona con IPv6.

IN ENGLISH PLEASE

Network Working Group
Request For Comments: 826

David C. Plummer
(DCP@MIT-MC)
November 1982

An Ethernet Address Resolution Protocol

-- or --

Converting Network Protocol Addresses to 48.bit Ethernet Address
for Transmission on Ethernet Hardware

Abstract

The implementation of protocol P on a sending host S decides, through protocol P's routing mechanism, that it wants to transmit to a target host T located some place on a connected piece of 10Mbit Ethernet cable. To actually transmit the Ethernet packet a 48.bit Ethernet address must be generated. The addresses of hosts within protocol P are not always compatible with the corresponding Ethernet address (being different lengths or values).
Presented here is a protocol that allows dynamic distribution of the information needed to build tables to translate an address A in protocol P's address space into a 48.bit Ethernet address.

#prendinota

In IPv6, ARP è stato sostituito dal protocollo **Neighbor Discovery** (RFC 4861), che utilizza i nuovi messaggi definiti in ICMPv6.

IL FORMATO DEL PACCHETTO ARP

Il protocollo ARP prevede solo due tipi di messaggi: **ARP Request**, per la richiesta di risoluzione di un indirizzo IP, e **ARP Reply**, per la risposta contenente l'indirizzo IP

richiesto. Quindi, il formato del pacchetto ARP è molto semplice: nella FIGURA 6 lo si mostra nel caso di una rete Ethernet (MAC address di 6 byte) e protocollo IPv4 (IP address di 4 byte).

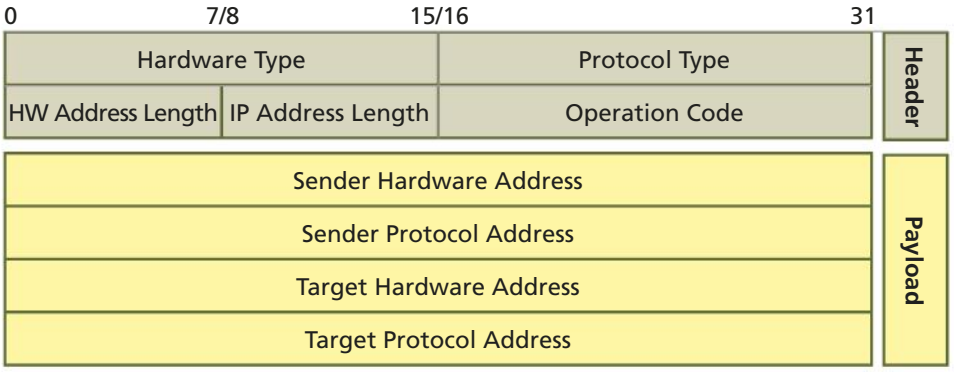


FIGURA 6 Il pacchetto ARP per Ethernet e IPv4

Nel pacchetto ARP, l'header e il payload contengono i seguenti campi:

1. **Header**
   - **Hardware Type**: indica il tipo di rete a livello Physical, 0x0001 sta per Ethernet;
   - **Protocol Type**: indica il tipo di protocollo a livello Network, 0x0800 sta per IP;
   - **Hardware Address Length**: è la lunghezza, in ottetti, dell'indirizzo fisico;
   - **IP Address Length**: è la lunghezza, in ottetti, dell'IP address;
   - **Operation Code**: specifica se il pacchetto è una ARP Request (1) o una ARP Reply (2).
2. **Payload**

   Il significato dei campi varia a seconda che il messaggio sia ARP Request o ARP Reply:

	ARP Request	ARP Reply
Sender Hardware Address	indirizzo fisico del mittente (MAC add.)	indirizzo fisico dell'**host richiesto** con la ARP Request
Sender Protocol Address	indirizzo logico del mittente (IP add.)	indirizzo logico del mittente
Target Hardware Address	*il campo non è valorizzato in quanto sconosciuto (è l'indirizzo fisico richiesto con ARP)*	indirizzo fisico del destinatario (cioè dell'host che ha inviato l'ARP Request)
Target Protocol Address	indirizzo logico del destinatario	indirizzo logico del destinatario

LA RISOLUZIONE DELL'INDIRIZZO IP

Un'implementazione TCP/IP utilizza di norma una **cache ARP** (detta anche ARP Table), dove ogni host mantiene e aggiorna una tabella con tutte le coppie IP-MAC a lui note.

Quando un host deve inviare dei pacchetti, controlla se nella cache ARP è presente l'indirizzo MAC corrispondente all'indirizzo IP del destinatario. Se non c'è, allora entra in gioco il protocollo ARP che stabilisce il seguente iter basato sui due tipi di messaggi, ARP Request e ARP Reply:

1. il mittente invia un pacchetto ARP contenente una ARP Request in cui specifica l'indirizzo IP del destinatario di cui vuole conoscere il corrispondente indirizzo MAC.

Questo pacchetto ARP viene mandato all'indirizzo MAC di broadcast Ethernet FF-FF-FF-FF-FF-FF, cioè a tutti i nodi della rete. Inoltre il mittente aggiunge anche il proprio IP e il proprio MAC affinché il destinatario possa aggiungere la coppia di indirizzi nella propria cache ARP;

2. tutti gli host della rete ricevono l'ARP Request e leggono l'IP in esso specificato:
   - se l'indirizzo IP non corrisponde al proprio, gli host ignorano il pacchetto e la sua richiesta;
   - se un host riscontra che l'IP specificato è il proprio indirizzo, allora prepara un pacchetto ARP di risposta contenente una ARP Reply in cui specifica l'indirizzo MAC corrispondente al proprio IP; inoltre aggiunge la coppia di indirizzi IP-MAC del mittente nella propria cache ARP;
3. il mittente, ricevuta la risposta, aggiorna la propria cache ARP e avvia la comunicazione.

Il procedimento ARP è differente se utilizzato su **reti remote**. Per dialogare con l'host remoto, l'host mittente si affida al gateway predefinito, impostato nelle proprietà del TCP/IP, al quale dirige tutto il traffico indirizzato all'host che non riesce a raggiungere. Se eventualmente poi si trattasse di un'implementazione TCP/IP che non prevede il gateway, il mittente invierebbe i pacchetti al router di rete.
In ogni caso il mittente può usare il pacchetto ARP per individuare gateway o router, qualora il loro MAC non fosse mappato nella sua cache ARP, nello stesso modo con cui individuava il MAC dell'host destinatario in locale.

È possibile conoscere il contenuto della propria cache ARP eseguendo il seguente comando dal prompt dei comandi di Windows:

```
arp –a
```

Nella FIGURA 7 vediamo, per esempio, la cache ARP dell'host 192.168.1.221.

FIGURA 7 Una cache ARP visualizzata con il comando arp -a

```
C:\Users>arp -a

Interfaccia: 192.168.1.11 --- 0xa
  Indirizzo Internet    Indirizzo fisico      Tipo
  192.168.1.1           64-59-f8-2b-64-e0     dinamico
  192.168.1.255         ff-ff-ff-ff-ff-ff     statico
  224.0.0.2             01-00-5e-00-00-02     statico
  224.0.0.22            01-00-5e-00-00-16     statico
  224.0.0.251           01-00-5e-00-00-fb     statico
  224.0.0.252           01-00-5e-00-00-fc     statico
  239.255.255.250       01-00-5e-7f-ff-fa     statico
  255.255.255.255       ff-ff-ff-ff-ff-ff     statico
```

#prendinota

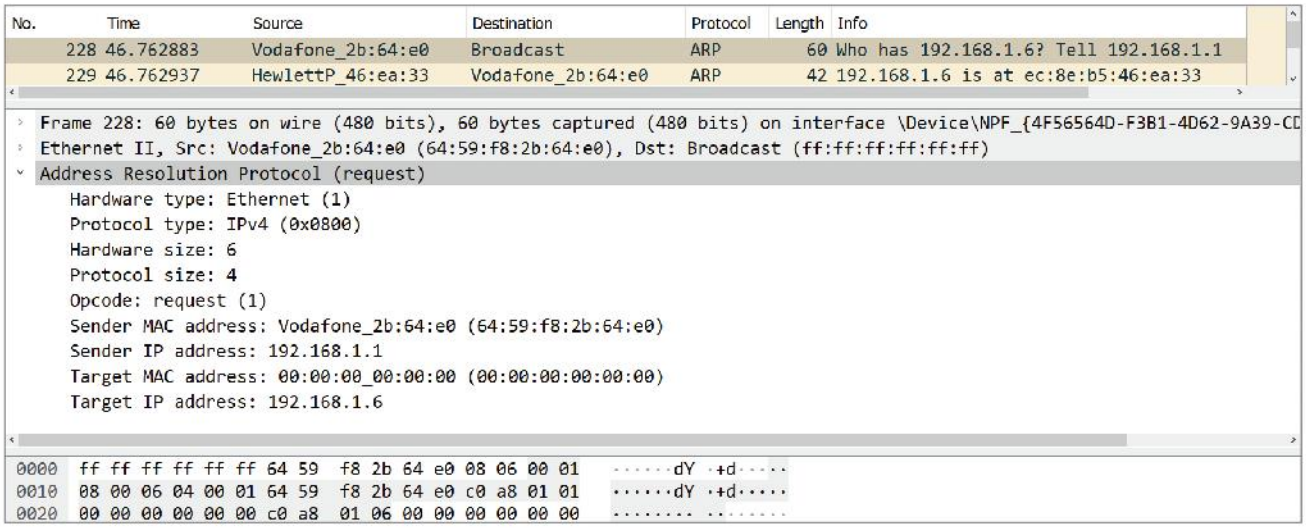
RARP è stato sostituito dal protocollo BOOTP che, a sua volta, è stato superato dal protocollo DHCP.

Esiste anche il protocollo **RARP (Reverse Address Resolution Protocol)** che permette l'operazione inversa, cioè consente a un host della rete che non conosce il proprio indirizzo IP (per esempio periferiche di rete che necessitano di un indirizzo IP) di chiederlo inviando il proprio MAC.
La richiesta va inoltrata però a un server RARP, l'unico in grado di avere nella propria cache ARP il MAC richiesto. Anche il pacchetto RARP è costituito dai due tipi di messaggi: RARP Request e RARP Reply.

4.2 Analisi di un pacchetto ARP con Wireshark

Utilizzando Wireshark (descritto nell'Unità 1) possiamo catturare ed esaminare il contenuto di tutti i pacchetti dati in transito sulle interfacce di rete utilizzate.
Nella FIGURA 8 vediamo nel dettaglio un pacchetto ARP di tipo ARP Request (campo Opcode = 1) in cui l'host 192.168.1.1 chiede all'host 192.168.1.6 di inviargli il suo MAC address (campo Target MAC address = 00:00:00:00:00:00).

FIGURA 8 Cattura di un pacchetto ARP Request



Nella FIGURA 9 vediamo la risposta di 192.168.1.6 mediante un pacchetto ARP Reply (Opcode = 2) in cui specifica il proprio MAC (campo Sender MAC address = ec:8e:b5:46:ea:33).

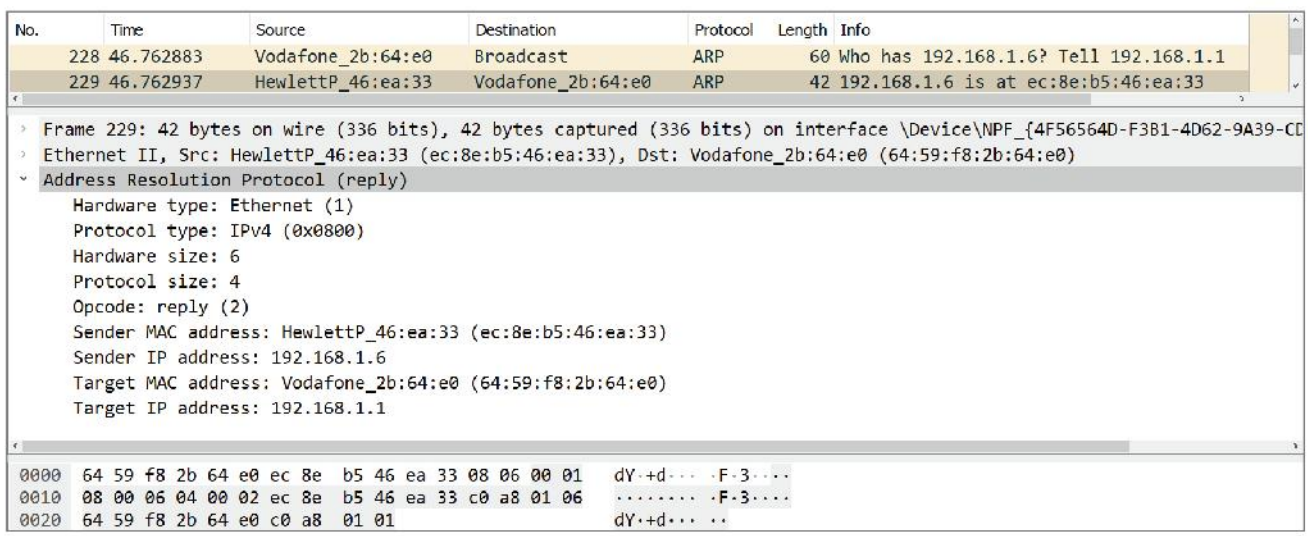


FIGURA 9 Cattura di un pacchetto ARP Reply

4.3 Le vulnerabilità di ARP

Il protocollo ARP è uno dei più vecchi protocolli sviluppati per la suite TCP/IP, quando ancora non si prevedeva la diffusione capillare di Internet e non sembrava necessario mettere in campo azioni preventive per la protezione delle reti.

Una delle conseguenze è che il compito svolto dal protocollo ARP è stato predisposto senza alcun meccanismo di autenticazione. Questo crea le premesse per numerose vulnerabilità.

#techwords

Spoofing
È la falsificazione dell'identità. Questa tecnica può essere utilizzata per falsificare diverse informazioni, come per esempio l'identità di un host all'interno di una rete o il mittente di un messaggio.

Lo **#spoofing** risulta relativamente semplice per un pirata informatico: è sufficiente che invii a un host di una rete X un pacchetto ARP contenente una ARP Reply in cui affermi che il proprio indirizzo MAC è associato a un indirizzo IP della rete X stessa.

Poiché non vi è alcun modo di verificare la veridicità di un'identità, chiunque può introdursi in una rete facendo credere di esserne un legittimo utente, ottenendo così accesso alle risorse della rete, per esempio al data base aziendale.
La scopo di questi attacchi è di ingannare lo switch, inquinandone la cache ARP (**ARP cache poisoning**), al punto da indurlo a inoltrare pacchetti verso destinazioni altrimenti non raggiungibili.

La FIGURA 10 mostra la sequenza di attacco di tipo spoofing al protocollo ARP:

1. l'host attaccante (192.168.1.2) invia un pacchetto ARP contenente una ARP Reply in cui afferma che: «Il mio indirizzo MAC è 00:00:33:00:00:CC (vero) e il mio indirizzo IP è 192.168.1.254 (falsa identità)»;
2. l'host sotto attacco si ritroverà la cache ARP inquinata e di conseguenza invierà il proprio traffico verso l'host attaccante convinto di mandarlo verso il proprio router;
3. l'host attaccante, dopo aver disposto a suo piacimento dei pacchetti ricevuti, li inoltrerà in modo trasparente al vero host che ha indirizzo 192.168.1.254 (nella Figura 10 è il router).

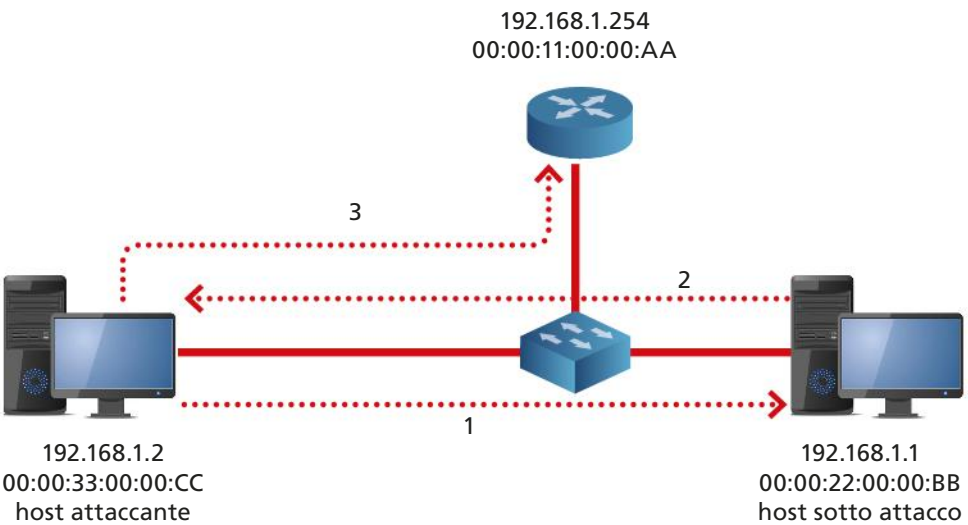


FIGURA 10 Attacco di tipo spoofing al protocollo ARP

LABORATORIO ONLINE

ANALISI DI IP, ARP E ICMP CON WIRESHARK
In questo laboratorio si usa l'applicazione Wireshark per analizzare l'header dei pacchetti dei protocolli IPv4, ARP e ICMP.

FISSA LE CONOSCENZE

- Qual è il compito di ARP? E che cosa contiene la sua cache?
- Quali sono i messaggi previsti nel protocollo ARP? Come sono usati?
- Descrivi il formato dei pacchetti ARP.
- A che cosa serve il protocollo RARP?
- Descrivi le vulnerabilità di ARP.

5 I COMANDI PING E TRACEROUTE

5.1 Il comando ping

Se si verificano problemi di connettività, è possibile utilizzare il comando **ping** per controllare la raggiungibilità di un qualsiasi indirizzo IP e visualizzare i risultati ottenuti. Il comando ping indica se è stata restituita una risposta dalla destinazione e quanto tempo è trascorso prima di riceverla.
Se si verifica un errore nella consegna, il comando ping visualizza un messaggio di errore.
Il comando ping usa i messaggi Echo Request ed Echo Reply di ICMP (Lezione 3 di questa unità).

esercizio

➜ PROBLEMA

Utilizzare il comando ping per stabilire se:
- TCP/IP è correttamente configurato sulla propria macchina;
- il router gateway è funzionante (rete locale raggiungibile);
- un indirizzo Internet è raggiungibile (host remoto raggiungibile).

➜ ANALISI DEL PROBLEMA

Per ottenere le informazioni sugli indirizzi IP locali si possono usare i seguenti comandi sui sistemi Windows e Linux:
- Windows: dal Prompt dei Comandi digitare **ipconfig /all**
- Linux: da terminale digitare **ip addr**

Sia Windows sia Linux impostano alcuni **parametri di default**, modificabili con le opzioni previste dal comando.

Windows:
- 4 è il numero di messaggi ECHO_REQUEST inviati;
- 32 byte è la lunghezza del campo dati del messaggio inviato;
- 128 è il valore del campo TTL del messaggio inviato;
- 4000 ms è il massimo tempo di attesa per ricevere il messaggio ECHO_REPLY; scaduto questo tempo verrà visualizzato il messaggio *Richiesta scaduta* (o *Request timed out*).

Linux:
- 56 byte è la lunghezza del campo dati del pacchetto da inviare ai quali si aggiungono gli 8 byte dell'header ICMP;
- 1 sec è il tempo di attesa tra l'invio di un pacchetto e l'invio successivo; la sequenza di invio può essere interrotta con CTRL+C.

➜ SVOLGIMENTO

Vediamo i 3 casi del problema, usando l'applicazione Command Prompt di Windows.

1) ping 127.0.0.1 (localhost)
Di default vengono inviati 4 pacchetti e attraverso i messaggi ICMP Echo Request/Echo Reply e Timestamp Request/Timestamp Reply si ottengono informazioni stati-

#prendinota

Mike Muuss, autore del programma **ping** (1983), affermò che il nome è nato per analogia con il suono che emette un sonar. Infatti, nel programma ping si usano i pacchetti Echo per misurare la distanza di un host remoto, che è un po' lo stesso metodo usato da un sistema sonar. Negli anni seguenti David Mills definì ping come acronimo di Packet IterNet Grouper.

stiche sull'esito della trasmissione e sui tempi di percorrenza (FIGURA 11). La risposta ottenuta indica che TCP/IP è correttamente configurato.

FIGURA 11 Ping all'indirizzo localhost

```
C:\Users>ping 127.0.0.1

Esecuzione di Ping 127.0.0.1 con 32 byte di dati:
Risposta da 127.0.0.1: byte=32 durata<1ms TTL=128
Risposta da 127.0.0.1: byte=32 durata<1ms TTL=128
Risposta da 127.0.0.1: byte=32 durata<1ms TTL=128
Risposta da 127.0.0.1: byte=32 durata<1ms TTL=128

Statistiche Ping per 127.0.0.1:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 0ms, Massimo =  0ms, Medio =  0ms
```

2) **ping all'indirizzo IP del gateway** (192.168.1.1 nell'esempio) (FIGURA 12):

FIGURA 12 Ping all'indirizzo del gateway

```
C:\Users>ping 192.168.1.1

Esecuzione di Ping 192.168.1.1 con 32 byte di dati:
Risposta da 192.168.1.1: byte=32 durata=1ms TTL=64
Risposta da 192.168.1.1: byte=32 durata=2ms TTL=64
Risposta da 192.168.1.1: byte=32 durata=2ms TTL=64
Risposta da 192.168.1.1: byte=32 durata=1ms TTL=64

Statistiche Ping per 192.168.1.1:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 1ms, Massimo =  2ms, Medio =  1ms
```

Notare come il TTL sia stato modificato.

3) **ping a un indirizzo pubblico**. È possibile usare anche l'hostname (*www.garr.it*) anziché l'IP (193.206.158.22) (FIGURA 13):

FIGURA 13 Ping verso un host remoto

```
C:\Users>ping www.garr.it

Esecuzione di Ping www.garr.it [193.206.158.22] con 32 byte di dati:
Risposta da 193.206.158.22: byte=32 durata=20ms TTL=56
Risposta da 193.206.158.22: byte=32 durata=20ms TTL=56
Risposta da 193.206.158.22: byte=32 durata=21ms TTL=56
Risposta da 193.206.158.22: byte=32 durata=19ms TTL=56

Statistiche Ping per 193.206.158.22:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 19ms, Massimo =  21ms, Medio =  20ms
```

Notare come il TTL sia ancora diminuito.

Se funziona il ping verso un indirizzo pubblico di Internet, vuol dire che funzionano anche i precedenti ping.

Vediamo ora cosa succede se facciamo un ping a un indirizzo inesistente o per qualche motivo irraggiungibile (FIGURA 14).

```
C:\Users>ping 192.168.1.125

Esecuzione di Ping 192.168.1.125 con 32 byte di dati:
Risposta da 192.168.1.11: Host di destinazione non raggiungibile.
Risposta da 192.168.1.11: Host di destinazione non raggiungibile.
Risposta da 192.168.1.11: Host di destinazione non raggiungibile.
Richiesta scaduta.

Statistiche Ping per 192.168.1.125:
    Pacchetti: Trasmessi = 4, Ricevuti = 3,
    Persi = 1 (25% persi),
```

FIGURA 14 Ping a un host non raggiungibile

Ci viene comunicato che la destinazione è irraggiungibile attraverso il servizio Destination Unreachable e che per l'ultimo pacchetto la richiesta è scaduta (tempo predefinito per la risposta: 1 secondo) attraverso il messaggio ICMP Time Exceeded.
Se si teme che la latenza della risposta sia maggiore di un secondo, è possibile utilizzare l'opzione –w nel comando ping per aumentare il valore di timeout. Per consentire per esempio risposte entro 5 secondi, utilizzare ping –w 5000.

5.2 Il comando traceroute o tracert

Un altro comando utile per verificare problemi di connettività è **traceroute** (su Windows è chiamato **tracert**).
Il comando traceroute, seguito da un indirizzo IP o da un hostname destinazione, visualizza le serie di router IP utilizzati per il recapito di pacchetti dal proprio computer alla destinazione e il tempo impiegato per ciascun passaggio (**hop**). Se i pacchetti non vengono recapitati alla destinazione finale, il comando traceroute visualizza l'ultimo router che ha inoltrato correttamente i pacchetti.
Occorre tener presente che, anche se l'ultimo router è visualizzato correttamente, non significa che la connessione sarà stabilita con successo, perché in ogni istante potrebbero insorgere problemi lungo il tragitto.
Quindi, per esempio, prima di garantire servizi real time, come una video conferenza, è consigliabile tracciare l'indirizzo verificando che i tempi di risposta siano accettabili: se la rete è già ai limiti della congestione, difficilmente riuscirà a trasmettere uno stream video-audio con un flusso uniforme.
In generale maggiore è il numero di salti, più è probabile incontrare colli di bottiglia, motivo per cui i siti più visitati mettono a disposizione dei #mirror.
Facendo un po' di prove e cambiando gli indirizzi finali, ci si accorge che i primi hop sono sempre gli stessi perché, qualunque sia la destinazione, partendo dal proprio host i pacchetti dovranno sempre transitare dal gateway e dal proxy, se presenti, e in ogni caso dagli host del provider.

Il meccanismo del comando traceroute è il seguente:
1. traceroute inizia inviando al primo router un TTL con valore uguale a 1;
2. il valore viene decrementato a 0 dal router e il pacchetto eliminato;
3. il mittente invia quindi un altro pacchetto con TTL pari a 2, così dopo il decremento sarà inoltrato al secondo hop ed eliminato;
4. si prosegue così fino al raggiungimento del destinatario.

#techwords

Mirror
I siti più visitati usano delle copie speculari (mirror, specchio in inglese) del web server su altri computer così da essere accessibile anche da altre fonti, garantendo prestazioni ottimali.

#prendinota

Su Windows oltre ai comandi **ipconfig**, **arp**, **ping** e **tracert**, ci sono altri comandi che vengono installati automaticamente quando si installa TCP/IP. I più utili sono:
- **hostname**: non ha parametri e visualizza il nome del computer usato dal DNS;
- **route**: consente di vedere e modificare le tabelle di routing della rete.

In questo modo è facile individuare l'eventuale punto in cui si è interrotto il tragitto del pacchetto.
Ogni router segnala al mittente l'avvenuta eliminazione con un messaggio Time Excedeed in Transit, nel quale è indicato l'orario corrente. Il tempo di transito è quindi calcolato come differenza tra l'orario di invio del pacchetto e quello di ritorno.

Il traceroute risolve gli indirizzi IP dei router in nomi e questo comporta un certo rallentamento.
In Windows se si scrive: **tracert -d** , i nodi attraversati sono visualizzati più velocemente perché tracert non tenta di risolvere i nomi dei router rilevati nel percorso.

Esistono dei tool in commercio che estendono le funzionalità di tracert con un'interfaccia grafica che mostra l'ubicazione degli hop su una mappa anziché semplicemente elencarli. Sono inoltre forniti di grafici statistici e danno informazioni aggiuntive sui tempi di percorrenza dei pacchetti. Tra le applicazioni più interessanti segnaliamo NeoTrace e VisualRoute, disponibili in versioni trial.
Il sito *www.visualroute.it* mette gratuitamente a disposizione un servizio di tracciamento direttamente via web. Basta inserire un hostname o un IP nell'apposita casella e attendere il risultato.

esercizio

➜ PROBLEMA

Trovare qual è il percorso seguito da un messaggio inviato dal proprio computer al sito web di Garr: *www.garr.it*.
Ripetere la prova più volte e confrontare i risultati ottenuti in termini di nodi attraversati e tempo di percorrenza.

➜ ANALISI DEL PROBLEMA

Per ottenere le informazioni richieste si usa il comando traceroute:
- Windows: dal Prompt dei Comandi digitare **tracert**;
- Linux: da terminale digitare **traceroute** (il programma potrebbe non essere installato di default nella distribuzione Linux usata).

Sia Windows sia Linux impostano alcuni **parametri di default**, modificabili con le opzioni previste dal comando.

Windows:
- 30 è il massimo numero di hop da seguire per arrivare a destinazione;
- 4000 ms è il massimo tempo di attesa per ricevere il messaggio ICMP Time Exceeded o ICMP Echo Reply; scaduto questo tempo verrà visualizzato un asterisco (vedi esempio nella Figura 15). Questo evento si verifica solitamente quando si attraversa un router che non restituisce messaggi ICMP Time Exceeded quando TTL = 0, risultando così *invisibile* al comando tracert.

Linux:
- 60 byte per IPv4 e 80 byte per IPv6 è la lunghezza totale del pacchetto di prova che viene inviato al router;
- 30 è il massimo numero di hop da seguire per arrivare a destinazione;
- 3 è il numero di pacchetti che sono inviati ogni volta a un router (con stesso TTL = n);

- 5000 ms è il massimo tempo di attesa per ricevere il messaggio ICMP Time Exceeded oppure ICMP Echo Reply; scaduto questo tempo verrà visualizzato un asterisco.

➜ SVOLGIMENTO

Supponiamo di lavorare con un computer Windows, dal Prompt dei comandi digitare:

```
tracert www.garr.it
```

```
C:\Users>tracert www.garr.it

Traccia instradamento verso www.garr.it [193.206.158.22]
su un massimo di 30 punti di passaggio:

  1     2 ms     1 ms     1 ms  vodafone.station [192.168.1.1]
  2     6 ms     7 ms     7 ms  net-37-119-7-1.cust.vodafonedsl.it [37.119.7.1]
  3    10 ms    14 ms    10 ms  185.210.48.52
  4     9 ms    10 ms    10 ms  185.210.48.53
  5    10 ms     9 ms    10 ms  garr.mix-it.net [217.29.66.39]
  6    20 ms    19 ms    19 ms  rx2-mi2-rx2-rm2.rm2.garr.net [90.147.80.18]
  7    30 ms    19 ms    20 ms  rx2-rm2-rx1-rm2.rm2.garr.net [90.147.81.49]
  8    20 ms    19 ms    20 ms  rx1-rm2-ru-dir-l1.rm2.garr.net [193.206.138.210]

  9    19 ms    19 ms    20 ms  eventi.dir.garr.it [193.206.158.22]

Traccia completata.
```

FIGURA 15
Esecuzione comando tracert

Il risultato che si ottiene sarà simile a quello mostrato nella **FIGURA 15**.
Il primo nodo visualizzato è quello dell'host che ha inviato il messaggio.
Dopo 9 hop il pacchetto spedito è giunto a destinazione.

5.3 Il comando pathping di Windows

Su Windows è disponibile anche il comando **pathping** che fa uso sia di ping sia di tracert, infatti identifica i router che si trovano lungo il percorso del pacchetto fino alla destinazione, come tracert, e invia periodicamente ping a tutti i router per elaborare statistiche sulla base del numero di risposte ricevute da ciascuno di essi. In questo modo, per ogni router, si hanno informazioni sui pacchetti persi (Lost) e si può così individuare se un router è in sovraccarico o se una sottorete ha problemi di congestione.
La sintassi è:

```
pathping [/n] [/h <maximumhops>] [/g <hostlist>] [/p <Period>] [/q <numqueries>]
[/w <timeout>] [/i <IPaddress>] [/4 <IPv4>] [/6 <IPv6>]<targetname>
```

Lo svantaggio nell'usare questo comando è il tempo di attesa per ricevere le statistiche: 25 secondi per hop, cioè per ciascun salto che fa il pacchetto passando da un router al successivo.

FISSA LE CONOSCENZE

- A che cosa serve il comando ping?
- Che cosa succede in caso di ping su un indirizzo inesistente o irraggiungibile?
- A che cosa serve il comando traceroute?
- Come funziona il tracert di Windows?

6 PACKET TRACER: CONFIGURARE UNA RETE IPv6

CONFIGURAZIONE DELLE INTERFACCE CON INDIRIZZI IPv6

In questa esercitazione di laboratorio vedremo, con il simulatore Packet Tracer, come configurare gli indirizzi IPv6 su host e router.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare uno scenario con due reti IPv6 i cui host siano in grado di comunicare tra loro.

➜ ANALISI DEL PROBLEMA

Esistono due modalità di mettere in comunicazione gli host di reti IPv6. La prima, detta **Link Local**, consente il collegamento tra host IPv6 della stessa rete. La seconda, detta **Global Unicast**, consente il collegamento tra host IPv6 di reti diverse.

➜ SVOLGIMENTO

Nella FIGURA 16 è mostrato un semplice scenario con due LAN collegate da un router.

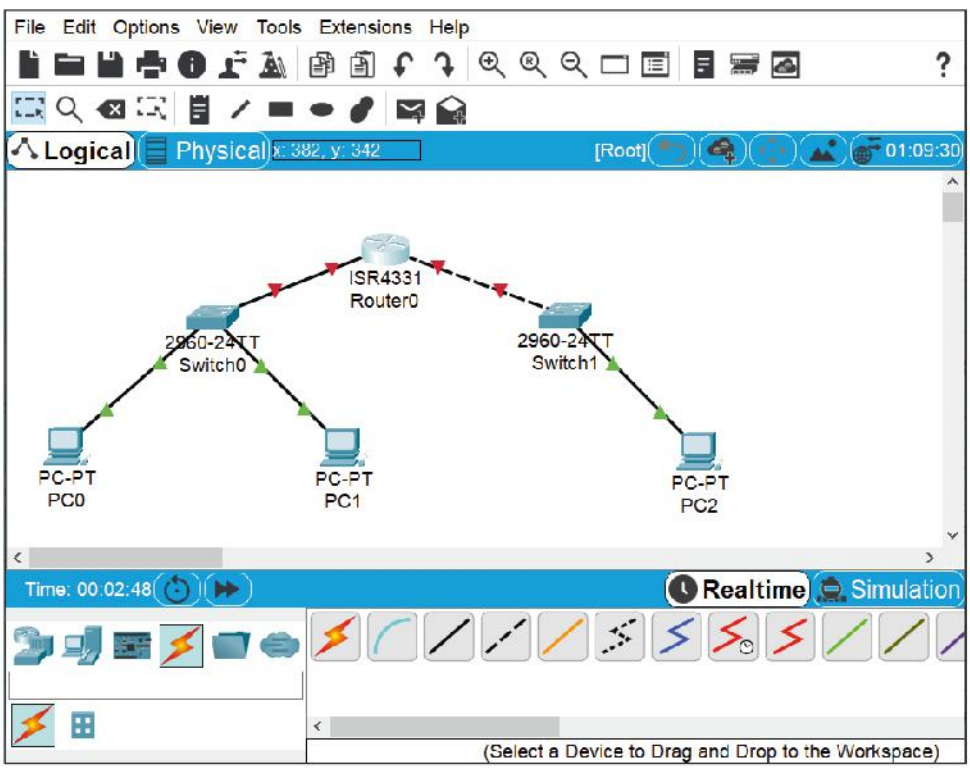


FIGURA 16 Scenario con due LAN collegate da un router

#prendinota

La maggior parte dei protocolli di livello 2 usano uno dei 3 range di numerazione regolati dall'IEEE e progettati per essere globalmente unici: **MAC-48**, **EUI-48** e **EUI-64**. EUI sta per **Extended Unique Identifier**.

Configurazione in Link Local

Non essendo stato ancora assegnato alcun indirizzo IP (né IPv4, né IPv6), gli host avranno solo l'indirizzo MAC. Partendo dall'indirizzo MAC viene creato di default il **Link Local Address IPv6** (passando attraverso l'EUI-64) che consente la comunicazione in locale tra gli host appartenenti alla stessa rete.

Vediamo come avviene la creazione del Link Local Address IPv6 per il PC0 (FIGURA 17):

1) il **MAC** (48 bit cioè 12 cifre esadecimali) è costituito da due parti: le prime 6 cifre esadecimali rappresentano l'OUI (Organization Unique Identifier) che individuano l'azienda produttrice della scheda, nel nostro esempio **00D0-D3**; le altre 6 cifre individuano il numero seriale della scheda, nel nostro esempio **AC-DB14**. Tra le due parti viene inserito **FFFE** ottenendo:

   00D0.D3FF.FEAC.DB14

   a questo punto viene **invertito il 7° bit**:
   – esadecimale 00D0 → binario 0000.0000.1101.0000
   – esadecimale 02D0 → binario 0000.0010.1101.0000

   Il risultato è l'**EUI-64**:

   02D0.D3FF.FEAC.DB14

2) all'EUI-64 vanno aggiunti altri 64 bit per arrivare ai 128 bit richiesti dall'IPv6.
   I 64 bit standard aggiunti sono FE80.0000.0000.0000 e quindi il Link Local Address IPv6 diventa:

   FE80::2D0:D3FF:FEAC:DB14

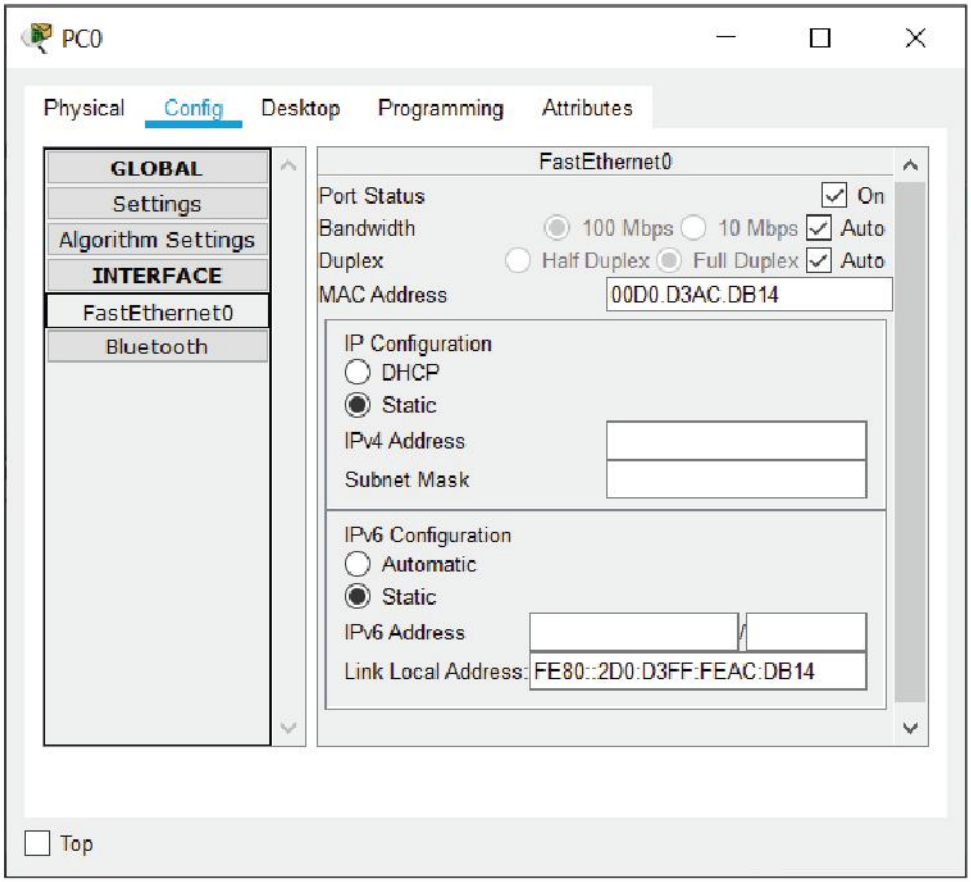


FIGURA 17 MAC address e Link Local Address IPv6 di PC0

Lo stesso automatismo creerà un IPv6 locale anche sugli altri PC. L'assegnamento automatico del Link Local Address IPv6 fa sì che i link sulle interfacce dei PC siano subito attive (triangolino verde nella Figura 16) e che quindi sia possibile effettuare un ping con successo tra PC0 e PC1 che sono nella stessa rete.

La FIGURA 18 mostra il ping da PC0 a PC1 che ha Link Local Address IPv6 uguale a FE80::290:2BFF:FE03:6EB9 (visibile dalla scheda Config di PC1 selezionando l'interfaccia FastEthernet0 o mediante il comando ipv6config dal prompt di PC1).

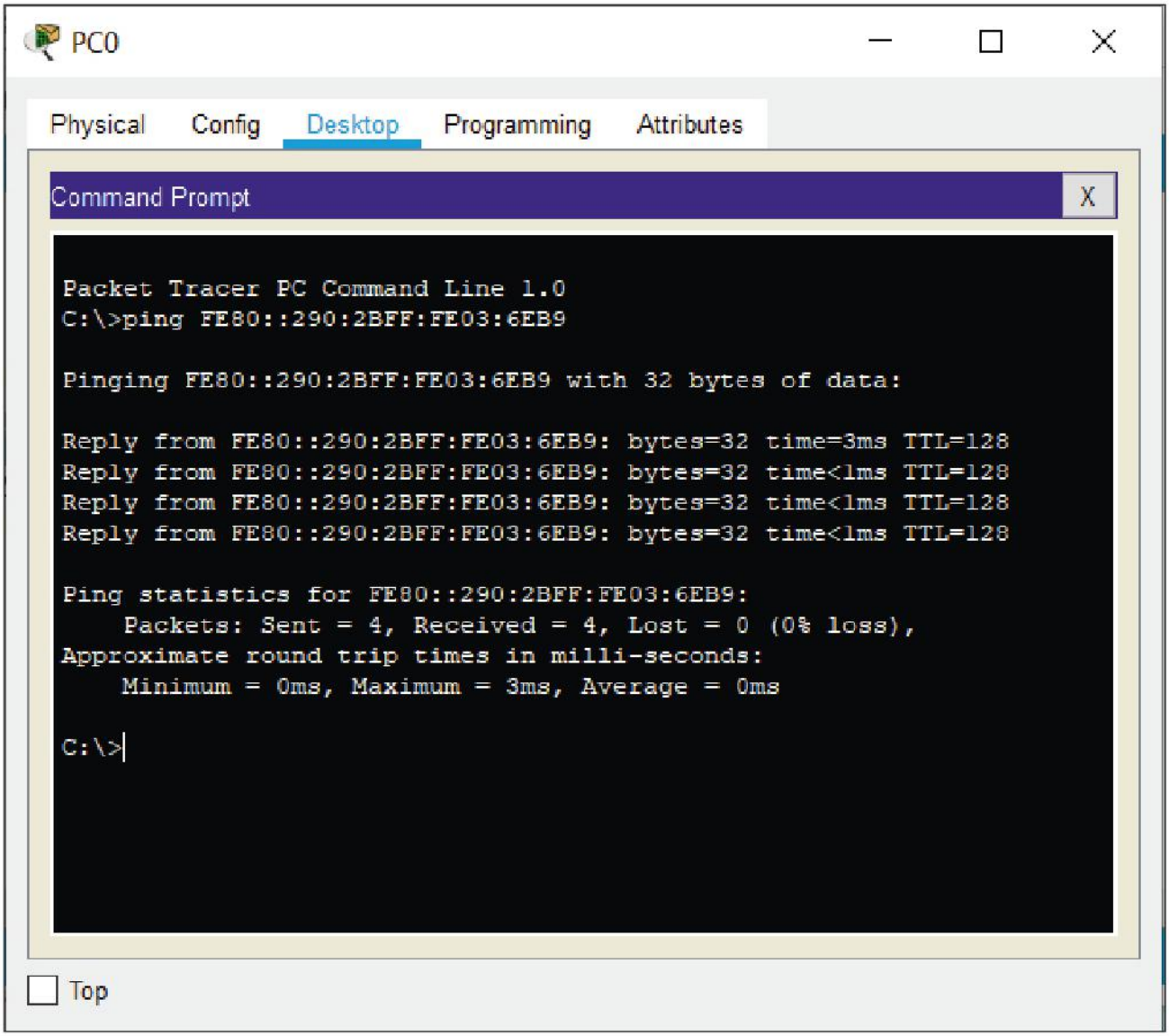


FIGURA 18 Ping da PC0 a PC1

Per quanto riguarda il router gateway invece la configurazione del Link Local Address IPv6 la facciamo esplicitamente su entrambe le interfacce GigabitEthernet usando la CLI.

Per l'interfaccia GigabitEthernet0/0/0 useremo i comandi:

```
Router # configure terminal
Router (config) # ipv6 unicast-routing
Router (config) # interface GigabitEthernet 0/0/0
Router (config-if) # ipv6 address FE80::1 link-local
Router (config-if) # no shutdown
```

mentre per l'interfaccia GigabitEthernet0/0/1 useremo i comandi:

```
Router # configure terminal
Router (config) # ipv6 unicast-routing
```

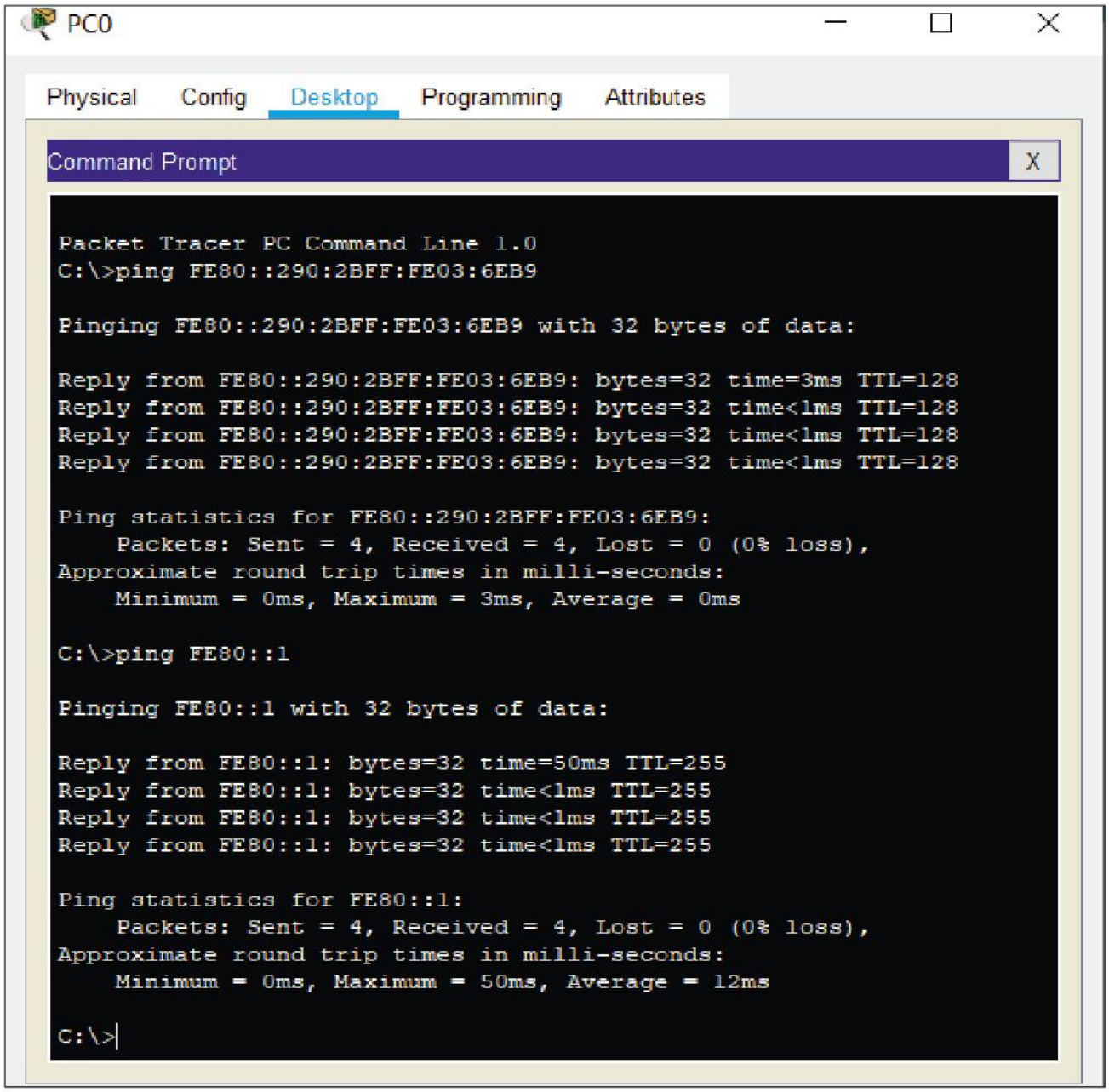
```
Router (config) # interface GigabitEthernet 0/0/1
Router (config-if) # ipv6 address FE80::1 link-local
Router (config-if) # no shutdown
```

Notare come sia possibile assegnare alle due interfacce lo stesso IPv6 (**FE80::1** dove le prime 4 cifre sono standard e l'ultima è tipica dei gateway) essendo tali indirizzi locali.

L'esecuzione dei comandi provocherà l'immediata attivazione delle interfacce del router che diventeranno verdi.

La **FIGURA 19** mostra il ping da PC0 al router che ha Link Local Address IPv6 pari a FE80::1.

FIGURA 19 Ping da PC0 al router



Configurazione in Global Unicast

La configurazione Link Local non consente il routing verso le altre reti, occorre passare alla configurazione Global Unicast per far sì che PC0 e PC1 possano comunicare con PC2.

Assegniamo quindi attraverso la CLI gli indirizzi globali alle due interfacce GigabitEthernet del router:

– GigabitEthernet0/0/0 → 2001:DB8:ACAD:**A**::1/64

– GigabitEthernet0/0/1 → 2001:DB8:ACAD:**B**::1/64

dove:

– 2001:DB8 è lo standard;

– ACAD sta a ricordare l'Academy CISCO;

– **A** e **B** identificano rispettivamente la rete a sinistra e quella a destra di Figura 16;

– l'1 finale è tipico dei router gateway;

– /64 sono i bit di prefix.

Per l'interfaccia GigabitEthernet0/0/0 useremo i comandi:

```
Router # configure terminal
Router (config) # interface GigabitEthernet 0/0/0
Router (config-if) # ipv6 address 2001:DB8:ACAD:A::1/64
Router (config-if) # no shutdown
```

Per l'interfaccia GigabitEthernet0/0/1 useremo i comandi:

```
Router # configure terminal
Router (config) # interface GigabitEthernet 0/0/1
Router (config-if) # ipv6 address 2001:DB8:ACAD:B::1/64
Router (config-if) # no shutdown
```

Agli host si potrebbero assegnare manualmente indirizzi IPv6 in sequenza:
2001:DB8:ACAD:**A::2**/64 per PC0 della rete A
2001:DB8:ACAD:**A::3**/64 per PC1 della rete A
2001:DB8:ACAD:**B::2**/64 per PC2 della rete B

Esiste però la possibilità di usare una forma di DHCP (detta **SAC**, State Address Configuration) che consente di assegnare automaticamente gli indirizzi IPv6 agli host.

Per fare questo è sufficiente dalla scheda **Desktop** di ogni PC selezionare **IP Configuration** e impostare **Automatic** nell'IPv6 Configuration (FIGURA 20).

Identico risultato si può ottenere col comando:

```
Router (config-if) # ipv6 address autoconfig
```

L'indirizzo IPv6 di ogni host viene generato automaticamente con lo stesso meccanismo che abbiamo visto in precedenza, cioè creando prima l'EUI partendo dal MAC address e aggiungendo i primi 64 bit del prefix di rete.

Questa tecnica consente anche di assegnare automaticamente il defaul gateway (cioè l'indirizzo dell'interfaccia del router, che è ancora il Link Local **FE80::1**) come mostrato in Figura 20.

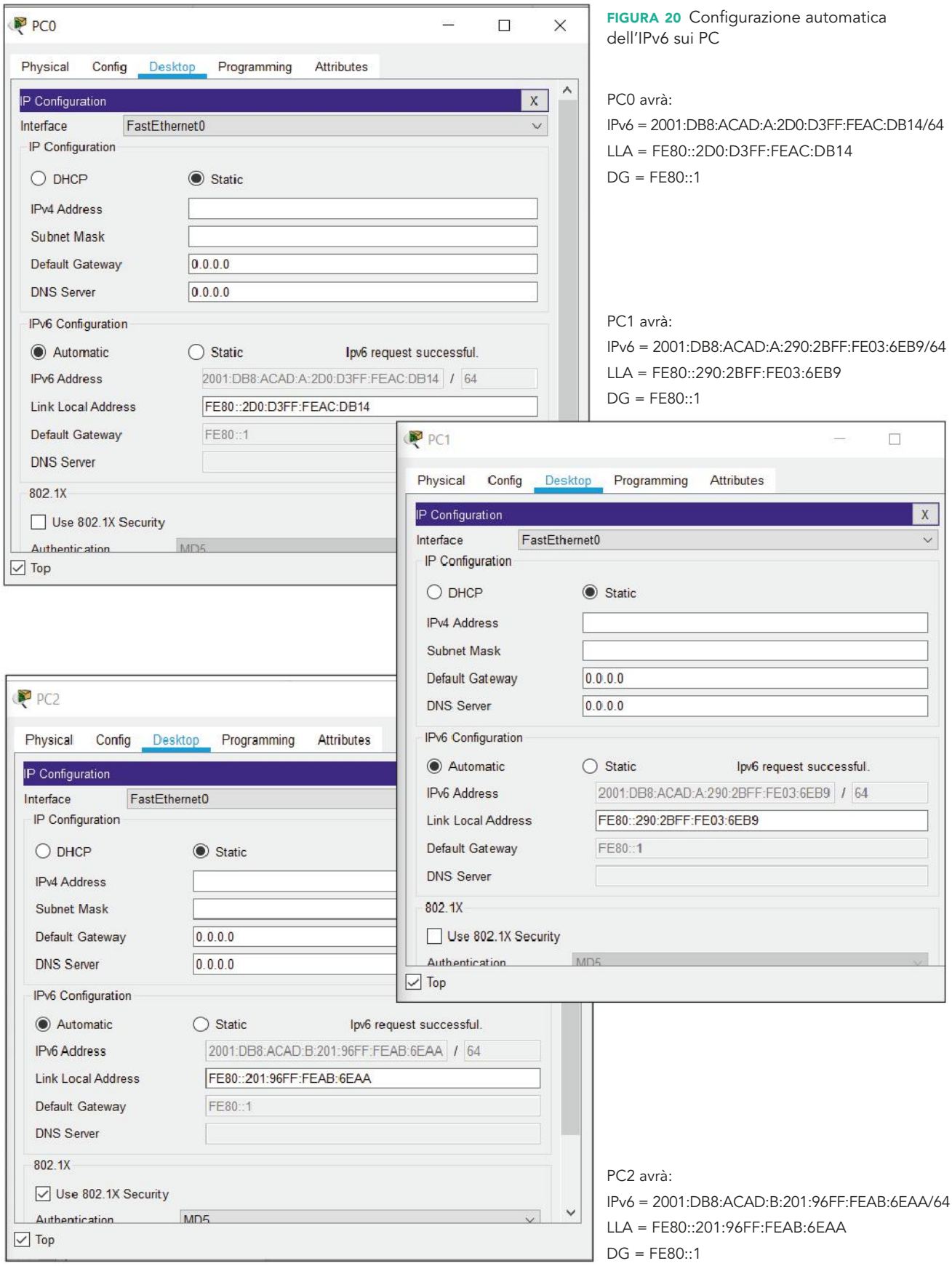


FIGURA 20 Configurazione automatica dell'IPv6 sui PC

PC0 avrà:
IPv6 = 2001:DB8:ACAD:A:2D0:D3FF:FEAC:DB14/64
LLA = FE80::2D0:D3FF:FEAC:DB14
DG = FE80::1

PC1 avrà:
IPv6 = 2001:DB8:ACAD:A:290:2BFF:FE03:6EB9/64
LLA = FE80::290:2BFF:FE03:6EB9
DG = FE80::1

PC2 avrà:
IPv6 = 2001:DB8:ACAD:B:201:96FF:FEAB:6EAA/64
LLA = FE80::201:96FF:FEAB:6EAA
DG = FE80::1

Possiamo anche controllare il riepilogo dei parametri assegnati dal Command Prompt dei PC mediante il comando **ipv6config** come mostrato in FIGURA 21 per PC0.

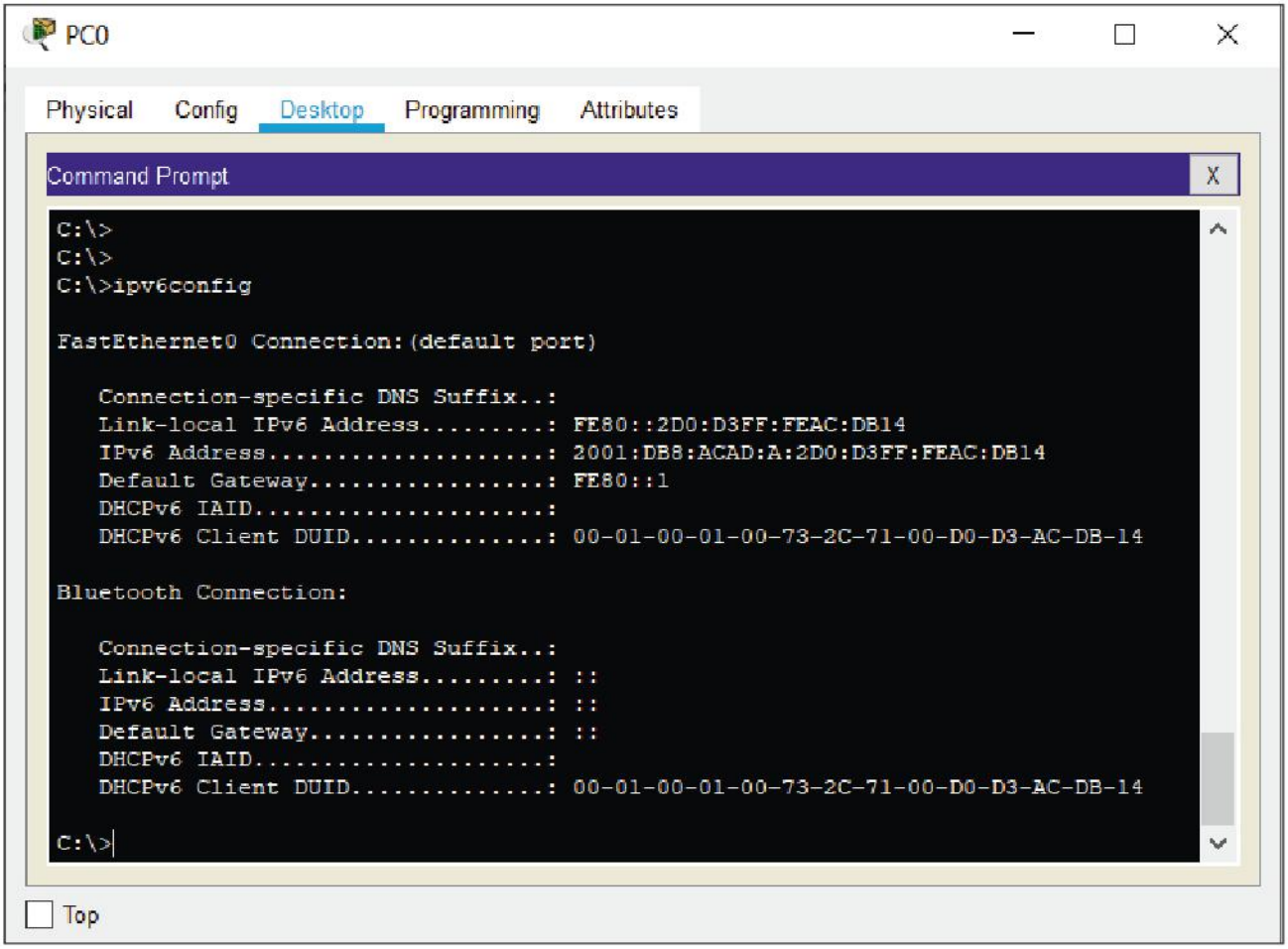


FIGURA 21 Parametri di configurazione di PC0

Fatto tutto ciò, possiamo effettuare il ping tra gli host delle due reti.

La FIGURA 22 mostra il risultato del ping tra PC0 della rete A e PC2 della rete B che ha Global Unicast IPv6 pari a FE80::201:96FF:FEAB:6EAA (visibile dalla scheda Config di PC2 selezionando l'interfaccia FastEthernet0 o mediante il comando ipv6config dal prompt di PC2).

IN ENGLISH PLEASE

DHCPv4 uses the MAC address and an optional Client ID to identify the client for purposes of assigning an address. Each time the same client arrives on the network, it gets the same address, if possible.
DHCPv6 uses basically the same scheme, but makes the Client ID mandatory and imposes structure on it. The Client ID in DHCPv6 consists of two parts: a DHCP Unique Identifier (**DUID**) and an Identity Association Identifier (**IAID**). The DUID identifies the client system (rather than just an interface, as in DHCPv4), and the IAID identifies the interface on that system.
As described in RFC 3315, an identity association is the means used for a server and a client to identify, group, and manage a set of related IPv6 addresses. A client must associate at least one distinct IA with each of its network interfaces, and then uses the assigned IAs to obtain configuration information from a server for that interface. However DUID+IAID is used if no Client ID is configured.

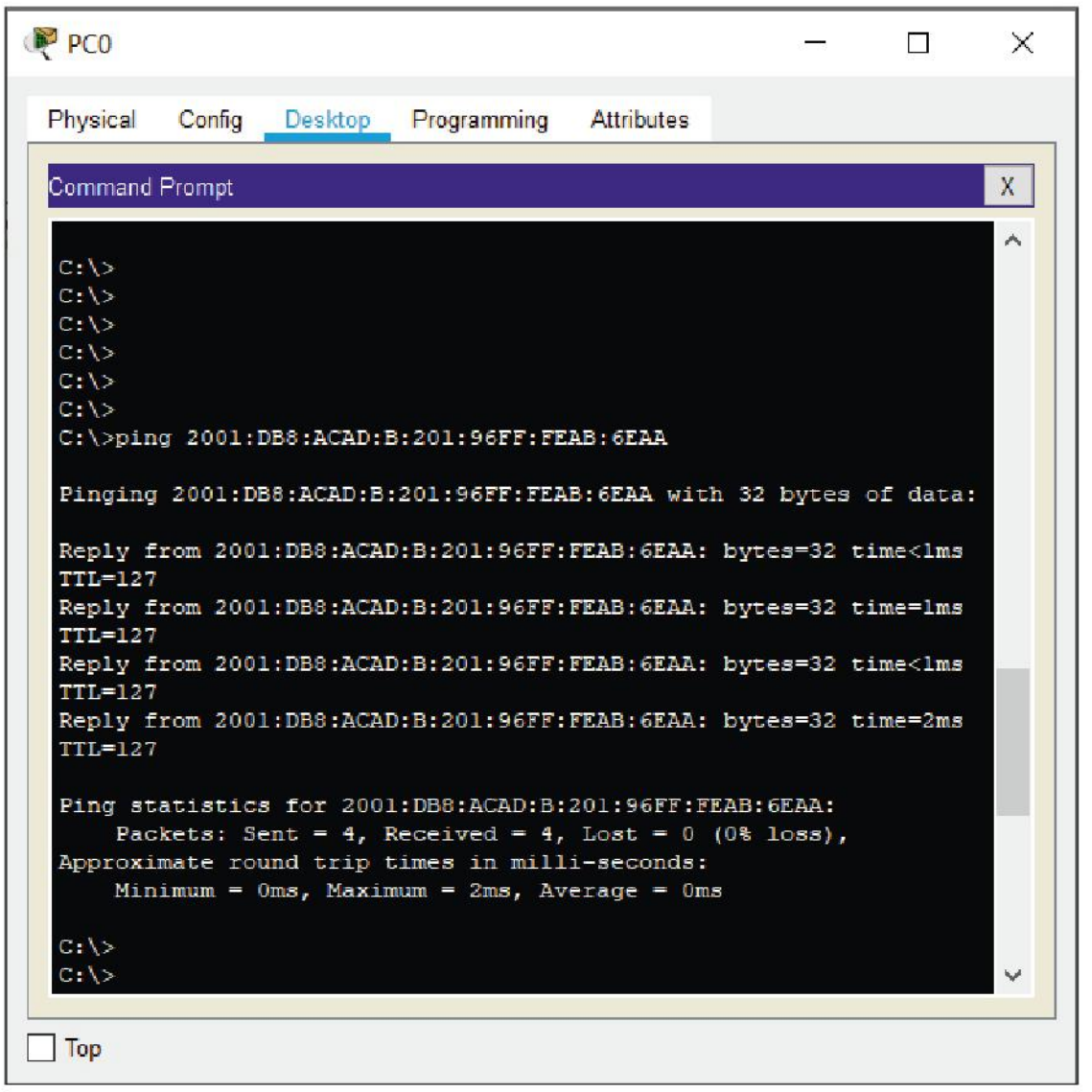


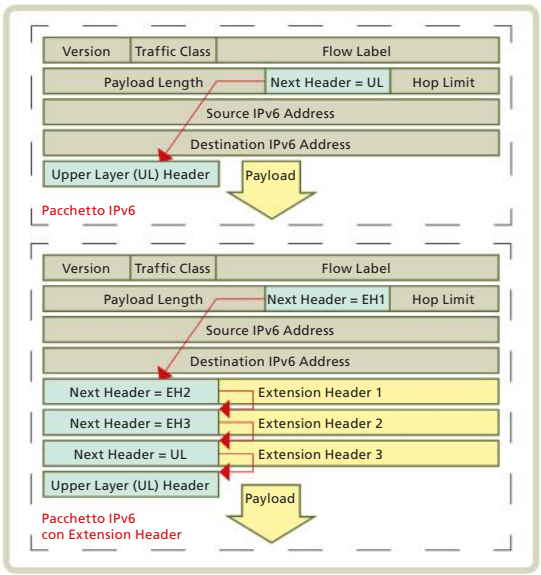
FIGURA 22 Ping da PC0 a PC2

Siamo quindi riusciti a collegare gli host delle due reti mediante il Global Unicast sulle interfacce del router e settando Automatic sulle interfacce dei PC.

FISSA LE CONOSCENZE

- Perché è possibile assegnare alle due interfacce del router lo stesso IPv6?
- Come avviene la creazione del Link Local Address IPv6?
- Quale comando dal prompt consente di sapere l'IPv6 di un PC?
- Che differenza c'è tra le modalità Link Local e Global Unicast?
- In che modo è possibile assegnare automaticamente gli indirizzi IPv6 agli host?

1 L'evoluzione del protocollo IP: IPv6

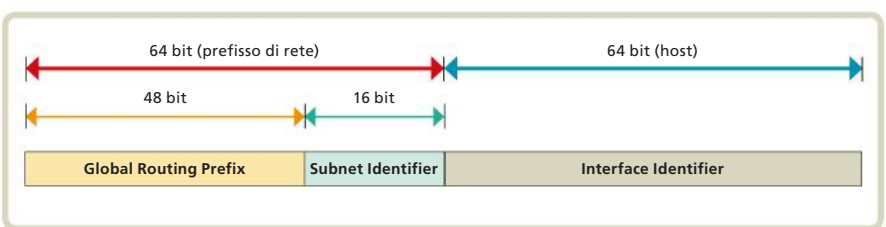


Nonostante le tecniche messe in campo per sopperire al limitato spazio degli indirizzi IPv4, la diffusione della rete Internet ha richiesto la progettazione di una nuova versione di IP che garantisse un numero di indirizzi sufficiente a soddisfare tutte le richieste. La nuova versione è stata definita dalle RFC 1883 e 1887 con il nome di **IPv6**, la cui caratteristica è quella di quadruplicare lo spazio degli indirizzi, portando il formato di un indirizzo da 4 a 16 byte (128 bit). IPv6 è compatibile con tutta la suite di protocolli TCP/IP, ma non con IPv4. Una tecnica per risolvere questo problema è il **tunneling**: quando i pacchetti IPv6 attraversano una rete IPv4, vengono incapsulati in un pacchetto IPv4. L'indirizzo di destinazione del pacchetto IPv4 è l'indirizzo di un sistema su cui sono attivi entrambi i protocolli. Un'importante innovazione di IPv6 è una gestione più efficiente delle opzioni presenti nell'header IP, mediante il meccanismo degli **extension header**.

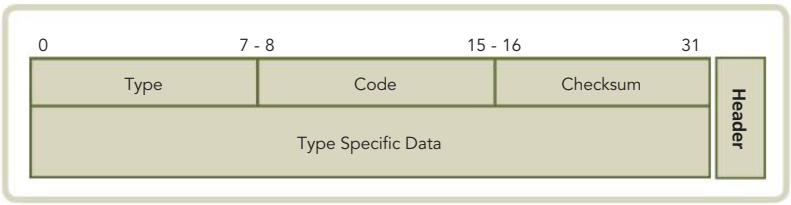
2 Gli indirizzi IPv6

In IPv6 gli indirizzi sono distinti in 3 categorie fondamentali: **unicast** (assegnato a un'interfaccia di rete singola), **anycast** (assegnato a più interfacce di rete distinte; un pacchetto inviato a un indirizzo anycast raggiunge *un'unica interfaccia* di rete, la prima che risponde) e **multicast** (simile a anycast, però un pacchetto inviato a un indirizzo multicast raggiunge *tutte le interfacce* di rete cui questo indirizzo è stato assegnato). Gli indirizzi anycast possono essere usati solo dai router.
Un indirizzo IPv6 è formato da 128 bit suddivisi in **8 gruppi di 16 bit** (detti hextet), separati dal carattere ":" e i 16 bit di ciascun hextet sono rappresentati con 4 cifre esadecimali.
La struttura di un generico indirizzo IPv6 di tipo unicast è la seguente:

- i primi 3 hextet sono detti **Global Routing Prefix** e sono assegnati dagli ISP ai client IPv6;
- il quarto hextet è il campo **subnet**, se non viene usato può essere lasciato a 0 o a 1 o a qualunque altro valore;
- gli ultimi 4 hextet formano l'**Interface ID** (IID), che corrisponde al campo HostID di IPv4.



3 Il monitoring della rete con il protocollo ICMP



Il protocollo **ICMP** (Internet Control Message Protocol) per il monitoraggio della rete è utilizzato prevalentemente dai router o dagli host destinatari per segnalare agli host mittenti eventuali insuccessi nell'instradamento dei pacchetti. In pratica ICMP consente ai router di scambiarsi informazioni di servizio (messaggi router-to-router) e di tenere sotto controllo le modalità con cui gli host generano i datagram IP, inviando loro messaggi per rallentare o dirottare altrove un flusso di pacchetti (messaggi router-to-host). Per quanto riguarda gli host, invece, ICMP consente loro di scambiarsi informazioni di servizio (messaggi host-to-host) e di richiedere ai router informazioni utili sul funzionamento e la topologia della rete (messaggi host-to-router). Due comandi molto utili per monitorare la rete sono **ping** e **traceroute** (o tracert).
La definizione del nuovo protocollo IPv6, ha comportato la specifica di una nuova versione di ICMP, **ICMPv6**, che ha mantenuto lo stesso formato dei messaggi, ma completamente rivisto i tipi e i codici, definendone di nuovi.

4 Indirizzi fisici e indirizzi IP: il protocollo ARP

A ogni nodo di una rete TCP/IP viene assegnato un indirizzo logico IP che lo identifica univocamente sulla rete. Affinché però due host possano comunicare tra loro questo non basta. Bisogna che le rispettive schede di rete siano capaci di localizzarsi reciprocamente. Occorre cioè che l'indirizzo fisico MAC del destinatario sia noto al mittente. In IPv4 chi si occupa di mappare un indirizzo IP noto nel corrispondente indirizzo MAC sconosciuto, è il protocollo **ARP** (Address Resolution Protocol). Un'implementazione TCP/IP utilizza di norma una cache ARP dove ogni host mantiene e aggiorna una tabella con tutte le coppie IP-MAC a lui note.
Il comando "arp -a", dato su un computer Windows, visualizza la sua cache ARP.
Il protocollo ARP si usa nelle reti IPv4, mentre in quelle IPv6 è stato sostituito dal protocollo **Neighbor Discovery** (RFC 4861), che utilizza i nuovi messaggi definiti in ICMPv6.

```
C:\Users>arp -a

Interfaccia: 192.168.1.11 --- 0xa
  Indirizzo Internet     Indirizzo fisico      Tipo
  192.168.1.1            64-59-f8-2b-64-e0     dinamico
  192.168.1.255          ff-ff-ff-ff-ff-ff     statico
  224.0.0.2              01-00-5e-00-00-02     statico
  224.0.0.22             01-00-5e-00-00-16     statico
  224.0.0.251            01-00-5e-00-00-fb     statico
  224.0.0.252            01-00-5e-00-00-fc     statico
  239.255.255.250        01-00-5e-7f-ff-fa     statico
  255.255.255.255        ff-ff-ff-ff-ff-ff     statico
```

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Il protocollo IPv6 è stato standardizzato da IEEE. V F
2. IPv6 e IPv4 sono perfettamente compatibili. V F
3. Gli indirizzi IPv6 si scrivono in esadecimale. V F
4. Gli indirizzi IPv6 si scrivono con le lettere minuscole. V F
5. Il campo IID dell'indirizzo IPv6 corrisponde al campo NetID dell'indirizzo IPv4. V F
6. L'ICMP fornisce un meccanismo di monitoraggio della rete. V F
7. Un messaggio ICMP è incapsulato all'interno del pacchetto IP. V F
8. Il protocollo ICMP è usato per mappare indirizzi IP e MAC. V F
9. Il protocollo ARP risolve un indirizzo IP in un indirizzo MAC. V F
10. Il protocollo ARP non è usato nelle reti IPv6. V F
11. Due comandi molto utili per monitorare la rete sono ping e tracert. V F
12. Il comando "ping" serve a verificare il percorso che seguono i pacchetti in rete. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **Il protocollo IPv6 è stato standardizzato per:**
   - A migliorare la bandwidth nella rete Internet
   - B risolvere il problema della carenza di indirizzi IP
   - C aumentare la velocità di trasmissione dei dati
   - D semplificare il routing

2. **Il protocollo IPv6 e il protocollo IPv4:**
   - A non possono essere usati insieme nella stessa rete
   - B possono coesistere nella stessa rete grazie alla tecnica del tunneling
   - C possono coesistere nella stessa rete senza alcun problema
   - D possono coesistere nella stessa rete grazie alla tecnica del supernetting

3. **In IPv6 quale indirizzo "di gruppo" si usa per inviare un pacchetto a tutte le interfacce che ne fanno parte?**
   - A Broadcast
   - B Anycast
   - C Unicast
   - D Multicast

4. **Nell'header IPv6 il campo TTL presente nell'header IPv4:**
   - A è stato ridenominato "Traffic Class"
   - B è stato eliminato
   - C è stato sostituito dal campo "Flow Label"
   - D è stato ridenominato "Hop Limit"

5. **La lunghezza in bit di un indirizzo IPv6 è:**
   - A 256
   - B 128
   - C 64
   - D 32

6. **Gli indirizzi IPv6 Link local possono essere usati:**
   - A su LAN interconnesse della stessa azienda
   - B nello stesso dominio di broadcast
   - C solo nello stesso dominio di collisione
   - D su Internet

7. **Un utente finale può richiedere un indirizzo IPv6 a:**
   - A IANA
   - B IETF
   - C RIR
   - D ISP

8. **Un host che riceve un pacchetto ICMP, come fa a capire di quale messaggio si tratta?**
   - A Leggendo il campo Type dell'header
   - B Leggendo il campo Code dell'header
   - C Leggendo il campo Checksum dell'header
   - D Leggendo l'header del pacchetto IP che lo contiene

9. **Chi può inviare messaggi ICMP?**
   - A sia i router sia gli host di destinazione
   - B solo gli host di destinazione
   - C solo i router

10. **In una rete IPv4, quando un router rileva che il TTL di un pacchetto è diventato 0, deve inviare al mittente un messaggio ICMP; quale?**

A Echo Request
B Time Exceeded
C Timestamp Request
D Timestamp Reply

11. **Il protocollo ARP realizza:**

A l'handshaking tra due host
B il limited broadcast
C il ping tra host e router
D la mappatura tra indirizzi IP e MAC

12. **Un router riceve un pacchetto IP da inviare nella rete locale, non conosce l'indirizzo MAC dell'host di destinazione, come fa a ottenerlo?**

A Consulta la sua cache ARP
B Invia un pacchetto ARP Request
C Consulta la sua Routing Table
D Esegue il comando ping

13. **Quando due host vogliono comunicare attraverso la rete, come fa il mittente a determinare l'indirizzo hardware (o fisico) dell'host destinatario?**

A RARP Request
B ARP Request
C Show Hardware Address Request
D Proxy Hardware Address Request

14. **Qual è l'indirizzo MAC del destinatario inserito in un pacchetto ARP Request?**

A 00-00-00-00-00-00
B FF-FF-FF-0A-0A-0A
C 0A-0A-0A-FF-FF-FF
D FF-FF-FF-FF-FF-FF

15. **Un host risponde a una richiesta ARP con il pacchetto ARP Reply, in cui il Target Hardware Address è:**

A l'indirizzo MAC dell'host da cui ha ricevuto la richiesta
B l'indirizzo IP dell'host da cui ha ricevuto la richiesta
C il suo indirizzo MAC
D il suo indirizzo IP

16. **Per verificare se il router gateway della rete locale è funzionante, un amministratore di rete può usare il comando:**

A icmp
B arp -a
C ipconfig
D ping

17. **Il comando ping utilizza i messaggi di tipo:**

A Address Mask Request e Address Mask Reply
B Information Request e Information Reply
C Echo Request ed Echo Reply
D Domain Name Request e Domain Name Reply

18. **Sui sistemi Windows, come si chiama l'utility che permette di verificare insieme la raggiungibilità di un host e il percorso seguito da un pacchetto?**

A traceroute
B ping
C pathping
D tracert

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Descrivi le principali differenza tra IPv4 e IPv6.**
2. LEZIONE 1 **Come viene gestita la frammentazione in IPv6?**
3. LEZIONE 1 **A che cosa serve e come funziona il meccanismo degli extension header?**
4. LEZIONE 2 **Descrivi le modalità unicast, anycast e multicast di IPv6.**
5. LEZIONE 2 **Descrivi formato e struttura di un indirizzo IPv6.**
6. LEZIONE 2 **Spiega la procedura di allocazione/assegnazione degli indirizzi IPv6.**
7. LEZIONE 3 **Quali sono i servizi offerti dal protocollo ICMP?**
8. LEZIONE 3 **Spiega come è cambiato ICMP nelle reti IPv6.**
9. LEZIONE 4 **Descrivi la risoluzione di un indirizzo IP con il protocollo ARP.**
10. LEZIONE 4 **Spiega la vulnerabilità ARP cache poisoning.**

IN ENGLISH, PLEASE

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

IP evolution and network monitoring

In the early 1990s, growth of the Internet led to various problems with IPv4, the main one is the running out of addresses.
IETF published in 1995 the first version of a new Internet Protocol, called IPv6, which used longer addresses and improved IP security and other functionalities. IPv6 has seen a number of enhancements and updates since then.
Nowadays most Operating System and most network devices support IPv6. The IPv6 address notation consists in eight groups of four hexadecimal digits with the groups separated by colons (there are some rules to abbreviate this notation).
The new version of IP has also affected other protocols, such as Internet Control Message Protocol (ICMP), for which a new version has been specified: ICMPv6.
ICMP provides a messaging capability for reporting different types of errors that can occur while processing datagrams.
Another control protocol is the Address Resolution Protocol (ARP), used for discovering the MAC address associated with a given IPv4 address. This mapping is stored in a table, called ARP cache, on each host of the local network. In IPv6 networks, ARP has been replaced by the new protocol Neighbor Discovery Protocol (NDP) and its extensions such as Secure Neighbor Discovery.

EXERCISES

Use the appropriate number to match words and meanings.

...	Anycast	1	Utility used to test the reachability of a host
...	Hextet	2	It identifies a single network interface
...	Traceroute	3	Technique used by an attacker to intercept data on a network
...	Unicast	4	A sixteen-bit aggregation
...	Ping	5	It is assigned to a group of interfaces, a packet is delivered to all ones
...	RARP	6	It is assigned to a group of interfaces, but a packet is delivered to just one
...	Multicast	7	Utility used to discover paths
...	ARP poisoning	8	It has been made obsolete by the DHCP

GLOSSARY

Allocate: provide Internet Registries with address space for the purpose of subsequent distribution.
Assign: provide ISP or End User with address space for specific use within the Internet infrastructure they operate. Assignments are not to be sub-assigned to other parties.
ARP cache: a table containing matched sets of MAC and IP addresses.
Hop: occurs when a packet moves from a network segment to the next, "**hop count**" refers to the number of routers the data passes through between source and destination.
Leading zero: any 0 digit that comes before the first nonzero digit in a string of numbers in positional notation. It is used to shorten an IPv6 address.
Link local address: address that is valid only for communications within a broadcast domain. IPv6 requires a link local address on every network interface on which the IPv6 protocol is enabled.
Local Internet Registry (LIR): an IR that primarily assigns address space to the users of the network services that it provides. LIRs are generally ISPs whose customers are primarily End Users.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper lavorare con gli indirizzi IPv6.
- Saper operare sui computer e sui router per configurare gli indirizzi Link local IPv6 delle interfacce di rete.
- Utilizzare la terminologia tecnica anche in lingua inglese.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Imparare a usare il protocollo IPv6 per la comunicazione in rete.
- Esporre i risultati del proprio lavoro alla classe.

tempi

- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione: 30 minuti.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Un computer con connessione a Internet.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Le reti IPv6

Al giorno d'oggi sono ormai molti i dispositivi e le applicazioni che supportano IPv6. Un amministratore di rete deve saper lavorare con gli indirizzi IPv6, individuandone le componenti e sapendo gestire le varie tipologie di indirizzi. Nelle Lezioni 1 e 2 di questa unità abbiamo descritto il protocollo IPv6, il nuovo header e la nuova struttura degli indirizzi. In particolare, per svolgere gli esercizi seguenti, è necessario conoscere i vari tipi di indirizzi IPv6, che si trovano riassunti nella Tabella 1 della Lezione 2.

1. Creare una tabella in cui indicare in una colonna i seguenti indirizzi IPv6 e nella colonna a fianco a quale tipologia appartengono:

- 2001:08d4:1:f55a::fe55:6789:c6a3
- ff02::4
- ::1
- fd00:32:f:1::3ab:7d38:76cc
- 2033:32:1:3:f5:a33d:d45a:6a34
- fe80::85a3:d47f:174d
- ff00::
- ff00::ac8:3d66:8ff4:5d4

2. Applicare le regole di compattazione degli indirizzi IPv6 per abbreviare o estendere i seguenti indirizzi:

- 2002:0ab0:0200:0003:0000:05dd:43de:07b5
- fe80:0000:0000:0001:0000:52ac:005a:8a43
- fe80::6134:b66f:a72c:5d32
- ff00::
- 2001:005a:0003:52ac:0000:5a1e:aa32:f628

3. Creare il Link local address IPv6 dell'interfaccia di un host che ha MAC address EC-8E-B5-17-FC-0F, configurando l'Interface ID (IID) con la tecnica EUI-64.

SVOLGIMENTO

1. Nella tabella seguente si riporta la tipologia di ciascun indirizzo IPv6:

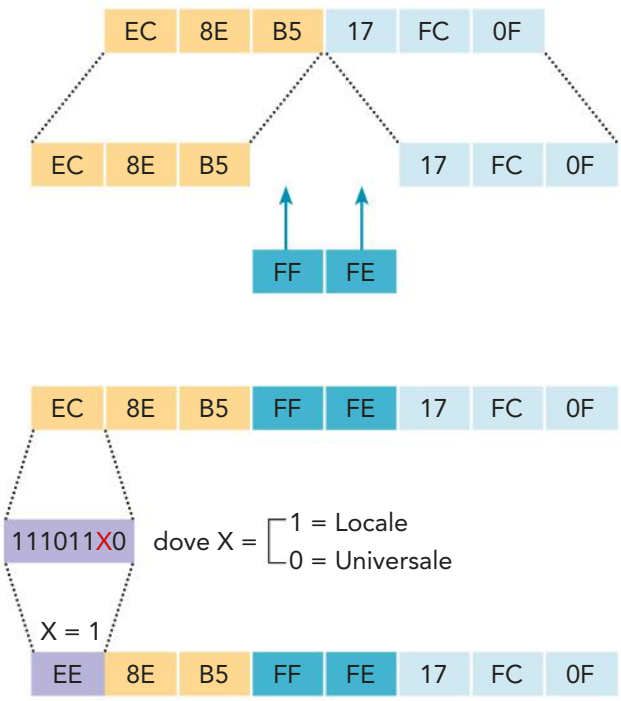
Indirizzo IPv6	Tipo di indirizzo
2001:08d4:1:f55a::fe55:6789:c6a3	Global Unicast address
ff02::4	Multicast address
::1	Loopback address
fd00:32:f:1::3ab:7d38:76cc	Unique Local address
2033:32:1:3:f5:a33d:d45a:6a34	Global Unicast address
fe80::85a3:d47f:174d	Link Local address
ff00::	Multicast address
ff00::ac8:3d66:8ff4:5d4	Multicast address

2. La tabella seguente mostra le notazioni richieste:

Indirizzo IPv6	Indirizzo abbreviato/non abbreviato
2002:0ab0:0200:0003:0000:05dd:43de:07b5	2002:ab0:200:3::5dd:43de:7b5
fe80:0000:0000:0001:0000:52ac:005a:8a43	fe80::1:0:52ac:5a:8a43
fe80::6134:b66f:a72c:5d32	fe80:0000:0000:0000:6134:b66f:a72c:5d32
ff00::	ff00:0000:0000:0000:0000:0000:0000:0000
2001:005a:0003:52ac:0000:5a1e:aa32:f628	2001:5a:3:52ac::5a1e:aa32:f628

3. Per ottenere l'IID dell'indirizzo IPv6 di tipo Link local, a partire dal MAC address dell'interfaccia, è necessario operare come spiegato nella Lezione 6, ossia dividendo a metà il MAC address e inserendo l'hextet **fffe**.
Il passo successivo consiste nel porre a 1 il settimo bit, a partire da sinistra, per indicare che questo identificatore è gestito localmente. Infatti, tutti gli indirizzi MAC hanno la parte OUI assegnata da IEEE che ne garantisce l'univocità a livello mondiale.
Il settimo bit viene detto **U/L bit (Universal/Local)** e indica se l'identificatore è gestito a livello globale (bit = 0) o locale (bit = 1). Essendo assegnato da IEEE, un indirizzo MAC avrà sempre U/L bit = 0.
Nel nostro caso, invece, l'identificatore è stato generato localmente e quindi non può essere definito universale, perciò si deve mettere **U/L bit = 1**, come indicato in figura.
Per creare l'indirizzo IPv6, ai 64 bit così ottenuti per l'IID, aggiungiamo il prefix standard ff80/10 e otteniamo il Link local address:

ff80::ee8e:b5ff:fe17:fc0f



A CASA

- Verifica su un computer con SO Windows la configurazione IPv6 delle interfacce di rete, nelle due modalità:
  – da interfaccia grafica: visualizzare la finestra delle proprietà dell'interfaccia di rete e selezionare la voce **Protocollo Internet Versione 6 (TCP/IPv6)** e cliccare su Proprietà;
  – nella finestra Prompt dei Comandi digitare il comando: **ipconfig/all**.
- Troverai un indirizzo Link local IPv6 con l'Interface ID generato in modo random. Sono presenti un indirizzo IPv6 Global Unicast, un indirizzo IPv6 Unique Local o un indirizzo del gateway IPv6?
- Analizza sul sito *https://stats.labs.apnic.net/ipv6* le statistiche sull'adozione di IPv6 nel mondo.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i risultati ottenuti.
- Riflettete sulle statistiche che riguardano l'adozione di IPv6 nel mondo e provate a ipotizzare se, e quanto tempo ci vorrà, si avrà un definitivo passaggio a IPv6 e non esisteranno più reti IPv4.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho capito come lavorare sul computer con gli indirizzi IPv6?	Ho avuto difficoltà nel trovare le informazioni. ☐	Con la guida del docente e la collaborazione dei compagni ho capito dove trovare i dati su IPv6. ☐	Ho trovato in autonomia i dati di IPv6 sul computer. ☐	Ho trovato facilmente le informazioni richieste su IPv6 e approfondito le varie tipologie di indirizzi. ☐
Ho interpretato i dati dell'analisi statistica sull'adozione di IPv6 nel mondo ed elaborato un'opinione sul futuro delle reti IP?	Ho avuto difficoltà nel capire i dati presenti sul sito indicato. ☐	Con la guida del docente e la collaborazione dei compagni ho capito come interpretare i dati statistici, ma non sono riuscito a farne un'elaborazione personale. ☐	Ho compreso in modo autonomo l'analisi statistica presentata sul sito, ma ho avuto un po' di difficoltà a ragionare sull'adozione completa di IPv6. ☐	Ho approfondito l'analisi statistica con altri dati recuperati da altri siti. Ho elaborato e sostenuto la mia opinione sul futuro delle reti IP. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ

5 INSTRADAMENTO E INTERCONNESSIONE DI RETI GEOGRAFICHE

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

DDI+ Didattica Digitale Integrata Plus

conoscenze

Comprendere le funzioni svolte dal Network Layer per garantire il percorso migliore ai pacchetti che transitano in rete.

Conoscere gli algoritmi e i protocolli di routing.

Conoscere le reti multiprotocollo (MPLS).

abilità

Saper scegliere i protocolli che individuano il percorso migliore per raggiungere la destinazione.

Essere in grado di verificare se la funzione di routing è correttamente configurata.

Saper usare semplici strumenti di diagnostica della rete.

competenze

Applicare un algoritmo di routing in una rete.

Gestire il corretto funzionamento dell'Internetworking.

FLIPPED CLASSROOM

A casa

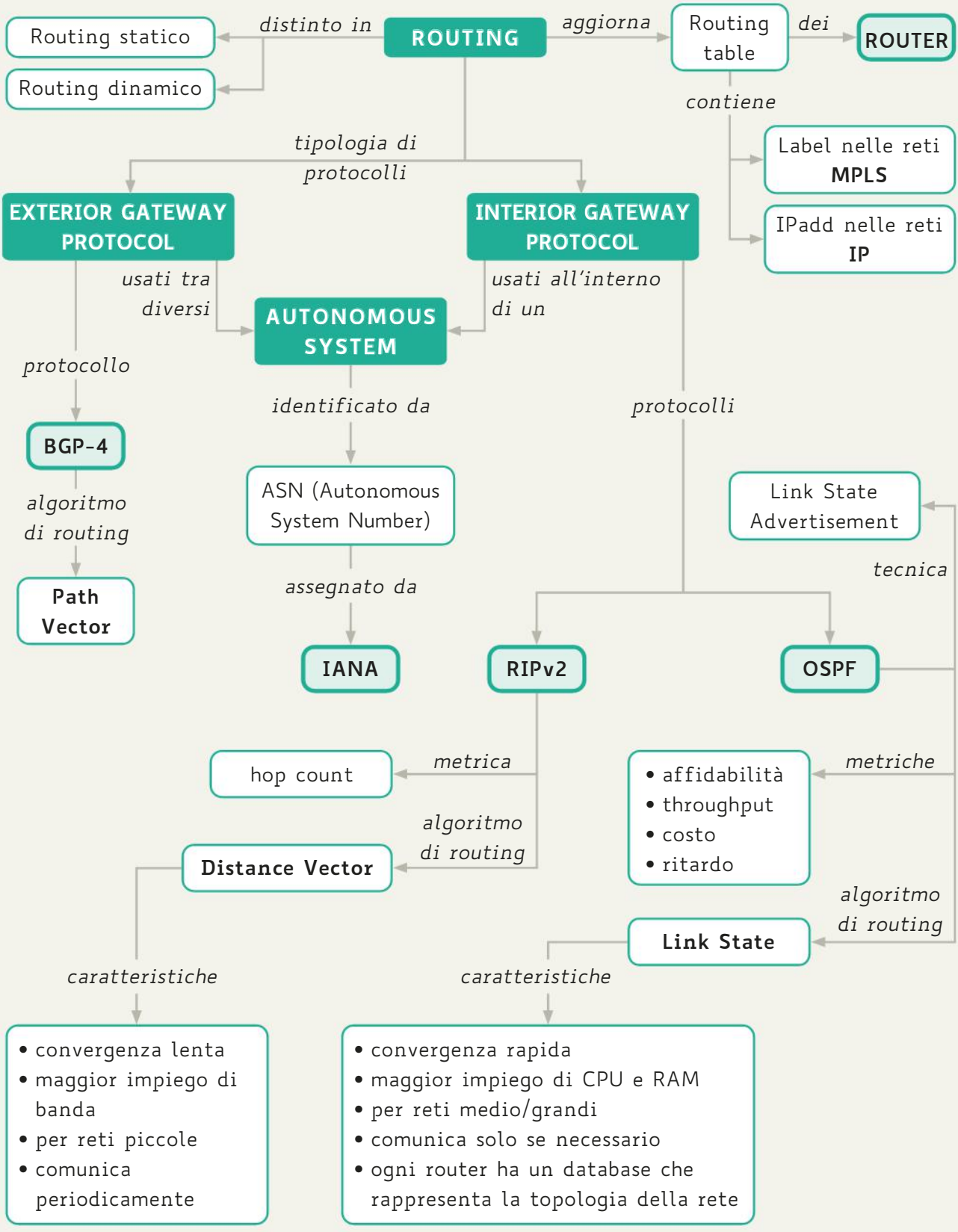
- Leggi la Lezione 1 di questa unità;
- ragiona sull'analogia esistente tra i pacchetti che sono trasferiti in una rete di telecomunicazioni e le automobili che transitano in una rete stradale e autostradale;
- trasferisci la tua analisi in una tabella in cui elenchi le problematiche dell'instradamento riferite ai pacchetti in rete e ai veicoli sulla strada.

In classe

- Confrontate i risultati descritti nelle tabelle;
- discutete i motivi che spiegano le eventuali differenze, al fine di comprendere meglio il funzionamento dell'instradamento nelle reti.

MAPPA CONCETTUALE

Mappa modificabile



1 PROBLEMATICA E SCENARI

1.1 Il routing e la routing table

Nell'Unità 3 abbiamo visto come il **routing** (instradamento) sia una funzione fondamentale del livello Network dell'architettura TCP/IP. Tale funzione viene svolta dal **router** (*intermediate system*), che, per poter ottimizzare il percorso dei pacchetti da instradare, deve conoscere ed eventualmente aggiornare una serie di informazioni:

- l'indirizzo del destinatario;
- i router adiacenti;
- l'insieme dei possibili percorsi (**route**) verso tutte le reti remote;
- il percorso migliore per ciascuna rete remota;
- il modo di mantenere e di verificare le informazioni necessarie per il routing.

Solo basandosi su queste informazioni il router può dare l'avvio al processo di **forwarding** per stabilire verso quale linea inviare il pacchetto.

Il router deve quindi costruire, nella propria memoria, una tabella di instradamento, detta **routing table**, che gli permetta di memorizzare i dati indispensabili per individuare il percorso ottimale verso le reti remote da raggiungere.

IN ENGLISH PLEASE

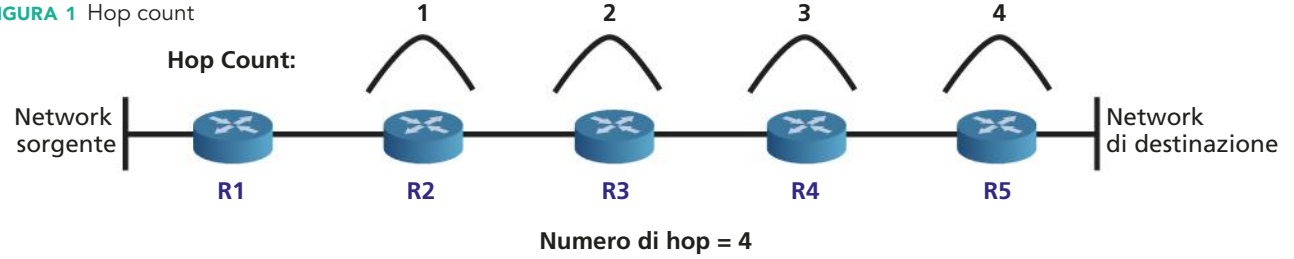
Routing is the task of finding a path from a sender to a desired destination. In the IP "Internet model" this reduces primarily to a matter of finding a series of routers between the source and destination networks.

La **routing table** è una lista di tutte le reti che il router può raggiungere insieme a informazioni sulle modalità di instradamento. Quando il router effettua il forwarding di un pacchetto, ricerca nella routing table l'indirizzo di rete corrispondente all'indirizzo IP di destinazione.

Il formato della tabella di routing varia a seconda del protocollo di routing utilizzato. In generale ogni riga (detta *entry*) della tabella presenta 4 campi:

- **network address**: contenente l'indirizzo IP di ciascuna rete raggiungibile;
- **next hop**: l'indirizzo del router successivo nel percorso verso il destinatario;
- **interface**: interfaccia del router a cui deve essere inoltrato il pacchetto per raggiungere il next hop (un router può avere più interfacce di rete);
- **metric**: è una misura utilizzata dal router per decidere quale percorso inserire nella routing table quando ci sono più alternative. Alla metrica si associa il concetto intuitivo di "costo": nella routing table si inserisce il percorso a costo minore (vedremo che il protocollo di routing OSPF sostituisce il termine "metric" proprio con "cost"). La metrica più semplice è quella del numero di hop necessari per raggiungere la destinazione (hop count, mostrato nella FIGURA 1).
  Altre metriche considerano parametri quali: l'ampiezza di banda, il numero di pacchetti persi o errati, il ritardo nella trasmissione.

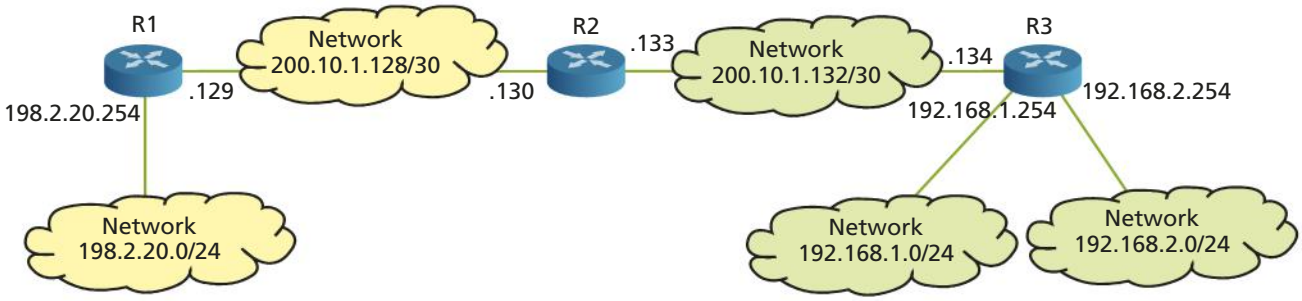
FIGURA 1 Hop count



esercizio

➜ PROBLEMA

Data la seguente topologia di rete definire la **routing table del router R1**, determinando il costo del percorso in base al numero di hop.



➜ ANALISI DEL PROBLEMA

Le reti 198.2.20.0, 192.168.1.0 e 192.168.2.0 sono reti di classe C, il prefix /24 specifica una subnet mask 255.255.255.0, quindi in nessuna delle 3 reti sono state create delle subnet.

Anche le reti 200.10.1.128 e 200.10.1.132 sono reti di classe C, il prefix /30 specifica una subnet mask 255.255.255.252, quindi è stato effettuato un subnetting: per ogni subnet sono disponibili solo 2 indirizzi di host, da assegnare alle interfacce dei 2 router.

Dalla figura ricaviamo gli indirizzi delle interfacce dei 3 router, che riportiamo nella tabella a fianco, dove:
FE = FastEthernet e GE = GigabitEthernet.

Router	Interfaccia	Indirizzo IP	Subnet mask
R1	FE0	198.2.20.254	255.255.255.0
R1	GE1	200.10.1.129	255.255.255.252
R2	GE0	200.10.1.130	255.255.255. 252
R2	GE1	200.10.1.133	255.255.255.252
R3	GE0	200.10.1.134	255.255.255.252
R3	FE0	192.168.1.254	255.255.255.0
R3	FE1	192.168.2.254	255.255.255.0

➜ SVOLGIMENTO

Il router R1 raggiunge con **instradamento diretto** le reti IP 198.2.20.0 e 200.10.1.128, che risultano quindi direttamente connesse con costo pari a 0 (zero hop). Il valore di next hop è l'indirizzo dell'interfaccia del router locale che sarà utilizzata per raggiungere quella destinazione. Per esempio: R1 raggiungerà tutte le destinazioni 198.2.20.0 attraverso la sua interfaccia FE0 con indirizzo 198.2.20.254; questo sarà il valore da inserire nel campo next hop.

Le reti IP non direttamente connesse sono quelle raggiungibili con l'**instradamento indiretto**, ossia i pacchetti da inviare a quelle reti devono essere trasmessi a un router che li farà proseguire verso quelle destinazioni. Per esempio: un host della rete 192.168.1.0 sarà raggiungibile da R1 con un instradamento indiretto attraverso il router R2, il valore di next hop sarà l'indirizzo IP dell'interfaccia di R2 direttamente connessa a R1, ossia 200.10.1.130.

La routing table del router R1 è illustrata nella tabella seguente.

Network address	Next hop	Interface	Cost (hop)
198.2.20.0/24	198.2.20.254	FE0	0
200.10.1.128/30	200.10.1.129	GE1	0
200.10.1.132/30	200.10.1.130	GE1	1
192.168.1.0/24	200.10.1.130	GE1	2
192.168.2.0/24	200.10.1.130	GE1	2

#prendinota

L'indirizzo del next hop deve sempre essere raggiungibile attraverso instradamento diretto dall'host IP in esame.

DEFAULT ROUTER

Il router deve estrarre dall'header IP del datagram l'indirizzo del destinatario e analizzare i bit dedicati all'indirizzo di rete. Nel caso più semplice in cui l'indirizzo di rete del destinatario sia presente nel campo network address della routing table, il pacchetto viene instradato verso la sua destinazione, in modo diretto o indiretto, come visto nell'esercizio precedente.
Può però succedere che il destinatario appartenga a una rete non presente nella tabella di routing. In questo caso il router inoltrerà il pacchetto verso un **router di default** che lo prenderà in carico col compito di farlo giungere a destinazione. Se nella routing table non è definita questa **default route**, il router invia al mittente un messaggio ICMP Destination Unreachable.

ROUTING STATICO E ROUTING DINAMICO

Il funzionamento di un router è caratterizzato dal modo in cui la tabella di routing viene creata:

- **manualmente**, se la tabella di routing è inserita dall'amministratore; in questo caso si parla di **routing statico** (utilizzabile per piccole reti);
- **automaticamente**, se il router costruisce da solo la tabella di routing in funzione delle informazioni ricevute attraverso i protocolli di routing; in quest'altro caso si parla di **routing dinamico**.

Il **routing statico** ha il vantaggio di non richiedere che i router si scambino informazioni per aggiornare i percorsi o eventualmente individuarne di nuovi, limitando così l'uso di banda. Inoltre essendo i percorsi già determinati una volta configurata la tabella, non è richiesto altro sforzo computazionale da parte del router stesso per calcolare i percorsi ottimali.
Il routing statico ha però l'inconveniente di richiedere sempre la riconfigurazione da parte dell'amministratore tutte le volte che si devono modificare le entry, sia in seguito a guasti, sia per inserire nuove route; inoltre tale metodo comincia a divenire abbastanza oneroso quando l'Internetwork contiene parecchi percorsi. Per questi motivi il routing statico viene usato quando il numero dei segmenti di LAN non è elevato.

> Il **routing dinamico** permette ai vari router di scambiarsi le informazioni necessarie a determinare i possibili percorsi per raggiungere destinazioni remote mediante dei protocolli, chiamati appunto **routing protocol**, che usano appropriati **algoritmi di routing**.

Il vantaggio di tale metodo è che richiede un minor controllo da parte dell'amministratore; per contro richiede un maggior uso di banda rispetto al routing statico perché, oltre al traffico relativo ai pacchetti, c'è un traffico relativo allo scambio delle informazioni indispensabili ai routing protocol. Inoltre, un altro notevole beneficio è dato dalla capacità di adattarsi automaticamente ai cambiamenti della topologia di rete: se si verifica un guasto lungo una connessione oppure ne viene attivata una nuova, gli aggiornamenti dei vari percorsi vengono automaticamente propagati a tutti i router.
Nella TABELLA 1 si mettono a confronto routing statico e dinamico sulla base delle principali caratteristiche del processo di routing.

TABELLA 1 Confronto tra routing statico e dinamico

Caratteristiche	Routing statico	Routing dinamico
Complessità di configurazione	Aumenta col crescere delle dimensioni della rete	È indipendente dalle dimensioni della rete
Conoscenze di amministrazione	Non sono richieste conoscenze specifiche	Sono richieste conoscenze avanzate
Cambiamenti nella topologia	Richiedono l'intervento dell'amministratore	La rete si adatta in automatico alla nuova topologia
Scalabilità	È adatto per semplici topologie di rete	È adatto per topologie semplici e complesse
Sicurezza	Più sicuro	Meno sicuro: invia in rete i pacchetti per l'aggiornamento delle route
Utilizzo delle risorse	Non richiede risorse aggiuntive	Utilizzo intenso di CPU e memoria del router e della bandwidth
Prevedibilità dei percorsi	Prevedibili: utilizza sempre la stessa route verso la stessa destinazione	Non prevedibili: utilizza route che dipendono dall'attuale topologia

1.2 Il problema della ricerca nella routing table

Un problema molto comune e assai studiato in letteratura è il cosiddetto **Routing Table Lookup Problem** (**#RTLP**), cioè il problema di dover decidere molto in fretta dove instradare i pacchetti per evitare rallentamenti e relative congestioni.
La difficoltà di RTLP risiede nel numero estremamente elevato di pacchetti che devono essere esaminati ogni secondo. Supponiamo di avere un canale da 1 Gbps, avremo che possono arrivare 1 milione di pacchetti da 1 kb in un secondo, quindi non è possibile dedicare più di 1 microsecondo a ciascun pacchetto per contenere il ritardo nell'ordine di un secondo. Il problema assume dimensioni ancora maggiori se si considera che un qualsiasi router ha più interfacce e che il tempo di accesso per una cache veloce è dell'ordine dei 50 nanosecondi: in queste condizioni, sono sufficienti pochi accessi alla memoria per esaurire il tempo disponibile per il routing. Dato che l'RTLP è un problema chiave per lo sviluppo della rete Internet, esso è stato affrontato intervenendo su più fattori con lo scopo di velocizzare al massimo il processo di routing: si sono studiate strutture dati per memorizzare la tabella di routing in maniera compatta ed efficiente, si sono ideati algoritmi per velocizzare la consultazione della tabella stessa, si è proposto di utilizzare sistemi paralleli allo scopo di suddividere il carico di lavoro tra più processori e altro ancora. Tuttavia, tutto questo non è ancora sufficiente: negli ultimi anni l'overhead legato alla gestione del traffico di pacchetti (packet processing) è diventato così critico per le prestazioni che si è cominciato a sviluppare hardware dedicato a tale gestione. In questo contesto, i network processor sono un promettente tentativo di ottenere elevate prestazioni mantenendo almeno parte della flessibilità dei microprocessori tradizionali e con essa la capacità di adattarsi a mutamenti degli algoritmi e dei protocolli di rete.

#techwords

L'**RTLP** si può ricondurre alla ricerca, nella tabella, del più lungo prefix corrispondente all'IP del destinatario (ricerca di prefisso di lunghezza massima). Per esempio, per i pacchetti indirizzati a 176.16.5.99, il router, tra 176.16.0.0/16 e 176.16.5.0/24, sceglierà quest'ultimo per l'inoltro, poiché ha la corrispondenza più lunga.

FISSA LE CONOSCENZE

- Che cos'è il forwarding?
- Descrivi la metrica per hop count.
- Spiega l'instradamento diretto e indiretto e il ruolo del default router.
- Descrivi il routing statico e dinamico e spiegane le differenze.
- Che cos'è il Routing Table Lookup Problem (RTLP)?

2 GLI ALGORITMI E I PROTOCOLLI DI ROUTING

2.1 Lo scopo di un protocollo di routing

Lo scopo di un protocollo di routing è quello di aggiornare dinamicamente le routing table. Per fare ciò, i router devono quindi condividere le informazioni sui percorsi (route) che ciascuno conosce. Questo scambio di dati è compiuto mediante pacchetti speciali chiamati **routing update**.
Prima di entrare nel dettaglio vediamo quali sono gli scopi che un routing protocol si deve prefiggere:

- **ottimalità**: deve essere in grado di fornire il percorso migliore o più veloce lungo l'Internetwork individuando percorsi alternativi, ciascuno con una velocità o livello di traffico diverso. Per esempio, un protocollo userà la banda e il conteggio dei salti (hop count), mentre un altro darà un peso maggiore alla banda;
- **imparzialità**: deve utilizzare tutte le linee disponibili per distribuire il traffico evitando le congestioni (occorre mediare tra ottimalità e imparzialità, spesso in conflitto);
- **flessibilità**: deve garantire capacità di adattarsi ai cambiamenti della topologia di rete;
- **convergenza veloce**: deve far sì che i cambiamenti all'interno dell'Internetwork si propaghino verso tutti i router nel minor tempo possibile;
- **robustezza**: deve essere in grado di funzionare anche nel caso di configurazioni non corrette e guasti di componenti;
- **semplicità**: deve essere semplice ed efficiente.

La gran parte dei protocolli che regolano il routing moderno utilizzano uno dei due seguenti algoritmi:
- Distance Vector Routing
- Link State Routing

Gli algoritmi di routing calcolano il percorso migliore o a costo minimo.

2.2 L'algoritmo di routing Distance Vector

#prendinota

La rete può essere considerata come un **grafo** in cui i router rappresentano i nodi e i canali di comunicazione gli archi pesati, che hanno cioè costi (pesi) diversi. Dunque la ricerca del percorso ottimale per i pacchetti si riconduce al problema del commesso viaggiatore che deve raggiungere la sua destinazione applicando strategie che minimizzino i costi e/o i tempi.

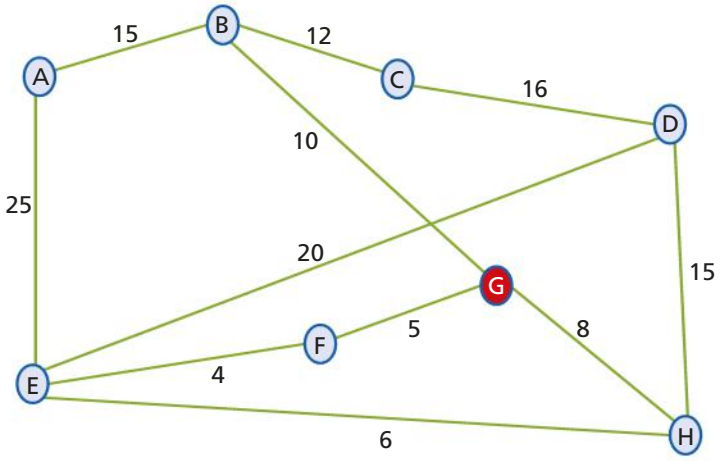
L'algoritmo Distance Vector si basa sull'algoritmo **Bellman-Ford** per calcolare il percorso migliore. Crea una tabella di routing costituita essenzialmente da due colonne: una contenente la **distanza** (il costo) stimata per raggiungere ogni nodo della rete e una che specifica l'**interfaccia** (la linea) da utilizzare. La distanza può essere calcolata secondo metriche diverse in base al protocollo in uso. Le righe della tabella saranno invece tante quanti sono i nodi della rete.
Essendo un algoritmo dinamico, le tabelle vengono aggiornate a intervalli di tempo prestabiliti.
Inizialmente ogni router invia ai router vicini (**neighbour**) un pacchetto di **ECHO** per calcolare la distanza che lo separa da ciascuno di essi e inserisce il valore nella tabella. Subito dopo i router vicini si scambiano un **vettore delle distanze**, cioè un array contenente le informazioni che ciascun router ha a disposizione riguardo i costi per raggiungere le varie destinazioni. A quel punto, ricevuti i vettori dai vicini, ciascun router aggiorna la propria tabella mediante un confronto tra i costi risultanti dalla

propria tabella e quelli ricevuti, modificando i valori laddove risultino inferiori e aggiornando le relative interfacce (linee d'uscita).

esercizio

➜ PROBLEMA

Si supponga di voler aggiornare la routing table del nodo G della rete magliata mostrata nella figura a fianco, dove la rete è rappresentata da un grafo con i nodi che rappresentano i router e i link le linee di collegamento. I numeri sui link indicano la distanza tra i due nodi.



➜ SVOLGIMENTO

G ha come nodi adiacenti B, F e H e quindi manderà innanzitutto i pacchetti di ECHO a questi 3 router.
Supponiamo che ottenga come risposte le seguenti distanze:
GB = 10; GF = 5; GH = 8.
G introdurrà questi valori nella sua tabella, come qui di seguito indicato:

Tabella delle distanze del router G		
Destinatario	**Distanza**	**Linea**
A		
B	10	B
C		
D		
E		
F	5	F
G	0	–
H	8	H

Dopo di che attenderà che dagli stessi 3 nodi gli arrivino i rispettivi vettori delle distanze, mostrati nella seguente tabella:

Vettori delle distanze ricevuti da G			
Destinatario	**B**	**F**	**H**
A	15	29	31
B	0	15	18
C	12	27	30
D	28	28	15
E	19	4	6
F	15	0	10
G	10	5	8
H	18	10	0

A questo punto G sarà in grado di costruirsi la tabella di routing selezionando le distanze più brevi per ogni destinazione.

Per esempio nello scegliere il percorso verso A valuterà le seguenti 3 possibilità legate alle 3 possibili sue linee d'uscita:

GB + BA = 10 + 15 = 25
GF + FA = 5 + 29 = 34
GH + HA = 8 + 31 = 39

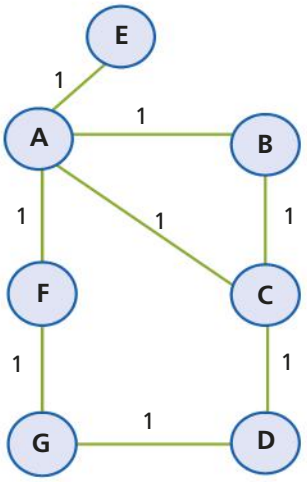
Ovviamente sceglierà quella col costo minore, cioè 25 attraverso il nodo B. Ripetendo questo procedimento per tutte le destinazioni della rete, G otterrà la tabella di routing riportata qui a fianco.

Naturalmente anche G a questo punto sarà in grado di inviare ai 3 router vicini il suo vettore dei costi, cioè la colonna Distanza della sua tabella di routing.

Routing table di G		
Destinatario	**Distanza**	**Linea**
A	25	B
B	10	B
C	22	B
D	23	H
E	9	F
F	5	F
G	0	-
H	8	H

esercizio

➜ PROBLEMA



Si consideri una piccola rete con 7 router e si supponga che i router utilizzino un algoritmo di tipo Distance Vector per costruire la routing table.

Valgono le seguenti condizioni:

- ogni link è a **costo 1** (metrica next hop), a un link inattivo (per esempio per via di un'interruzione di linea o perché è guasto il router a cui è connesso) è assegnato un costo infinito;
- ogni router costruisce un vettore che contiene i costi (in termini di distanza) per raggiungere gli altri router e invia questo vettore ai suoi "vicini" (ossia i router adiacenti).

Determinare il vettore delle distanze che ogni router si costruisce all'inizio e quello memorizzato alla fine dopo aver concluso il processo di scambio e confronto dei Distance Vector con gli altri router.
Infine scrivere la routing table per il **router B** e quella per il **router F**.

➜ SVOLGIMENTO

Mostriamo la tabella che riporta tutti i Distance Vector memorizzati nei router all'inizio:

Dati memorizzati nei router	Distanza per raggiungere i router						
	A	**B**	**C**	**D**	**E**	**F**	**G**
A	0	1	1	?	1	1	?
B	1	0	1	?	?	?	?
C	1	1	0	1	?	?	?
D	?	?	1	0	?	?	1
E	1	?	?	?	0	?	?
F	1	?	?	?	?	0	1
G	?	?	?	1	?	1	0

La procedura attivata è la seguente:

1) ogni router invia un messaggio al router adiacente (neighbour) con il suo Distance Vector. Per esempio A invia le sue informazioni ai router B, C, E e F;
2) i router che ricevono i Distance Vector aggiornano le proprie informazioni nel caso scoprano percorsi più brevi (cioè a costo minore) di quelli che hanno memorizzato nel loro Distance Vector. Per esempio: il router B scopre, dal Distance Vector che ha ricevuto dal router A, che il router E può essere raggiunto da A a costo 1. B sa già che può raggiungere A a costo 1, quindi memorizza che può raggiungere il router E a costo 2 passando per il router A;
3) dopo che ogni router ha scambiato i propri Distance Vector con i router vicini, tutti i router vengono a conoscenza dei percorsi più brevi per raggiungere tutti gli altri router della rete;
4) infine, oltre ad aggiornare i propri vettori delle distanze, i router mantengono traccia del router che ha consentito di conoscere il percorso migliore per raggiungere un altro router e inseriscono questa informazione nella propria routing table. Per esempio B sa che per raggiungere E deve usare il link che lo connette ad A.

Mostriamo la tabella che riporta tutti i Distance Vector memorizzati nei router alla fine.

Dati memorizzati nei router	Distanza per raggiungere i router						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Routing table del **router B**

Destinazione	Distanza	Next hop
A	1	A
B	0	-
C	1	C
D	2	C
E	2	A
F	2	A
G	3	A

Routing table del **router F**

Destinazione	Distanza	Next hop
A	1	A
B	2	A
C	2	A
D	2	G
E	2	A
F	0	-
G	1	G

I PRINCIPALI PROBLEMI DEL DISTANCE VECTOR

I due principali problemi che possono presentarsi con il Distance Vector sono:

- **routing loop**: si verifica quando un pacchetto è inoltrato su un percorso circolare senza mai giungere a destinazione; in questi casi il problema si risolve grazie al contatore TTL (Time To Live) il cui valore si riduce a ogni hop e quando arriva a 0 il pacchetto viene scartato;

- **count to infinity**: si verifica quando il costo per il raggiungimento di una destinazione viene progressivamente incrementato (normalmente avviene quando una destinazione non è più raggiungibile per via di un guasto di cui il mittente non è a conoscenza); in questi casi è il percorso che finisce con l'essere scartato visti i costi sempre crescenti.

Entrambi i problemi sono legati al fatto che il Distance Vector **non conosce la topologia della rete** e che la convergenza della rete (cioè la propagazione delle informazioni) può richiedere molto tempo.

LE MODIFICHE ALL'ALGORITMO DISTANCE VECTOR

Si può migliorare l'algoritmo Distance Vector, limitando così i due problemi, mediante modifiche all'algoritmo originale. Le più note varianti sono:

- **split horizon**: serve a prevenire il loop tra due nodi adiacenti. In pratica un router che riceve informazioni relative a una certa destinazione da un router adiacente non può rispedire indietro informazioni su quella stessa destinazione;
- **poison reverse**: può essere considerato come uno split horizon leggermente modificato, infatti talvolta è chiamato **split horizon with poison reverse**. Con questa tecnica il router spedisce ugualmente informazioni su certe route a chi le ha inviate, ma attribuisce loro una metrica infinita, per cui la destinazione viene considerata come irraggiungibile;
- **route poisoning**: blocca tutte le route che aumentano di costo supponendo che si tratti di un loop. Lo svantaggio è che potrebbe non essere un loop ma un legittimo aumento dovuto magari a una temporanea congestione;
- **hold down**: serve a limitare il count to infinity. Tutte le volte che un link è rimosso dalla routing table, il router non accetta aggiornamenti relativi al link stesso, se prima non ha aspettato un certo periodo di tempo (hold down timer);
- **triggered updates**: consente di inviare update non più a intervalli regolari ma non appena si verifica un cambiamento nella rete.

2.3 L'algoritmo di routing Link State

L'algoritmo Link State supera la principale limitazione del Distance Vector cioè la mancata conoscenza della topologia della rete.

> Ogni router ha una descrizione completa e diretta della topologia della rete poiché scambia le informazioni sulle distanze direttamente con **tutti** i router della rete e non solo coi vicini.

Questo avviene tramite l'invio di pacchetti, detti **LSP** (Link State Packet), da parte di ogni router a **tutti** gli altri router della rete. La trasmissione avviene in **flooding**, cioè un pacchetto viene inoltrato verso tutte le linee, tranne quella da cui è arrivato. Il pacchetto LSP è solitamente inviato solo quando avviene un cambiamento nella rete (come guasti o aggiunta di nuovi nodi), anche se alcuni gestori ne prevedono l'invio periodicamente.
Il pacchetto LSP contiene, per ogni mittente, l'elenco e la distanza da ogni vicino.
Ogni router esamina il numero di sequenza del pacchetto in arrivo e se risulta minore o uguale a quello memorizzato nel database, lo scarta. Se invece è maggiore lo memorizza e lo ritrasmette in flooding.

Tramite questi pacchetti, ogni router si costruisce un suo database con le informazioni sull'intera rete e, dopo aver ricevuto i pacchetti da tutti i router, è in grado di costruire un **grafo pesato** che rappresenta la rete stessa. A questo punto è possibile applicare un algoritmo per la ricerca dei cammini a costo minimo (il più noto è quello di **Dijkstra**).

#prendinota

Nel Distance Vector ogni nodo dice tutto ciò che sa ai suoi vicini, nel Link State ogni nodo dice ciò che sa dei suoi vicini a tutti.

Le caratteristiche del Link State si possono riassumere così:

- dispone della mappa della rete;
- ha una convergenza rapida poiché le informazioni si propagano velocemente senza alcuna elaborazione intermedia (comunicazione diretta tra tutti i nodi e non attraverso informazioni di "seconda mano");
- difficilmente genera loop e comunque è in grado di identificarli e interromperli facilmente;
- tutti i nodi hanno basi di dati identiche;
- è facilmente scalabile (all'aumentare del numero di router).

Il principale svantaggio di un algoritmo Link State è la complessità di realizzazione, anche dovuta alla notevole capacità di memoria (il database di tutta la rete) e velocità di elaborazione (ricerca dei cammini a costo minimo) richiesti.

2.4 Distance Vector e Link State a confronto

Nella TABELLA 2 si riassumono le caratteristiche dei due algoritmi di routing.

TABELLA 2 Confronto tra Distance Vector e Link State

Caratteristiche	Distance Vector	Link State
Tipo di routing	Dinamico	Dinamico
Convergenza	Lenta	Veloce
Utilizzo della banda	Basso, i pacchetti sono piccoli e non usa il flooding	Alto, usa il flooding e i pacchetti LSP sono di grandi dimensioni
Conoscenza della rete	Locale, conoscenza basata sulle informazioni provenienti dai router vicini	Globale, conosce la topologia dell'intera rete
Condivisione delle informazioni con i router vicini	A intervalli regolari di tempo	Solo quando c'è stato un cambiamento nella rete
Algoritmo con cui è costruita la tabella di routing	Bellman-Ford	Dijkstra SPF (Shortest Path First)
Problemi	Count to infinity (si risolve con split horizon) Loop persistenti (risolvibili con TTL)	Loop, causati dall'elevato traffico in rete generato con flooding, che può causare loop infiniti (risolvibile con TTL)
Protocolli che lo implementano	RIP, IGRP	OSPF, IS-IS

FISSA LE CONOSCENZE

- Quali sono gli scopi che un routing protocol si deve prefiggere?
- Quali sono i due principali problemi che possono presentarsi con il Distance Vector e come possono essere limitati?
- Quali sono le caratteristiche dell'algoritmo Link State?

3 GLI AUTONOMOUS SYSTEM E IL ROUTING GERARCHICO

3.1 Gli Autonomous System

#prendinota

Nell'Internetwork ogni rete è indipendente da tutte le altre e per questo sono chiamate *Autonomous System*.

Nei primi anni Ottanta, Internet era considerata come una *single network* cioè un'unica rete in cui tutti i router dovevano predisporre una routing table contenente una voce per ogni rete raggiungibile e l'indirizzo del router attraverso cui raggiungerla (neighbour router). L'introduzione di algoritmi e protocolli che permettessero ai router di adattarsi dinamicamente al mutare della topologia non risolveva il problema della crescita delle tabelle di routing.

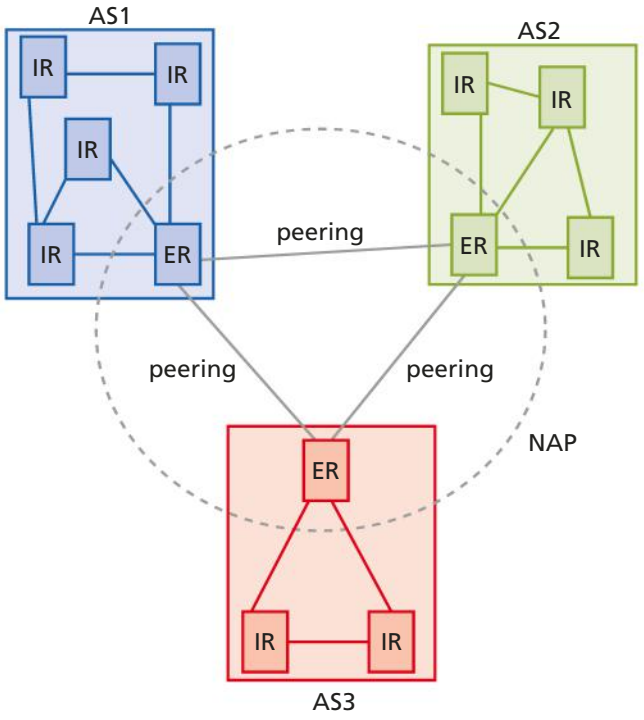
La decisione intrapresa fu quella di abbandonare il modello single network a favore di una rete modulare.

Internet è stata suddivisa in un certo numero di aree denominate **Autonomous System** (**AS**), ognuna gestita da un network provider (o da un gruppo di amministratori). Un AS è costituito da un insieme di router e LAN raggruppati secondo criteri topologici e organizzativi, che assicurano che i router all'interno di ogni AS siano reciprocamente raggiungibili.

La FIGURA 2 mostra la comunicazione all'interno di un AS e tra AS diversi:

- le informazioni di raggiungibilità all'interno di un AS sono scambiate mediante uno o più protocolli adattivi detti **Interior Protocol**;
- gli AS si scambiano informazioni sulla rispettiva raggiungibilità utilizzando un protocollo opportuno genericamente designato come **Exterior Protocol**.

FIGURA 2 Autonomous System con IR e ER



Se un router deve instradare un messaggio a un altro router appartenente allo stesso AS (in questo caso si parla di comunicazione tra **Interior Router, IR**), avrà nella propria tabella di routing l'informazione di raggiungibilità idonea.
Se al contrario è necessario raggiungere un router che appartiene a un differente AS, il messaggio viene inviato attraverso una coppia di router particolari detti **Exterior Router** (**ER**), almeno uno per ogni AS. Ciascun ER conosce le reti raggiungibili utilizzando i link che lo collegano agli altri ER, ma non conosce il modo in cui queste reti sono di fatto connesse all'interno dei rispettivi AS.

Gli Exterior Router di un AS hanno visibilità su tutte le destinazioni all'interno dell'AS e sugli Exterior Router degli altri AS.

Ciascun AS è gestito indipendentemente dagli altri, in particolare per quanto riguarda l'instradamento dei pacchetti IP al suo interno. Gli accordi, amministrativi e tecnici, tra i gestori di AS differenti per stabilire le politiche di transito e raggiungibilità, sono detti accordi di **peering**.

> Una relazione di peering si stabilisce tutte le volte che un Exterior Protocol viene attivato tra due AS differenti.

Per esempio la rete dell'Università di Bologna e la rete del Politecnico di Torino comunicano con il resto del mondo tramite il **GARR** e le sue scelte di peering (stesse politiche di routing) e non c'è bisogno di avere un AS per ogni ateneo. Infatti, il GARR e tutte le reti connesse a esso costituiscono un unico AS (AS137).
Esistono dei siti dedicati dove viene creato un punto di contatto tra diversi AS detti **NAP** (Neutral Access Point). Un NAP dal punto di vista pratico è un sottosistema di comunicazione (come fosse una LAN) che interconnette router appartenenti ad AS differenti, tra i quali viene opportunamente configurato un Exterior Protocol.
La rete GARR e gli Internet Exchange Provider (IXP) sono stati descritti nell'Unità 8 del volume del terzo anno, dedicata alle reti geografiche WAN.

TIPI DI AUTONOMOUS SYSTEM

Si possono distinguere tre tipi diversi di Autonomous System:

- **multi-homed AS**: si connette a due o più AS, così da garantire la connessione a Internet in caso di guasto del collegamento verso un AS;
- **stub AS** (o single-homed): si connette a un solo altro AS, sebbene possa avere delle proprie connessioni private non visibili al resto di Internet;
- **transit AS**: agisce come collegamento tra due o più AS, permettendo il transito dei dati, anche provenienti da reti non associate (può offrire così ai propri utenti accesso ad altre reti).

AUTONOMOUS SYSTEM NUMBER

Abbiamo visto che gli Autonomous System sono delle singole autorità amministrative, e in quanto tali devono essere identificate univocamente all'interno di Internet. A livello internazionale si è quindi deciso di mantenere un elenco di "numeri" da assegnare via via a chi ne fa richiesta. Questi identificatori sono detti **Autonomous System Number (ASN)** e sono allocati a blocchi da **IANA** alle Regional Internet Registries (**RIR**) che li assegnano ai gestori di rete.
L'elenco è disponibile alla pagina: *www.iana.org/assignments/as-numbers*.

Esistono due diversi formati di ASN:

- 2 byte ASN: sono numeri a 16 bit, da 0 a 65.535, di questi IANA riserva un blocco di 1.023 per uso privato;
- 4 byte ASN: sono numeri a 32 bit, da 0 a 4.294.967.295, di questi IANA riserva un blocco di 94.967.295 per uso privato.

#prendinota

Fino al 2007 i numeri ASN erano a 16 bit, attualmente non c'è più distinzione tra i due formati e tutti gli ASN sono considerati di **4 byte**.

3.2 Il routing gerarchico

Come abbiamo sottolineato nelle precedenti Lezioni, tabelle di routing molto grandi, oltre a rischiare di saturare la memoria del router, finiscono col rallentare le trasmissioni costringendo la CPU dei router a lunghe elaborazioni per trovare i percorsi

ottimali. Inoltre, l'aggiornamento delle routing table dei router comporta numerosi e continui scambi di informazioni tra innumerevoli router e il traffico di rete può risentirne. Una rete con 1.000 router richiederebbe una tabella con 1.000 voci, una per ogni router.
Se una rete cresce fino al punto in cui i router non hanno più spazio nella tabella per tutte le possibili destinazioni, occorre riorganizzare la rete al di là dei possibili algoritmi e protocolli utilizzati.
In questi casi si usa la classica idea del "divide et impera". Secondo questo paradigma, applicato alle reti, anziché creare un'unica rete molto grande, si creano tante piccole reti unite tra loro. Nell'ambito di ognuno di questi medio/piccoli domini, detti **regioni**, la comunicazione segue i protocolli standard. A sua volta l'interconnettività tra le varie regioni è garantita da router dedicati, che nuovamente comunicano tra loro come delle reti che seguono protocolli standard. Questa tecnica è detta **routing gerarchico** e consiste nel riproporre su scala geografica quanto implementato su un dominio limitato.

Secondo il routing gerarchico è necessario partizionare la rete in regioni e interconnetterle fra loro.
Con questa tecnica, ogni router conosce nel dettaglio la propria regione di appartenenza ma nulla sa della topologia delle altre regioni, per questo motivo le prestazioni ne risentono, in quanto i percorsi non sono ottimizzati.

#prendinota

Una regione deve sempre essere **connessa**, cioè deve essere sempre possibile trasferire un messaggio tra due router appartenenti alla stessa regione senza farlo uscire da quella regione.

Un router appartenente a una qualsiasi regione può essere interno (**internal router**) o di frontiera (**boundary router**). Se è interno conterrà nella sua tabella di routing l'indicazione del router di frontiera della sua stessa regione a cui affidarsi a seconda della destinazione che vuole raggiungere in altra regione. Il router di frontiera inoltrerà i pacchetti verso la regione di destinazione o a sua volta affiderà i pacchetti a un router di frontiera di un'altra regione affinché li inoltri verso la regione di destinazione. Non conoscendo la topologia dell'intera rete, la scelta dell'instradamento potrebbe essere penalizzante. Infatti i router di frontiera di una regione non comunicano direttamente con tutti gli altri router di frontiera delle altre regioni, ma sono predisposti per inoltrare il traffico verso una determinata regione che potrebbe non rappresentare la scelta migliore.
Il vantaggio è la riduzione della tabella di routing poiché tutte le destinazioni appartenenti a una certa regione sono riassunte nell'unico indirizzo del router di frontiera.

QUANTI LIVELLI GERARCHICI?

Con questa tecnica si presenta il problema di stabilire quanti livelli gerarchici realizzare.

Occorrerà mediare tra la necessità di avere routing table di dimensioni ragionevoli e la necessità di svolgere un routing il più performante possibile.

All'aumentare dei livelli gerarchici, le tabelle si snelliscono e dunque si velocizzano gli algoritmi e si limita l'occupazione di banda dovuta allo scambio di tabelle tra router. Al contempo però si perde sempre più di vista la topologia di rete e quindi il percorso dei pacchetti rischia fortemente di non essere ottimizzato.
Un router che apprenda la stessa destinazione da più protocolli dà la preferenza a uno di essi secondo una gerarchia predefinita, senza altri calcoli.

In sostanza, reti con migliaia di router sparsi per i 5 continenti non possono limitarsi a 2 soli livelli gerarchici. Supponiamo per esempio che un'azienda come FCA da un router di Torino debba trasmettere pacchetti alla sua filiale di San Paolo del Brasile. Il router di Torino conoscerà la topologia locale ma si appoggerà, per esempio, a un router di Roma per tutto il traffico nazionale. Il router di Roma a sua volta gestirà direttamente il traffico nazionale, ma si affiderà a un router a Parigi per il traffico al di fuori dell'Italia. Allo stesso modo il router di Parigi sarà in grado di indirizzare verso la destinazione il traffico in Europa, ma si affiderà a un router di Buenos Aires per il traffico verso il Sud America. Solo arrivati a Buenos Aires i pacchetti potranno essere instradati verso San Paolo.
Per evitare problemi di gestione dei router di frontiera spesso si fa in modo che ogni regione utilizzi almeno 2 router di frontiera che si interfacciano a 2 regioni distinte (nella precedente Figura 2, gli ER hanno 2 interfacce verso 2 diversi AS). Questo consente di distribuire il traffico e predisporre le regioni non esclusivamente secondo parametri geografici, ma anche in base al cosiddetto **routing domain** (dominio di routing), cioè al fatto che un gruppo di router sia accomunato dall'utilizzo dello stesso protocollo di routing.

PROTOCOLLI DIVERSI E METRICHE DIVERSE

Se due regioni utilizzano protocolli diversi non possono scambiarsi informazioni direttamente, ma dovranno ricorrere alla tecnica della **redistribuzione**.
In pratica un router può redistribuire, su un dominio che utilizza un certo protocollo, le informazioni apprese da un altro dominio che ha un diverso protocollo: in questi casi si parla di **router multiprotocollo a metriche diverse** (di norma protocolli diversi usano metriche diverse).
Per consentire una corretta interpretazione dei costi dei percorsi verso i diversi domini di routing, è quindi necessario inserire all'interno dei pacchetti di routing un identificativo di protocollo. Solo se si conosce il protocollo è possibile trasformare la metrica in modo coerente.

esempio

La figura a fianco mostra un esempio di **routing gerarchico**:

– la Regione 1 (R1) è connessa alle Regioni 2 e 4 (R2 ed R4);

– queste, a loro volta, sono connesse alla Regione 3 (R3).

Se si usasse un protocollo di routing basato su Distance Vector o su Link State, in cui ogni router riesce, indirettamente o direttamente, ad avere informazioni su tutte le destinazioni, avremmo la tabella di routing per il **router A** della Regione 1 (R1A) riportata nella pagina seguente.

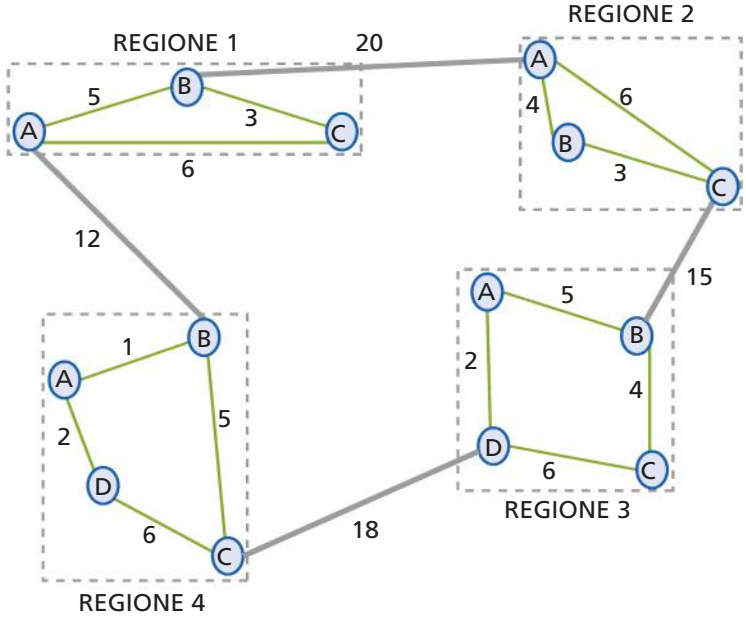


TABELLA DI ROUTING PER IL ROUTER R1A

Destinatario	Distanza	Linea
R1A	-	-
R1B	5	R1B
R1C	6	R1C
R2A	25	R1B
R2B	29	R1B
R2C	31	R1B
R3A	37	R4B
R3B	42	R4B
R3C	41	R4B
R3D	35	R4B
R4A	13	R4B
R4B	12	R4B
R4C	17	R4B
R4D	15	R4B

La suddivisione in regioni limita la dimensione della tabella di routing di R1A, che risulta così con poche righe (entry), come mostra la tabella qui sotto.

TABELLA DI ROUTING DI R1A CON ROUTING GERARCHICO

Destinatario	Distanza	Linea
R1A	-	-
R1B	5	R1B
R1C	6	R1C
R2	25	R1B
R3	35	R4B
R4	12	R4B

Le limitate dimensioni della tabella ne velocizzano la consultazione ma, come si può notare, si perde la visione dell'intera rete e la metrica, chiamata Distanza in questo esempio, non dà informazioni sull'esatto costo dell'intero percorso.

3.3 I protocolli di routing interior ed exterior

Come abbiamo già visto, il routing può essere eseguito o all'interno di una singola rete, anche eventualmente suddivisa in sottoreti, oppure nell'Internetwork, cioè tra reti eterogenee e che magari utilizzano protocolli diversi. Le due situazioni di routing sono simili: in entrambi i casi si tratta di costruire il grafo dei router e applicare l'algoritmo desiderato, tipo il Distance Vector o il Link State.
Ovviamente nell'Internetwork le difficoltà sono maggiori, legate al fatto che ogni router ha una scelta molto ampia potendo instradare i pacchetti verso qualsiasi altro router connesso alla rete.

Questo fa sì che i protocolli di routing, che verranno approfonditi nelle prossime due Lezioni, vengano suddivisi in due tipi (FIGURA 3):

- **IGP (Interior Gateway Protocol)**, usati all'interno di una rete;
- **EGP (Exterior Gateway Protocol)**, usati tra più reti.

Da notare l'uso del termine gateway e non router, come potremmo aspettarci, considerato che gli apparati di rete coinvolti sono, appunto, i router. Per tradizione, con il termine **gateway** (portone, passaggio) si indica il servizio di inoltro dei pacchetti da una rete verso l'esterno:

- nelle reti più semplici è presente un solo gateway che inoltra tutto il traffico diretto all'esterno, verso la rete Internet;
- in reti più complesse, in cui sono presenti parecchie sottoreti, ognuna di queste fa riferimento a un gateway che si occuperà di instradare il traffico verso le altre sottoreti o a indirizzarlo verso altri gateway.

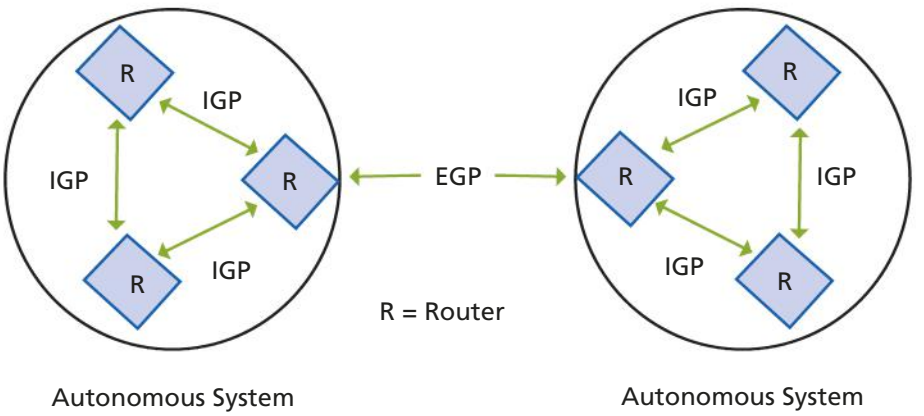


FIGURA 3 IGP ed EGP tra Autonomous System

Considerato che queste funzioni sono tipiche dei router, si potrebbero usare i due termini come sinonimi. In realtà, spesso i gateway non si limitano a fornire la funzione di routing, ma integrano altri servizi come proxy DNS, firewall, NAT, ecc., che sono servizi dei livelli superiori della rete TCP/IP.

esercizio

➜ PROBLEMA

Impostare il gateway predefinito sui sistemi Windows e Linux.

➜ SVOLGIMENTO

Windows

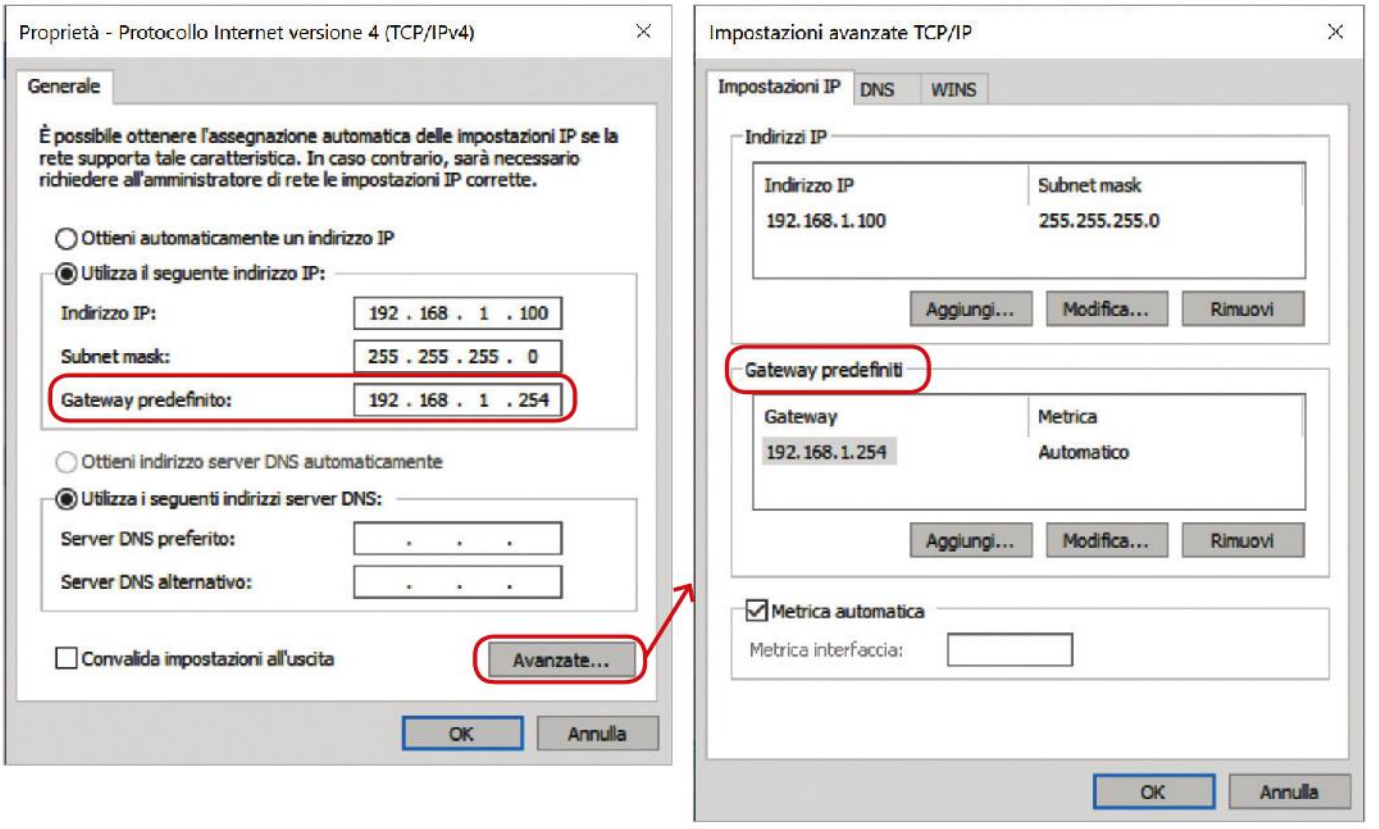
Su un sistema Windows 10 si imposta l'indirizzo IP del gateway predefinito, operando sulle proprietà del Protocollo Internet della scheda di rete interessata, per esempio Ethernet o Wi-Fi. Si può seguire il percorso partendo dal tradizionale Pannello di Controllo o da Impostazioni di Windows: selezionando con il tasto destro la scheda di rete, si apre la finestra delle Proprietà.

Nell'elenco selezionare **Protocollo Internet versione 4 (TCP/IPv4)** e cliccare sul pulsante Proprietà, nella finestra che compare cliccare su **Utilizza il seguente indirizzo IP** (configurazione statica degli indirizzi IP) e inserire i dati relativi a indirizzo IP e subnet mask della scheda di rete e Gateway predefinito.

Se si clicca poi sul pulsante **Avanzate...** si apre una finestra in cui è possibile aggiungere un nuovo gateway o modificare/rimuovere uno già presente, come mostrato nelle seguenti figure.

#prendinota

Per impostazione predefinita, TCP/IP calcola automaticamente la **metrica** di interfaccia in base alla velocità dell'interfaccia stessa.

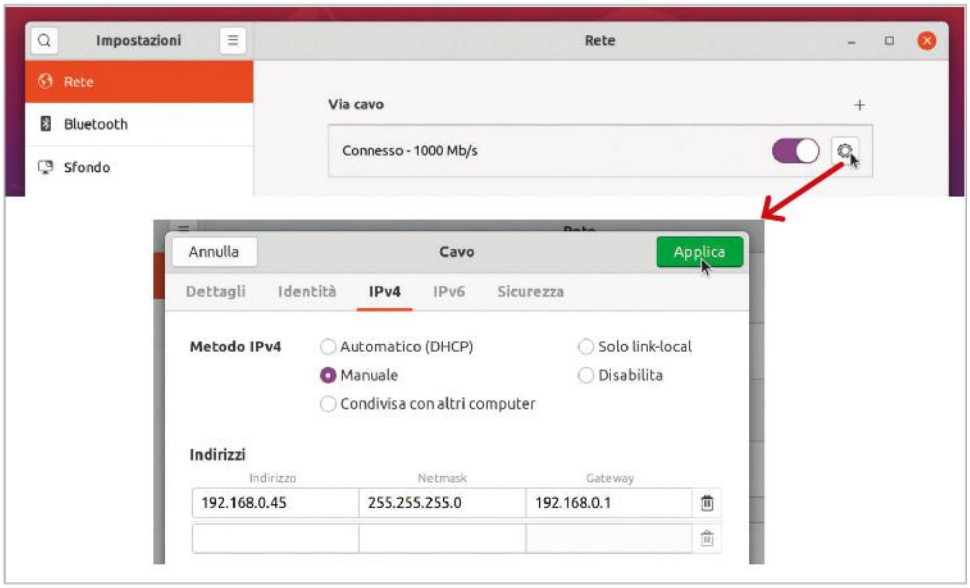


Linux

Prendiamo per esempio il sistema Ubuntu Desktop, possiamo procedere in modo analogo a quanto visto per Windows, tramite interfaccia grafica, cliccando sull'icona della rete in alto a destra.
Nel menu che compare espandere la voce relativa al collegamento di interesse, Via cavo o Wi-Fi, e cliccare su **Impostazioni rete**. Nella finestra visualizzata cliccare sull'icona a forma di ingranaggio, quindi selezionare la scheda **IPv4** e il metodo Manuale, a questo punto si possono immettere/modificare i dati relativi all'indirizzo IP del PC e del Gateway, come mostrato nella figura seguente.

#prendinota

In Ubuntu Desktop si può anche procedere con la modifica del file di configurazione da terminale, come si fa con Ubuntu Server, ma è consigliabile usare l'interfaccia grafica. Infatti, se si modifica il file di configurazione, sparisce l'icona della rete in alto a destra, rendendo così necessario continuare a operare tramite terminale.



Gli host dipendono dai gateway predefiniti per la maggior parte delle comunicazioni con host residenti in segmenti di reti remote. In questo modo, non è necessario che i singoli host gestiscano e aggiornino costantemente i dati di routing relativi ai vari segmenti delle reti IP remote. Solo il router che agisce come gateway predefinito deve gestire questo livello di informazioni per ottimizzare il routing volto a raggiungere segmenti di reti nell'Internetwork.

FISSA LE CONOSCENZE

- Che cosa sono gli Autonomous System (AS)?
- Spiega a che cosa serve l'Autonomous System Number e qual è il suo formato.
- Descrivi le caratteristiche del routing gerarchico.
- Che cosa sono e in cosa differiscono i protocolli IGP ed EGP?
- Ogni rete ha un solo gateway?
- Perché il gateway predefinito consente di alleggerire le routing table degli host di una rete?

4 PROTOCOLLI DI ROUTING IGP

4.1 Gli Interior Gateway Protocol

In questa Lezione analizzeremo i protocolli di routing di tipo **IGP** (Interior Gateway Protocol) noti come **protocolli intradominio** poiché vengono usati per regolare l'instradamento dei pacchetti tra gli host interni a un Autonomous System.
Questi protocolli possono essere classificati in base all'algoritmo di routing, Distance Vector o Link State, che utilizzano. I principali IGP sono:

1. con Distance Vector:
   - **RIP**, Routing Information Protocol, definito in ambito IETF, usa la metrica hop count;
   - **IGRP**, Interior Gateway Routing Protocol, proprietario Cisco, è stato creato per superare le limitazioni di RIP, supportando anche più metriche (bandwidth, delay, load, reliability);
   - **EIGRP**, Enhanced IGRP, proprietario Cisco, ha sostituito IGRP introducendo una serie di miglioramenti, però continua a usare le stesse metriche.
2. con Link State:
   - **OSPF**, Open Shortest Path First;
   - **Integrated IS-IS**, Integrated Intermediate System-Intermediate System, è uno standard che è stato definito prima in ISO e poi in IETF.

Analizziamone uno per tipologia, scegliendo quelli definiti come standard in ambito IETF e, come tali, usati in tutte le reti: RIP e OSPF.

4.2 Il protocollo RIP (Routing Information Protocol)

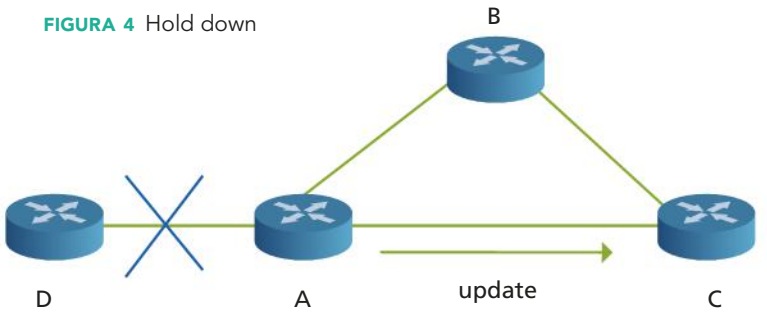
È uno dei protocolli più vecchi essendo stato sviluppato dalla Xerox e poi adattato alla suite TCP/IP nel 1982. La sua metrica è il numero di hop (**hop count**), la più semplice di tutte, ma anche la meno efficiente. RIP è un protocollo Distance Vector che opera in modo molto simile all'algoritmo descritto nella Lezione 2, dove per semplicità il costo di un percorso è stato definito tra coppie di router. In RIP i costi sono in realtà calcolati dal router sorgente alla subnet di destinazione.

> Nella specifica di RIP, la distanza è definita come il numero di subnet attraversate lungo il percorso più breve da un router sorgente a una subnet di destinazione, includendo anche quest'ultima.

Solo il percorso più breve verso la destinazione viene memorizzato nella routing table. Il numero massimo di hop consentiti prima che il pacchetto venga scartato è pari a 15. Di conseguenza, il RIP non può essere impiegato in un contesto in cui esistano subnet separate da più di 15 router. Per questo motivo il RIP è solitamente utilizzato in LAN non troppo grandi.
Il RIP prevede che le tabelle di routing vengano aggiornate ogni 30 secondi, quindi ogni router invia la propria tabella completa a tutti i vicini direttamente collegati, generando grandi quantità di traffico su reti a bassa capacità trasmissiva. Un'intera rete garantisce l'aggiornamento dei percorsi verso tutti i suoi router (convergenza) in circa 3 minuti.

Opzionalmente il RIP applica l'**hold down**, cioè non accetta alcun update per 60 secondi relativamente a route guaste.
Per esempio, come mostrato nella FIGURA 4, se il router A manda un update al router C comunicando che il router D non è più raggiungibile, dovranno passare 60 secondi prima che il router C accetti dal router B (o da qualunque altro router) update riguardanti il router D. Questo per evitare che B, non ancora informato del guasto su D, faccia credere a C che D sia tornato raggiungibile. I 60 secondi servono proprio a garantire la convergenza evitando che si propaghino informazioni inconsistenti.

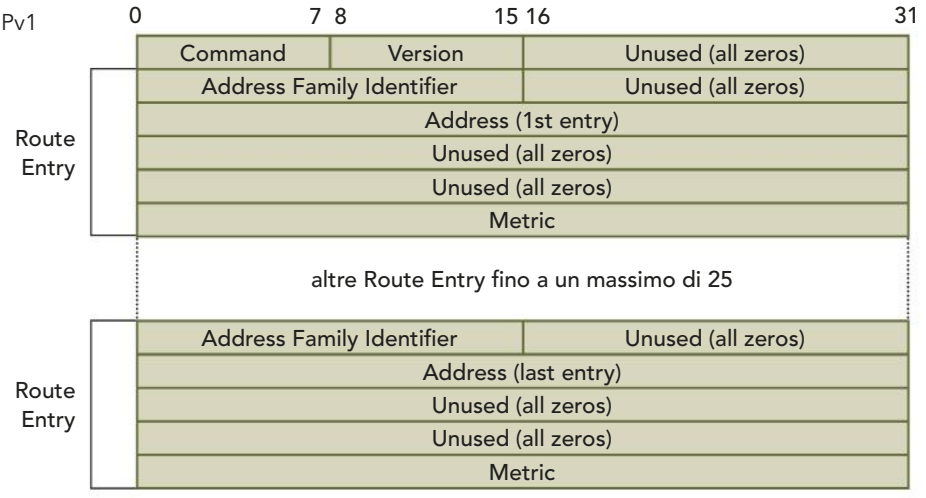
FIGURA 4 Hold down



Il modo con cui il RIP evita il loop tra 2 nodi adiacenti è lo **split horizon**, tipicamente utilizzato dai protocolli basati sul Distance Vector.

In un pacchetto RIP non è presente alcun campo dati, in quanto il protocollo RIP non è stato progettato per lo scambio di informazioni tra host. In pratica le informazioni contenute in tali pacchetti riguardano solo il routing.
Vediamo in dettaglio il formato di un pacchetto RIPv1 (FIGURA 5):

FIGURA 5 Pacchetto RIPv1



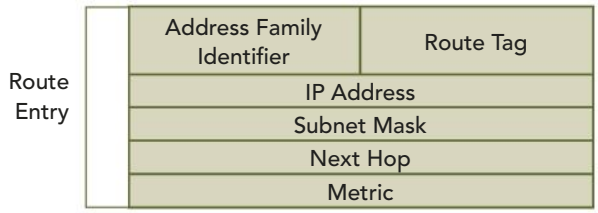
Ogni pacchetto RIPv1 contiene:

- un campo **Command:** indica il tipo di messaggio: 1 = Request, 2 = Response; un pacchetto Response può essere inviato come risposta a una richiesta o a seguito di un aggiornamento della tabella di routing (update);
- un campo **Version**: contiene il numero della versione di RIP: 1 = RIPv1 e 2 = RIPv2;
- fino a 25 **route entry**: corrispondono alle entry della routing table del mittente. Nel caso in cui il router avesse una routing table con più di 25 entry, invierà più pacchetti. Ogni route entry occupa 20 ottetti che trasportano i seguenti dati:
  - **Address Family Identifier**: individua il tipo di indirizzo contenuto nel campo Address, nel caso di IP il valore è 2;
  - **Address**: è l'indirizzo del destinatario della route (network, subnetwork o host);
  - **Metric**: è espressa in hop count, l'unica metrica usata da RIPv1.

Il pacchetto RIP è imbustato in UDP (User Datagram Protocol a livello Transport) usando il numero di porta 520.

Ci sono 3 versioni del protocollo: RIPv1, RIPv2 e RIPng. La versione **RIPv1** (RFC 1058) effettua un routing classfull, infatti non trasporta la subnet mask: tutte le sottoreti di una rete hanno la stessa subnet mask. Questo problema viene risolto nella versione **RIPv2** (RFC 2453): nel pacchetto è presente il campo Subnet Mask che contiene appunto la subnet mask corrispondente alla sottorete considerata (FIGURA 6).
Un'ulteriore novità introdotta da RIPv2 riguarda i messaggi di update: con RIPv1 tali messaggi vengono inviati in broadcast e quindi favoriscono il sovraccarico nella rete e un tempo di convergenza lento, mentre con RIPv2 i messaggi vengono inviati in multicast, interessando così solo certi router. Inoltre, in RIPv2 si implementano meccanismi per l'autenticazione. **RIPng** (RIP next generation, RFC 2080) è un'estensione del protocollo originale RIPv1 per supportare IPv6.

FIGURA 6 Route entry in RIPv2



IN ENGLISH PLEASE

Network Working Group — G. Malkin
Request for Comments: 2453 — Bay Networks
Obsoletes: 1723, 1388 — November 1998
STD: 56
Category: Standards Track

RIP Version 2

[...]

1. Justification

With the advent of OSPF and IS-IS, there are those who believe that RIP is obsolete. While it is true that the newer IGP routing protocols are far superior to RIP, RIP does have some advantages. Primarily, in a small network, RIP has very little overhead in terms of bandwidth used and configuration and management time. RIP is also very easy to implement, especially in relation to the newer IGPs. Additionally, there are many, many more RIP implementations in the field than OSPF and IS-IS combined. It is likely to remain that way for some years yet. Given that RIP will be useful in many environments for some period of time, it is reasonable to increase RIP's usefulness. This is especially true since the gain is far greater than the expense of the change.

4.3 Il protocollo OSPF (Open Shortest Path First)

OSPF è un protocollo di routing (RFC 2328 e successivi aggiornamenti) basato sulla creazione di un database distribuito che rappresenti la topologia della rete sotto forma di grafo. Utilizza quindi l'algoritmo Link State. I protocolli SPF-based hanno la caratteristica dell'uniformità del database relativo all'Autonomous System (AS), cioè tutti i router dell'AS hanno le stesse informazioni relative allo stato delle connessioni. Questa caratteristica, ovviamente, la si riscontra anche in OSPF che quindi è in grado di rispondere prontamente alle variazioni topologiche dell'AS con un rapido aggiornamento delle routing information, mediante un traffico di protocollo assai ridotto.

Come per RIP, anche di OSPF è stata definita una nuova versione, **OSPFv3**, per il supporto di IPv6: **OSPF for IPv6**, RFC 5340.

IN ENGLISH PLEASE

Network Working Group — J. Moy
Request for Comments: 2328 — Ascend Communications, Inc.
STD: 54 — April 1998
Obsoletes: 2178
Category: Standards Track

OSPF Version 2

Abstract

This memo documents version 2 of the OSPF protocol. OSPF is a link-state routing protocol. It is designed to be run internal to a single Autonomous System. Each OSPF router maintains an identical database describing the Autonomous System's topology. From this database, a routing table is calculated by constructing a shortest-path tree. OSPF recalculates routes quickly in the face of topological changes, utilizing a minimum of routing protocol traffic. OSPF provides support for equal-cost multipath. An area routing capability is provided, enabling an additional level of routing protection and a reduction in routing protocol traffic. In addition, all OSPF routing protocol exchanges are authenticated.

Il protocollo OSPF si fonda sul concetto di **area**: un'area è costituita da una o più reti contigue. Ogni area ha il proprio database topologico ed esegue la propria copia dell'algoritmo di routing.
La caratteristica di ogni area è che la propria struttura interna è nascosta alle altre aree, così come i router interni all'area non conoscono in dettaglio la topologia delle aree esterne.
La conseguenza immediata, che è anche il vantaggio di tale organizzazione distribuita, è la sensibile riduzione del traffico di routing, rispetto a un'organizzazione centralizzata. Altra conseguenza immediata è che i router non hanno tutti lo stesso database topologico, ma ognuno possiede quello relativo alla propria area; i router collegati a più aree avranno un database per ogni area.

Con questo tipo di organizzazione si hanno due tipi di routing, anzi due livelli: **routing intra-area**, quando la sorgente e la destinazione dei pacchetti si trovano nella stessa area; **routing inter-area**, quando la destinazione si trova in un'area differente da quella della sorgente.

Nel primo livello di routing intra-area non si utilizzano informazioni di routing provenienti dall'esterno.
Tra le aree ce n'è una particolare, detta **#backbone** o **area zero** (FIGURA 7).
Tale area è formata da tutte quelle reti che non appartengono a nessuna area, più i router di confine di ogni area. È quindi possibile passare da un'area a una qualunque altra area dello stesso AS passando dalla backbone area. Nel caso un'area non sia fisicamente connessa alla backbone area si configura un collegamento virtuale (**virtual link**) attraverso un'area di transito (Figura 7).
Dal punto di vista del grafo, si ha che il costo del link si ottiene come somma delle distanze intra-area tra i due router; si parla di distanza intra-area perché si utilizza solo il routing intra-area per i link virtuali. Dal punto di vista topologico la backbone area ha le stesse caratteristiche e proprietà di tutte le altre.

#techwords

Backbone
È il termine usato ogniqualvolta si fa riferimento a una rete principale, *core*, che collega altre reti. Il traffico dati di queste reti passa attraverso uno o più nodi del backbone.

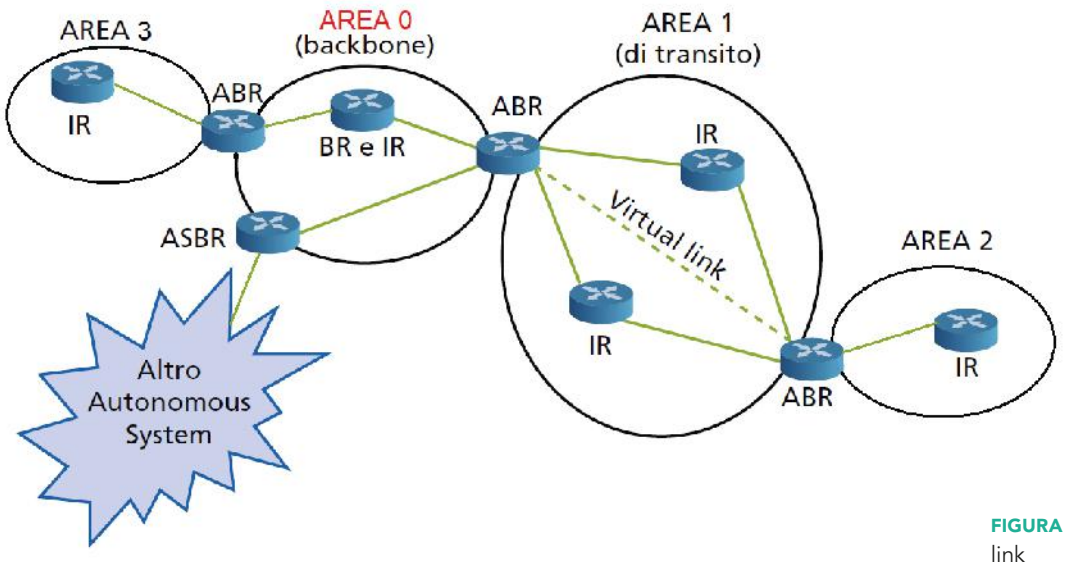


FIGURA 7 Backbone con virtual link

I router si classificano come segue:

- **router interni** (**IR**, Internal Router): router che sono connessi direttamente alle reti appartenenti alla stessa area; fanno parte di questa categoria anche quei router che sono nell'area backbone;
- **router di confine dell'area** (**ABR**, Area Border Router): router collegati a più aree, una deve essere la backbone area. Essi eseguono una copia dell'algoritmo per ogni area collegata, più una copia per la backbone area, dal momento che i router di confine sono anche router di backbone (BR);
- **router backbone** (**BR**, Backbone Router): router che hanno un'interfaccia sulla backbone area. Da sottolineare che i router che hanno solo interfacce sulla backbone sono da considerare router interni;
- **router di confine dell'AS** (**ASBR**, Autonomous System Boundary Router): router che scambiano informazioni con altri AS. Tutti i router dell'AS conoscono i cammini che portano a questi router, che sono gli unici a poter instradare l'informazione verso l'esterno.

La Figura 7 mostra un esempio di ubicazione di tali router.
Per mandare un pacchetto a un'area esterna, all'interno dello stesso AS, occorre utilizzare la backbone area; si può dividere il cammino che effettua il pacchetto in 3 parti:

- la prima parte è il cammino intra-area dall'IR fino all'ABR della propria area;
- la seconda parte è il cammino nell'area backbone tra l'ABR dell'area sorgente e l'ABR dell'area destinazione;
- la terza parte è il cammino intra-area dall'ABR dell'area destinazione fino all'IR destinatario.

Questi cammini devono essere, ovviamente, quelli con costo minimo. È dunque la topologia backbone a stabilire i cammini fra le aree, cammini che, come detto, possono essere creati anche con link virtuali. Il router di confine dell'area (ABR) che ha un'interfaccia sulla backbone area, ha il compito fondamentale di fornire sia i costi per il collegamento a tutte le reti esterne all'area, sia i costi interni all'area che si sommano ai precedenti.

Le informazioni EGP, esterne a un AS, vengono propagate al suo interno senza subire modifiche.
Tutti i router avranno quindi tutti gli indirizzi necessari per inviare i pacchetti alle reti esterne.
I router di confine tra gli AS (ASBR) devono avere a bordo un'istanza del protocollo IGP e una istanza del protocollo EGP. L'amministratore del sistema deve predisporre opportunamente la propagazione delle informazioni tra i due protocolli (redistribuzione).

Nell'ambito delle **route esterne** all'AS si considerano due tipi di metriche:

- **metrica di tipo 1**: praticamente identica alla metrica utilizzata all'interno dell'AS, cioè data dalla somma dei costi delle interfacce da attraversare relativi a un percorso per giungere a destinazione. Se esistono più percorsi per una singola destinazione, viene scelto quello con costo minore;
- **metrica di tipo 2**: viene scelto il router di confine dell'AS che comunica il minor costo per il raggiungimento di un network esterno, a prescindere dal costo relativo alla route interna all'AS. A parità di costo esterno viene scelto il router che ha metrica di tipo 1 minore.

#prendinota

Se esistono cammini multipli a pari costo per una data destinazione, OSPF li utilizza tutti, distribuendo il traffico equamente sui percorsi disponibili.

OSPF supporta il routing in base agli IP Type Of Services (TOS), ciò rende possibile l'associazione di un costo differente a un'interfaccia in base al TOS. Tutti i router OSPF hanno questa capacità e calcolano un SPF (Shortest Path First) per ogni TOS in modo che ci siano delle route già configurate nel caso in cui giungano pacchetti per un dato TOS. I tipi di TOS supportati da OSPF sono:

- default
- affidabilità
- throughput
- costo
- ritardo

In base a questi TOS vengono poi smistati i pacchetti. Se una TOS-route non è stata configurata, allora tutti i pacchetti ivi indirizzati verranno smistati sulle route del TOS di default (TOS 0).

LINK STATE ADVERTISEMENT

IN ENGLISH PLEASE

Type of Service (TOS) has been removed and is not encoded within OSPF for IPv6's Link State Advertisement (LSA).

OSPF utilizza la tecnica **LSA** (Link State Advertisement) per condividere le informazioni di instradamento tra i nodi. Questa tecnica consiste nel rendere pubbliche le informazioni sullo stato dei collegamenti inviando dei pacchetti, detti appunto **LSA packet**, ai router. Sono stati definiti vari tipi di pacchetti LSA generati dai diversi router: interni, designated, ABR e ASBR, per inviare in flooding i messaggi nella stessa area interna o nelle aree esterne.
L'header OSPF (FIGURA 8) è costituito da 24 byte suddivisi in 8 campi.

- **Version**: 1 byte, nelle reti IPv4 si usa la versione 2, nelle reti IPv6 si usa la versione 3 (OSPF for IPv6);
- **Packet Type**: 1 byte, con valori:
  1 Hello
  2 Database Description (DD)
  3 Link State Request
  4 Link State Update
  5 Link State Acknowledgment

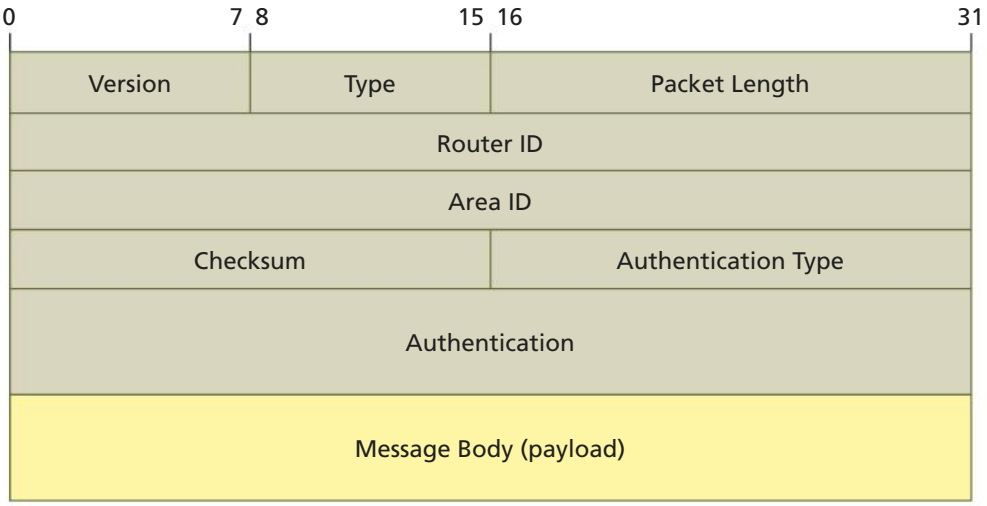


FIGURA 8 Header OSPF

- **Packet Length**: 2 byte, dimensione del pacchetto in byte, header incluso;
- **Router ID**: 4 byte, identificatore del router che ha generato il messaggio;
- **Area ID**: 4 byte, identificatore dell'area OSPF a cui appartiene il messaggio, ogni messaggio appartiene a una sola area;
- **Checksum**: 2 byte, calcolo della checksum, in modo simile al protocollo IP, su tutto il messaggio tranne il campo Authentication;
- **Authentication Type**: 2 byte, indica il tipo di autenticazione usato per il messaggio:
  - 0 nessuna autenticazione;
  - 1 autenticazione tramite semplice password;
  - 2 autenticazione con crittografia;
- **Authentication**: 8 byte, contiene dati utili per il tipo di autenticazione da applicare, definito nel campo Authentication Type:
  - password semplice, è presente una password di 64 bit in chiaro. Questa modalità rende vulnerabile la rete e compromette la sicurezza del dominio di routing OSPF, infatti la password può essere letta da qualunque dispositivo connesso fisicamente alla rete;
  - uso della crittografia, in tutti i router è configurata una chiave segreta, usata per generare o verificare la stringa di autenticazione (*digest*) del pacchetto, che viene inserita al fondo di ogni pacchetto OSPF.

#prendinota

In OSPF for IPv6 i due campi relativi all'autenticazione non sono più presenti, infatti le tecniche descritte in OSPFv2 sono state sostituite da Authentication header e da Encapsulating Security Payload header (descritti nell'Unità 4).

Il **Message Body** contiene i dati relativi al tipo di pacchetto. Per esempio, nel caso di pacchetti di Hello, contiene l'elenco dei Router ID dei router vicini.

Vediamo nel dettaglio i **Packet Type**.

1. **Pacchetto Hello**

   Il protocollo di Hello è usato per creare e mantenere le relazioni di vicinato. Viene anche usato per l'elezione del Designated Router. All'accensione il router inizializza alcune strutture dati di supporto e aspetta indicazioni dai protocolli di basso livello sulla corretta funzionalità delle sue interfacce. Appena si è assicurato che le sue interfacce funzionano usa un protocollo di Hello per acquisire informazioni sui nodi vicini: invia pacchetti di Hello e li riceve da questi. Successivamente i pacchetti di Hello sono usati come pacchetti *keep-alive* per verificare i vicini attivi.

#prendinota

Designated Router (DR)
È un router eletto tramite protocollo Hello allo scopo di ridurre il traffico di rete. Il DR invia gli aggiornamenti agli altri router della sua area usando il multicast.

2. Pacchetto Database Description (DD)

Una coppia di router adiacenti si scambia pacchetti di Database Description (DD) per descrivere il contenuto del proprio database topologico (Link State Database). È una tecnica di sincronizzazione tra i database. Mediante un meccanismo basato sugli ID dei router uno dei due viene eletto **master** e l'altro **slave**. Appena un router riceve il primo pacchetto di Hello da un suo vicino, gli invia un pacchetto DD che fornisce la descrizione di ciascuna entry del suo database. Il router, esaminando il DD, invia uno o più pacchetti Link State Request per richiedere informazioni relative a tutti i collegamenti citati nel DD.

Si usa una procedura di poll-response:
- il master invia pacchetti DD (poll);
- lo slave riscontra i pacchetti DD ricevuti inviando a sua volta pacchetti DD (response).

Le risposte sono collegate alle richieste dal numero di sequenza dei pacchetti DD.
Il pacchetto DD contiene una lista di LSA.
Quando uno dei due (master o slave) ha finito continua a inviare DD vuoti finché non finisce anche l'altro.

3. Pacchetto Link State Request (LSR)

Sono usati da un router per richiedere a un router vicino l'invio di uno o più pacchetti LSA. Questi vengono scambiati solo dopo che un router scopre (esaminando i pacchetti DD) che parti del suo database topologico non sono aggiornate.

4. Pacchetto Link State Update (LSU)

Contiene una lista di LSA che il router invia ai suoi router vicini. Sono usati per rispondere a un pacchetto di Link State Request, per la diffusione periodica o in seguito a cambiamenti topologici.
Sono inviati con la tecnica flooding e richiedono un riscontro da fornire con un messaggio di Link State Acknowledgment.

5. Pacchetto Link State Acknowledgment (LSA)

Indica l'avvenuta ricezione di un pacchetto di Link State Update. Eventuali ritrasmissioni si inviano direttamente al vicino che le ha richieste.

#prendinota

Attenzione a non confondere la tecnica LSA (Link State Advertisement) e i relativi pacchetti informativi, con i pacchetti Link State Acknowledgment.

FISSA LE CONOSCENZE

- Elenca i principali IGP che utilizzano il Distance Vector e poi quelli che utilizzano il Link State.
- Qual è la metrica utilizzata dal RIP?
- Descrivi l'header del pacchetto RIPv1.
- Quali sono le caratteristiche fondamentali dell'OSPF?
- Come vengono suddivisi i router nell'OSPF?
- Spiega il ruolo dei virtual link usati nelle aree OSPF.
- Descrivi l'header del pacchetto OSPF.
- Che cos'è la tecnica LSA?

5 PROTOCOLLI DI ROUTING EGP

5.1 Gli Exterior Gateway Protocol

In questa Lezione analizzeremo i protocolli di routing di tipo **EGP** (Exterior Gateway Protocol) noti come **protocolli extradominio** poiché vengono usati per regolare l'instradamento dei pacchetti tra host appartenenti ad Autonomous System diversi.
I protocolli di tipo EGP differiscono da quelli di tipo IGP soprattutto perché ogni AS vuole mantenere una propria autonomia e indipendenza dagli altri e non vuole subire decisioni prese da altri.
Per esempio alcuni AS possono non volere che altri AS instradino il traffico attraverso le loro reti. In altri casi è necessario operare tenendo conto degli accordi internazionali.

I principali EGP sono:

- **EGP**, Exterior Gateway Protocol, è stato il primo EGP a essere standardizzato, disponibile su tutti i router, ormai è considerato un protocollo obsoleto;
- **BGP**, Border Gateway Protocol, definito in ambito IETF, è attualmente il protocollo di tipo EGP usato su Internet, nella sua versione BGP-4;
- **IDRP**, Inter-Domain Routing Protocol, specificato da ISO, è della famiglia Path Vector come BGP.

Nel seguito esaminiamo il protocollo più utilizzato: BGP.

5.2 Il protocollo BGP (Border Gateway Protocol)

BGP (Border Gateway Protocol) è un protocollo di routing tra Autonomous System, attualmente utilizzato sul backbone di Internet ed è in pratica il successore del protocollo EGP.
Le specifiche di BGP sono state più volte riviste: RFC 1105 del 1988 è la prima versione, BGP-2 fu specificato in RFC 1163 nel 1990 e BGP-3 in RFC 1267 nel 1991. La versione BGP-4 è del 2006 (RFC 4271) e ha subito alcuni aggiornamenti, ma resta l'ultima versione specificata.

> BGP è un protocollo molto complesso, usato soprattutto su Internet dove diversi AS sono collegati attraverso gli Internet Service Provider (ISP).

Il protocollo BGP costruisce un grafo di Autonomous System basato sulle informazioni che si scambiano i router in cui ciascun AS viene identificato con il suo AS Number (ASN). La connessione tra due AS si chiama **path** e una collezione di path forma a sua volta un path che viene utilizzato per raggiungere la destinazione.

> BGP è un protocollo di tipo **Path Vector**, evoluzione del Distance Vector, dove nel vettore dei percorsi si elencano tutti gli AS da attraversare per raggiungere una destinazione.

IN ENGLISH PLEASE

Network Working Group — Y. Rekhter, Ed.
Request for Comments: 4271 — T. Li, Ed.
Obsoletes: 1771 — S. Hares, Ed.
Category: Standards Track

January 2006

A Border Gateway Protocol 4 (BGP-4)

Abstract

This document discusses the Border Gateway Protocol (BGP), witch is an inter-Autonomous System routing protocol. The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. This network reachability information includes information on the list of Autonomous Systems (ASs) that reachability information traverses. This information is sufficient for constructing a graph of AS connectivity for this reachability from which routing loops may be pruned, and, at the AS level, some policy decisions may be enforced.

BGP-4 provides a set of mechanisms for supporting Classless Inter-Domain Routing (CIDR). These mechanisms include support for advertising a set of destinations as an IP prefix, and eliminating the concept of network "class" within BGP. BGP-4 also introduces mechanisms that allow aggregation of routes, including aggregation of AS paths.

Il Path Vector risolve il problema dei percorsi ciclici ed è più consono a definire le politiche di routing tra AS rispetto alla semplice distanza.
I cicli vengono evitati poiché quando un router di frontiera di un AS riceve un Path Vector, controlla se il suo AS è già elencato al suo interno:

- se lo è significa che esiste la possibilità di un loop e quel Path Vector non viene considerato;
- altrimenti il Path Vector viene aggiornato con l'indicazione dell'AS di appartenenza e comunicato ai vicini, in quanto considerato corretto.

La **FIGURA 9** mostra un esempio di diffusione del Path Vector.

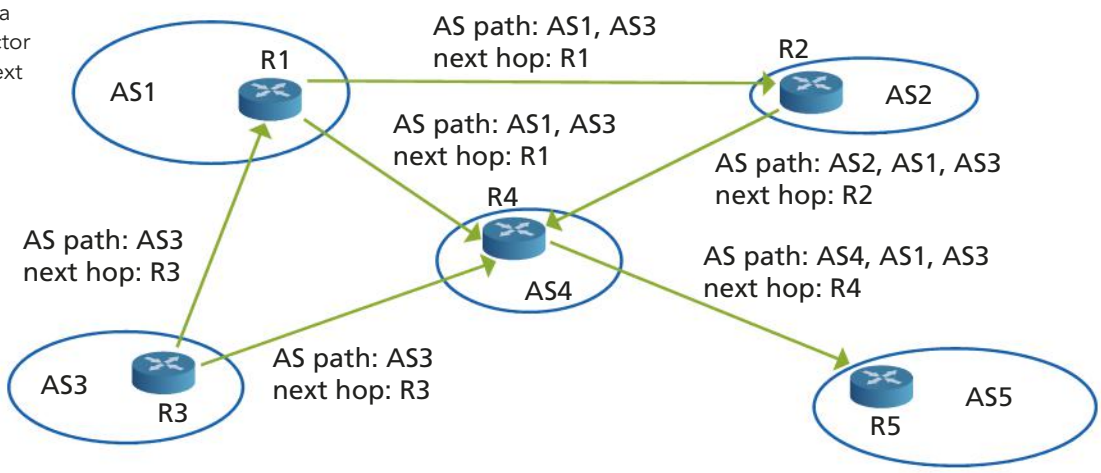


FIGURA 9 Diffusione senza loop da AS3 dei Path Vector con attributi AS path e next hop

A ciascun Path Vector sono associati degli **attributi** che ne specificano la natura. Un determinato attributo può avere le seguenti caratteristiche:

- **well-known**: riconoscibile da tutte le implementazioni BGP, deve essere inoltrato assieme al Path Vector (dopo un eventuale aggiornamento);
- **mandatory**: deve essere presente nel Path Vector;
- **discretionary**: può anche non essere indicato;
- **optional**: può non essere riconosciuto da alcuni router;
- **transitive**: deve essere inoltrato anche se non riconosciuto;
- **non-transitive**: deve essere ignorato se non riconosciuto;
- **partial**: si tratta di un attributo optional-transitive che è stato ritrasmesso senza modifiche da un router perché non lo ha riconosciuto (indica se un determinato Path Vector è stato riconosciuto o meno da tutti i router attraversati).

> Quando un router annuncia un indirizzo di rete attraverso una sessione BGP, include anche alcuni attributi: in BGP si chiama **route** l'indirizzo di rete insieme ai suoi attributi.

Gli attributi più significativi sono i seguenti:

- **origin** (Code = 1) è well-known mandatory e può valere:
  - 0 = IGP: l'informazione è stata ottenuta direttamente dal protocollo di routing operante all'interno dell'AS in cui si trova la destinazione e per cui la si ritiene veritiera;
  - 1 = EGP: l'informazione è stata appresa dal protocollo EGP, che non funziona se vi sono cicli (un percorso caratterizzato da questo valore è peggiore di uno di tipo IGP);
  - 2 = incomplete: serve a indicare che il percorso è stato determinato in altro modo (per esempio statico) oppure è utilizzato per marcare un percorso di AS che è stato troncato perché la destinazione è al momento non raggiungibile;
- **AS path** (Code = 2) è well-known mandatory e consiste nell'elenco degli AS da attraversare lungo il percorso verso la destinazione;
- **next hop** (Code = 3) è well-known mandatory e indica l'indirizzo IP del router di bordo dell'AS che deve essere usato come next hop verso la destinazione specificata.

La precedente Figura 9 mostra un esempio di Path Vector con attributi AS path e next hop.

L'header dei messaggi BGP è lungo 19 ottetti: i primi 16 non sono usati e devono essere messi tutti a 1; i successivi 2 ottetti indicano la lunghezza totale del messaggio e l'ultimo ottetto contiene il codice relativo al tipo di messaggio.

I messaggi possono essere di 4 tipi:

- **Open** è il primo messaggio trasmesso quando viene attivata una connessione TCP verso un router BGP vicino, contiene:
  - informazioni di identificazione dell'AS di chi trasmette;
  - durata del timeout per considerare un vicino non più attivo;
  - dati di autenticazione;
- **UpDate** contiene il Path Vector e i relativi attributi;
- **Notification** è un messaggio di notifica di errori e/o di chiusura della connessione;
- **KeepAlive** è usato per comunicare a un router BGP vicino, in assenza di nuove informazioni di routing, che il trasmettitore è comunque attivo, anche se silente.

LE SESSIONI DI BGP

I router BGP si scambiano informazioni attraverso connessioni TCP (porta 179) chiamate **sessioni BGP** in cui le comunicazioni sono affidabili e le funzionalità di controllo degli errori vengono demandate al livello Transport.

Si distinguono due tipi di sessioni BGP (FIGURA 10):

- sessioni BGP esterne (**eBGP**) instaurate tra router BGP appartenenti ad AS diversi (adiacenze eBGP, i router BGP devono essere connessi direttamente);
- sessioni BGP interne (**iBGP**) instaurate tra router BGP appartenenti allo stesso AS (adiacenze iBGP, i router BGP possono anche non essere direttamente connessi).

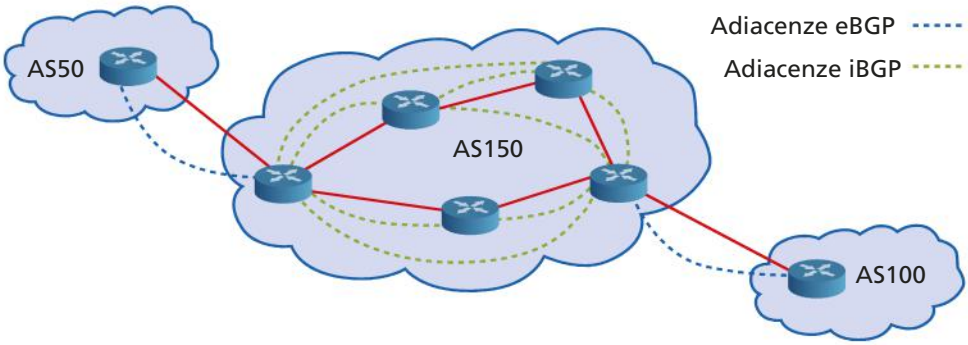


FIGURA 10 Adiacenze eBGP e iBGP

#techwords

Nel protocollo BGP sono chiamati **peer** i router adiacenti che stabiliscono una sessione per lo scambio delle route. L'amministratore di rete deve configurare manualmente il peering sui router BGP in quanto non sono in grado di scoprirsi in automatico.

Quando si instaura la connessione TCP tra due router BGP (**#peer**), nello scambio dei dati che avviene inizialmente, entrambi i router inviano l'uno all'altro la loro intera tabella di instradamento. Man mano che si presentano variazioni della tabella, il router invia aggiornamenti al peer. BGP non richiede periodici invii dell'intera tabella, quindi è necessario che ogni dispositivo che comunica mediante BGP gestisca e mantenga in memoria tutta la tabella di ognuno dei suoi peer. Le informazioni scambiate riguardano la raggiungibilità di reti IP secondo lo schema classless (CIDR).

Le politiche di routing tra AS possono essere di due tipi:

- **export policies**: si comunicano ai vicini solo i Path Vector relativi alle destinazioni verso le quali si vuole permettere il transito;
- **import policies**: dal Path Vector è possibile risalire agli AS da attraversare per raggiungere una destinazione: se nel Path Vector ricevuto da un vicino sono presenti uno o più AS incompatibili con le politiche di routing stabilite, esso viene ignorato.

FISSA LE CONOSCENZE

- Descrivi il principio di funzionamento del BGP.
- Come vengono evitati i loop con il BGP?
- Descrivi qualche attributo associato al Path Vector.
- Descrivi i tipi di messaggi previsti in BGP.
- Che differenza c'è tra sessioni eBGP e iBGP?

6 LE RETI MULTIPROTOCOLLO: MPLS

6.1 MPLS e la tecnica Label Switching

Nelle precedenti Lezioni abbiamo visto le tecniche e i protocolli utilizzati nelle reti IP per l'instradamento in rete dei pacchetti. Ora vediamo un'altra tecnologia, sviluppata in ambito IETF a partire dalla metà degli anni Novanta, che cambia la tecnica di instradamento, con l'obiettivo di renderla più generale, ossia valida non solo per IP, ma per qualsiasi protocollo di livello Network, quindi multiprotocollo!
Per realizzare questo obiettivo è necessario svincolarsi dalla modalità di inoltro dei pacchetti basata sull'indirizzo IP di destinazione. In IETF si è quindi seguita la strada di definire un nuovo standard: **MPLS, Multiprotocol Label Switching**, che utilizza la tecnica di commutazione di etichetta (**label switching**). L'architettura MPLS è specificata in RFC 3031 del 2001 e successivi aggiornamenti.

> La tecnica **label switching** consiste nell'utilizzare un'etichetta (**label**) per identificare ogni pacchetto che entra nella rete core MPLS e di effettuare il forwarding usando la label al posto dell'indirizzo IP di destinazione.

La rete MPLS distingue i router in due categorie:
- **CLSR** (Core Label Switching Router): router interni al dominio MPLS;
- **ELSR** (Edge Label Switching Router): router che si trovano al confine del dominio MPLS.

La FIGURA 11 mostra una possibile disposizione dei router in un dominio MPLS.

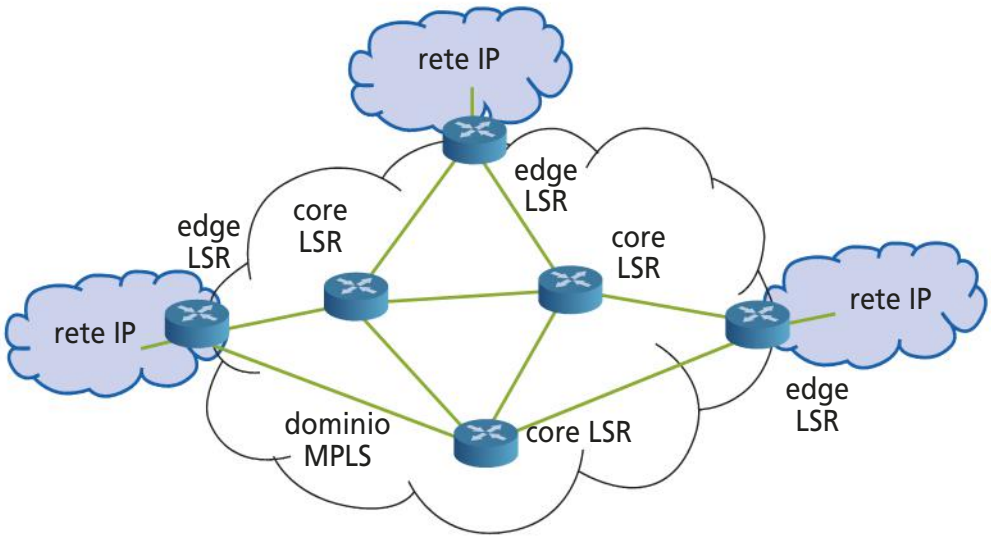


FIGURA 11 Il dominio MPLS

I pacchetti provenienti dalla rete IP vengono aggregati in una classe di appartenenza, che prende il nome di **FEC** (Forwarding Equivalent Class), in base a due parametri:
- l'ELSR di destinazione;
- il tipo di servizio richiesto.

In base a questi due parametri, l'ELSR di ingresso inserisce nel pacchetto una label costituita da due parti contenenti:

- l'identificativo dell'interfaccia d'uscita (**port**) del router ELSR verso il primo CLSR che il pacchetto dovrà attraversare (**next hop**) dentro il dominio MPLS;
- l'identificativo del nodo di uscita, cioè dell'ELSR di destinazione.

Dall'ELSR di ingresso a quello di uscita dal dominio MPLS, tutte le operazioni di forwarding verranno effettuate utilizzando solo la label e non l'header IP, che quindi non verrà più letto fino all'ELSR di destinazione.
I CLSR attraversati leggono la label, analizzano una tabella di corrispondenza **FEC ↔ label** (simile alle tabelle di routing di IP ma di dimensioni più ridotte), modificano la label in base all'interfaccia d'uscita verso il successivo CLSR o ESLR individuato (next hop) e inoltrano il pacchetto.
L'ultimo nodo attraversato all'interno del dominio MPLS, ELSR d'uscita, elimina la label e instrada il pacchetto nuovamente sulla base dell'indirizzo IP.

6.2 La gestione delle label in MPLS

#techwords

Stack
Una struttura dati a cui si accede in modalità LIFO (Last In First Out): i dati sono estratti in modalità inversa a quella in cui sono stati inseriti.

L'architettura MPLS prevede un modello molto flessibile di uso delle label, in cui un singolo pacchetto può trasportare un numero m, con $m \geq 1$, di label organizzate in m livelli gerarchici (**label #stack**).
La label di livello gerarchico più basso sarà indicata come label di livello 1, mentre la label di livello gerarchico più elevato sarà indicata come label di livello m (FIGURA 12).

FIGURA 12 Label stack

In ogni caso, qualsiasi sia il numero di label trasportate, l'instradamento di un pacchetto in un LSR dipende sempre ed esclusivamente dalla label di livello gerarchico maggiore trasportata dal pacchetto stesso.
La disponibilità di un numero qualsiasi di livelli di etichetta è particolarmente utile per la creazione di tunnel MPLS e per la definizione di livelli gerarchici all'interno di un dominio MPLS.

Una label MPLS (RFC 3032 e successivi aggiornamenti) ha una lunghezza uguale a 32 bit e il suo formato è mostrato nella FIGURA 13.

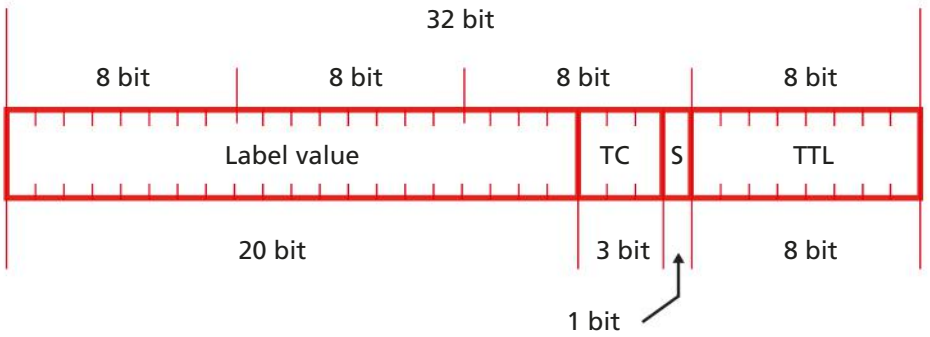


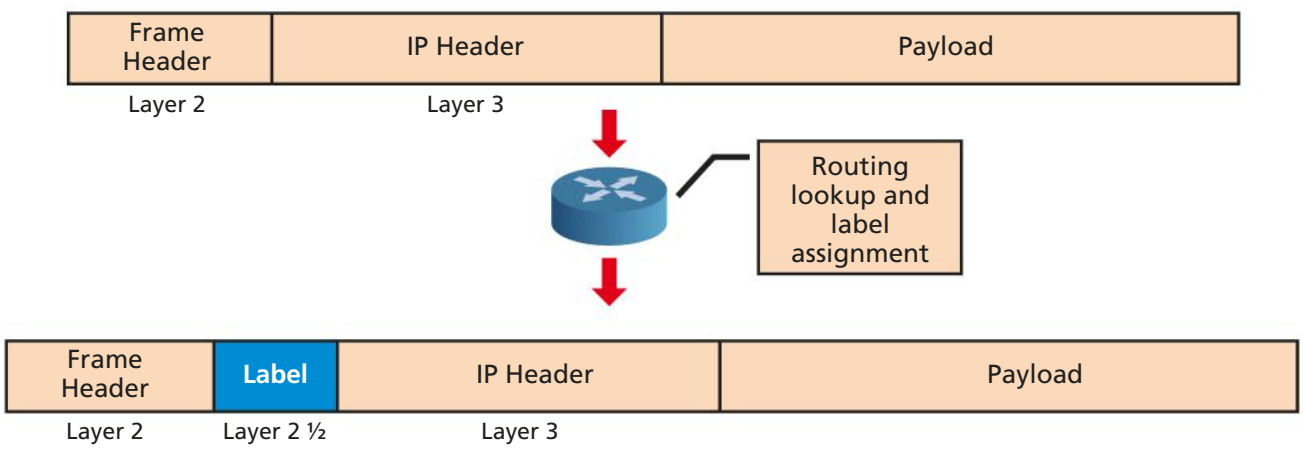
FIGURA 13 Formato di una label MPLS

Il significato dei campi è il seguente:

- **Label value** (20 bit): indica il valore della label ed è utilizzato come **indice** per l'accesso alla tabella di routing dell'LSR in cui sono indicati l'identificatore della porta d'uscita (**port**) verso cui deve essere instradato il pacchetto e il valore della label sulla tratta successiva;
- **Traffic Class** (TC) (3 bit): in origine l'utilizzo di questo campo non era stato definito (si chiamava Experimental Use), in seguito l'RFC 5462 assegnò il nuovo nome Traffic Class specificando che può essere usato per indicare una classe di servizio per la QoS;
- **Bottom of Stack** (S) (1 bit): è un flag che informa il router del fatto che seguono altre label nello stack (bit = 0) oppure che è l'ultima label dello stack (bit = 1);
- **Time To Live** (TTL) (8 bit): il campo ha un significato del tutto analogo a quello del campo omonimo presente nell'header di un datagramma IP e indica il numero massimo di salti che ancora il pacchetto può eseguire in rete prima che raggiunga la destinazione o venga scartato. Il valore di tale campo è decrementato di uno ogni volta che viene elaborato da un LSR.

La label viene inserita tra l'header di livello 2 e l'header di livello 3 del modello ISO/OSI e di conseguenza è nota come **label di livello 2,5** (FIGURA 14).

FIGURA 14 Label di livello 2,5



LABEL SWITCHING PATH (LSP)

Il cammino seguito da un pacchetto in una rete MPLS è denominato **Label Switching Path** (LSP).

In particolare, per uno specifico pacchetto P, un LSP di livello m è definito dalla sequenza di LSR (R_1, R_2, …, R_n) che soddisfano le seguenti proprietà:

- l'ELSR R_1, che rappresenta il punto di inizio dell'LSP, è il router che applica la label di ordine m al pacchetto P;
- in tutti i CLSR R_i ($1 < i < n$) il pacchetto P sarà instradato attraverso l'esame della label di ordine m e, inoltre, il numero di label contenute nel pacchetto sarà sempre superiore a m;
- se nel transito tra due LSR, R_i ed R_{i+1}, il pacchetto P attraversa altri elementi di rete che effettuano l'instradamento sulla base di una label diversa da quella di ordine m, per esempio di ordine $m + k$, ciò avviene solo se altre label k addizionali sono state aggiunte dal router R_i e da altri elementi di rete intermedi.

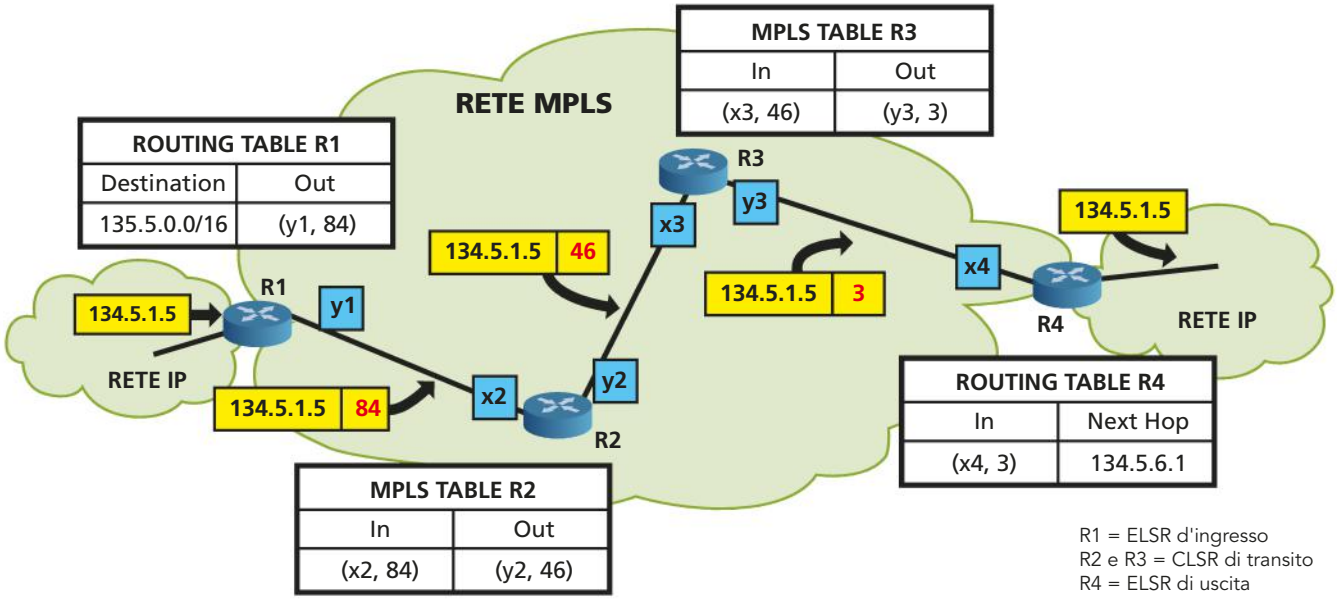
In altre parole, un LSP di ordine *m* per un pacchetto P è dato dalla sequenza di router in cui il primo LSR (ELSR di ingresso) applica la label di ordine *m*; i router intermedi (CLSR di transito) eseguono l'instradamento elaborando e sostituendo la label di ordine *m*; l'ultimo router (ELSR di uscita) provvede all'eliminazione della label di ordine *m*.

Riassumendo, le operazioni che vengono effettuate sul pacchetto in transito in una rete MPLS in relazione alle label sono sostanzialmente 3:

- **pushing**: inserimento della label effettuata dall'ELSR d'ingresso;
- **swapping**: aggiornamento della label, effettuata contestualmente all'operazione di commutazione, dai CLSR interni;
- **popping**: eliminazione della label, effettuata dall'ELSR d'uscita.

esempio

La figura sotto illustra i concetti esposti nel caso di gestione di etichette di ordine $m = 1$, mostrando il percorso di un pacchetto con indirizzo IP di destinazione 134.5.1.5/16.



Il percorso di un pacchetto all'interno di tale rete MPLS è riassumibile nei seguenti passi:

- **pushing**: l'ELSR di ingresso, R1, legge la propria routing table e in base all'IP contenuto nel pacchetto identifica la porta d'uscita (y1) verso cui deve essere instradato il pacchetto e aggiunge la label con il valore per la tratta successiva (Label value = 84). Da qui in poi il pacchetto procederà con l'instradamento MPLS e non IP;
- **swapping**: il primo CLSR di transito, R2, utilizza il Label value ricevuto (84) come **indice** per l'accesso alla propria tabella di routing e quindi identifica la porta d'uscita (y2) verso cui deve essere instradato il pacchetto e il Label value per la tratta successiva (46);
- identico lavoro (**swapping**) farà il secondo CLSR di transito, R3: grazie all'**indice** 46 userà la porta y3 e scriverà nel Label value il valore 3;
- **popping**: giunto all'uscita della rete MPLS, l'ELSR di uscita, R4, eliminerà la label di livello 1 e il pacchetto tornerà a procedere verso la destinazione con l'instradamento IP.

6.3 I servizi offerti dalle reti MPLS

La tecnica MPLS è in grado di fornire a una rete IP, oltre a un miglioramento delle prestazioni, una serie di funzionalità addizionali che elenchiamo di seguito:

1. **supporto alla qualità del servizio**: una rete connectionless, quale è una rete IP, non è in grado di soddisfare in modo pieno e affidabile le richieste di trasferimento con garanzia di determinati parametri di qualità. Una rete basata su un modo di trasferimento orientato alla connessione ha invece la possibilità di gestire in maniera molto efficiente gli aspetti di qualità del servizio; la tecnica MPLS, che definisce una modalità di trasferimento connection-oriented in una rete IP, costituisce quindi la base per la fornitura flessibile di servizi con prefissati livelli di qualità;
2. **ingegneria del traffico**: instaurazione di cammini in rete in modo da ottimizzare l'utilizzo delle sue risorse. Questa funzione, normalmente indicata con il termine Ingegneria del traffico (Traffic Engineering, TE), non può essere realizzata, almeno in modo semplice, mediante le tecniche tradizionali di instradamento utilizzate nelle reti IP. La ragione di questo risiede nel fatto che nelle reti IP il traffico tra due punti segue sempre un'unica via, mentre con MPLS tra due punti può essere utilizzata una pluralità di cammini; i flussi di traffico possono essere quindi instradati utilizzando tutti i cammini disponibili in modo da ottenere una distribuzione uniforme del traffico sulle risorse di rete e, di conseguenza, un miglioramento complessivo delle prestazioni di rete;
3. **realizzazione di VPN**: definire reti private virtuali (Virtual Private Network, VPN) all'interno di una rete IP. Mediante questo servizio il traffico tra punti d'accesso remoti può transitare in modo trasparente e completamente isolato dagli altri flussi di traffico all'interno della rete IP con conseguenti vantaggi sia per la gestione che per i requisiti di sicurezza;
4. **fault tolerance**: al momento dell'instaurazione di un cammino in rete, vengono anche predisposti cammini alternativi, detti di protezione o di backup, da utilizzare in caso di guasto di uno o più tratte del cammino principale per il nuovo instradamento del flusso di traffico. Questa soluzione consente di raggiungere dei tempi di riconfigurazione dei cammini molto inferiori a quelli ottenibili utilizzando i normali protocolli di routing IP e praticamente comparabili con le tipiche tecniche a commutazione di circuito.

FISSA LE CONOSCENZE

- Che cosa si intende per multiprotocol riferito alle reti MPLS?
- Quali sono le due categorie di router nelle reti MPLS?
- Quali sono i due parametri che determinano la classe di appartenenza (FEC)?
- Da cosa è costituita la label inserita nel pacchetto?
- Chi ha il compito di eliminare la label?
- Quali sono le tre operazioni che vengono effettuate su un pacchetto in transito?
- Che cosa contiene il campo Label value e com'è usato?
- Che cos'è il Label Switching Path (LSP)?

7 LA GESTIONE DELLE TABELLE DI ROUTING

In questa Lezione vedremo i comandi CLI usati nei sistemi Windows e Linux per la gestione delle tabelle di routing di un host, mentre nelle successive lavoreremo sul sistema IOS dei router con Packet Tracer, sia da interfaccia grafica, sia da CLI.
Inoltre, l'ultimo paragrafo presenta la gestione del routing a livello di AS con database distribuiti gestiti dai vari RIR (Regional Internet Register).

7.1 Il comando route nei sistemi Windows

Il comando **route** consente di visualizzare, impostare o modificare le entry della routing table di un host di una rete locale, anche in presenza di più schede di rete.

Il formato del comando route è il seguente:

ROUTE [-f] [-p] [-4 | -6] comando [destinazione] [MASK netmask] [gateway] [METRIC metrica] [IF interfaccia]

Vediamo in dettaglio come si compone:

- **opzioni**:
  -f cancella tutte le entry nella tabella di routing, tranne quelle con netmask 255.255.255.255, quelle di loopback e quelle di multicast. Se -f si usa insieme a uno dei comandi, le tabelle vengono cancellate prima dell'esecuzione del comando;
  -p usata insieme al comando **ADD** rende **persistente** una route in caso di riavvio del sistema; infatti, di default, al riavvio le route definite con il comando route ADD non sono mantenute. Se si usa con il comando PRINT visualizza tutte le entry persistenti;
  -4 impone l'utilizzo di IPv4;
  -6 impone l'utilizzo di IPv6;
- **comando**:
  PRINT visualizza una route;
  ADD aggiunge una route;
  DELETE elimina una route;
  CHANGE modifica una route esistente;
- **destinazione**: specifica l'indirizzo IP della rete o dell'host di destinazione oppure la default route 0.0.0.0;
- **netmask**: specifica una maschera di sottorete da associare alla route entry. Se non specificato viene utilizzato il valore predefinito 255.255.255.255;
- **gateway**: specifica il gateway da usare per raggiungere la destinazione;
- **metrica**: un valore intero tra 1 e 9999, che indica il costo della route, è utilizzato nel caso sia necessario scegliere tra più route presenti nella routing table;
- **interfaccia**: numero di interfaccia, identifica l'interfaccia dalla quale si raggiunge la destinazione; se omesso, si individua l'interfaccia dall'indirizzo del gateway.

Windows determina in automatico la metrica sulla base dell'ampiezza di banda dell'interfaccia. In questo modo le interfacce più veloci creano route con il costo più basso. È possibile disabilitare questa funzionalità andando nelle proprietà avanzate di TCP/IP di ciascuna delle interfacce di rete presenti sul computer.
Nei comandi PRINT e DELETE possono essere usati i caratteri jolly * e ? (#wildcard) per i parametri destinazione e gateway, per visualizzare solo le route corrispondenti.

#prendinota

Case-insensitive
Ricordiamo che la CLI di Windows non fa differenza tra caratteri maiuscoli e minuscoli.

#techwords

Wildcard
È un carattere che si inserisce all'interno di una stringa per rappresentare non se stesso, bensì un insieme di altri caratteri o sequenze di caratteri.

L'asterisco significa "una qualsiasi stringa", mentre il punto interrogativo indica "un qualsiasi carattere". Esempi di uso di wildcard sono:
170.*.1 192.168.* 127.* *224* 10.0.0.10? 192.168.1.1?0

Vediamo ora alcuni esempi di utilizzo del comando route.

- **route print** senza altri parametri, visualizza l'intero contenuto della routing table. Un esempio dell'output di questo comando è mostrato nella Figura 15.
- **route print** [destinazione] [gateway] visualizza una route specifica, relativa a un dato indirizzo IP di destinazione e/o indirizzo IP del gateway di inoltro:

```
route print 192.168.*
```

- **route -p print** [destinazione] [gateway] visualizza gli instradamenti persistenti.
- **route add** [destinazione] [**mask** netmask] [gateway] aggiunge una nuova route:

```
route add 192.168.1.48 mask 255.255.255.0 192.168.1.1
route add 0.0.0.0 mask 0.0.0.0 192.168.1.1     // aggiunge una default route
```

- **route -p add** [destinazione] [**mask** netmask] [gateway] aggiunge una nuova route persistente.
- **route add** [destinazione] [**mask** netmask] [gateway] [**if** interfaccia] aggiunge una nuova route verso la destinazione 192.168.1.48 usando l'interfaccia identificata dal numero 3 (l'elenco delle interfacce e del corrispondente numero, si ottiene con il comando route print):

```
route add 192.168.1.48 mask 255.255.255.0 192.168.1.1 if 3
```

- **route delete** [destinazione] [gateway] cancella un instradamento:

```
route delete 192.168.1.48 mask 255.255.255.0
route delete 192.168.*  // cancella tutte le route che iniziano con 192.168
```

- **route -f** cancella tutta la routing table.
- **route change** [destinazione] [**mask** netmask] [gateway] si usa per modificare un instradamento, per esempio se si vuol cambiare il gateway:

```
route change 192.168.1.48 mask 255.255.255.0 192.168.1.254
```

esercizio

➜ PROBLEMA

Visualizzare le informazioni di routing su un host Windows e descrivere quelle più significative distinguendo tra i dati IPv4 e quelli IPv6.

➜ SVOLGIMENTO

Per visualizzare la tabella di routing, il sistema Windows mette a disposizione il comando route, eseguibile dall'applicazione Prompt dei comandi.

La FIGURA 15 mostra il risultato dell'esecuzione del comando **route print**.

In riferimento alla Figura 15, vediamo alcune righe significative:

1) nella sezione IPv4:

- **0.0.0.0** è la **default route**, quella che permette di scambiare traffico con altre reti, e punta al default gateway 192.168.1.1;
- la riga con **192.168.1.0** è quella del traffico diretto a tutti gli host della **LAN**, mentre l'host **192.168.1.9** è il computer che ha emesso il comando;
- il **directed broadcast 192.168.1.255** indica solo che si trova anch'esso sulla stessa interfaccia (.9);

FIGURA 15 Il comando route print su Windows

```
C:\>route print
===========================================================================
Elenco interfacce
 11...ec 8e b5 46 ea 33 ......Realtek PCIe GbE Family Controller
 14...b8 81 98 36 66 a0 ......Microsoft Wi-Fi Direct Virtual Adapter
 20...ba 81 98 36 66 9f ......Microsoft Wi-Fi Direct Virtual Adapter #3
  8...b8 81 98 36 66 a3 ......Bluetooth Device (Personal Area Network)
 10...b8 81 98 36 66 9f ......Intel(R) Dual Band Wireless-AC 3165
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete            Mask          Gateway     Interfaccia Metrica
          0.0.0.0          0.0.0.0    192.168.1.1      192.168.1.9     25
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
      192.168.1.0    255.255.255.0         On-link       192.168.1.9    281
      192.168.1.9  255.255.255.255         On-link       192.168.1.9    281
    192.168.1.255  255.255.255.255         On-link       192.168.1.9    281
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link       192.168.1.9    281
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link       192.168.1.9    281
===========================================================================
Route permanenti:
  Nessuna

IPv6 Tabella route
===========================================================================
Route attive:
 Interf Metrica Rete Destinazione      Gateway
  1    331 ::1/128                  On-link
 11    281 fe80::/64                On-link
 11    281 fe80::75ba:c72a:40f7:f0b1/128
                                    On-link
  1    331 ff00::/8                 On-link
 11    281 ff00::/8                 On-link
===========================================================================
Route permanenti:
  Nessuna
```

2) nella sezione IPv6:

- **::1/128** è l'equivalente del 127.0.0.1 visibile nella sezione IPv4 e indica l'indirizzo IP di loopback (o localhost);
- la riga con **fe80::/64** è l'indirizzo **Link local** della **LAN** collegata all'interfaccia **11**;
- la riga con **fe80::75ba:c72a:40f7:f0b1/128** è l'indirizzo **Link local dell'host** che ha dato il comando; si noti che i 64 bit dell'Interface ID, non contengono al centro la sequenza ff:fe, bensì 2a:40, a indicare che Windows ha calcolato l'ID in modo random e non in base all'algoritmo EUI-64 visto nella Lezione 2 dell'Unità 4 (in WindowsXP si usava EUI-64);
- **ff00::/8** sono gli indirizzi IPv6 riservati al multicast, equivalgono a quelli del tipo 224.0.0.0 presenti nella sezione IPv4.

Nella colonna Gateway la stringa "On-link" indica che i relativi indirizzi di host hanno avuto accesso diretto trovandosi nella stessa subnet e non necessitano di essere inviati al gateway.

7.2 Il comando route nei sistemi Linux

Molti amministratori e utenti dei sistemi Linux usano i comandi:

- **ifconfig**, per gestire la configurazione delle interfacce di rete;
- **route**, per gestire le tabelle di routing.

Da un po' di anni, questi e altri comandi della shell di Linux sono stati #deprecati a favore del comando **ip**: un unico comando con cui gestire la configurazione delle interfacce, il routing e il tunneling.

La sua sintassi è:

```
ip [ OPTIONS ] OBJECT { COMMAND | help }
```

L'object può essere scritto in forma completa o abbreviata (short), per esempio: **address** oppure **addr**.
Il comando ip deve essere dato da utente superuser o facendolo precedere dal comando sudo.

Nel caso del routing, si usa l'object **route** e alcuni sottocomandi, i più usati sono: **list**, **add** e **del**.

- **ip route**: visualizza e modifica la routing table del kernel Linux; ha sostituito il comando netstat –r.
- **ip route add**: configura una route.
  Per esempio:
  - `ip route add default via 192.168.1.1`
    aggiunge l'indirizzo del default gateway nella tabella di instradamento;
  - `ip route add 10.0.0.0/24 via 172.16.10.1`
    imposta una route per raggiungere la rete 10.0.0.0 attraverso il gateway 172.16.10.0;
  - `ip -6 route add 2000::/3 via 2001:0d58:0:6a5f`
    imposta una route con indirizzi IPv6.

Il kernel Linux è in grado di gestire molteplici routing table diverse.
Il comando:

```
ip rule show
```

permette di visualizzare le regole definite.

7.3 Internet Routing Registries: il comando whois

Gli **IRR** (Internet Routing Registry) sono un insieme, sempre più grande, di database distribuiti contenenti le politiche di routing degli AS allocati dai diversi Regional Internet Registry (RIR) e da altre organizzazioni nazionali e internazionali.
Un esempio di IRR è **RADb – Routing Assets Database**, gestito da Merit Networks, un'organizzazione no-profit. RADb è un registro pubblico contenente informazioni sul routing di rete. Offre alle aziende gli strumenti per registrare le loro risorse relative al routing, tenere traccia delle modifiche e determinare le prestazioni delle risorse di rete.

#prendinota

iproute2
Il comando **ip** fa parte del package iproute2 che si trova già installato sulle distribuzioni più recenti di Linux.

#techwords

Deprecato
Indica un software, una procedura, un comando, ecc. ormai superato, di cui si sconsiglia l'uso a favore di una versione aggiornata.

Collegandosi alla home page *www.radb.net* è possibile usufruire di un servizio di **query** in grado di interrogare il registro pubblico e fornire le informazioni sull'AS o l'indirizzo IP specificato.

Gli IRR possono essere interrogati tramite il **client whois**:

- nella shell **Linux** whois è già disponibile;
- da web client, per esempio: *www.radb.net*;
- per i sistemi **Windows**: occorre installare un client whois che fa parte delle **SysInternals Network Utilities**:
  https://docs.microsoft.com/en-us/sysinternals/downloads/networking-utilities.
  Nella cartella scaricata si trovano gli eseguibili, copiare quello adatto al proprio sistema, per esempio **whois64.exe**, nella cartella System32 di Windows, così potrà essere richiamato nella finestra del Prompt dei comandi:

```
whois [-v] domainname [whois.server]
```

Con il comando whois è possibile richiedere informazioni su un indirizzo IP o un numero di Autonomous System:
whois <network_IP>
whois AS <Autonomous_System_Number>

Di seguito mostriamo alcuni esempi di query:

```
whois -h whois.radb.net 128.223.0.0/16
whois -h whois.radb.net AS8
```

Il risultato della query è una breve descrizione testuale in forma libera dell'oggetto. Le informazioni possono essere usate per contattare gli amministratori.
Anche sui siti web dei RIR è disponibile un client whois: nella FIGURA 16 si mostra la pagina del web client whois di **RIPE** NCC (Réseaux IP Européens Network Coordination Centre) all'indirizzo *www.ripe.net/whois*.

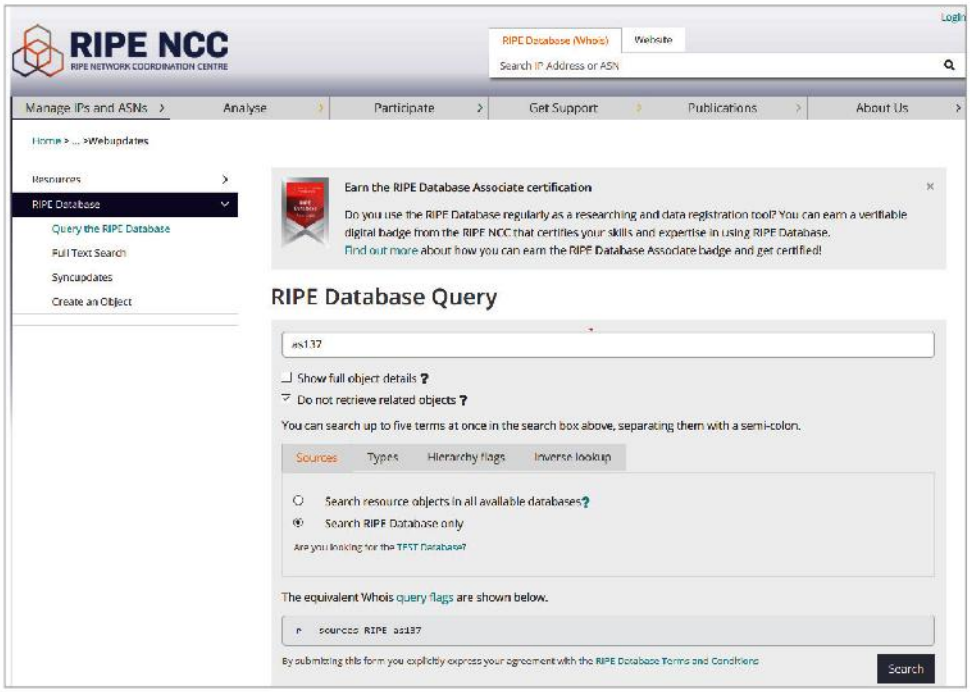


FIGURA 16 RIPE Database Query

Per esempio la richiesta di informazioni sull'**AS137** (che è l'AS Number della GARR Italian Academic and Research Network) produce la risposta riportata nella **FIGURA 17**.

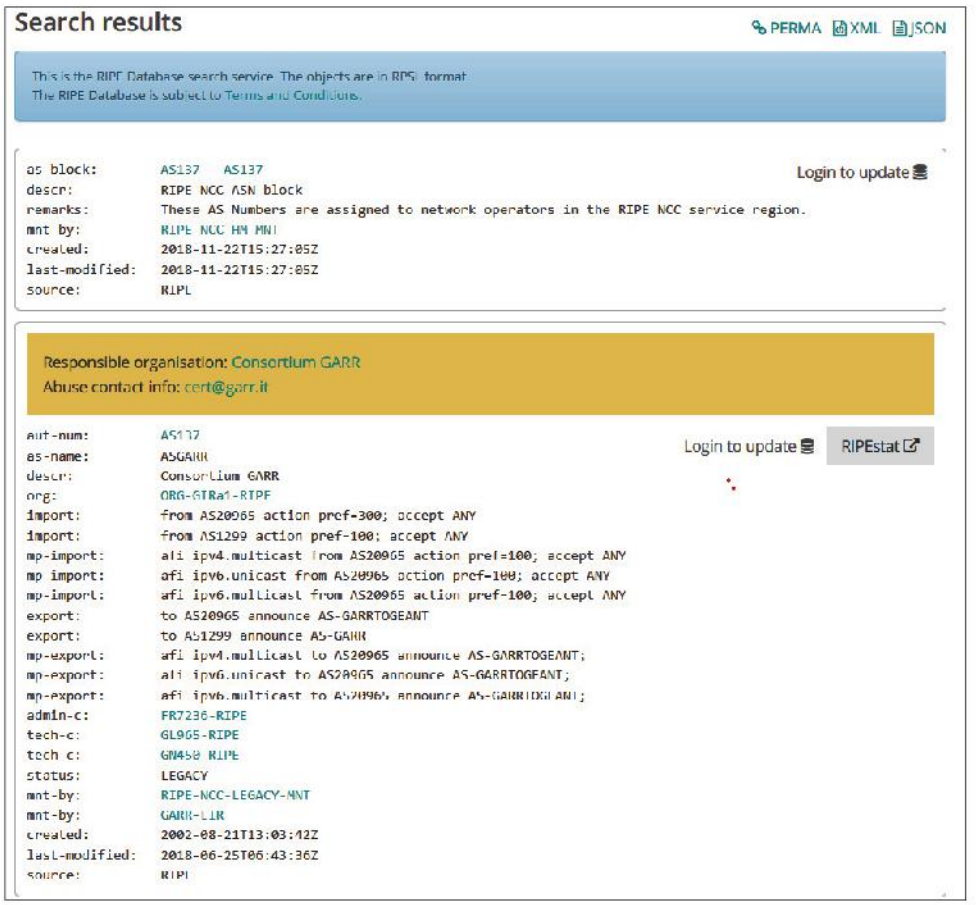


FIGURA 17 Le informazioni sulla GARR (AS137)

Se si facesse la stessa richiesta su RADb si otterrebbero le stesse informazioni, tranne quelle sugli amministratori, che RADb omette rimandando al source dell'informazione, che nel caso del GARR è il RIPE, presso cui l'AS è registrato.

Come descritto nella Lezione sul protocollo BGP, ogni espressione sulle politiche di import è specificata usando un attributo import.
Infatti, nella riga 5 della Figura 17 troviamo:

import: from AS20965 action pref=300; accept ANY

significa che le route di AS20965 sono accettate da AS137 con preferenza 300.

Lo stesso vale per l'attributo export per le politiche di export.
Per esempio la riga 11:

export: to AS1299 announce AS-GARR

significa che la GARR annuncerà le route verso AS1299.

FISSA LE CONOSCENZE

- A che cosa serve il comando route?
- Quale comando occorre usare per aggiungere un instradamento persistente?
- A che cosa serve il comando whois?

8 PACKET TRACER: CONFIGURAZIONE DEL ROUTING STATICO

8.1 Definire la routing table manualmente

Nel seguente esercizio vedremo come impostare le route statiche su un router, così da rendere raggiungibili alcune reti che non sono direttamente connesse al router.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Configurare 2 router e connetterli tramite linea seriale. Il primo router è presso una piccola azienda (Small Office) e collega 2 reti LAN con vari computer, il secondo è il router del provider (ISP). Per semplicità supponiamo che il server su cui lavorano in rete alcuni computer sia direttamente collegato al router ISP. Una volta effettuati tutti i collegamenti e verificato che il livello Physical sia funzionante, procedere con le seguenti operazioni:

1) verificare con il comando ping la mancanza di connettività fra i Personal Computer sulle LAN collegate al router Small Office e il server nella rete locale del router ISP;
2) configurare sul router ISP il routing statico verso le LAN di Small Office;
3) configurare sul router Small Office il routing statico verso la LAN di ISP;
4) verificare con il comando ping la connettività fra gli host delle reti di Small Office e il server sulla rete di ISP.

➜ ANALISI DEL PROBLEMA

Quando deve trasmettere un pacchetto verso una destinazione, il router consulta la propria **routing table** per:

- conoscere l'esistenza della rete di destinazione (funzione di routing);
- conoscere attraverso quale delle sue interfacce inviare i dati per raggiungere tale rete (funzione di forwarding).

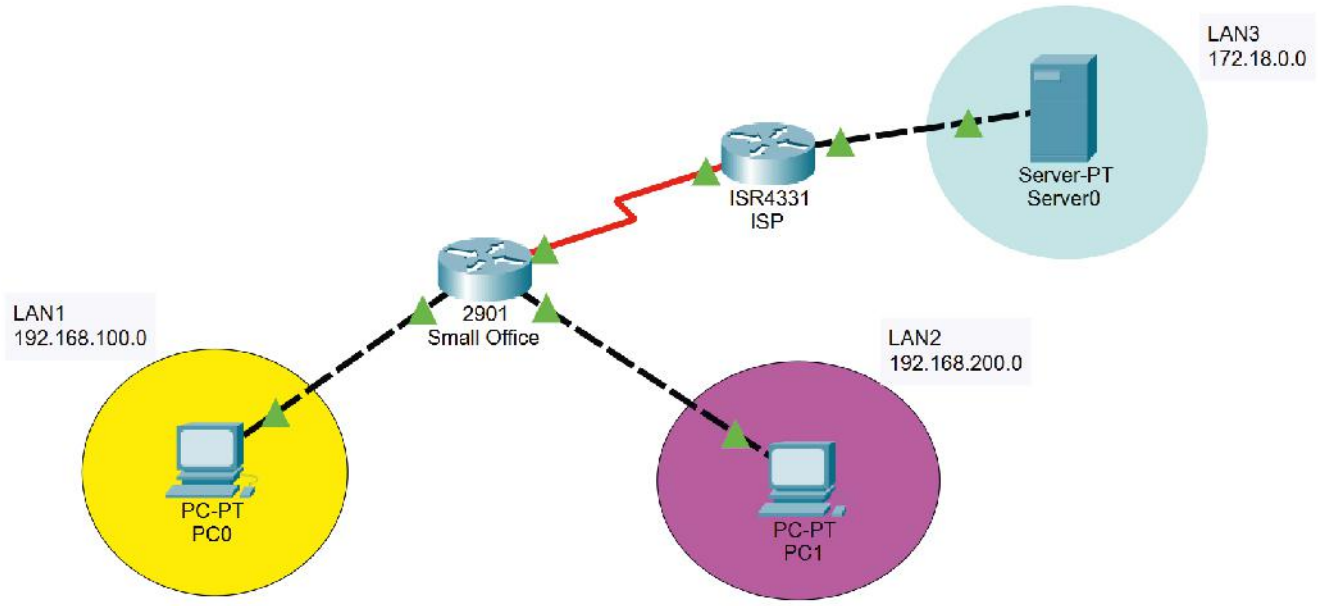
La routing table è dinamica e viene aggiornata:

- in automatico, per le reti collegate alle interfacce del router (**reti direttamente connesse**);
- manualmente, con opportune istruzioni inserite dall'amministratore di rete (**routing statico**);
- tramite opportuni protocolli che permettono ai router direttamente interconnessi di scambiarsi informazioni sulla raggiungibilità delle reti (**routing dinamico**).

Nella routing table può essere presente la **default route**, usata dal router gateway per instradare tutto il traffico generato dalla LAN verso Internet.
Per quanto riguarda le reti direttamente connesse, sui router Cisco gli indirizzi IP di rete vengono caricati nella tabella di routing quando le corrispondenti interfacce passano allo stato di attività piena (**interface up**) e connettività piena a livello Data Link (**protocol up**). Infine, ogni qual volta il router non è in grado di definire una route verso la rete di destinazione, scarta i pacchetti senza inoltrarli.

➜ SVOLGIMENTO

Realizziamo con Packet Tracer il seguente scenario, collegando i vari dispositivi (per un ripasso su collegamenti e configurazioni vedi la Lezione 7 dell'Unità 1 e gli esercizi di laboratorio delle Lezioni 6 e 7 dell'Unità 3).



Per il router Small Office scegliamo il modello **2901** e per il router ISP il modello **ISR 4331**. In entrambi sono presenti delle interfacce GigabitEthernet che useremo per i collegamenti LAN, mentre per il collegamento WAN tra i 2 router useremo un'interfaccia seriale.
Non essendo presente nella configurazione di base, sarà necessario inserire una scheda di rete in uno slot libero dei router. Nella finestra di dialogo del router selezionare quindi la scheda **Physical**, spegnere il router, installare il modulo **NIM-2T** sul router ISP e **HWIC-2T** sul router Small Office. Riaccendere i router e mettere up ciascuna interfaccia di rete utilizzata nei collegamenti (Config → Port Status "on").

Il piano di indirizzamento delle 3 reti locali è il seguente:

Rete locale	Network address	Broadcast address	Host address range
LAN1	192.168.100.0/24	192.168.100.255	192.168.100.1 ... 192.168.100.254
LAN2	192.168.200.0/24	192.168.200.255	192.168.200.1 ... 192.168.200.254
LAN3	172.18.0.0/16	172.18.255.255	172.18.0.1 ... 172.18.255.254

Su ciascun dispositivo si effettuerà la seguente configurazione di rete:

Host	Interface	IP address	Default gateway
PC0	FastEthernet0	192.168.100.2/24	192.168.100.1
PC1	FastEthernet0	192.168.200.2/24	192.168.200.1
Server0	FastEthernet0	172.18.0.10/16	172.18.0.1
Small Office	GigabitEthernet0/0	192.168.100.1/24	
Small Office	GigabitEthernet0/1	192.168.200.1/24	
Small Office	Serial0/3/0	197.85.8.2/24	
ISP	GigabitEthernet0/0/0	172.18.0.1/16	
ISP	Serial0/1/0	197.85.8.1/24	

#prendinota

Per semplicità inseriamo un solo computer in ciascuna LAN, così da focalizzare la trattazione sulla gestione del routing. In uno scenario più realistico avremmo più PC per ogni LAN collegati a uno o più switch, a loro volta connessi al router, secondo la tipica topologia a stella estesa.

La colorazione **verde** di tutti i triangolini presenti alle estremità dei collegamenti segnala che c'è la connettività a livello fisico.

1) Test sulla connettività di rete

Per eseguire i test di connettività, partendo dall'interno della finestra di dialogo per la configurazione del PC, occorre selezionare la scheda **Desktop** e l'icona **Command Prompt**, quindi digitare il comando **ping** seguito dall'indirizzo di rete dell'host da raggiungere.

- Eseguire i seguenti test di connettività prima da PC0 e poi da PC1:
  – verso il proprio default gateway;
  – verso il PC sull'altra LAN di Small Office;
  – verso l'interfaccia Serial0/3/0 di Small Office.
  Tali test dovrebbero dare tutti esito **positivo**.
- Eseguire poi i seguenti test:
  – da PC0 a interfaccia Serial0/1/0 sul router ISP;
  – da PC1 a interfaccia Serial0/1/0 sul router ISP;
  – da PC0 verso Server0;
  – da PC1 verso Server0.
  Tali test dovrebbero avere esito **negativo**.

I test di ping da un host su LAN1 o LAN2 verso la seriale del router ISP falliscono perché **ISP non conosce un percorso** (route) di ritorno verso le reti 192.168.100.0/24 e 192.168.200.0/24; di conseguenza non è in grado di instradare verso PC0 oppure PC1 le risposte ai messaggi ping da essi generati.
Il test ping verso il server fallisce in quanto il router **Small Office non conosce un percorso** (route) verso la rete 172.18.0.0/16 dove risiede Server0.

2) Configurazione su ISP del routing statico verso LAN1 e LAN2

La procedura di configurazione del routing statico prevede il caricamento manuale delle route verso le reti remote su ogni singolo router. Per definire ogni route occorre conoscere:

- l'indirizzo IP della rete remota e subnet mask;
- l'indirizzo IP dell'interfaccia del router attraverso cui deve passare la route (next hop).

Sul router ISP è necessario configurare 2 route verso LAN1 e LAN2:
– al termine del boot del router, selezionare la scheda **Config**;
– alla voce ROUTING, selezionare **Static**;
– nella finestra che si apre inserire i parametri della destinazione remota, per **LAN1**:
Network 192.168.100.0 Mask 255.255.255.0 Next Hop 197.85.8.2
– clic sul pulsante **Add**;
– inserire i parametri per **LAN2**:
Network 192.168.200.0 Mask 255.255.255.0 Next Hop 197.85.8.2
– clic sul pulsante **Add**;
– alla voce GLOBAL Settings, **NVRAM**, cliccare su **Save** per salvare la configurazione impostata come backup nella memoria non volatile del router.

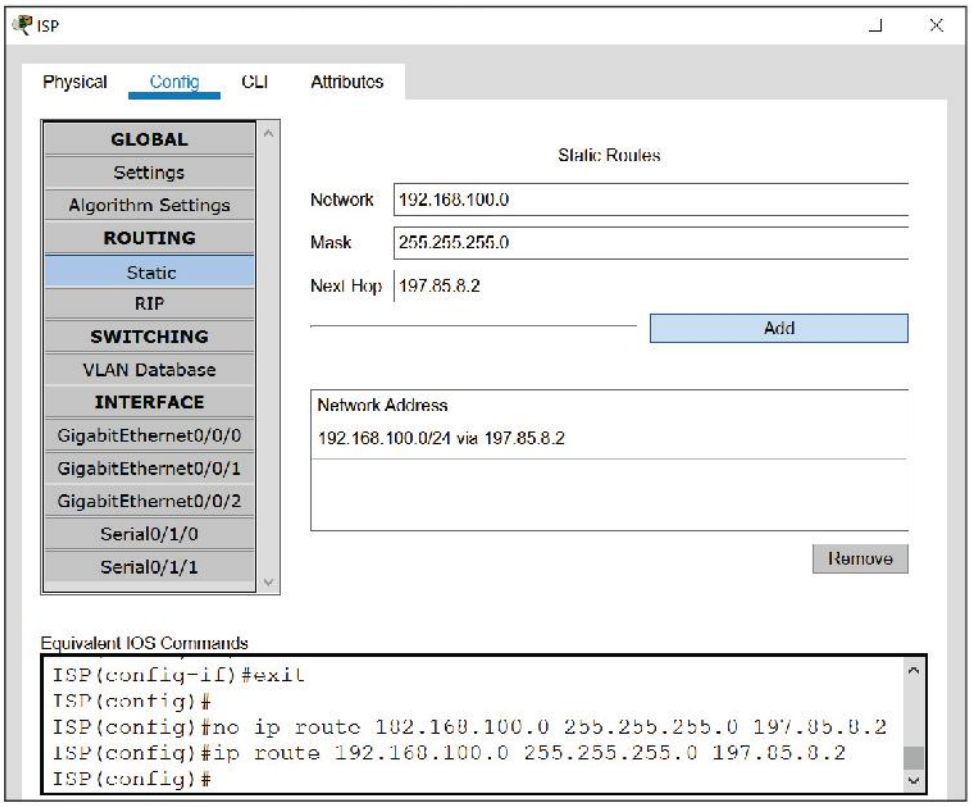
Notare i comandi IOS mostrati in basso in "Equivalent IOS Commands":

```
# ip route  192.168.100.0  255.255.255.0  197.85.8.2
```

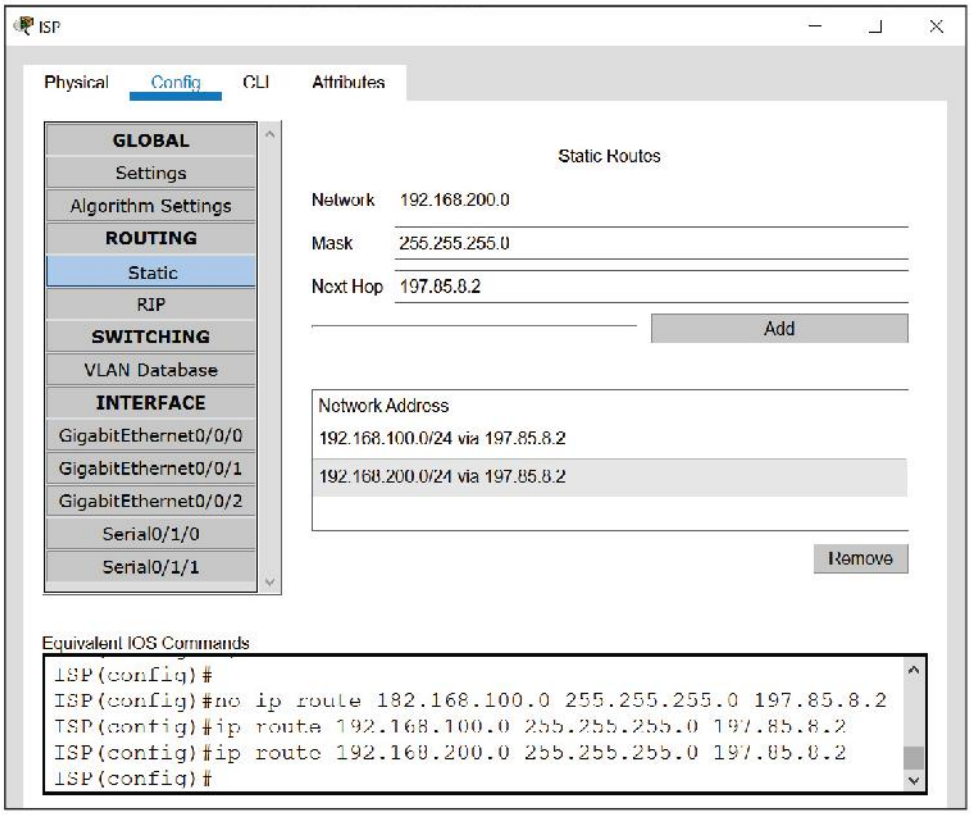
Per creare queste entry nella routing table del router ISP si può anche procedere da CLI e inserirle con il comando **ip route**.

Le due figure seguenti mostrano i passaggi di configurazione del routing statico su ISP.

Route da ISP verso rete 192.168.100.0 (LAN1)



Route da ISP verso rete 192.168.200.0 (LAN2)



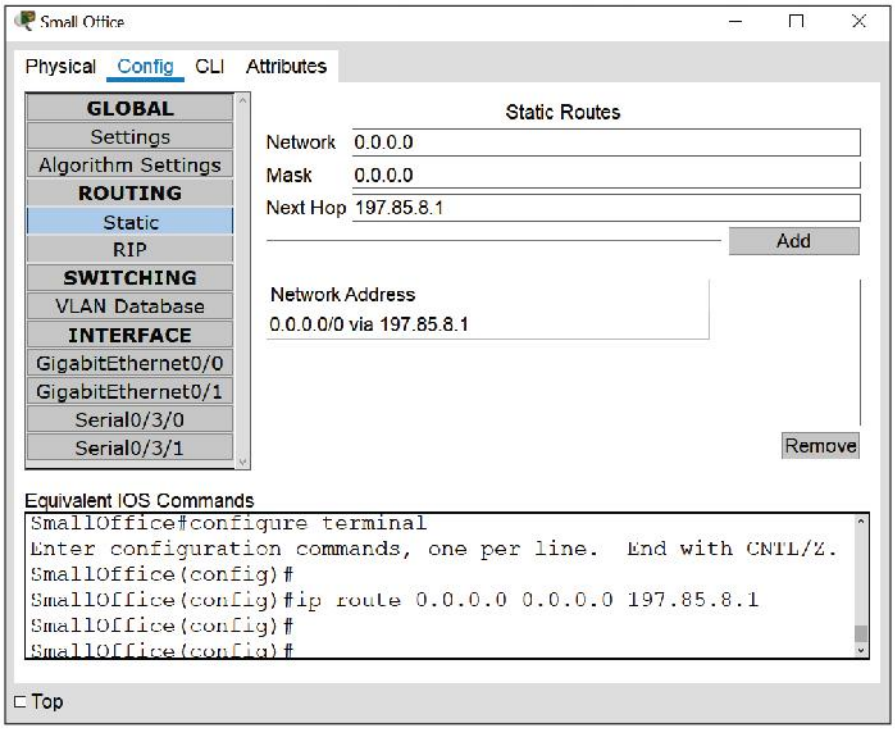
3) Configurazione su Small Office del routing statico verso LAN3

L'amministratore di rete può eseguire sul router Small Office la configurazione della route statica verso LAN3, come descritto per il router ISP.

Trattandosi, però, di un gateway verso l'esterno (cioè verso Internet), potrebbe anche impostare una **static default route** per dirigere verso ISP tutto il traffico destinato a Internet. In questo caso, i parametri rete remota e maschera sono per convenzione:

network address 0.0.0.0 subnet mask 0.0.0.0

che si traduce in: "**verso tutte le reti con tutte le maschere di sottorete**".



4) Test sulla connettività di rete

Eseguiamo nuovamente i seguenti test di connettività con il comando **ping**:

– da PC0 a interfaccia Serial0/1/0 sul router ISP;
– da PC1 a interfaccia Serial0/1/0 sul router ISP;
– da PC0 verso Server0;
– da PC1 verso Server0.

Tali test ora dovrebbero avere esito positivo.

8.2 La verifica della routing table

Con Packet Tracer è possibile verificare le route configurate nella routing table, così da verificarne la correttezza.

esercizio

File sorgenti
Scarica il file

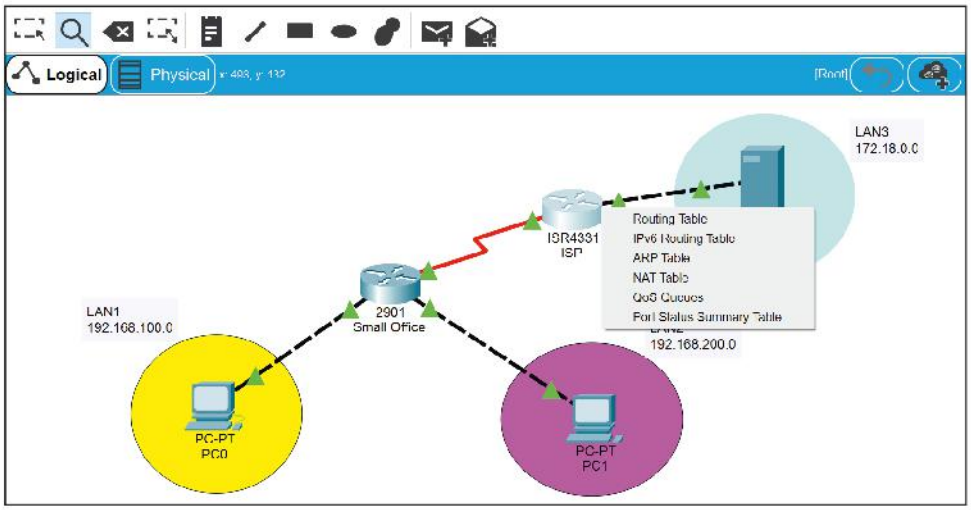
➜ PROBLEMA

Verificare sui router Small Office e ISP la correttezza della routing table.

➜ ANALISI DEL PROBLEMA

Per visualizzare la routing table si può procedere in due modi.

- Da CLI: selezionare la scheda **CLI** e dare il comando **show ip route.**
- Da interfaccia grafica: selezionare nella toolbar secondaria, a sinistra, l'icona della lente e posizionarsi sul router selezionandolo con un clic, comparirà l'elenco delle strutture dati del router, selezionare Routing Table.



➜ SVOLGIMENTO

Per verificare la correttezza delle routing table dei router Small Office e ISP si possono usare i due metodi presentati nell'analisi. Scegliamo di svolgere l'esercizio usando l'interfaccia CLI.

Router Small Office

Da linea di comando digitare: **show ip route**.

Se tutto funziona, a video compare la tabella di routing del router Small Office.

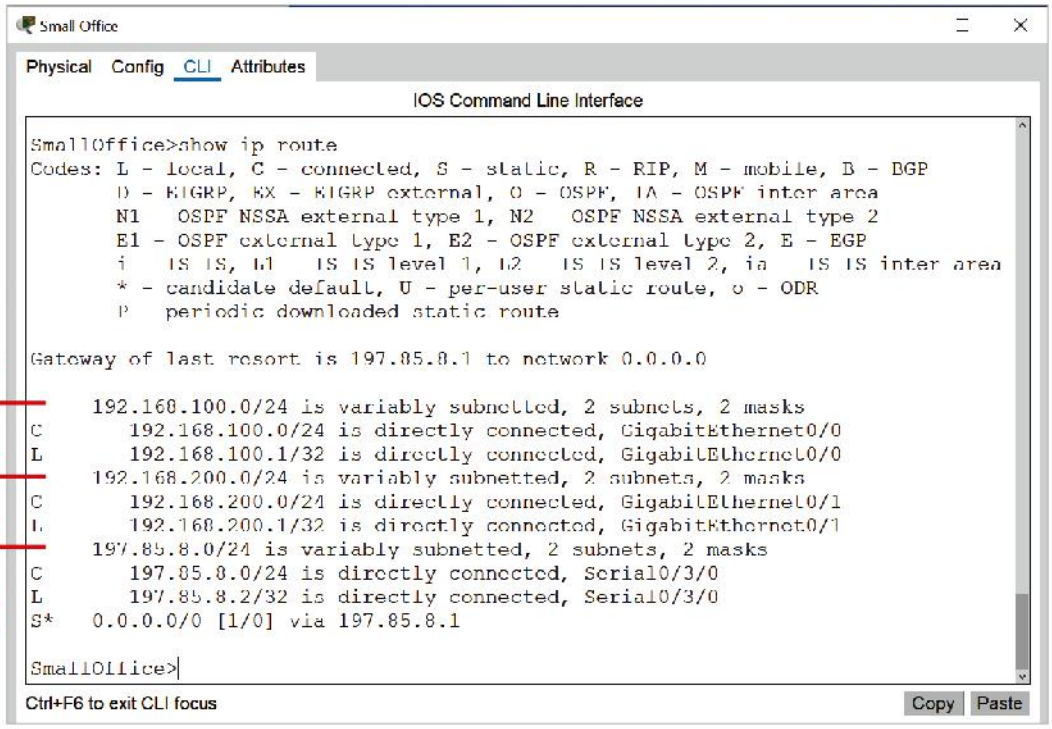
Indicazione del next hop per tutto il traffico instradato dalla default route

LAN1

LAN2

rete locale tra i due router

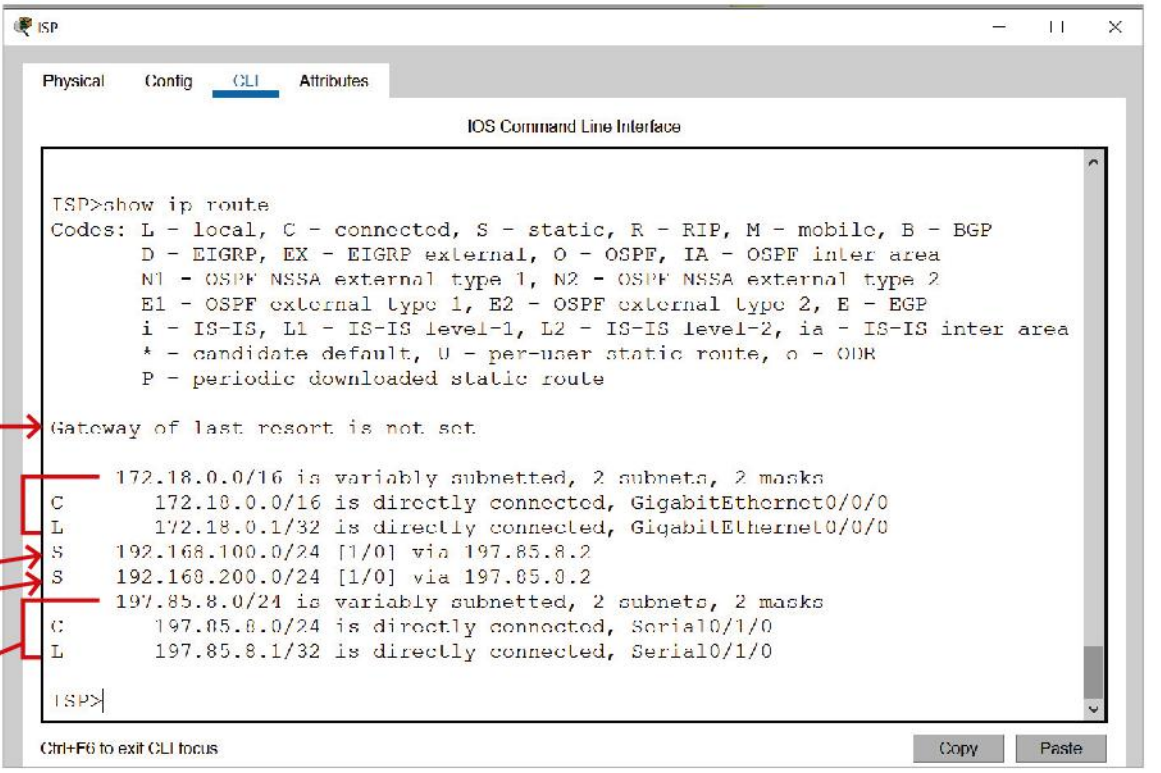
default static route



Router ISP

Da linea di comando digitare: **show ip route**.

Se tutto funziona, a video compare la tabella di routing del router ISP.



Indicazione sull'assenza della default route

LAN3

route verso LAN 1

route verso LAN 2

rete locale tra i due router

Case study
Routing statico

FISSA LE CONOSCENZE

- Di quali informazioni deve disporre il router per poter svolgere il suo compito di *instradatore*?
- Quando gli indirizzi IP vengono caricati nella routing table?
- Spiega il significato di *default route*.
- Quali parametri si devono impostare per il routing statico su un router?
- Come si può visualizzare la routing table con Packet Tracer?

9 PACKET TRACER: CONFIGURAZIONE DEL ROUTING DINAMICO

9.1 Definire la routing table dinamicamente

Nella Lezione precedente abbiamo visto come popolare in modalità statica la tabella di routing. Questa modalità presenta alcuni inconvenienti:

- è di difficile implementazione in reti molto complesse; infatti, ogni destinazione possibile deve essere caricata manualmente sul router con il corrispondente next hop e questo deve essere fatto per ogni router della rete;
- ogni qual volta una route diventa non più valida, questa deve essere rimossa o aggiornata manualmente dall'amministratore di rete.

Nelle prime Lezioni di questa Unità sono stati descritti gli algoritmi e i protocolli di routing che permettono ai router di acquisire le informazioni sulle route in automatico e scegliere la route migliore, per ogni destinazione di rete. Il contesto è quindi quello del **routing dinamico**. La scelta della route migliore avverrà in base a un parametro di costo minimo definito **metrica**; nel caso di più cammini verso una stessa destinazione, verrà scelto quello con metrica minore.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Riprendere lo scenario creato nella Lezione precedente, eliminare le route statiche sui due router e configurare il **routing dinamico** usando il protocollo **RIP** (Routing Information Protocol).
Verificare con il comando ping la connettività fra gli host delle reti di Small Office e il server sulla rete di ISP.

➜ ANALISI DEL PROBLEMA

Per avviare RIP su un router, occorre segnalare al protocollo le reti direttamente connesse al router che parteciperanno allo scambio di informazioni. Quindi, l'indirizzo IP della rete direttamente connessa verrà annunciato agli altri router e tramite la medesima rete verranno inviate le informazioni di routing ai router stessi.

➜ SVOLGIMENTO

Eliminazione del routing statico sul router ISP:

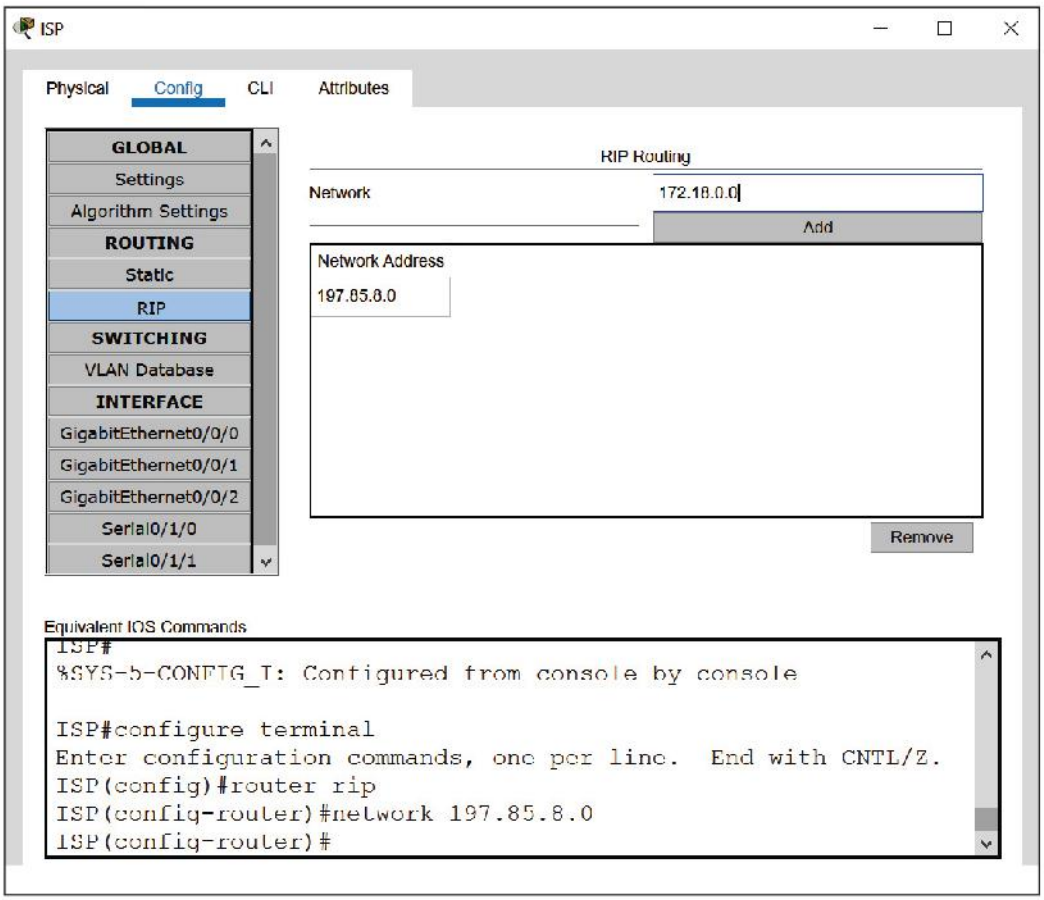
- aprire il file creato durante la precedente esercitazione di laboratorio;
- al termine del boot del router, selezionare la scheda **Config** e alla voce ROUTING, selezionare **Static**: nella finestra che si apre, in basso a destra sono elencate le route verso le reti connesse al router Small Office;
- selezionare una alla volta le route e cliccare sul pulsante Remove: in questo modo la route statica viene cancellata dalla programmazione del router e dalla tabella di routing.

Se provassimo ora a eseguire dei test di connettività con il comando ping, darebbero esito negativo, in quanto al router ISP manca la conoscenza delle route per LAN1 e LAN2.

Configurazione del routing dinamico sul router ISP:

- selezionare la scheda **Config**, alla voce ROUTING, selezionare **RIP**: si apre la finestra di dialogo per la configurazione del protocollo di routing;
- nel box a destra della scritta Network, inserire gli indirizzi di rete IP delle **reti direttamente connesse a ISP**:
  – digitare l'indirizzo IP della rete seriale che collega ISP a Small Office: 197.85.8.0 e cliccare su Add;
  – digitare l'indirizzo IP della rete locale di ISP: 172.18.0.0 e cliccare su Add.

  Ogni volta che viene selezionato il pulsante Add, la rete digitata nel box Network compare nel sottostante box Network Address:



Notare i **comandi IOS** che compaiono nel box Equivalent IOS Command. Per inserire una route dinamica con il protocollo IP si usano i seguenti comandi:

```
Router # configure terminal
Router (config) # router rip
Router (config-router) # network indIP_della_rete
```

Eliminazione del routing statico sul router Small Office:
procedere all'eliminazione della default route con la stessa modalità svolta per eliminare le route statiche in ISP.

Impostazione del routing dinamico sul router Small Office:

- selezionare la scheda **Config**, alla voce ROUTING, selezionare **RIP**;

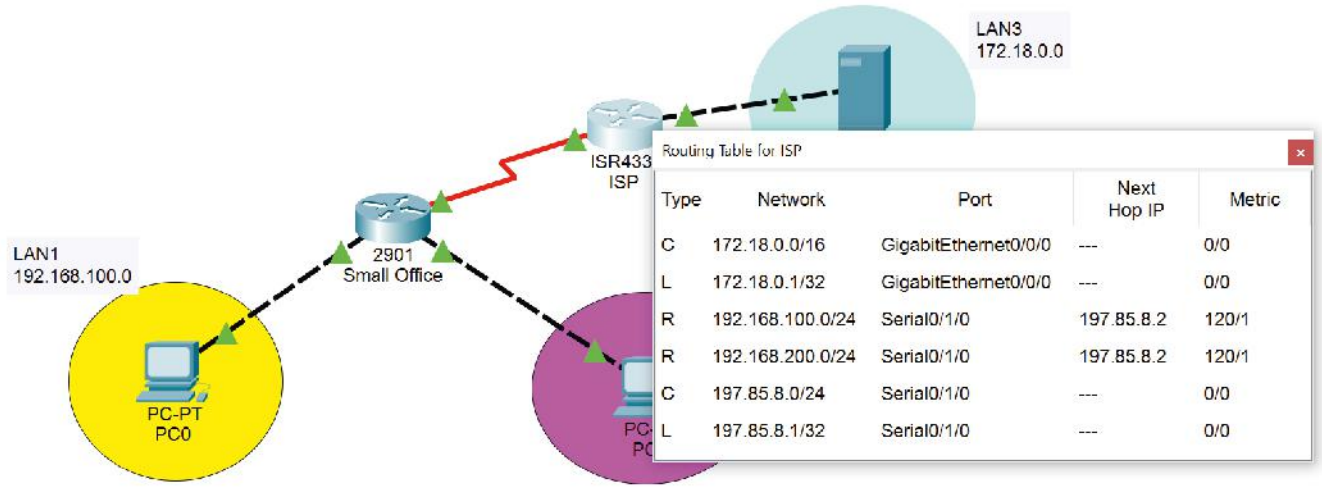
- nel box a destra della scritta Network, inserire gli indirizzi di rete IP delle **reti direttamente connesse a ISP**:
  – digitare l'indirizzo IP della rete seriale che collega Small Office a ISP: 197.85.8.0 e cliccare su Add;
  – digitare l'indirizzo IP della LAN1 di Small Office: 192.168.100.0 e cliccare su Add;
  – digitare l'indirizzo IP della LAN2 di Small Office: 192.168.200.0 e cliccare su Add.

Test sulla connettività di rete

Eseguire i test di connettività con il comando ping da LAN1 e LAN2 verso ISP e da LAN3 verso LAN1 e LAN2.
Tali test dovrebbero avere esito positivo.

Verifica della routing table da interfaccia grafica

Per vedere il contenuto della routing table dall'interfaccia grafica di Packet Tracer, si deve usare lo strumento Inspect raffigurato con l'icona della lente d'ingrandimento (nella toolbar in alto a sinistra). Quindi si seleziona il router d'interesse e compare un box con i dati contenuti nella routing table.
Eseguendo questa procedura sul **router ISP** otteniamo:



Type	Network	Port	Next Hop IP	Metric
C	172.18.0.0/16	GigabitEthernet0/0/0	---	0/0
L	172.18.0.1/32	GigabitEthernet0/0/0	---	0/0
R	192.168.100.0/24	Serial0/1/0	197.85.8.2	120/1
R	192.168.200.0/24	Serial0/1/0	197.85.8.2	120/1
C	197.85.8.0/24	Serial0/1/0	---	0/0
L	197.85.8.1/32	Serial0/1/0	---	0/0

Verifica della routing table da CLI

Routing table del router Small Office:

- selezionare la scheda **Config**, in GLOBAL Settings alla voce NVRAM cliccare su Save;
- selezionare la scheda **CLI** e da linea di comando digitare:

```
SmallOffice# show ip route
```

se tutto funziona, a video dovrebbe comparire la tabella di routing del router Small Office che comprende:
– le due reti locali 192.168.100.0/24 e 192.168.200.0/24;
– la rete connessa alla seriale, 197.85.8.0/24;
– la rete locale di ISP 172.18.0.0/16, con next hop 197.85.8.1 e l'interfaccia locale di "uscita" per i pacchetti Serial0/3/0.

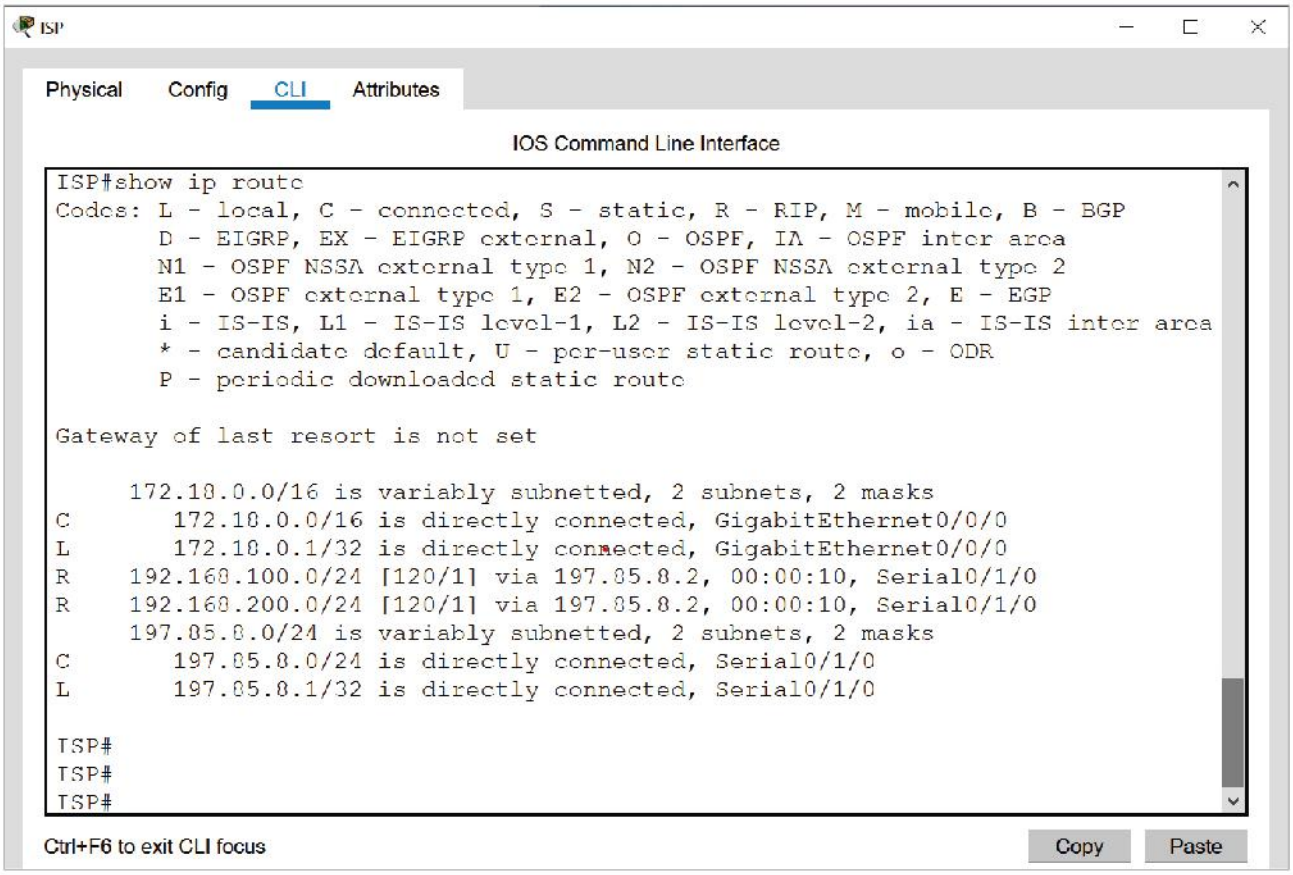
Routing table del router ISP:

- selezionare la scheda **Config**, in GLOBAL Settings alla voce NVRAM cliccare su Save;

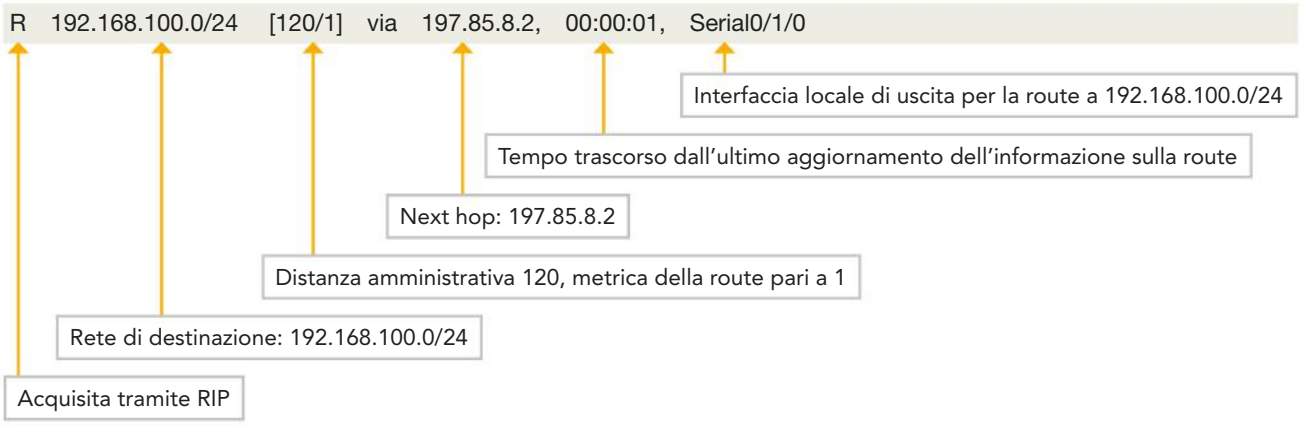
- selezionare la scheda **CLI** e da linea di comando digitare:

```
ISP# show ip route
```

Nella tabella di routing che compare troviamo:
– la rete locale di ISP 172.18.0.0/16;
– le due reti locali 192.168.100.0/24 e 192.168.200.0/24 con next hop 197.85.8.2 e l'interfaccia locale di "uscita" per i pacchetti Serial0/1/0;
– la rete connessa direttamente alla seriale, 197.85.8.0/24.



Esaminiamo ora in dettaglio la struttura di una route così come compare nella routing table di ISP:

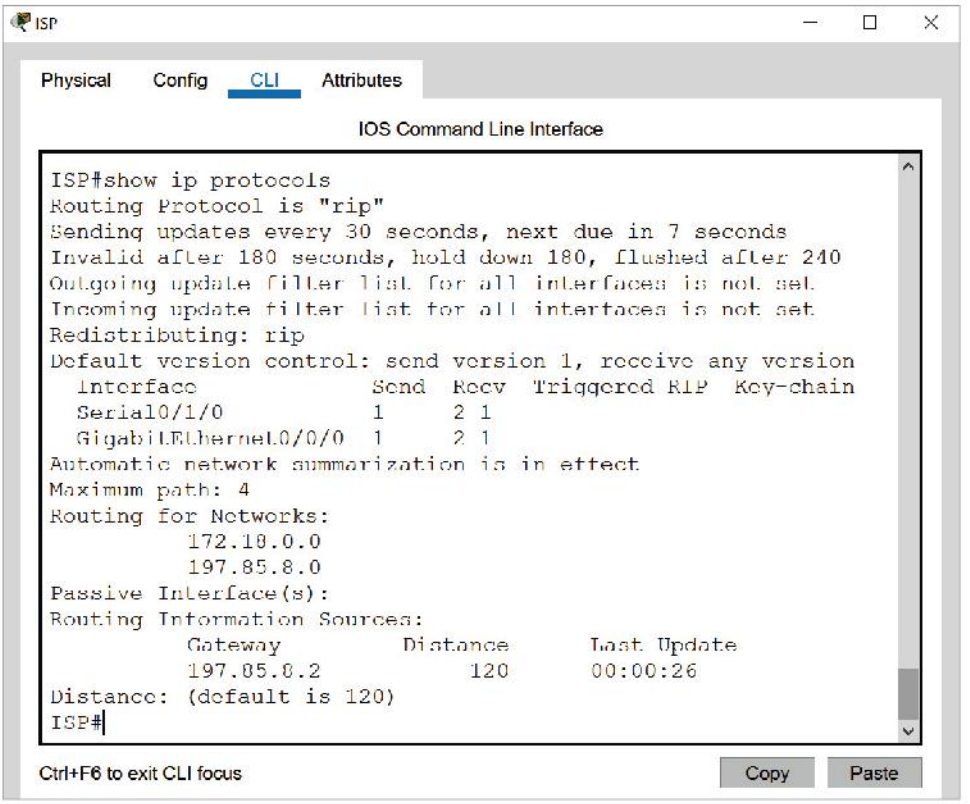


9.2 Debugging di RIP

Tra i comandi di IOS per il routing, ve ne sono alcuni utili per la diagnostica (debugging), per capire se un router invia informazioni di routing errate.

Per esempio, il seguente comando:

```
Router#show ip protocols
```



fornisce in output i parametri che il protocollo di routing sta usando per inviare e ricevere gli aggiornamenti, la metrica utilizzata e le reti vicine alle quali annuncia i cambiamenti nel routing.

Le informazioni principali che ricaviamo dall'output del comando sono:

- quale protocollo di routing si sta usando: RIP;
- le informazioni sulle route sono inviate ogni 30 secondi e il prossimo invio avverrà tra 7 secondi, ciò significa che il router ha ricevuto l'ultimo aggiornamento 23 secondi fa;
- nella riga successiva sono indicate le azioni intraprese dal router: se non riceve un aggiornamento entro 180 secondi segna la route come da non usare e la rimuove dopo 240 secondi;
- è presente anche un'informazione sulla versione di RIP usata: il router trasmette con RIPv1, ma può ricevere (ed elaborare) sia v2 sia v1;
- importanti per fini diagnostici sono i dati scritti in Routing for Networks, qui sono indicate le reti adiacenti al router, quelle con cui si scambia le informazioni sulle route.

Altri comandi IOS per il debug sono:

Router#show ip rip database //per visualizzare le informazioni di routing

Router#debug ip rip //per abilitare il debug di RIP

Router#no debug ip rip //per disabilitare il debug di RIP

FISSA LE CONOSCENZE

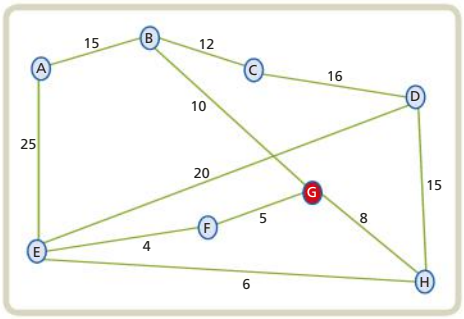
- Quali sono i problemi che possono sorgere con il routing statico?
- Spiega in cosa consiste il routing dinamico e fai un esempio di protocollo di routing.
- Perché si devono inserire nel router gli indirizzi delle reti direttamente connesse?
- Quali sono i comandi IOS per inserire una route dinamica per RIP?

1 Problematica e scenari

Il routing (instradamento) è una funzione del livello Network del TCP/IP. Tale funzione viene svolta da un dispositivo di rete chiamato router. Il routing può essere diretto se il router è direttamente connesso alla rete di destinazione o indiretto se il pacchetto deve passare attraverso altri router prima di arrivare a destinazione. Nella **routing table** di un router si memorizzano le reti a lui note, inserendo l'indirizzo IP di network e il next hop, ossia l'indirizzo IP del successivo router a cui inviare il pacchetto. Oltre a queste informazioni si memorizza l'interfaccia del router su cui trasmettere il pacchetto (forwarding) e il costo del percorso. La metrica più semplice per calcolare il costo è "hop count", che "conta" il numero di nodi che devono essere attraversati.

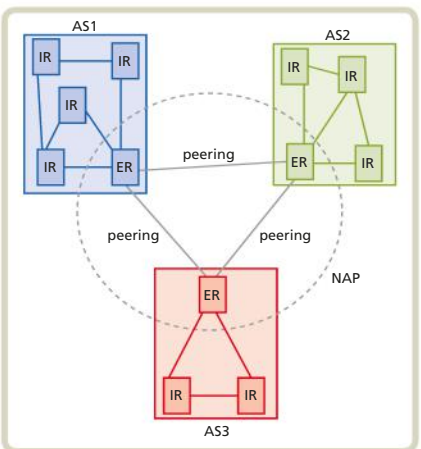
2 Gli algoritmi e i protocolli di routing

Lo scopo di un protocollo di routing è quello di mantenere dinamicamente le routing table. Per fare ciò, i router devono condividere le informazioni sui percorsi (route) che ciascuno conosce. La gran parte dei protocolli che regolano il routing moderno utilizzano gli algoritmi Distance Vector o Link State. Il primo si basa sulla costruzione di un vettore delle distanze che ogni router periodicamente invia ai propri vicini e sulla base del quale costruisce la propria routing table. È un algoritmo adatto a reti di piccole dimensioni per via dei tempi di convergenza lunghi. Al contrario, l'algoritmo Link State ha un tempo di convergenza molto basso, quindi si adatta bene a reti di grandi dimensioni. Il Link State fornisce a ogni router l'esatta topologia della rete in cui si trova, infatti ogni nodo invia dei pacchetti (LSP) a tutti i nodi della rete, che contengono informazioni sui suoi nodi adiacenti.



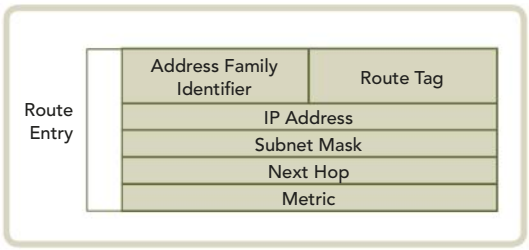
3 Gli Autonomous System e il routing gerarchico

Nei primi anni Ottanta, Internet era considerata come una *single network* cioè un'unica rete in cui tutti i router dovevano predisporre una routing table contenente una voce per ogni rete raggiungibile e l'indirizzo del router attraverso cui raggiungerla. La decisione intrapresa fu quella di abbandonare il modello single network per suddividere Internet in un certo numero di Autonomous System, ognuno costituito da un insieme di router e LAN raggruppati secondo criteri topologici e organizzativi. Ogni AS è individuato da un numero assegnato da IANA. I router che collegano tra loro gli AS sono spesso chiamati gateway a sottolineare il servizio che svolgono di inoltro dei pacchetti da una rete verso l'esterno.



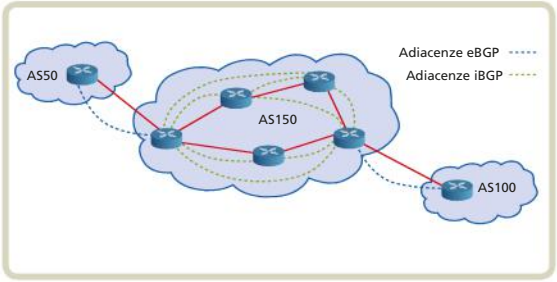
4 Protocolli di routing IGP

I protocolli di routing di tipo IGP (Interior Gateway Protocol) sono noti come protocolli intradominio poiché vengono usati per regolare l'instradamento dei pacchetti tra gli host interni a un Autonomous System. Questi protocolli possono essere classificati in base all'algoritmo di routing che utilizzano. Il protocollo RIP implementa il Distance Vector routing, usando, nella versione 1, la metrica hop count per scegliere il percorso migliore. Inoltre il RIP prevede che le tabelle di routing siano aggiornate a intervalli di tempo prefissati. Il protocollo OSPF implementa il Link State routing, ciò gli consente di avere un database distribuito che rappresenta la topologia dell'intera rete dell'AS sotto forma di grafo. I nodi della rete aggiornano il database solo a fronte di cambiamenti nella rete, garantendo così tempi di convergenza molto ridotti rispetto a quelli del protocollo RIP.



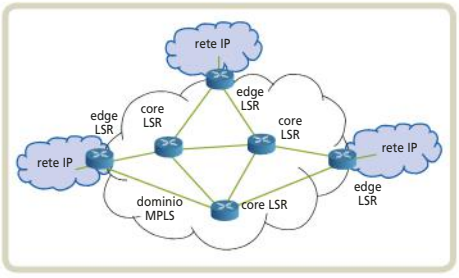
5 Protocolli di routing EGP

I protocolli di routing di tipo EGP (Exterior Gateway Protocol) sono noti come protocolli extradominio poiché vengono usati per regolare l'instradamento dei pacchetti tra host appartenenti ad Autonomous System diversi. Quello usato attualmente sul backbone di Internet è BGP, aggiornato alla versione BGP-4. Questo protocollo si basa su un algoritmo di routing di tipo Path Vector, simile al Distance Vector, dove con path si indica il percorso che connette due AS, quindi nel vettore si elencano tutti gli AS da attraversare per raggiungere una destinazione.



6 Le reti multiprotocollo: MPLS

Nelle reti MPLS, la tabella di routing dei nodi del backbone non memorizza gli indirizzi IP, bensì degli identificatori di 32 bit, detti **label**, con cui si identificano i pacchetti che transitano nel backbone. La tecnica di commutazione è infatti detta **label switching**. I router che si trovano al confine del dominio MPLS hanno il compito di aggiungere la label al pacchetto che entra nella rete MPLS, così che possa procedere con l'instradamento MPLS e non IP, finché non uscirà dalla rete MPLS e verrà nuovamente instradato con l'indirizzo IP di destinazione.



VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Il next hop rappresenta l'indirizzo del router successivo per proseguire verso la rete di destinazione. V F
2. Il Routing Table Lookup Problem (RTLP) è il problema di dover decidere molto in fretta dove instradare i pacchetti. V F
3. Il routing update consiste nell'aggiornamento del software di gestione di un router. V F
4. Il Distance Vector e il Link State sono algoritmi di routing di tipo statico. V F
5. Il Distance Vector conosce la topologia dell'intera rete. V F
6. Lo split horizon serve a prevenire il loop tra due nodi adiacenti. V F
7. Il route poisoning blocca tutte le route che aumentano di costo supponendo che si tratti di un loop. V F
8. Gli Autonomous System sono delle singole autorità amministrative. V F
9. Deve essere sempre possibile trasferire un messaggio tra due router appartenenti alla stessa regione senza farlo uscire da quella regione. V F
10. IGP ed EGP sono tipi di protocolli dedicati al trasferimento di file. V F
11. Il modo con cui il RIP (Routing Information Protocol) evita il loop tra due nodi adiacenti è lo split horizon. V F
12. In un pacchetto RIP il campo principale è il campo dati. V F
13. Il protocollo OSPF (Open Shortest Path First) è basato sulla creazione di un database distribuito che rappresenta la topologia della rete sotto forma di grafo. V F
14. Le aree OSPF sono costituite da reti non contigue. V F
15. Il protocollo BGP (Border Gateway Protocol) è un protocollo di tipo Path Vector, evoluzione del Distance Vector. V F
16. I router che si occupano dell'operazione di label swapping sono quelli di core (CLSR). V F
17. Le label MPLS contengono indirizzi IP. V F
18. Il pushing MPLS consiste nell'eliminazione di una label. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **La convergenza di un algoritmo di routing è:**
   - A l'operazione di inoltro dei pacchetti nelle reti MPLS
   - B il tempo che impiega un pacchetto ad attraversare tutti gli hop
   - C un'operazione di connessione a un router remoto
   - D l'intervallo di tempo che trascorre prima che tutti i router della rete abbiano aggiornato le routing table

2. **Quale protocollo di routing si adatta meglio a reti che cambiano spesso la topologia?**
   - A RIPv1
   - B RIPv2
   - C OSPF
   - D BGP

3. **Con "default router" si identifica:**
   - A un router di backup, da mettere in funzione quando si guasta un router della rete
   - B un router a cui inviare un pacchetto la cui destinazione non è presente nella routing table
   - C un router che mantiene una configurazione di default dei parametri di rete

4. **Quale tra le seguenti non è una metrica per il calcolo del costo di una route?**
   - A Bandwidth
   - B Hop count
   - C CPU frequency
   - D Delay

5. **L'ente che assegna i numeri di identificazione degli Autonomous System è:**
   - A IETF
   - B IANA
   - C ISO
   - D IEEE

6. **Che cosa è scritto nel campo next hop della routing table?**

A L'indirizzo IP del router a cui inoltrare i pacchetti
B Come si è ricavata la route verso quella destinazione
C L'interfaccia di output su cui effettuare il forwarding dei pacchetti
D Il valore di hop count

7. **Il Link State Routing prevede l'invio di un pacchetto LSP:**

A al router più vicino
B al default gateway per l'inoltro nelle reti remote
C in flooding
D ai soli router adiacenti che non lo devono inoltrare in rete

8. **Quale tra i seguenti non è un tipo di Autonomous System?**

A Multi-homed
B Transit
C Stub
D Peering

9. **Quante entry di una routing table possono essere inviate, al massimo, in un pacchetto RIP?**

A 10 B 16 C 25 D 30

10. **Quale tra i seguenti protocolli di routing supporta IPv6?**

A OSPFv2
B RIPv1
C OSPFv3
D RIPv2

11. **Un Autonomous System scambia informazioni con altri AS attraverso un router:**

A ASBR
B ABR
C DR
D BR

12. **Quale tra le seguenti non è un'operazione effettuata da un router MPLS sulle label?**

A Popping
B Pushing
C Swapping
D Finding

13. **Quale tra le seguenti funzioni è realizzabile con una rete MPLS?**

A Assegnazione dinamica degli indirizzi
B Risoluzione dei nomi
C Traffic Engineering
D Address resolution

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Quali strategie si possono utilizzare per rendere veloce la decisione sull'instradamento dei pacchetti?**

2. LEZIONE 1 **Quali sono i campi generalmente presenti in una routing table?**

3. LEZIONE 2 **Quali sono i due più diffusi algoritmi di routing dinamico e cosa li distingue?**

4. LEZIONE 3 **In che cosa consiste il peering tra Autonomous System?**

5. LEZIONE 3 **Qual è la convenienza di realizzare un routing gerarchico?**

6. LEZIONE 3 **Che cosa vuol dire che una regione o un'area deve essere connessa?**

7. LEZIONE 3 **Qual è lo scopo del gateway predefinito?**

8. LEZIONE 4 **Che cos'è la metrica e quali tipi di metriche conosci?**

9. LEZIONE 4 **In che cosa consiste l'hold down opzionalmente utilizzato dal protocollo RIP?**

10. LEZIONE 4 **Spiega il concetto di area su cui si basa il protocollo OSPF.**

11. LEZIONE 4 **Che cos'è la backbone area?**

12. LEZIONE 4 **Quali tipi di pacchetti sono definiti nel protocollo OSPF?**

13. LEZIONE 5 **Descrivi l'algoritmo di routing Path Vector.**

14. LEZIONE 5 **Quali sono le due politiche di routing attuabili con BGP?**

15. LEZIONE 6 **Che differenza c'è tra un router core e un router edge nelle reti MPLS?**

16. LEZIONE 6 **Che cosa viene utilizzato come indice della routing table dagli LSR?**

IN ENGLISH, PLEASE

CLIL

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

Routing and Interconnection of Networks

Routing is a Network Layer function of the TCP/IP model. This function is performed by a device called network router, also known as intermediate system, using algorithms to analyse appropriate routing schedules and decide on which outgoing line packets should be routed. The purpose of a routing protocol is to maintain the routing tables dynamically. To do so, routers must share information about the routes that each of them knows. In the early 80s the Internet was considered a single network on which all routers had to prearrange a routing table including every reachable network along with the next hop router address to reach it. The enormous growth of the Internet has made such arrangement no longer applicable. Modern routing may be carried out either within a single network, which may be splitted into smaller units, or in Internetwork, that is, between heterogeneous networks which may even use different protocols. A large network or group of networks with a single routing policy forms an Autonomous System (AS), identified by an assigned unique identifier (AS number).

Most of the protocols regulating routing use the Distance Vector Routing algorithm or else the Link State Routing algorithm. A routing protocol specifies how routers communicate with each other to distribute information for selecting routes between any two nodes. Many routing protocols are defined by IETF, such as RIP, OSPF, BGP.

EXERCISES

Use the appropriate number to match words and meanings.

...	Route	1	A router function
...	Cost	2	It must be implemented
...	Forwarding	3	Within an autonomous system
...	Exterior	4	A short identifier
...	Interior	5	An adjacent node
...	Mandatory	6	Outside an autonomous system
...	Neighbour	7	A value based on metrics which is used to measure a route
...	Label	8	Path through an Internetwork

GLOSSARY

Adjacency: the relationship formed between neighbouring routers and end nodes for routing information exchange.

Autonomous System: a group of one or more IP prefixes (lists of IP addresses reachable on a network) run by one or more network operators that maintain a single routing policy.

Black hole: it happens when a route sends data to the wrong place or can't get to the destination. Packets go in and they don't come out.

Convergence: a routing protocol is said to converge when all routers in a network know all routes to all destinations.

Flooding: a router forwards a packet to every other node connected to the router except the node from which the packet arrived. It is a way to distribute routing information updates quickly to every node in a large network.

Label swap: the basic forwarding operation, that consists in looking up an incoming label to determine the outgoing label, encapsulation, port, and other data handling information.

Metrics: routing metrics are a scoring system for routes. They measure how good or bad the route is. Metrics can include hop count, bandwidth, delay, etc.

Multiprotocol: a technology that can work with many different protocols.

Peering: when two or more Autonomous Systems interconnect directly with each other to exchange traffic.

Reachability: a network is reachable by a host when the host can send a packet towards a destination on that network.

Routing table: the table stored in a router that contains routes to some network destinations and, in some cases, the metrics associated with those routes.

COMPETENZE IN GIOCO

Competenze disciplinari

- Realizzare scenari di rete WAN usando un simulatore di rete in cui testare le funzionalità di routing.
- Saper configurare su un router il protocollo RIP usando anche i comandi IOS.
- Saper gestire le tabelle di routing.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Imparare a usare sia la modalità grafica sia quella da terminale (CLI) per configurare e gestire gli apparati di rete.
- Esporre i risultati del proprio lavoro alla classe.

tempi

- Personale risoluzione del tema proposto: 2 ore.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

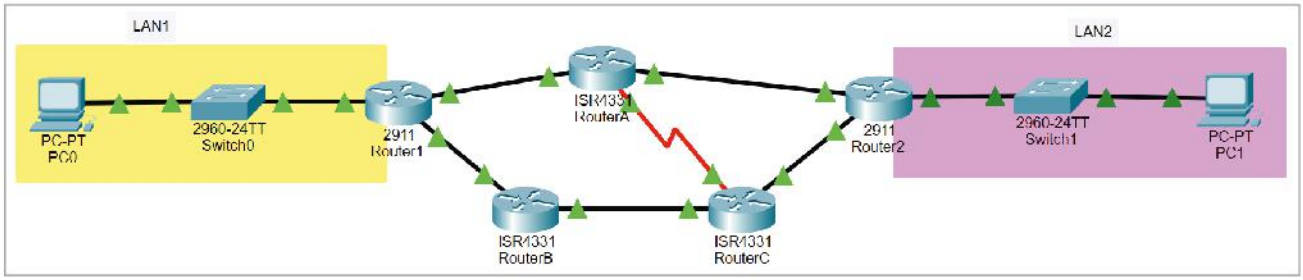
strumenti

- Libro di testo.
- Applicazione Cisco Packet Tracer.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

File sorgenti
Scarica il file

Configurare il routing dinamico in una rete LAN-WAN.
Realizzare uno scenario di rete in cui siano presenti 2 reti locali interconnesse attraverso una rete WAN composta da 3 router (intermediate system):



Definire il piano di indirizzamento IP della rete usando l'indirizzo IP pubblico **200.1.50.0** e applicando la tecnica del subnetting per creare le reti che interconnettono i vari router. Per le 2 reti locali LAN1 e LAN2 usare indirizzi privati.
Configurare il routing dinamico con protocollo **RIPv2** su tutti i router e verificare le relative routing table.
Infine, provare, più volte e da punti diversi della rete, i percorsi dei pacchetti con il comando **traceroute**.

SVOLGIMENTO

Prima di realizzare lo scenario di rete aziendale con il simulatore Packet Tracer, è opportuno definire il piano di indirizzamento IP della rete.
Come richiesto, usiamo l'indirizzo pubblico 200.1.50.0 per creare delle subnet: una per ogni connessione tra 2 router. Ogni subnet avrà solo 2 host: le interfacce dei 2 router collegati.
L'indirizzo IP è di classe C, quindi si possono creare $2^6 = 64$ subnet, ciascuna con $2^2 - 2 = 2$ host.

Ogni sottorete avrà quindi la subnet mask: **255.255.255.252** (prefix **/30**).
Per le 2 LAN usiamo un indirizzo privato di classe C: **192.168.100.0/24** per LAN1 e **192.168.200.0/24** per LAN2.

La tabella seguente riassume il piano di indirizzamento:

RETI	Ind. di rete	Subnet mask	Range indirizzi di host	Ind. di broadcast
LAN1	192.168.100.0	255.255.255.0	Da 192.168.100.1 a 192.168.100.254	192.168.100.255
LAN2	192.168.200.0	255.255.255.0	Da 192.168.200.1 a 192.168.200.254	192.168.200.255
R1-RA	200.1.50.0	255.255.255.252	Da 200.1.50.1 a 200.1.50.2	200.1.50.3
R1-RB	200.1.50.4	255.255.255.252	Da 200.1.50.5 a 200.1.50.6	200.1.50.7
R2-RA	200.1.50.8	255.255.255.252	Da 200.1.50.9 a 200.1.50.10	200.1.50.11
R2-RC	200.1.50.12	255.255.255.252	Da 200.1.50.13 a 200.1.50.14	200.1.50.15
RB-RC	200.1.50.16	255.255.255.252	Da 200.1.50.17 a 200.1.50.18	200.1.50.19
RA-RC	200.1.50.20	255.255.255.252	Da 200.1.50.21 a 200.1.50.22	200.1.50.23

Legenda: R1=Router1, R2=Router2, RA=RouterA, RB=RouterB, RC=RouterC

Creiamo ora lo scenario con Packet Tracer: inseriamo un PC connesso a uno switch, nelle reti LAN1 e LAN2. Gli switch saranno collegati a un'interfaccia del router, scegliendo quelle a più alta velocità: GigabitEthernet.
Scegliamo di usare il modello di router 2911 per quelli che hanno il ruolo di gateway per LAN1 e LAN2, mentre per gli altri router, che rappresentano router di backbone, usiamo il modello ISR 4331.

Configuriamo l'indirizzo IP e la subnet mask su PC e router, ricordando che le interfacce del router allo startup sono tutte down e devono essere accese (scheda Config, Interface, flag Port Status).
A questo punto possiamo creare i collegamenti tra i vari dispositivi presenti nel workspace usando i cavi Copper straight-through, tranne che nel collegamento tra RouterA e RouterC che si usa un cavo seriale.
Per questi ultimi 2 router sarà necessario inserire il modulo **NIM-2T** per configurare un'**interfaccia seriale**, operando in Config, scheda Physical, come visto nella Lezione 7 dell'Unità 3.
La colorazione verde dei triangolini presenti alle estremità dei cavi segnala il corretto funzionamento della connessione fisica.
La seguente tabella mostra gli indirizzi IP assegnati alle interfacce dei router e dei 2 PC con le rispettive connessioni.

Host	Interfaccia	Ind. IP	Interfaccia a cui è connesso
PC1	FastEthernet 0	192.168.100.10 /24	**Switch0** FastEthernet 0/1
R1	GigabitEthernet 0/0	192.168.100.254 /24	**Switch0** GigabitEthernet 0/1
	GigabitEthernet 0/1	200.1.50.1 /30	**RA** GigabitEthernet 0/0/0
	GigabitEthernet 0/2	200.1.50.5 /30	**RB** GigabitEthernet 0/0/0
PC2	FastEthernet 0	192.168.200.10 /24	**Switch1** FastEthernet 0/1
R2	GigabitEthernet 0/0	192.168.200.254 /24	**Switch1** GigabitEthernet 0/1
	GigabitEthernet 0/1	200.1.50.9 /30	**RA** GigabitEthernet 0/0/1
	GigabitEthernet 0/2	200.1.50.13 /30	**RC** GigabitEthernet 0/0/1

RA	GigabitEthernet 0/0/0	200.1.50.2 /30	**R1** GigabitEthernet 0/1
	GigabitEthernet 0/0/1	200.1.50.10 /30	**R2** GigabitEthernet 0/1
	Serial 0/1/0	200.1.50.21 /30	**RC** Serial 0/1/0
RB	GigabitEthernet 0/0/0	200.1.50.6 /30	**R1** GigabitEthernet 0/2
	GigabitEthernet 0/0/1	200.1.50.17 /30	**RC** GigabitEthernet 0/2
RC	GigabitEthernet 0/0/0	200.1.50.14 /30	**R2** GigabitEthernet 0/2
	GigabitEthernet 0/0/1	200.1.50.18 /30	**RB** GigabitEthernet 0/2
	Serial 0/1/0	200.1.50.22 /30	**RA** Serial 0/0/0

Su PC0 e PC1 impostiamo anche il **default gateway**, inserendo l'indirizzo IP dell'interfaccia del router, Router1e Router2, che fa parte della LAN1 e LAN2, rispettivamente.
Verificare la raggiungibilità dei vari dispositivi usando il comando ping dal Command Prompt nei PC (scheda Desktop) o dalla scheda CLI nei router. Non avendo impostato alcuna route, i 2 PC non comunicheranno.

Configurazione del routing dinamico con RIP

Sui router configuriamo il protocollo RIP così da avere l'aggiornamento dinamico delle tabelle di routing. Come impostazione predefinita, i router in Packet Tracer usano RIPv1 che non gestisce, però, il subnetting.

> È necessario abilitare l'uso di RIPv2 in tutti i router, dal momento che abbiamo creato delle subnet. A differenza di RIPv1, in RIPv2 la subnet mask è presente nei pacchetti scambiati tra i router.

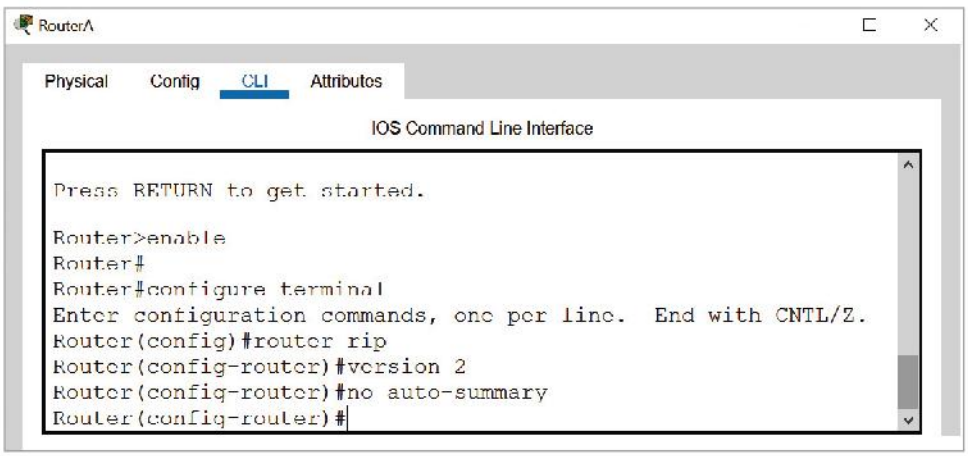
Per capire come è configurato su un router RIPv2, eseguire il comando IOS dall'interfaccia CLI:

```
RouterA # show ip protocol
```

Per impostare l'uso di RIPv2 è necessario usare i comandi IOS dall'interfaccia CLI:

```
RouterA (config) # router rip
RouterA (config-router) # versione 2
RouterA (config-router) # no auto-summary
```

Il comando **no auto-summary**, disponibile solo con RIPv2, indica al router di non aggregare le reti con lo stesso indirizzo di rete (NetID), così nella routing table vedremo anche le varie subnet (NetID + SubnetID) con le relative subnet mask.
Questa operazione è da ripetere su tutti e 5 i router del nostro scenario.



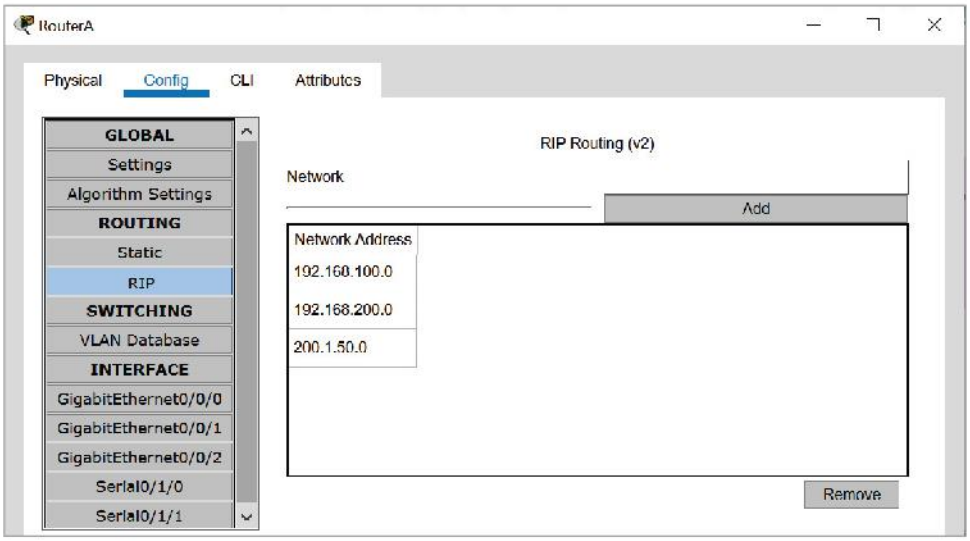
#prendinota

Auto-summary
Questa funzionalità, presente nei router Cisco, consente ai router di "annunciare" solo le reti classfull, raggruppando più subnet in una sola: nella routing table viene inserito solo l'indirizzo di rete (NetID). Il vantaggio è una riduzione del numero di entry nella tabella, lo svantaggio la perdita del subnetting.

Il passo successivo è l'inserimento delle reti che ogni router conosce, in quanto adiacenti a esso. Si può operare in Packet Tracer da **CLI** come segue:

```
RouterA (config) # router rip
RouterA (config-router) # network 192.168.100.0
RouterA (config-router) # network 192.168.200.0
RouterA (config-router) # network 200.1.50.0
```

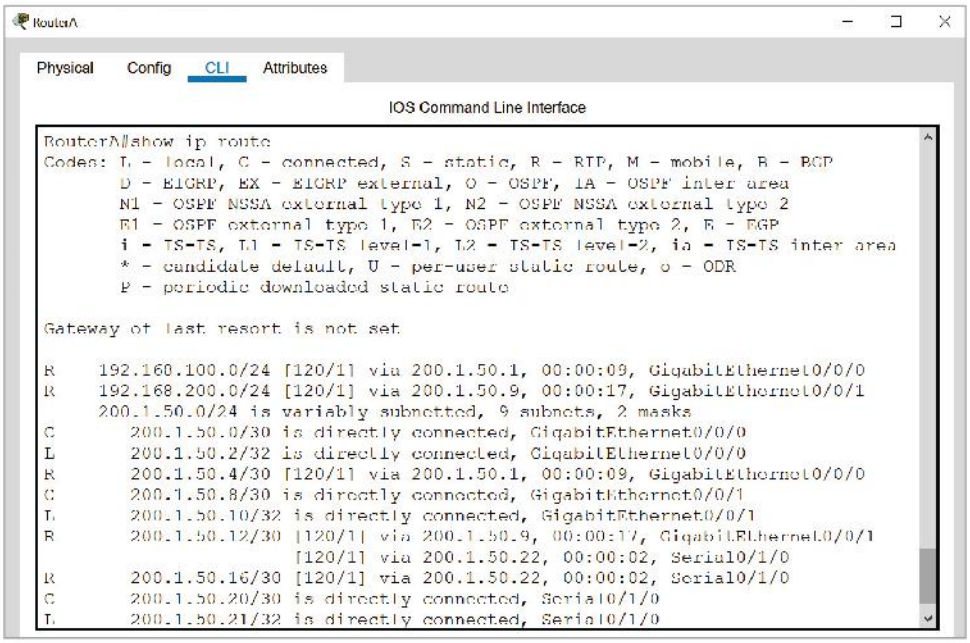
Oppure lavorare con l'interfaccia grafica: selezionare **RIP** nella scheda **Config** e inserire gli indirizzi di rete, cliccando ogni volta su **Add**.



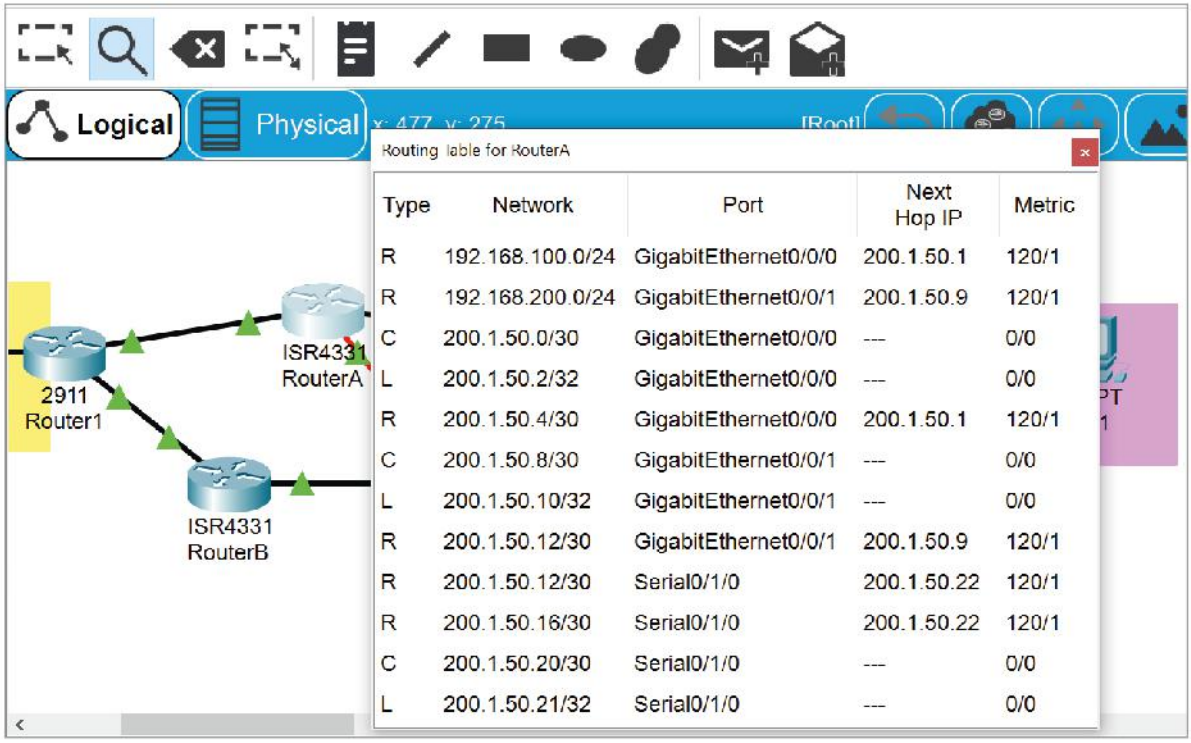
Verifica delle routing table

Per ogni router è possibile leggere il contenuto della routing table, da CLI inserendo il comando

```
RouterA # show ip route
```



Oppure da interfaccia grafica usando lo strumento Inspect raffigurato dall'icona della lente d'ingrandimento. Cliccare con la lente su un router; nel menu che compare si seleziona la voce Routing table:



Type	Network	Port	Next Hop IP	Metric
R	192.168.100.0/24	GigabitEthernet0/0/0	200.1.50.1	120/1
R	192.168.200.0/24	GigabitEthernet0/0/1	200.1.50.9	120/1
C	200.1.50.0/30	GigabitEthernet0/0/0	---	0/0
L	200.1.50.2/32	GigabitEthernet0/0/0	---	0/0
R	200.1.50.4/30	GigabitEthernet0/0/0	200.1.50.1	120/1
C	200.1.50.8/30	GigabitEthernet0/0/1	---	0/0
L	200.1.50.10/32	GigabitEthernet0/0/1	---	0/0
R	200.1.50.12/30	GigabitEthernet0/0/1	200.1.50.9	120/1
R	200.1.50.12/30	Serial0/1/0	200.1.50.22	120/1
R	200.1.50.16/30	Serial0/1/0	200.1.50.22	120/1
C	200.1.50.20/30	Serial0/1/0	---	0/0
L	200.1.50.21/32	Serial0/1/0	---	0/0

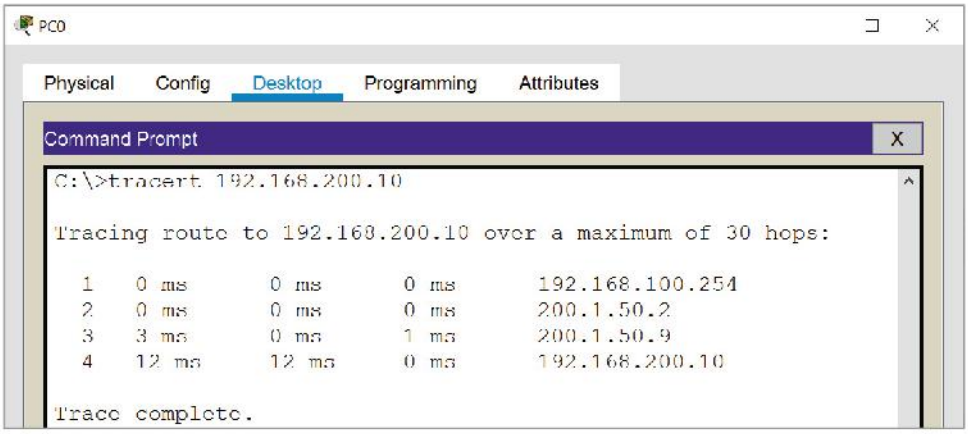
Nella routing table del RouterA (RA) troviamo le seguenti informazioni:

- **Type**: informa su come è stata ottenuta la route. Sono stati definiti vari codici associati alle diverse sorgenti di informazione; nella routing table di RA troviamo:
  - **R** = RIP, il router non è connesso direttamente a questa rete, le informazioni le ha ricevute da altri router tramite il protocollo RIP;
  - **C** = Connected, il router è direttamente connesso a questa rete, quindi per questa route non si memorizza un next hop e la metrica vale 0; per esempio: 200.1.50.0/30 è l'indirizzo di una rete a cui RA è collegato direttamente;
  - **L** = Local, identifica l'indirizzo IP dell'interfaccia di un router di destinazione, collegato direttamente; per esempio: 200.1.50.2/32 è l'indirizzo dell'interfaccia GigabitEthernet0/0/0 del router R1 a cui RA è collegato direttamente tramite un cavo UTP. Questa route è più specifica di quella indicata con il codice C, in quanto permette di indirizzare l'host e non la rete, migliorando l'efficienza nella consegna dei pacchetti nella rete locale. Non è da usare per reti non connesse direttamente;
- **Network**: indica la rete di destinazione, specificando il prefix;
- **Port**: è l'interfaccia del router su cui inviare i pacchetti verso la destinazione indicata in Network;
- **Next Hop IP**: è l'indirizzo IP del nodo di rete verso cui verranno instradati i pacchetti inviati dall'interfaccia specificata in Port;
- **Metric**: è il costo della route; nel caso dei router Cisco, troviamo indicata la **distanza amministrativa** usata quando il router riceve le informazioni di routing, verso una stessa destinazione, da più protocolli di routing diversi. La scelta di quale route inserire nella routing table è fatta sulla base dell'affidabilità del protocollo. Cisco supporta un certo numero di protocolli di routing ai quali ha assegnato un valore di default per la distanza. Eccone alcuni:
  - 0 è il costo più basso, assegnato a una route quando il router è direttamente collegato al Next Hop
  - 1 è usato per le route statiche

- 5 EIGRP
- 20 BGP
- 100 IGRP
- 110 OSPF
- 120 RIP

Questi valori predefiniti possono essere modificati da CLI nel caso in cui l'amministratore volesse dare una maggiore priorità a un protocollo. Per esempio: un router riceve 2 route verso la stessa destinazione, una da IGRP e una da RIP; per dare priorità a RIP è necessario assegnargli una distanza amministrativa inferiore a 120 oppure assegnare a IGRP un valore maggiore di 120. Questa operazione di modifica deve, però, essere svolta con particolare attenzione, in quanto potrebbe generare problemi di loop e black hole.
È importante sottolineare che la distanza amministrativa ha significato solo localmente al router, non viene inviata agli altri router.

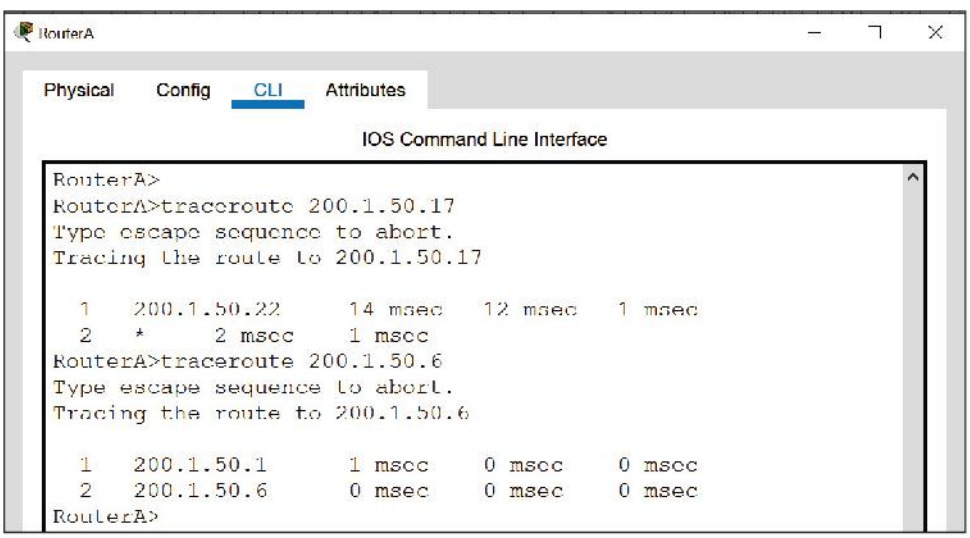
Verifica dei percorsi con il comando traceroute
Sui PC per conoscere i percorsi si esegue il comando **tracert**, dal Command Prompt (scheda Desktop).
Per esempio, verifichiamo il percorso che seguono i pacchetti inviati da PC0 a PC1.



Come si può osservare, il percorso passa dal RouterA, in quanto è quello a costo minore (solo 2 hop).

Sui router, si usa il comando **traceroute** inviato da CLI. Nel seguente esempio si è verificato il percorso che segue un pacchetto dal RouterA al RouterB.

Qui a fianco è mostrato il risultato, differente a seconda che si usi l'indirizzo IP dell'interfaccia GigabitEthernet 0/0/1 (200.1.50.17) oppure quello dell'interfaccia GigabitEthernet 0/0/0 (200.1.50.6).



Provare a fare ulteriori analisi, eseguendo traceroute da e verso varie interfacce della nostra rete.

A CASA

- Ipotizza una tua soluzione al tema proposto, prova a utilizzare un **piano di indirizzamento IP che non usi il subnetting**. Puoi quindi configurare RIP secondo le impostazioni predefinite per RIPv1, tra cui la funzionalità di auto-summary. Verifica come cambiano le informazioni nelle routing table.
- **Elimina il RouterA**, prova a verificare come è cambiata la routing table di Router1: all'istante e a distanza di un minuto. Verifica anche le routing table degli altri router. Verifica con **traceroute** i percorsi.
- Raccogli i tuoi risultati in una presentazione (massimo 3 slide per il primo punto e 6 slide per il secondo punto).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i piani di indirizzamento presentati e come sono cambiate le tabelle di routing con l'eliminazione del RouterA.
- Trovate una spiegazione sui dati contenuti nelle tabelle di routing.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho creato il nuovo piano di indirizzamento senza difficoltà?	Ho avuto difficoltà nel capire come configurare le interfacce. ☐	Con la guida del docente e la collaborazione dei compagni ho creato i nuovi indirizzi. ☐	Ho creato il nuovo piano di indirizzamento autonomamente. ☐	Ho creato il piano di indirizzamento senza difficoltà. ☐
Ho saputo verificare i cambiamenti nelle routing table a seguito della cancellazione del RouterA?	Ho avuto difficoltà a capire dove verificare i dati nelle routing table e a eseguire il traceroute. ☐	Con la guida del docente e la collaborazione dei compagni ho trovato le routing table. ☐	Ho verificato le routing table di tutti i router e i nuovi percorsi nella rete, in modo autonomo. ☐	Ho verificato tutte le routing table ed eseguito traceroute da vari punti, analizzando i cambiamenti. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ

6 IL TRANSPORT LAYER DEL TCP/IP

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

conoscenze	abilità	competenze
Organizzare il software di comunicazione in livelli. Conoscere gli standard internazionali definiti per il livello Transport. Conoscere il funzionamento delle porte e delle socket. Conoscere i protocolli fondamentali del livello Transport.	Saper usare i numeri di porta opportuni per le comunicazioni client-server tra applicativi. Saper distinguere servizi connectionless e servizi connection-oriented. Saper affrontare le vulnerabilità dei protocolli del livello Transport.	Classificare una rete e i servizi offerti con riferimento agli standard tecnologici e utilizzando correttamente la relativa terminologia. Saper scegliere il tipo di protocollo di trasporto in base al grado di affidabilità, alla velocità e alla sicurezza del servizio che si vuole offrire.

Didattica Digitale Integrata Plus

FLIPPED CLASSROOM

A casa

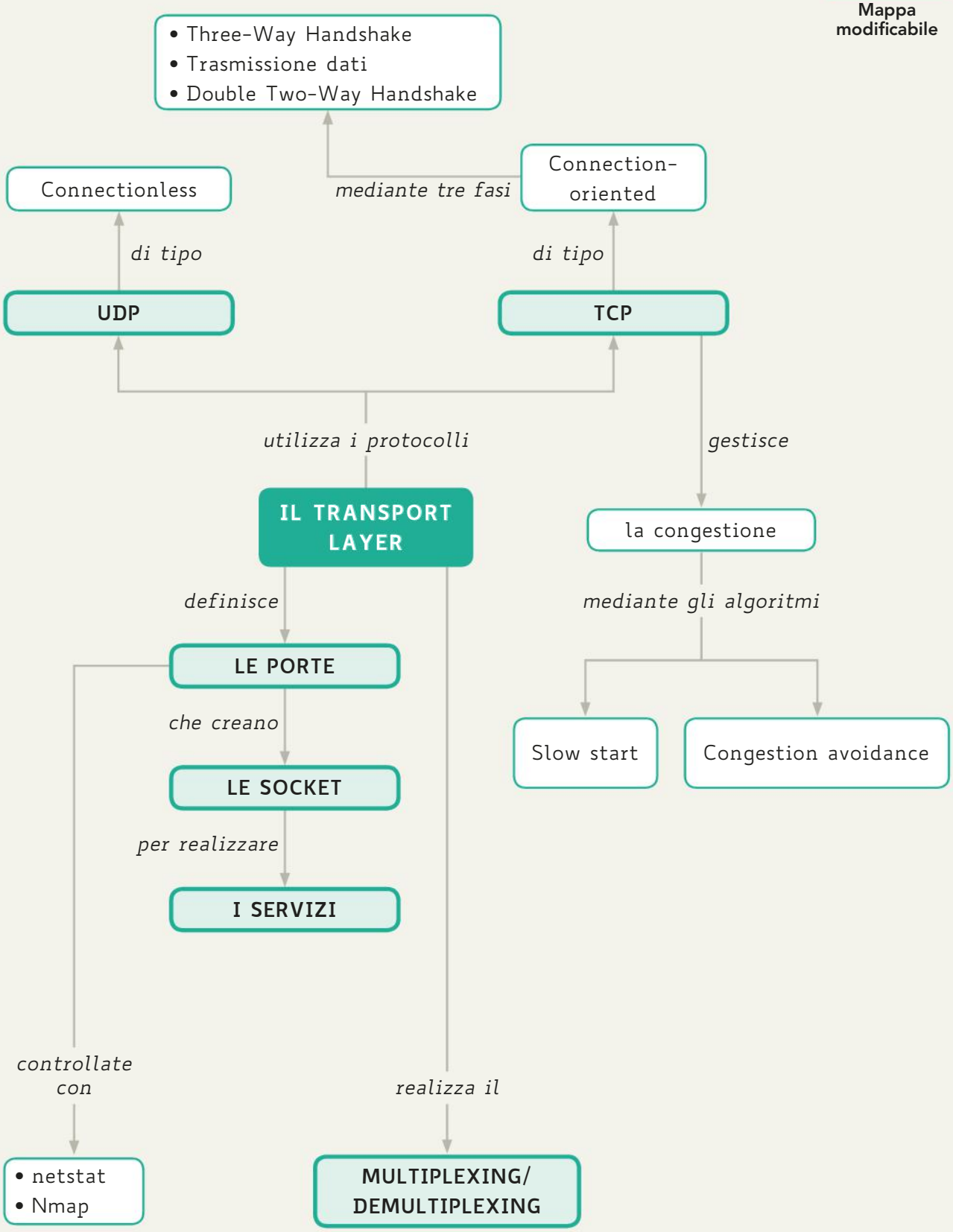
- Leggi la Lezione 8 di questa unità;
- prova il comando *netstat -na* sul tuo PC o sul PC del laboratorio;
- approfondisci i servizi che trovi in ascolto (*listening*) verificando in particolare quelli associati alle Well Known Ports;
- raccogli i risultati in una presentazione (massimo 5 slide).

In classe

- Confrontate i risultati raccolti;
- valutate se tra queste vi sono differenze;
- discutete i motivi che spiegano le eventuali differenze.

MAPPA CONCETTUALE

Mappa modificabile



1 LE PORTE, LE SOCKET E I SERVIZI

1.1 Le porte

#prendinota

Uno dei principali compiti dei protocolli del livello Transport è quello di individuare, tra i tanti applicativi che risiedono su un computer host, quale sia il destinatario del pacchetto ricevuto dalla rete.

Nelle precedenti Unità sono stati descritti i livelli inferiori dello stack TCP/IP, che si occupano della comunicazione host-nodo e nodo-nodo. Grazie all'organizzazione a livelli, i protocolli del livello di trasporto (**Transport Layer**) non necessitano di conoscere i dettagli della rete che utilizzano.
In particolare, si è visto l'importante ruolo del protocollo IP nel trasferire i datagram attraverso Internet. Una volta arrivato a destinazione, però, l'header del datagram IP non contiene alcuna informazione utile all'host ricevente per individuare l'applicazione destinataria del messaggio.

Infatti i protocolli del livello Network svolgono la funzione di far attraversare la rete al pacchetto per poi consegnarlo al sistema di destinazione; a questo punto il loro compito è finito.
Poiché su un host ci possono essere più applicazioni aperte contemporaneamente che che generano un **#processo** di richiesta o anche più di uno, si pone il problema per l'host destinatario di sapere a quale processo attivo (di sistema o utente) consegnare il pacchetto ricevuto.

#techwords

Un **processo** è una sequenza di attività (task) generata da un programma applicativo utente o di sistema, eseguito su un processore sotto la gestione del rispettivo Sistema Operativo.

Il problema di consegnare il messaggio all'applicazione finale potrebbe essere risolto semplicemente individuando il processo relativo a quella applicazione, ma la questione è tutt'altro che semplice. Infatti i processi vengono creati ed eliminati dinamicamente, non possono quindi essere noti ai potenziali mittenti che si trovano in Internet.
Per esempio, un host mittente deve poter richiedere una pagina web che risiede su un computer server di cui conosce l'indirizzo, senza dover sapere qual è il processo che implementa la funzione di web server.
Non solo, uno stesso processo potrebbe gestire più funzionalità, ma una sola è quella destinataria del messaggio.

> Le applicazioni finali devono allora poter essere individuate non in base al processo che le esegue, bensì alle **funzioni** che richiedono (client) o che svolgono (server). Occorre quindi trovare un sistema che consenta di individuare in modo univoco ogni comunicazione in rete in base alla sua funzione (web server, mail server, comunicazione tra App, accesso a social network, ...). Sulla base di questo approccio, il primo passo è stato stabilire che in ogni host siano definiti dei **punti di accesso**, chiamati **porte** (ports), che vengono associati al processo che sta svolgendo in quel momento la funzione richiesta e al protocollo utilizzato. Solo alla porta predisposta verranno consegnati i pacchetti che arrivano dalla rete.

Ogni porta viene identificata da un numero intero positivo codificato in 16 bit (range: 0 - 65.535).
I **numeri di porta** sono assegnati a livello internazionale da IANA e suddivisi in tre gruppi:

- Well Known Ports (da 0 a 1.023);
- Registered Ports (da 1.024 a 49.151);
- Dynamic and/or Private Ports (da 49.152 a 65.535).

Nel seguito si riporta l'introduzione del documento ufficiale che contiene il registro dei numeri di porta.

IN ENGLISH PLEASE

PORT NUMBERS

The port numbers are divided into three ranges: the Well Known Ports, the Registered Ports, and the Dynamic and/or Private Ports.

The Well Known Ports are those from 0 through 1023.
Well Known Ports SHOULD NOT be used without IANA registration.
The registration procedure is defined in [RFC4340], Section 19.9.

The Registered Ports are those from 1024 through 49151.
Registered ports SHOULD NOT be used without IANA registration.
The registration procedure is defined in [RFC4340], Section 19.9.

The Dynamic and/or Private Ports are those from 49152 through 65535.

A value of 0 in the port numbers registry below indicates that no port has been allocated.

Per quanto possibile, IANA ha assegnato le stesse porte sia per UDP sia per TCP, i due principali protocolli definiti per il livello Transport (successive Lezioni 3 e 4).

L'host client e l'host server, oltre al reciproco indirizzo IP, devono conoscere i rispettivi numeri di porta utilizzati dall'applicazione che ha generato il processo relativo a una determinata funzione.

Tali porte sono dette:

- **source port**: numero della porta presente sull'host client, sulla quale il processo client è in attesa di risposta dal server (è un'informazione utilizzata dal server);
- **destination port**: numero della porta su cui il processo server è in ascolto o su cui fornisce il servizio (è un'informazione utilizzata dal client).

Nella FIGURA 1 si mostra un esempio di utilizzo delle porte nel caso di un'applicazione di trasferimento file con FTP (File Transfert Protocol), protocollo del livello applicazione che affronteremo nell'Unità 8.

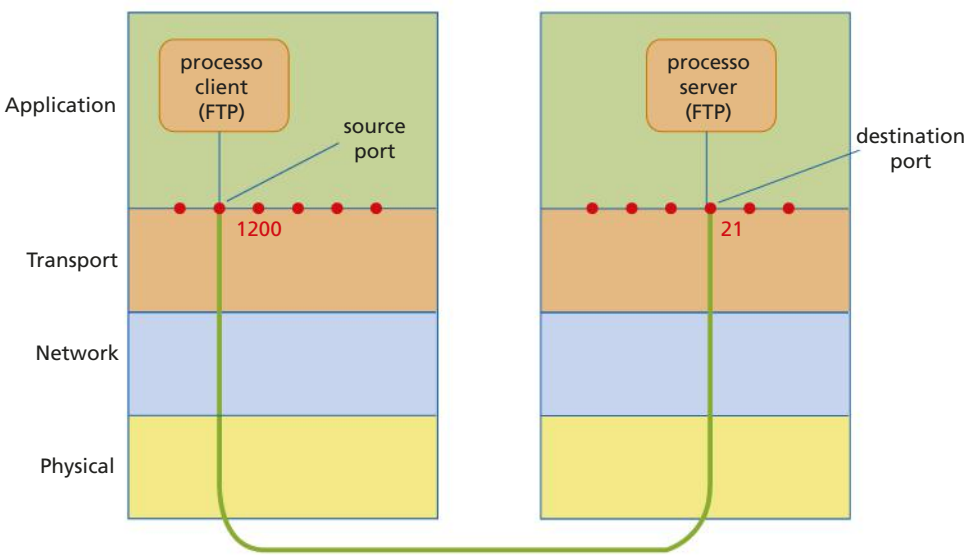


FIGURA 1 Porte a livello Transport

1.2 Le socket

Come abbiamo detto le porte sono il primo passo per stabilire un canale bidirezionale univocamente individuabile tra processo client e processo server relativamente a una funzione.
Un client che si rivolge a un server, in ascolto su una propria porta, dovrà innanzitutto fare una richiesta di connessione al servizio comunicando il proprio indirizzo IP e la porta su cui vuole ricevere la risposta. Il server, dopo un'eventuale fase di impostazione utile al trasferimento successivo delle informazioni, stabilirà la connessione comunicando a sua volta il proprio indirizzo IP e la porta su cui avverrà la comunicazione, diversa da quella su cui era in ascolto (FIGURA 2).

Un pacchetto che viaggia nella rete deve quindi contenere 4 informazioni fondamentali:

(Client IP address, Server IP address, Client Port number, Server Port number)

Specificando anche il protocollo per la comunicazione si riesce a descrivere univocamente la connessione con una **quintupla** detta **Association**:
(protocollo, ind. logico locale, porta locale, ind. logico remoto, porta remota)

esempio: (TCP, 200.18.6.12, 3000, 58.163.138.10, 4000)

La quintupla viene in realtà costituita a partire da due elementi simmetrici, un'associazione locale e una remota:

- protocollo, ind. logico locale, porta locale (TCP, 200.18.6.12,3000);
- protocollo, ind. logico remoto, porta remota (TCP, 58.163.138.10,4000).

Ciascuna di queste parti è una **socket** su un host diverso.
A ogni processo che richiede o risponde a una comunicazione in rete viene assegnata una socket che ha il compito di interfacciare l'applicazione client che ha generato il processo (di richiesta) con l'applicazione server che fornisce il servizio (in risposta).

#prendinota

Più client possono comunicare attraverso la stessa socket, ma solo un server può essere associato a una definita socket.

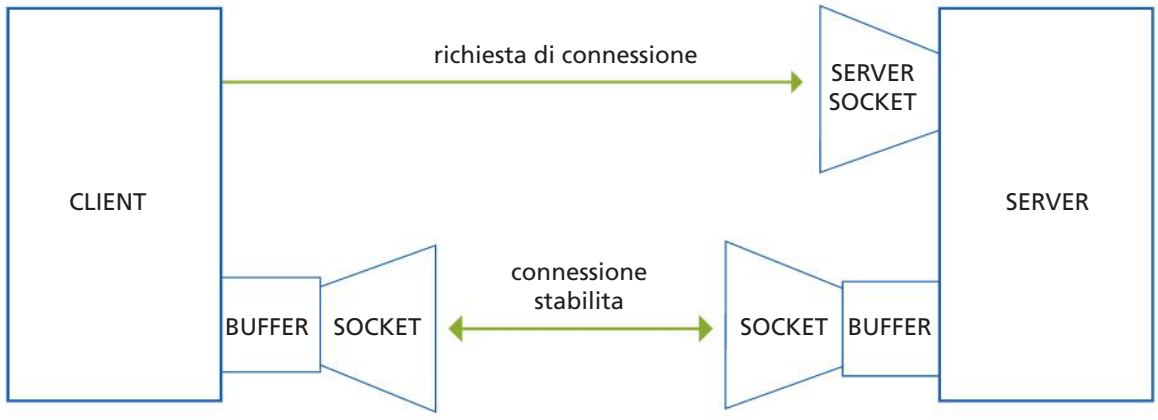


FIGURA 2 Apertura delle socket tra processo server e processo client

#techwords

Con **primitiva** si intende una chiamata di sistema a una routine del kernel.

Agli inizi degli anni Ottanta, l'ARPA con l'Università di Berkeley creò un'interfaccia per far comunicare applicazioni che girano su macchine UNIX connesse in rete. Fu deciso anche di adoperare il sistema di **chiamate a funzioni** esistente in UNIX e di aggiungere un nuovo sistema di chiamate con apposite **#primitive** per supportare le funzioni TCP/IP (TABELLA 1). Il risultato fu una interfaccia socket BSD (Berkeley Software Distribution) UNIX.

PRIMITIVA	DESCRIZIONE	APPLICAZIONE CHE LA USA
SOCKET	Crea un nuovo punto di comunicazione (end point). I parametri specificano il formato degli indirizzi, il tipo di servizio (per esempio: reliable byte stream) e il protocollo (per esempio: TCP). Restituisce un descrittore del file da usare nelle chiamate successive, in modo analogo a quanto fatto da una normale Open di un file.	SERVER CLIENT
BIND	Associa (bind) un indirizzo locale alle socket, così da consentire ai client remoti di connettersi.	SERVER
LISTEN	Alloca spazio per accodare le chiamate entranti nel caso in cui più client cerchino di connettersi al server contemporaneamente. Listen non è bloccante.	SERVER
ACCEPT	Blocca il server finché non riceve da un client una Connect Request. Il TCP crea una nuova socket con le stesse caratteristiche di quello usato con la LISTEN e viene restituito il relativo descrittore di file. A questo punto il server crea un nuovo processo (tramite fork) o un thread per gestire la connessione sulla nuova socket e ritorna in attesa di una nuova richiesta di connessione sulla socket originaria inviando una primitiva ACCEPT.	SERVER
CONNECT	Questa primitiva è usata dal client dopo che ha creato una socket. Essa blocca il client e attiva il processo che gestisce l'instaurazione della connessione con il server. Quando questa è avvenuta (a seguito della ricezione della primitiva di conferma dal server), il processo client è sbloccato e la connessione è attiva.	CLIENT
SEND	Invia i dati sulla connessione tra client e server.	SERVER, CLIENT
RECEIVE	Blocca l'applicazione (client o server) in attesa di ricevere i dati dalla connessione.	SERVER, CLIENT
CLOSE	Rilascio simmetrico della connessione: sia l'applicazione client che quella server devono chiudere la connessione.	SERVER, CLIENT

TABELLA 1 Le *Berkeley Sockets*: le primitive UNIX Berkeley utilizzate per TCP

LABORATORIO ONLINE

LA PROGRAMMAZIONE SOCKET IN JAVA

In questo laboratorio si descrivono due esercitazioni (una con TCP e una con UDP) utili a comprendere l'uso dei socket nella realizzazione di programmi Client-Server. Usiamo il linguaggio Java, in quanto fornisce un'API object-oriented completa e semplice.

Nel 1991 Microsoft decise di definire una **#API** (Application Program Interface) standard per applicazioni TCP/IP in ambiente Windows **(FIGURA 3)** che permette di semplificare il dialogo tra applicazioni residenti su host diversi.

#techwords

Con **API** si indica un insieme di procedure (in genere raggruppate per strumenti specifici) predisposte all'espletamento di un dato compito; spesso tale termine designa le librerie software di un linguaggio di programmazione. Le API sono quindi un'interfaccia per programmatori.

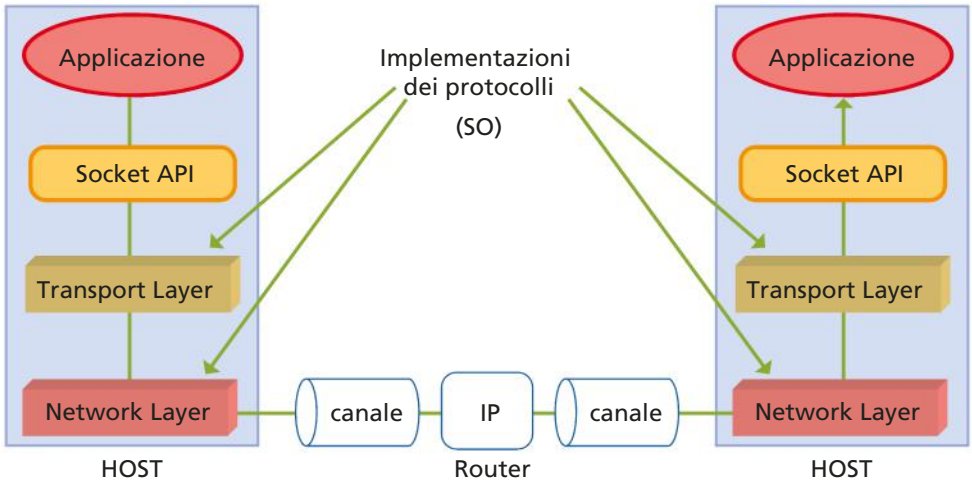


FIGURA 3 Socket API

LABORATORIO ONLINE

LA PROGRAMMAZIONE SOCKET IN C#

In questo laboratorio riproponiamo l'esercizio del laboratorio online precedente utilizzando il linguaggio C# in ambiente .NET Framework che fornisce anch'esso un'API object-oriented completa e semplice. Si potrà in questo modo cogliere le similitudini tra le tecnologie C# e Java.

Ogni linguaggio di programmazione, che sia di alto o basso livello, deve fornire delle API che permettano di lavorare con le socket.
In pratica le API permettono agli sviluppatori software di accedere a determinate funzioni, altrimenti inaccessibili, di un programma o servizio web in modo da creare un nuovo utilizzo.
Tantissimi servizi Internet hanno ormai le loro librerie API e le forniscono in maniera open. Per esempio Facebook, Google Maps o Twitter hanno le loro e sono utilizzate per tantissimi altri servizi o da innumerevoli altri siti.

1.3 I servizi

I protocolli implementati a livello Transport svolgono funzioni simili a quelli del livello Data Link, per esempio si occupano del controllo degli errori, della sequenza corretta dei dati, del controllo di flusso, ecc. La differenza fondamentale tra i due tipi di protocollo è lo scenario di rete in cui operano: a livello Data Link la connessione tra il router che invia il pacchetto a un altro router è diretta, infatti i due router comunicano attraverso un canale fisico di trasmissione; a livello Transport invece la connessione avviene attraverso l'intera rete, si tratta di un canale *logico* di trasmissione che unisce il computer host mittente con il computer host destinatario.

Per questo motivo si dice che il livello Transport si occupa della comunicazione **end-to-end**, dove i due estremi della connessione sono gli host che vogliono comunicare (FIGURA 4) o più precisamente le socket predisposte sui due host.

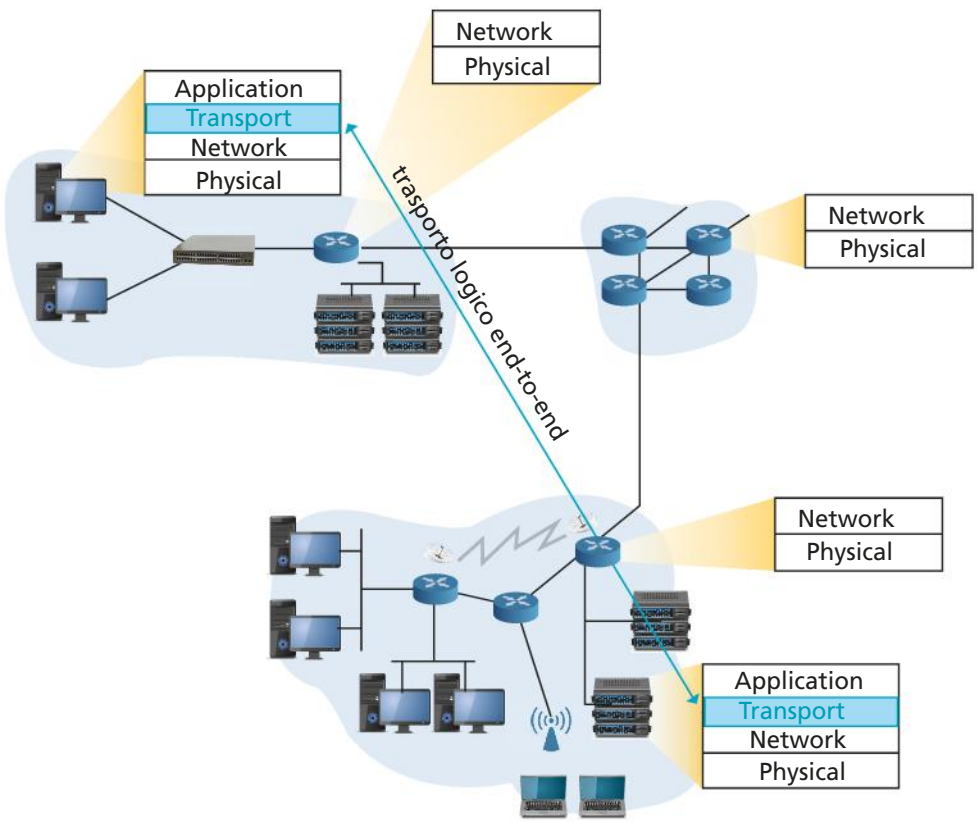


FIGURA 4 La comunicazione end-to-end a livello Transport

Il livello Transport svolge quindi una funzione "cuscinetto" tra il livello Application e i livelli inferiori, che si occupano della trasmissione in rete dei dati. In questo modo le applicazioni non necessitano di conoscere i dettagli operativi e organizzativi della rete che utilizzano, non interessa loro sapere il tipo di computer di destinazione, il percorso che faranno i dati, quanto è grande la rete che utilizzano o quale mezzo trasmissivo verrà usato. Analogamente, anche i livelli inferiori non sono a conoscenza che esistono molte applicazioni che inviano dati in rete, loro hanno solo il compito di consegnare i messaggi a destinazione.

Nella FIGURA 5 viene evidenziato il ruolo del livello Transport nella comunicazione tra applicazioni differenti, installate su diverse tipologie di dispositivi (device).

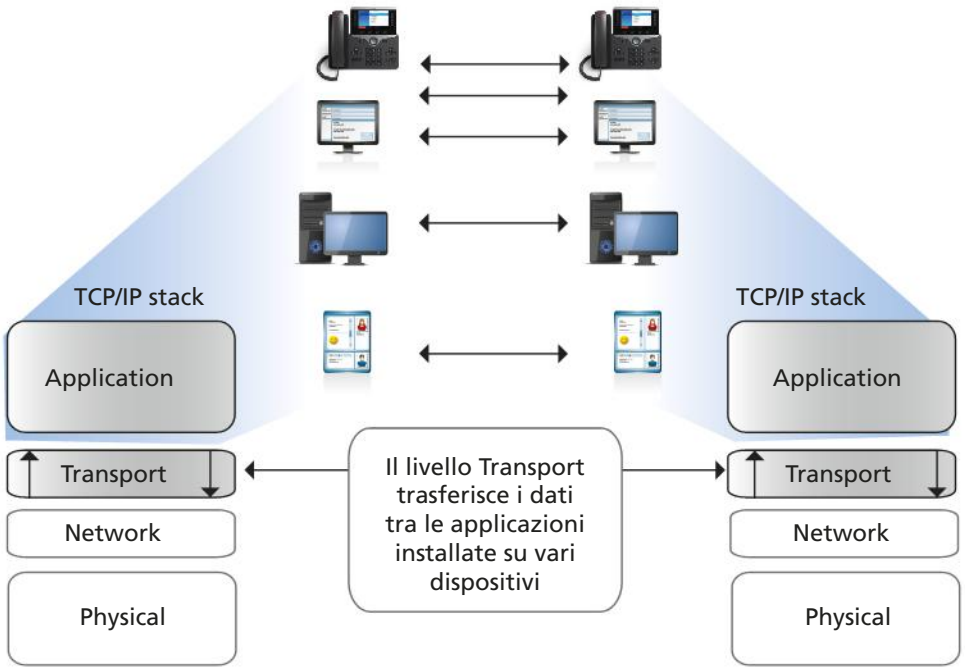


FIGURA 5 Il livello Transport permette la comunicazione tra applicazioni/processi attraverso la rete

Esistono vari protocolli di trasporto che sono stati standardizzati per soddisfare le differenti esigenze delle diverse applicazioni. Nel seguito si esamineranno i due protocolli di trasporto più diffusi: **UDP** e **TCP**.

UDP e TCP sono anche stati tra i primi a essere standardizzati. In generale il livello Transport non offre servizi di tipo real time (come lo streaming audio-video) né una banda garantita. Vi sono però dei protocolli sviluppati dalla IETF che svolgono le funzioni del livello di trasporto (come TCP o UDP) appoggiandosi a un servizio di rete a pacchetto come IP. Tra i più recenti c'è per esempio SCTP (Stream Control Transmission Protocol) sviluppato nel 2000 per supportare la telefonia su reti IP e il cui uso si è poi allargato alle applicazioni multistreaming e **#multihoming**.

SCTP combina diverse caratteristiche dei protocolli TCP e UDP, anch'essi utilizzati per il trasferimento dati, e include meccanismi per il controllo di congestione e il miglioramento della tolleranza ai guasti durante l'invio di pacchetti. Grazie alla sua elevata flessibilità, l'SCTP è utilizzato anche in altri modi (per esempio nella gestione e nell'amministrazione di pool di server).

Nei documenti RFC di IETF, che descrivono i protocolli UDP e TCP, la Transport Protocol Data Unit (TPDU) è indicata con nomi differenti: **datagram** in UDP e **segment** in TCP.

#techwords

Con **multihoming** si intende la possibilità di avere più indirizzi di rete in un computer, di solito per interfacciare reti diverse.

I protocolli del livello Transport mettono in comunicazione le applicazioni, o meglio i processi, presenti su due host remoti offrendo un servizio denominato **multiplexing/demultiplexing** (descritto nella Lezione 2), insieme al controllo dell'integrità dei dati.

Inoltre, un protocollo come TCP fornisce anche la garanzia di consegna dei dati, effettuando il controllo della congestione (descritta nella Lezione 5) e il controllo di flusso (sliding windows).

La FIGURA 6 mostra i servizi offerti dal livello Transport alle applicazioni.

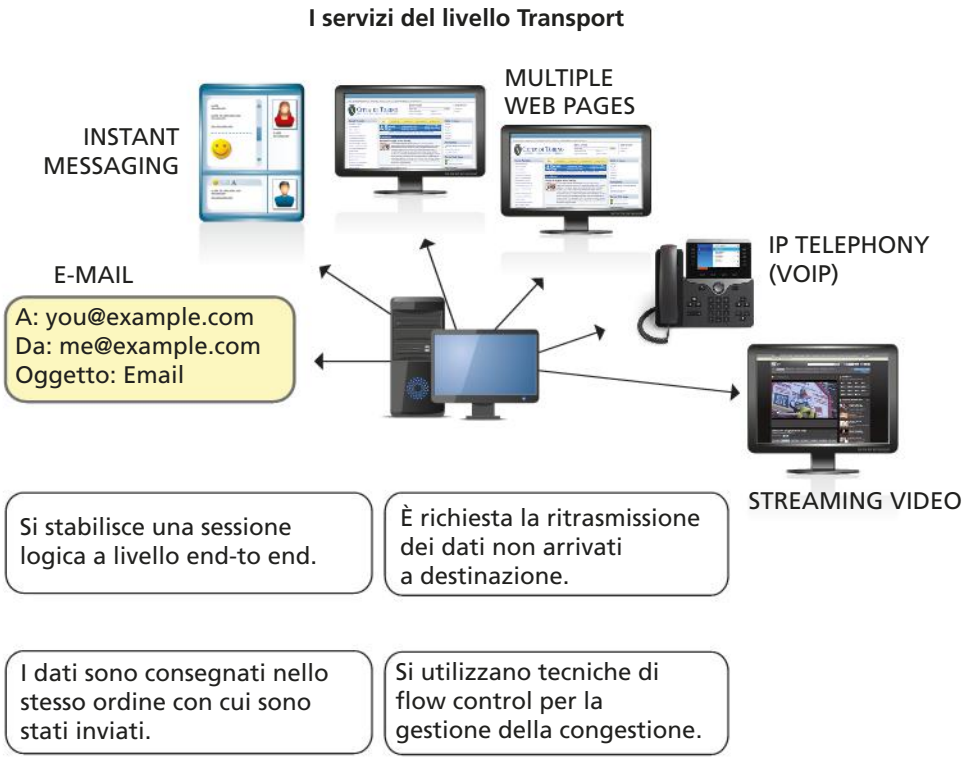


FIGURA 6 I servizi di connessione e affidabilità del livello Transport

Non tutti i protocolli di questo livello offrono i servizi elencati. Infatti vedremo che il protocollo UDP non prevede l'uso di una connessione end-to-end e non offre un servizio affidabile.

FISSA LE CONOSCENZE

- Qual è la differenza fondamentale della comunicazione a livello Transport rispetto a quella dei livelli inferiori?
- Qual è il problema che si pone nella consegna del datagram IP, una volta raggiunto l'host di destinazione?
- Che cosa sono e a cosa servono le porte?
- Che cosa sono e a cosa servono le socket?
- Che cosa va definito in una comunicazione end-to-end?

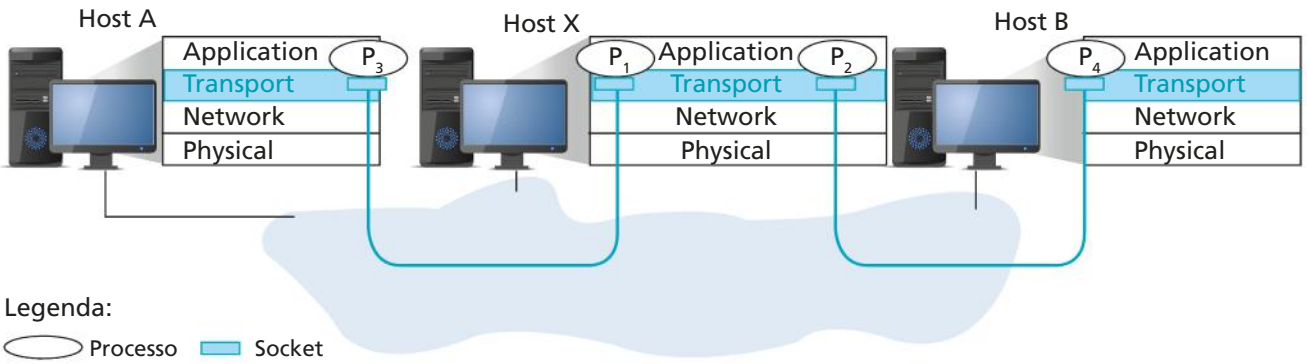
2 LE FUNZIONALITÀ DI MULTIPLEXING E DEMULTIPLEXING

La creazione delle socket permette di realizzare le funzionalità di multiplexing e demultiplexing:

- **multiplexing**: in trasmissione il livello Transport riceve i dati dalle socket e aggiunge le proprie informazioni di controllo (header);
- **demultiplexing**: in ricezione il livello Transport legge l'header e determina a quale socket consegnare i dati.

Per aver un'idea più chiara di questa funzionalità, che è particolarmente critica, si può far riferimento al servizio postale tradizionale che recapita più lettere nella cassetta della posta di una famiglia (il cui indirizzo di abitazione equivale all'indirizzo IP dell'host). L'attività di demultiplexing consiste nel prendere le lettere dalla cassetta e consegnarle ai vari membri della famiglia destinatari di ciascuna di esse. Al contrario, l'attività di multiplexing consiste nel ricevere varie lettere dai diversi membri della famiglia e inserirle in un'unica cassetta di spedizione.
La seguente **FIGURA 7** mostra l'operazione di demultiplexing effettuata dal livello Transport dell'host X, che riceve i dati dal processo P3, residente sull'host A, per il processo P1 e dal processo P4, residente sull'host B, per il processo P2. I dati ricevuti vengono inviati ai processi destinatari tramite l'interfaccia socket (ossia la destination port) specificata nel messaggio ricevuto.

FIGURA 7 Demultiplexing, i dati vengono consegnati al processo applicativo destinatario del messaggio



Le operazioni di multiplexing/demultiplexing possono avvenire sia in presenza, che in assenza, di una connessione:

- **multiplexing/demultiplexing connectionless:** è il caso del protocollo UDP (descritto nella Lezione 3) in cui è previsto che più client accedano allo stesso servizio sullo stesso server. La socket è individuata in questo caso solo dalla *coppia*:

  <Server IP address><Server Port number>

  di 32 e 16 bit, rispettivamente;
- **multiplexing/demultiplexing connection-oriented:** è il caso del protocollo TCP (descritto nella Lezione 4) in cui è previsto che più client accedano allo stesso servizio sullo stesso server e che uno stesso client possa attivare più sessioni dello

#prendinota

L'aver definito la comunicazione a livello Transport come *end-to-end*, non deve far pensare che obbligatoriamente si debba instaurare una connessione tra host mittente e host ricevente.
TCP instaura una connessione, UDP no.

stesso servizio. Le connessioni che vengono stabilite sono identificate dalla socket completa cioè dalla *quadrupla*:

<Client IP address><Server IP address><Client Port number><Server Port number>

di 32, 32, 16 e 16 bit rispettivamente.

Si noti che, nei due punti sopra elencati, si è fatto uso della terminologia client-server al posto di host mittente (source) e host ricevente (destination), in quanto è proprio a livello Transport che si inizia a individuare come la comunicazione in Internet avvenga tipicamente tra un'applicazione client e un'applicazione server, presente su un computer remoto.

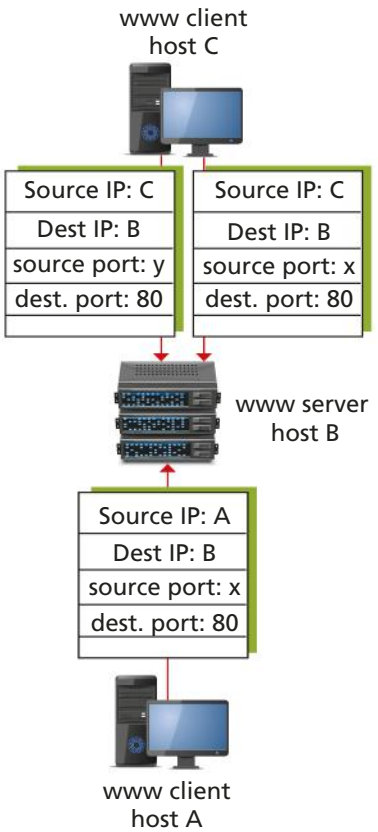


FIGURA 8 Esempio di multiplexing/demultiplexing per applicazioni web

Questo concetto è esemplificato nella **FIGURA 8**, dove si fa riferimento al servizio World Wide Web presente sul computer B, al quale si connettono le applicazioni client presenti sui computer A e C. In questo esempio è necessario usare il multiplexing/demultiplexing connection-oriented.
Nell'esempio raffigurato, il PC www client, host A, invia al www server, host B, un segmento con **destination port number** 80 (il numero di porta assegnato da IANA alle applicazioni che usano il protocollo HTTP), mentre come **source port number** specifica un numero (x in Figura 8) che non è usato da nessun altro processo su quel computer. Ogni segmento che l'host A invierà all'host B avrà source port = x e destination port = 80.
Quando il segmento arriva al server B, questi potrà consegnarlo al processo applicativo corretto, associato alla destination port 80 (porta standard del processo HTTP sul server B) e potrà rispondere con un messaggio indirizzato alla source port x (porta specifica del processo HTTP che gestisce la richiesta sul client A). Quindi, simmetricamente, i segmenti che il server B invia al client A avranno source port = 80 e destination port = x; tale coppia permetterà ad A di identificare il processo applicativo corretto a cui consegnare il segmento ricevuto.
Il livello Transport trova automaticamente un numero di porta libero da usare come source port x, in modo trasparente all'applicazione. Su alcuni sistemi esiste comunque la possibilità che l'applicazione richieda in modo esplicito di usare un determinato numero di porta (per esempio nei sistemi Unix-like si usa a questo scopo la system call bind() descritta nella Tabella 1 della Lezione 1).
Dal momento che la scelta del source port number è fatta a livello del singolo client, può capitare che due client utilizzino lo stesso numero (Figura 8, dove sia A che C hanno scelto il numero x), questo però non causa conflitti in quanto il server usa anche l'indirizzo IP (e i due host, A e C, hanno un indirizzo IP distinto).

FISSA LE CONOSCENZE

- In che cosa consistono le operazioni di multiplexing e demultiplexing svolte dal livello Transport?
- È sempre necessario stabilire una connessione logica tra source host e destination host?
- Perché per le applicazioni web si usa il multiplexing/demultiplexing connection-oriented?

3 UN PROTOCOLLO DI TRASPORTO CONNECTIONLESS: UDP

3.1 UDP

User Datagram Protocol è un protocollo del livello Transport che non prevede l'uso di una connessione tra host mittente e host destinatario: infatti ciascun datagram UDP è trattato in modo indipendente.

Il servizio offerto da UDP è di tipo Best Effort: i datagram UDP possono essere persi o arrivare fuori sequenza, non si ha quindi alcuna garanzia sulla consegna dei dati trasmessi (FIGURA 9).

A prima vista quindi sembrerebbe non offrire un servizio significativamente diverso da quello offerto dal protocollo IP, in realtà non è così: UDP fornisce le funzionalità tipiche del livello Transport in termini di multiplexing, grazie all'uso delle porte, e di controllo dell'integrità dei dati (il campo Checksum, però, è opzionale).

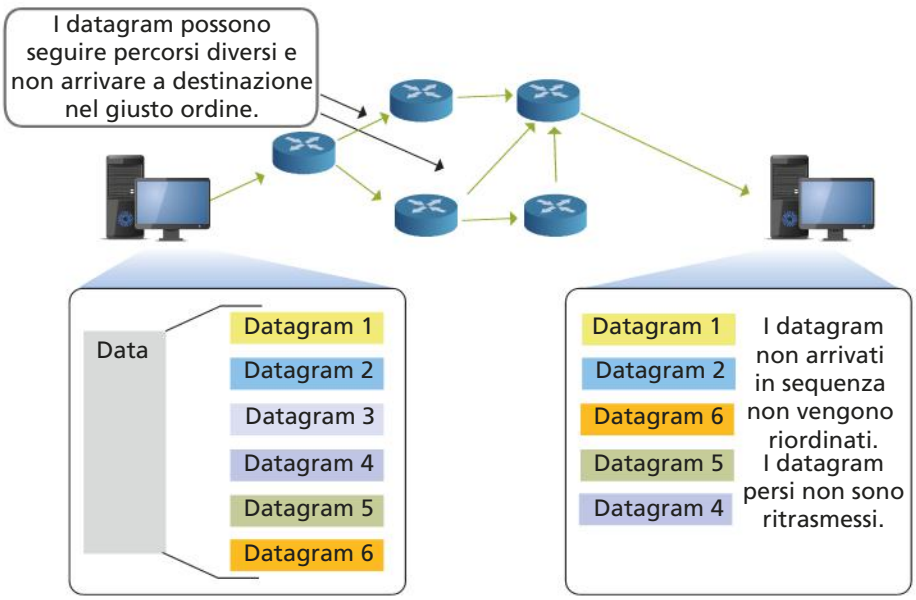


FIGURA 9 Le caratteristiche del protocollo UDP (senza connessione e non affidabile)

IN ENGLISH PLEASE

Network Working Group — J. Postel
RFC 768 — ISI
Category: Standard — 28 August 1980

User Datagram Protocol

Introduction

This User Datagram Protocol (UDP) is defined to make available a datagram mode of packet-switched computer communication in the environment of an interconnected set of computer networks. This protocol assumes that the Internet Protocol (IP) [1] is used as the underlying protocol.

This protocol provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism. The protocol is transaction oriented, and delivery and duplicate protection are not guaranteed. Applications requiring ordered reliable delivery of streams of data should use the Transmission Control Protocol (TCP) [2].

3.2 Il datagram UDP

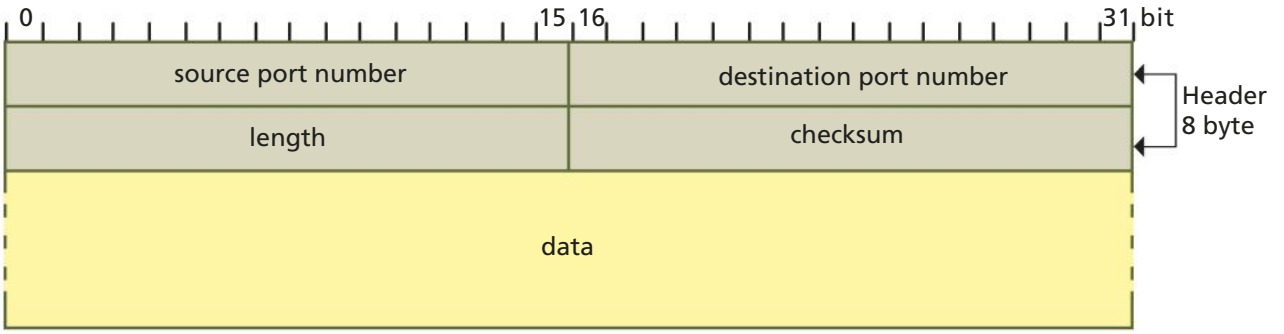


FIGURA 10 Datagram UDP

La FIGURA 10 mostra il formato del datagram del protocollo UDP, i cui campi sono:

- **source port number** (16 bit): numero di porta sull'host del mittente;
- **destination port number** (16 bit): numero di porta sull'host del destinatario;
- **length** (16 bit): contiene la lunghezza totale in byte del datagram UDP (header + dati);
- **checksum** (16 bit, opzionale): contiene il codice di controllo del datagram UDP. L'algoritmo di calcolo è definito nell'RFC del protocollo e copre il datagram UDP e parte dell'header IP. Se dal calcolo della checksum risulta che il datagram UDP è stato danneggiato, esso viene scartato;
- **data**: contiene le informazioni trasmesse/ricevute.

La dimensione massima di un datagram UDP è 65.508 byte. Infatti deve essere contenuto in un pacchetto IP che ha al massimo 64 KB (65.536 byte) ai quali si devono togliere i 20 byte minimi dell'header IP, che porta ad avere un campo dati IP al massimo di 65.516 byte ai quali si devono ancora togliere gli 8 byte dell'header UDP.
L'opzionalità del campo checksum consente alle implementazioni di lavorare con la massima velocità quando usano UDP su una rete altamente affidabile (come le reti LAN).
Dal momento, però, che il protocollo IP non calcola la checksum sulla parte dei dati del pacchetto, il calcolo della checksum UDP permette di verificare se tutti i dati sono arrivati integri. Quindi omettere questo campo vuol dire eliminare del tutto la possibilità di sapere se il messaggio è arrivato a destinazione in modo corretto.

Per quanto riguarda il **multiplexing/demultiplexing connectionless** del protocollo UDP, possiamo distinguere le due fasi: trasmissione in multiplexing e ricezione in demultiplexing.

I passi e l'utilizzo dei campi del datagram nelle due fasi sono le seguenti.

Multiplexing:

- il datagram UDP viene formato aggiungendo l'header al messaggio ricevuto dal livello applicativo:
  - il campo source port viene preso dalla socket attraverso cui è stato inviato il messaggio;
  - il campo destination port è un parametro della primitiva Data_Request ricevuta dal livello applicativo;
  - viene calcolato il campo length;
  - viene calcolato il campo checksum;

- il datagram UDP così costruito viene inviato al livello Network, che vi aggiunge il proprio header in cui inserirà il numero 17 che identifica il protocollo di trasporto UDP.

Demultiplexing:
- viene ricevuto dal livello Network un datagram UDP;
- se ne verifica l'integrità calcolando il suo valore di controllo e confrontandolo con quello contenuto nel campo checksum;
- si cerca la socket UDP associata al numero di porta contenuto nel campo destination port;
- la parte data, del datagram UDP, viene inviata alla socket individuata; se non viene trovata allora il datagram è respinto e viene generato un messaggio ICMP di porta non raggiungibile.

VANTAGGI DI UDP

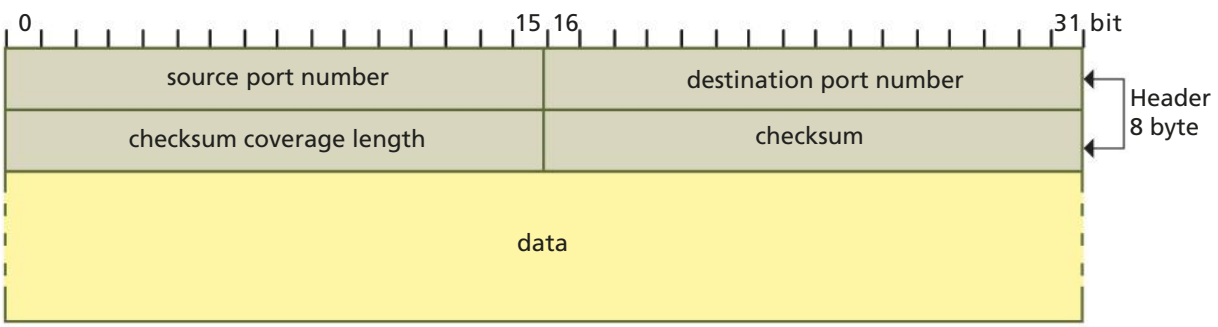
Alcuni dei vantaggi derivanti dall'uso di UDP sono:
- non richiede di stabilire una connessione: non introduce un ritardo dovuto alla fase di setup della connessione;
- non mantiene lo stato della connessione: un server può supportare molti più client attivi;
- il sovraccarico dovuto all'intestazione del pacchetto è minimo: solo 8 byte contro i 20 di un protocollo connection-oriented;
- il controllo del livello applicativo è più efficace: in mancanza di un controllo della congestione, il mittente non viene mai bloccato.

3.3 UDP-Lite

Nel corso degli ultimi anni è sorta l'esigenza di non scartare i datagram UDP che risultano errati dopo il controllo del campo checksum. Infatti, per certi tipi di applicazioni ricevere parte dei dati è meglio che non ricevere nulla. Tipici esempi sono le applicazioni VoIP e di streaming audio/video che usano pacchetti con una gran quantità di dati. Con il protocollo UDP tradizionale, è sufficiente avere un byte errato per scartare tutto il datagram, con conseguente impatto sulla qualità delle informazioni ricevute; con UDP-Lite, invece, si può salvare la parte di dati arrivata correttamente. Per esempio, in una comunicazione VoIP il ricevente riesce comunque a capire il contenuto informativo trasmesso da chi gli sta parlando.

UDP-Lite fa parte del kernel di Linux dalla versione 2.6.20 in poi.
Nella FIGURA 11 è mostrato il formato del datagram UDP-Lite.

FIGURA 11
Datagram UDP-Lite



Il controllo checksum come minimo si deve applicare agli 8 byte dell'header, al massimo a tutto il datagram UDP, in quest'ultimo caso il protocollo UDP-Lite si comporta esattamente come il protocollo UDP.

Per soddisfare questa esigenza si è dovuto dare un'interpretazione diversa al campo length dell'header, che nel protocollo UDP-Lite assume il significato di **checksum coverage length**, ossia si deve specificare quanti byte del datagram UDP saranno controllati: solo 8 (header) o tutti.

Si noti che non si perde nessuna informazione circa la lunghezza del datagram UDP (che era il significato originario del campo length), in quanto questa può essere dedotta dalla lunghezza del pacchetto IP ricevuto a cui si devono sommare gli 8 byte dell'header UDP.

IN ENGLISH PLEASE

Network Working Group
Request for Comments: 3828
Category: Standards

L-A. Larzon
Lulea University of Technology
Track M. Degermark
S. Pink
The University of Arizona
L-E. Jonsson, Ed.
Ericsson
G. Fairhurst, Ed.
University of Aberdeen

July 2004

The Lightweight User Datagram Protocol (UDP-Lite)

Abstract

This document describes the Lightweight User Datagram Protocol (UDP-Lite), which is similar to the User Datagram Protocol (UDP) (RFC 768), but can also serve applications in error-prone network environments that prefer to have partially damaged payloads delivered rather than discarded. If this feature is not used, UDP-Lite is semantically identical to UDP.

FISSA LE CONOSCENZE

- Che cosa significa che UDP è un protocollo di tipo connectionless?
- Descrivi i campi del datagram UDP.
- Perché la lunghezza massima di un datagram UDP è di 65.508 byte?
- Qual è la differenza tra i due protocolli UDP e UDP-Lite?

4 UN PROTOCOLLO DI TRASPORTO CONNECTION-ORIENTED: TCP

4.1 TCP

Transmission Control Protocol è un protocollo di trasporto molto diffuso, più dell'UDP, in quanto offre un servizio **connection-oriented** e **affidabile**, garantendo quindi la consegna dei dati in modo ordinato (infatti si parla di *data stream* inteso come *flusso di byte*).

La FIGURA 12 mostra come i numeri di sequenza dei segmenti TCP siano usati per ricostruire il messaggio originale ponendo i segmenti nel giusto ordine.

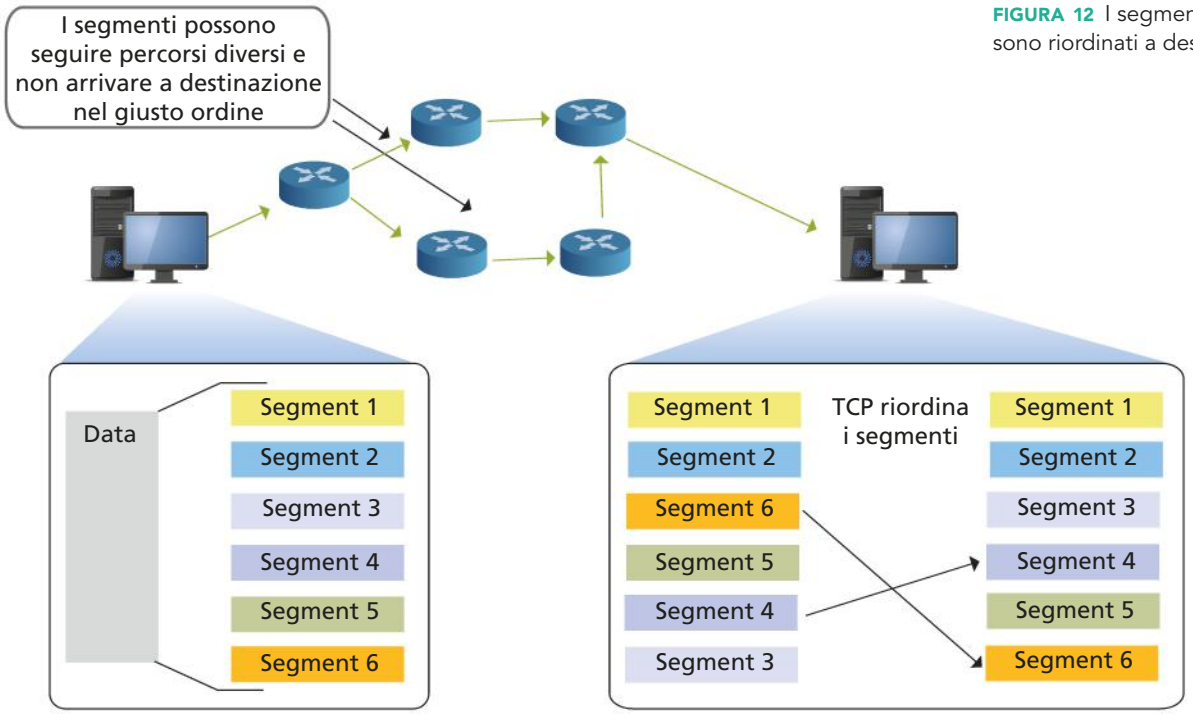


FIGURA 12 I segmenti TCP sono riordinati a destinazione

La connessione che TCP stabilisce tra mittente e destinatario offre all'applicazione, che si trova al livello superiore, l'impressione di usufruire di una linea dedicata. La connessione può, quindi, essere intesa come un canale logico le cui caratteristiche sono:

- **full-duplex**: sulla stessa connessione si può trasmettere e ricevere in contemporanea;
- **point-to-point**: un solo mittente e un solo destinatario;
- richiede l'inizializzazione di **variabili di stato** da parte del mittente e del ricevente (i due processi si devono accordare prima di iniziare il trasferimento dati).

TCP usa più risorse del computer host rispetto al protocollo UDP, sia in termini di CPU (l'elaborazione è piuttosto complessa), sia in termini di memoria (ci sono maggiori informazioni di stato da memorizzare). Inoltre necessita di maggiore capacità trasmissiva (banda) per via della ritrasmissione e dell'header più grande (20 byte rispetto agli 8 di UDP).

IN ENGLISH PLEASE

RFC: 793

TRANSMISSION CONTROL PROTOCOL

DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION

The Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in packet-switched computer communication networks, and in interconnected systems of such networks.

TCP si usa con applicazioni che richiedono una trasmissione affidabile dei dati, per esempio le applicazioni che utilizzano i protocolli FTP (trasferimento file), SMTP (posta elettronica) e HTTP (trasferimento pagine web) che verranno descritti nell'Unità 8 dedicata all'Application Layer.
Nel corso degli anni, sono stati definiti molti standard per TCP, tra questi: **RFC 793**, la prima specifica stabile; **RFC 7323**, che prende in considerazione il problema delle prestazioni (performance) definendo delle estensioni per migliorare l'efficienza; **RFC 5681**, che descrive gli algoritmi per il controllo della congestione della rete.

4.2 La comunicazione tra TCP e processo applicativo

Di regola i Sistemi Operativi permettono di accedere alle porte in modo sincrono, ciò implica che l'esecuzione del processo si interrompa durante un'operazione di accesso alla porta.

A volte i dati ricevuti vengono scartati perché:
- il processo di destinazione non è pronto a riceverli o il numero di porta di destinazione non esiste;
- non c'è spazio sufficiente nel buffer di ricezione per contenere tutti i dati.

#prendinota

Quando un processo accede in lettura alla porta per estrarne i dati, il Sistema Operativo lo mette in attesa (stato di *waiting* o *blocked*) finché non arrivano i dati. Quando i dati sono disponibili, il processo può ritornare attivo (stato di *ready*) e riceverli.

Un processo, che vuole inviare un messaggio, scrive il testo da spedire in un buffer nel suo spazio in memoria, inserisce le informazioni di controllo (header) e passa il controllo a TCP. Analogamente, un processo che deve ricevere un messaggio, definisce un buffer di ricezione nel suo spazio di memoria e passa il controllo a TCP (FIGURA 13).

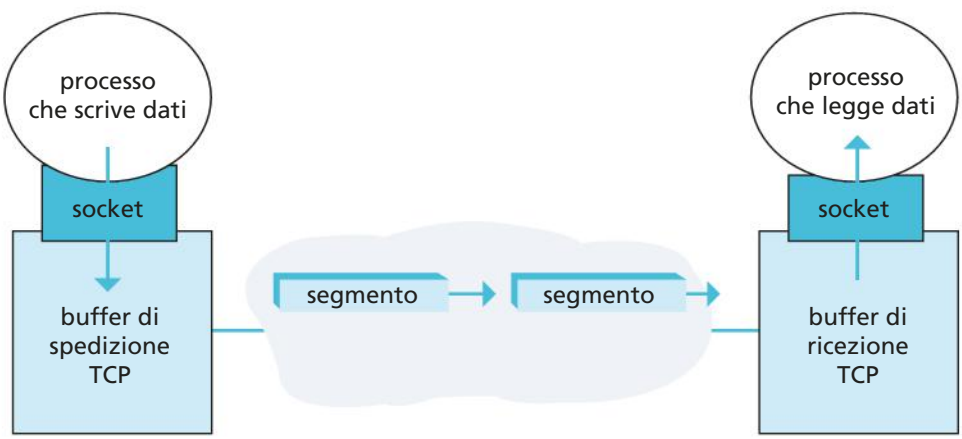


FIGURA 13 I segmenti TCP sono riordinati a destinazione

Le informazioni di controllo, che il processo deve passare al TCP, comprendono:
- **source address**: indirizzo completo del mittente (network + host + port);
- **destination address**: indirizzo completo del destinatario (network + host + port);
- **next packet sequence number**: il numero di sequenza che TCP deve assegnare al prossimo pacchetto che trasmetterà da quella porta;
- **current buffer size**: la dimensione del buffer del mittente;
- **next write position**: indirizzo dell'area del buffer in cui il processo pone i nuovi dati da trasmettere;
- **next read position**: indirizzo dell'area del buffer da cui TCP deve leggere i dati per costruire il prossimo segmento da inviare;

- **timeout/flag**: indica il tempo dopo il quale i dati non riscontrati (unacknowledged) devono essere ritrasmessi; il flag è usato per sincronizzare TCP e processo (per esempio tramite l'uso di semafori), per segnalazioni di stato, ecc.

Quando un'applicazione passa dei dati a TCP, questi può salvarli in un **buffer** oppure spedirli subito; la bufferizzazione consente di migliorare l'efficienza della comunicazione: i dati verranno spediti quando si raggiunge una certa quantità (Figura 13).

Lo standard prevede due eccezioni alla bufferizzazione:

- l'applicazione richiede che i dati vengano spediti immediatamente, allora imposta a 1 il flag PUSH, nell'header TCP, per indicare a questi di non bufferizzare e di inviare subito i dati;
- l'applicazione imposta a 1 il flag URG dell'header TCP: i dati non vengono accumulati nel buffer e TCP trasmette immediatamente tutto ciò che riguarda quella connessione. In ricezione, quando arrivano i dati con il flag URG = 1, l'applicazione interrompe la sua attività per esaminare immediatamente i dati urgenti.

4.3 Il segment TCP

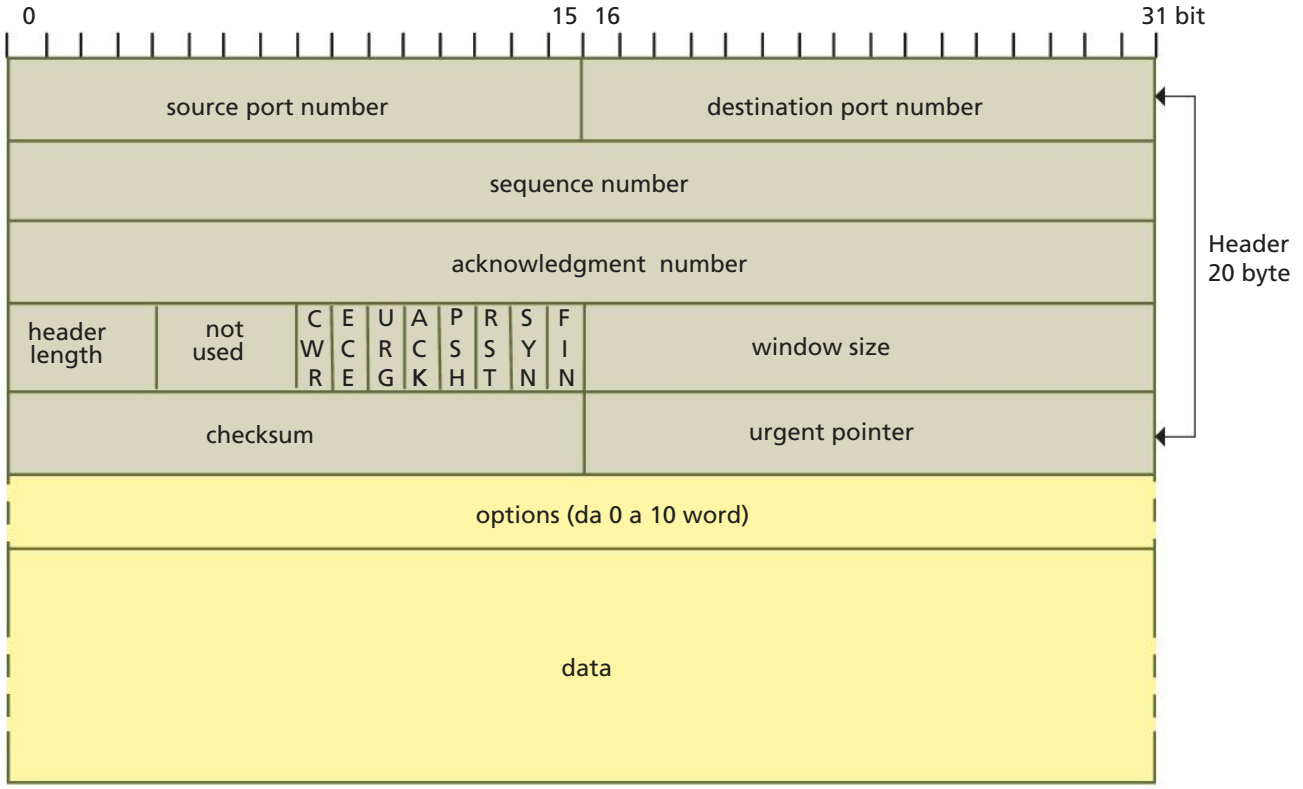


FIGURA 14 Segment TCP

I campi del segmento TCP (FIGURA 14) sono:

- **source port number** (16 bit): numero di porta sull'host del mittente;
- **destination port number** (16 bit): numero di porta sull'host del destinatario;
- **sequence number** (32 bit): numero di sequenza progressivo del primo byte di dati contenuto nel segmento;

- **acknowledgment number** (32 bit): numero di riscontro, ha significato solo se il flag ACK è impostato a 1, conferma la ricezione di una parte del flusso di dati indicando il valore del prossimo sequence number atteso (implicitamente si conferma che i byte precedenti sono stati ricevuti correttamente);
- **header length** (4 bit): indica la lunghezza (in word da 32 bit) dell'header del segmento TCP; tale lunghezza può variare da 5 word (20 byte) a 15 word (60 byte) a seconda della presenza e della dimensione del campo facoltativo Options. Serve quindi a indicare l'inizio dei dati del segmento;
- **not used** (4 bit): bit attualmente non utilizzati; devono essere impostati a zero;
- **flags** (8 bit): bit utilizzati per il controllo del protocollo:
  - **CWR** (Congestion Window Reduced) se impostato a 1 indica che l'host sorgente ha ricevuto un segmento TCP con il flag ECE impostato a 1 e ha di conseguenza abbassato la sua velocità di trasmissione per ridurre la congestione ed evitare la perdita di pacchetti;
  - **ECE** (ECN-Echo) se impostato a 1 indica che l'host supporta ECN (Explicit Congestion Notification) durante il Three-Way Handshake;
  - **URG** se impostato a 1 indica che nel flusso sono presenti *dati urgenti* e quindi si deve leggere il campo urgent pointer;
  - **ACK** se impostato a 1 indica che il segmento TCP in questione è in risposta a un altro ricevuto, che conteneva dati, di conseguenza indica che il campo acknowledgment number è valido e si devono leggere le informazioni in esso contenute;
  - **PSH** (push) se impostato a 1 indica che i dati in arrivo non devono essere bufferizzati, ma passati subito ai livelli superiori dell'applicazione;
  - **RST** (reset) se impostato a 1 indica che la connessione non è valida; viene utilizzato in caso di grave errore; a volte utilizzato insieme al flag ACK, per la chiusura di una connessione;
  - **SYN** (SYNchronize sequence numbers) è usato nella fase di instaurazione di una connessione; se impostato a 1 indica che l'host mittente del segmento vuole aprire una connessione TCP con l'host destinatario e specifica nel campo sequence number il valore dell'Initial Sequence Number (ISN), per sincronizzare i numeri di sequenza dei due host. L'host ricevente invia poi la risposta SYN-ACK con SYN = 1. Questo flag deve essere usato solo in questi due casi;
  - **FIN** (final) se impostato a 1 indica che l'host mittente del segmento non ha più dati da inviare e vuole *chiudere la connessione* TCP. Il ricevente invia la conferma di chiusura con un FIN-ACK. A questo punto la connessione è ritenuta chiusa in un verso: l'host che ha inviato il FIN non potrà più inviare dati, mentre l'altro host ha il canale di comunicazione ancora disponibile. Quando anche l'altro host invierà il pacchetto con FIN = 1 la connessione, dopo il relativo FIN-ACK, sarà considerata completamente chiusa;
- **window size** (16 bit): è usato dall'host ricevente per dire al mittente quanti dati può ricevere in quel momento (*finestra di ricezione*), cioè il numero di byte che il mittente può spedire a partire dal byte confermato (specificato dall'acknowledgment number). Il valore 0 indica di non inviare altri dati per il momento; quando il ricevente sarà di nuovo in grado di ricevere dati invierà al mittente un segmento con window size diverso da 0 ma con lo stesso acknowledgment number;
- **checksum** (16 bit): è utilizzato per la verifica della validità del segmento. L'algoritmo di calcolo è definito nell'RFC ed è del tutto simile a quello di UDP, l'unica

#prendinota

L'RFC 3168 del settembre 2001 ha introdotto l'uso del flag ECE nell'header del protocollo TCP e IP al fine di rilevare la congestione in rete, prima che si verifichino perdite di pacchetti. Infatti chi riceve pacchetti o segmenti con ECE impostato a 1, riduce la propria velocità di trasmissione, evitando la perdita di pacchetti dovuti alla congestione. Il rilevamento della congestione, prima che si verifichino perdite, consente di aumentare la velocità effettiva generale (infatti gli host TCP non devono ritrasmettere i pacchetti persi).

differenza è che per TCP il campo è obbligatorio. Se dal calcolo della checksum risultasse che il segmento TCP è stato danneggiato, non verrebbe riscontrata la sua ricezione e quindi il mittente lo dovrebbe ritrasmettere;

- **urgent pointer** (16 bit): ha significato solo se il flag URG = 1 contiene il numero che deve essere sommato (offset) al sequence number per ottenere il numero dell'ultimo byte urgente nel campo dati del segmento. Questo campo consente all'applicazione di usare messaggi che interrompono la normale elaborazione;
- **options**: campo facoltativo che può avere dimensione variabile da 0 a 10 word (word = 32 bit). L'opzione più importante è quella che consente a un host di specificare la dimensione massima del segmento che è in grado di accettare. Il default è 536 byte di dati e 20 byte di header (quindi ogni host deve poter gestire segmenti di almeno 556 byte);
- **data**: contiene i dati da trasmettere provenienti dal livello superiore o, nell'altra direzione, i dati ricevuti dal livello inferiore.

#prendinota

La dimensione massima di un segmento TCP è 65.495 byte. Infatti, deve essere contenuto in un pacchetto IP che ha al massimo 64 KB ($2^{16} - 1 = 65.535$ byte) ai quali si devono togliere i 20 byte minimi dell'header IP, che porta ad avere un campo dati IP al massimo di 65.516 byte ai quali si devono ancora togliere i 20 byte minimi dell'header TCP.

IN ENGLISH PLEASE

Options: TCP includes a generic mechanism for including one or more sets of optional data in a TCP segment. Each of the options can be either one byte in length or variable in length. The first byte is the *Option-Kind* subfield, and its value specifies the type of option, which in turn indicates whether the option is just a single byte or multiple bytes. Options that are many bytes consist of three fields:

Subfield Name	Size (bytes)	Description
Option-Kind	1	Specifies the option type.
Option-Length	1	The length of the entire option in bytes, including the *Option-Kind* and the *Option-Length* fields.
Option-Data	Variable	The option data itself. In at least one oddball case, this field is omitted (making *Option-Length* equal to 2).

FISSA LE CONOSCENZE

- Che tipo di servizio offre il protocollo TCP?
- Spiega come avviene la comunicazione tra TCP e applicazione.
- Descrivi i primi 4 campi dell'header TCP.
- Descrivi i flag dell'header TCP.

5 LA GESTIONE DELLA CONGESTIONE

SLOW START E CONGESTION AVOIDANCE

Come detto in precedenza l'architettura di rete TCP/IP adotta il modello di comportamento Best Effort: la rete fa del suo meglio per consegnare i pacchetti e non rifiuta mai nuovi utenti (come avviene invece nella tradizionale rete telefonica). La conseguenza di questo comportamento è che possono verificarsi delle congestioni in rete: se la coda di ricezione del router è piena e non è più in grado di accettare nuovi pacchetti, questi verranno scartati.
Prima di gestire la congestione è necessario rilevarla e a questo scopo TCP utilizza dei **timer** per misurare il tempo intercorso tra l'invio di un segmento e la ricezione del relativo ACK. Se quest'ultimo non arriva entro un dato tempo, si genera un timeout.

I problemi che possono aver causato la perdita dei dati (o dell'ACK) potrebbero avere varie origini, per esempio una tratta radio con elevato tasso di errore, ma TCP è nato per essere usato su reti fisse, ipotizza che la causa della perdita sia la congestione e agisce di conseguenza. Questa ipotesi, più che plausibile dal momento che le moderne reti hanno una bassissima probabilità di errori di trasmissione, è comune a molte versioni di TCP.
Negli ultimi anni sono nate nuove versioni di TCP, che tengono conto anche della possibilità che la perdita dei dati possa essere causata da errori di trasmissione, quindi non intervengono diminuendo la quantità di dati inviati, ma cercano di mantenere alte le prestazioni.

La funzione TCP di gestione della congestione è particolarmente critica proprio perché si basa su deduzioni che avvengono agli end system (si ricordi che TCP è un **protocollo end-to-end**) e non su dati precisi prelevati in rete. Non ci sono quindi garanzie che queste deduzioni siano sempre esatte.
Negli anni sono stati proposti diversi algoritmi che si adattano alle varie situazioni ed effettuano deduzioni a partire da varie informazioni. Per esempio c'è chi deduce la perdita di un segmento dallo scadere di un timer, altri al timeout aggiungono anche la ricezione di ACK duplicati e così via.
TCP lavora applicando congiuntamente diversi algoritmi e configurandone i parametri da usare. Esistono perciò diverse implementazioni del protocollo, che differiscono in base alle opzioni scelte.
L'RFC 5681 specifica 4 algoritmi che cooperano tra loro per realizzare il controllo della congestione.

IN ENGLISH PLEASE

RFC 5681
This document defines TCP's four intertwined congestion control algorithms: **slow start, congestion avoidance, fast retransmit** and **fast recovery**.

Tutti questi algoritmi fanno uso di una nuova variabile: la **finestra di congestione** che viene utilizzata dal mittente per ogni connessione attiva e serve per avere un'indicazione sul massimo numero di byte non riscontrati che si possono trovare ancora nella rete.

In particolare vale quanto segue:

maxWindow = **min**(FinestraDiCongestione, FinestraDiRicezione)

dove:

- **FinestraDiCongestione** (**congestion window**): è il massimo numero di byte che la rete è in grado di trasmettere senza che si verifichino dei timeout che segnalano la perdita di dati;

- **FinestraDiRicezione** (**advertised window**): indica quanti byte il destinatario è in grado di ricevere;
- **maxWindow**: quando un host deve inviare dei dati prenderà come numero massimo di byte che può trasmettere il minimo tra i due valori delle finestre.

L'idea alla base di questi algoritmi è diminuire la finestra di congestione quando un pacchetto è scartato dalla rete e aumentarla quando un pacchetto è riscontrato. Infatti se si sono persi pacchetti a causa della congestione, essi dovranno essere ritrasmessi, aumentando così il traffico in una rete che ne ha già troppo. Quindi perché la rete non collassi è importante che il mittente sia drastico nel ridurre il traffico che genera e cauto nell'aumentarlo.

Dei 4 algoritmi descritti nelle specifiche, nel seguito si prendono in esame **slow start** e **congestion avoidance**, che vengono usati solitamente insieme in molte implementazioni di TCP, prima fra tutte quella di Berkeley da cui deriva il nome *TCP Berkeley* dato alle versioni che implementano i due algoritmi.
Si inzia a trasmettere lentamente, *esplorando* la rete, per poi accelerare, prima in modo esponenziale, poi lineare, finché non c'è una perdita di dati che comporta un rallentamento nell'invio di nuovi byte.

I punti seguenti spiegano come lavorano i due algoritmi, iniziando con **slow start**:

- al momento della creazione della connessione TCP tra due host, il mittente imposta la finestra di congestione alla massima quantità di byte che la rete può spedire;
- ogni volta che viene inviato un segmento TCP, e ricevuto l'ACK di conferma ricezione da parte del destinatario, il valore della finestra di congestione viene raddoppiato. Avviene in questo modo una crescita esponenziale del numero di byte che possono essere inviati in rete, fino al raggiungimento di una certa soglia (**threshold**) che viene consigliato di impostare inizialmente a 64 KB;
- da questo punto in poi la dimensione della finestra di congestione è regolata dall'algoritmo **congestion avoidance** che effettua incrementi di tipo lineare, infatti viene di volta in volta sommato il valore assegnato inizialmente alla finestra di congestione;
- quando si genera un timeout (che per questa versione di TCP significa congestione), si effettuano le seguenti operazioni:
  - la soglia viene impostata alla metà del valore della finestra di congestione che ha generato il timeout;
  - la finestra di congestione viene riportata al suo valore iniziale.

esempio

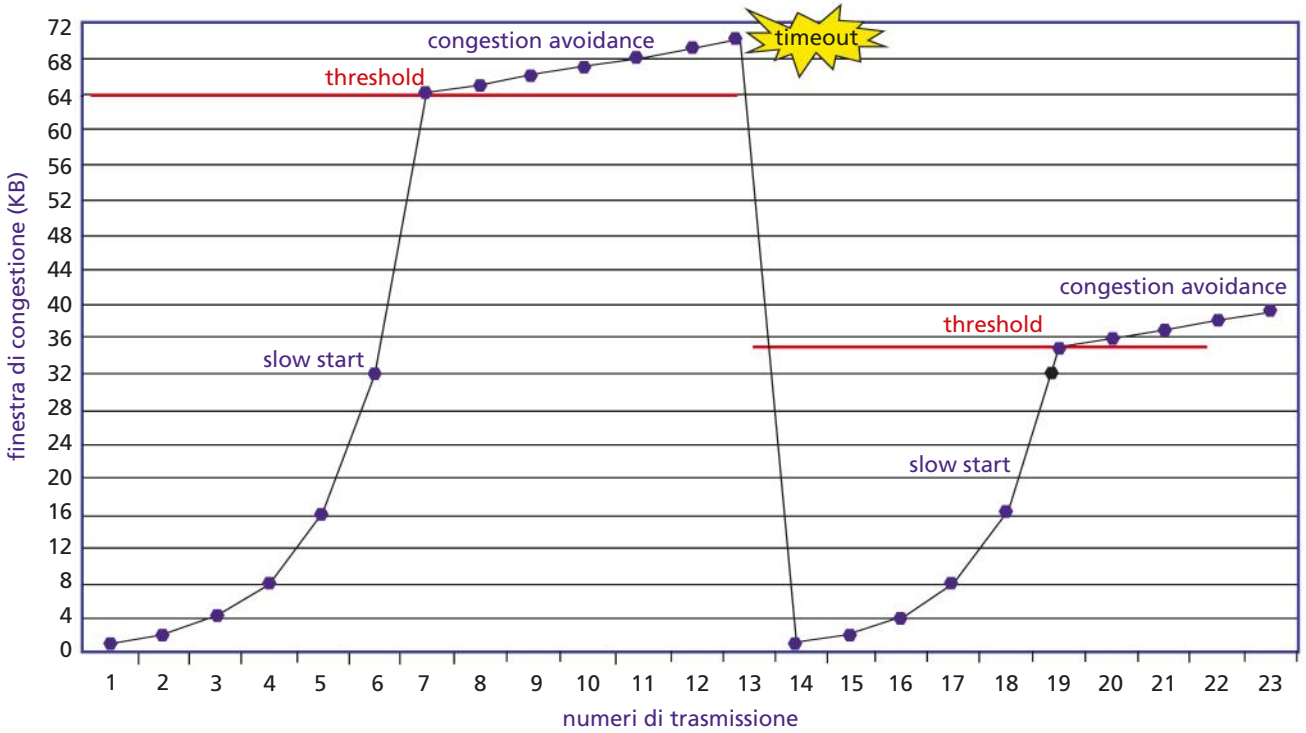
Si consideri la seguente situazione:

- all'inizio della trasmissione si imposta la finestra di congestione alla dimensione del segmento pari a 1 KB;
- si usa l'algoritmo *slow start* e ogni volta che arriva un ACK di conferma il valore è raddoppiato: 2 KB, 4 KB, 8 KB, 16 KB, 32 KB, 64 KB, con crescita esponenziale;
- la prima soglia è impostata a 64 KB, da lì in poi procede sommando 1 KB (cioè il valore iniziale della finestra di congestione) applicando l'algoritmo *congestion avoidance*;

- quando arriva a 70 KB si ipotizza un timeout (è scaduto il timer impostato per la ricezione dell'ACK), quindi si suppone che si sia verificata una situazione di congestione della rete e:
  – il valore della finestra di congestione ritorna quello iniziale (cioè 1 KB);
  – si dimezza il valore della soglia che diventa pari a 35 KB;
  – la trasmissione riprende applicando nuovamente l'algoritmo *slow start*.

Nella **FIGURA 15** è mostrato il grafico dell'andamento della dimensione della finestra di congestione che si viene a formare con il nostro esempio.

FIGURA 15 Slow start e congestion avoidance



FISSA LE CONOSCENZE

- Come TCP effettua il controllo della congestione di rete?
- Che cos'è la threshold?
- Come si può intervenire per evitare il collasso della rete?
- Cosa contiene maxWindow?
- Quando si passa dallo slow start al congestion avoidance?
- Quando si passa dal congestion avoidance allo slow start?

6 L'HANDSHAKING TCP

6.1 Instaurazione della sessione TCP: Three-Way Handshake

La comunicazione tra host mittente (tipicamente un client) e host destinatario (tipicamente un server) a livello TCP è di tipo connection-oriented, quindi sono previste 3 fasi:

1. **instaurazione della sessione TCP (Three-Way Handshake)**: è la fase in cui avviene il colloquio iniziale tra client e server, in cui si scambiano i dati di impostazione, utili al trasferimento successivo delle informazioni;
2. **trasmissione dati**: è la fase in cui avviene la trasmissione delle informazioni vere e proprie. Poiché TCP offre un servizio affidabile (reliable) si attuano meccanismi per il **controllo degli errori**, per il **controllo di flusso** e per il **controllo della congestione**, al fine di prevenire la perdita dei dati trasmessi, mantenere la giusta sequenza ed eliminare eventuali duplicati;
3. **abbattimento della sessione TCP (Double Two-Way Handshake)**: questa fase è anche detta *disconnessione*, il protocollo prevede più modalità per terminare la comunicazione tra client e server. Principalmente si ha il Three-Way Handshake se la chiusura è contemporanea, il Double Two-Way Handshake se la chiusura avviene in tempi diversi.

Prima di vedere le 3 fasi ricordiamo che il dialogo avviene attraverso le socket con l'utilizzo delle primitive.

La TABELLA 2 mostra un tipico insieme di **primitive** usate per implementare i servizi del TCP. Mittente e destinatario comunicano mediante messaggi detti **TPDU** (Transport Protocol Data Unit), che nelle specifiche vengono anche chiamati **segmenti**.

TABELLA 2 Insieme di primitive e relative TPDU

Primitive	TPDU inviata	Descrizione
accept()	nessuna	blocca finché non arrivano le richieste di connessione da parte dei processi
connect()	Connection Request	prova a stabilire una connessione
send()	Data	invio di informazioni
receive()	nessuna	blocca finché non si riceve una Data TPDU
disconnect()	Disconnection Request	rilascio della connessione da un lato

Affinché si possa instaurare una sessione TCP tra host 1 e host 2 occorre che quest'ultimo acconsenta mediante una sequenza che prevede tre passi e che viene chiamata Three-Way Handshake (stretta di mano a tre vie) illustrata in FIGURA 16.

1. L'host 1 invia un segmento TCP con il flag SYN impostato a **1**. Invia inoltre un numero di sequenza, scelto in modo casuale, che diventa il suo sequence number (**X**). Primitiva usata: connect().
2. L'host 2, se acconsente a stabilire la connessione, risponde con una conferma mediante il flag ACK.F impostato a **1** e l'acknowledgment number impostato al valore ricevuto del sequence number +1 (**X + 1**).

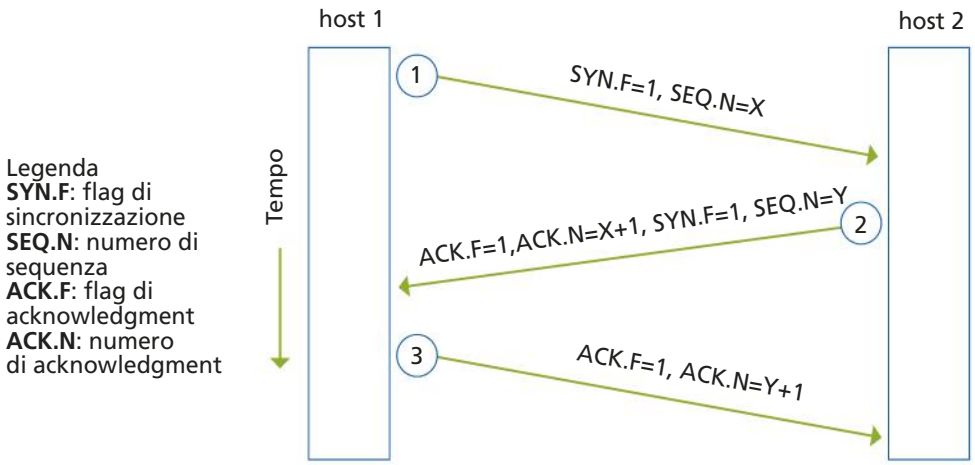


FIGURA 16 Three-Way Handshake

Inoltre, per stabilire la connessione nella direzione inversa (full-duplex), l'host 2 imposta il proprio flag SYN.F a **1** e genera un suo numero di sequenza (**Y**), scelto in modo casuale, da inviare all'host 1. Primitiva usata: send().

3. L'host 1 risponde con un'ulteriore conferma mediante il flag ACK.F impostato a **1** e l'acknowledgment number impostato al valore, ricevuto dall'host 2, del sequence number incrementato di 1 (**Y + 1**). Primitiva usata: send().

Se sull'host 2 non c'è nessun processo in ascolto sulla porta specificata nel campo destination port number, il passo **2** non viene eseguito e l'host 2 invia un segmento di risposta con flag RST = 1 per rifiutare la connessione.
Nella fase di setup è quindi importante che ogni host conosca il sequence number iniziale dell'altro. Altra informazione che si scambiano è la dimensione massima del segmento (**MSS** = Maximum Segment Size) che ogni host invierà all'altro; verrà scelta la dimensione minore e questo dato sarà particolarmente utile per il controllo della congestione. Infine si scambiano la dimensione della finestra TCP (windows size) che fornisce indicazioni sulla dimensione del buffer utilizzato per memorizzare i segmenti ricevuti.

MTU = Maximum Transfer Unit, è la dimensione massima del campo dati nel frame a livello Data Link; è un valore che caratterizza la rete di trasmissione utilizzata (per esempio per le reti Ethernet MTU = 1.500 byte).
Ogni volta che IP deve inviare un pacchetto più grande della MTU è costretto a frammentare. TCP tiene conto di questo e, per avere maggiori prestazioni, fa coincidere la dimensione massima del segmento con la MTU.

MRU = Maximum Receive Unit, è la MTU del destinatario.

MSS = Maximum Segment Size, è la dimensione massima che possono avere i segmenti che si scambiano i due host, mittente e ricevente, ottenuta come il minimo tra i due valori MTU e MRU meno i 20 byte dell'header IP:

$$\mathbf{MSS = min(MTU, MRU) - 20\ byte}$$

In caso di mancanza di informazioni (dipende dalle implementazioni del TCP) viene utilizzato come valore di default 536 byte ottenuto nel seguente modo:

576 byte (default pacchetto IP) – 20 byte (header IP) – 20 byte (header TCP) =
= 536 byte (MSS).

6.2 Trasmissione dati

La seconda fase della comunicazione TCP consiste nella trasmissione dei dati.
TCP gestisce il **controllo di flusso** e gli **errori di trasmissione** (o perdita di pacchetti) con lo stesso meccanismo: il protocollo sliding window (a finestra scorrevole) affrontato nel volume del terzo anno.
Il protocollo utilizzato è quindi simile a quello visto per il controllo di flusso a livello Data Link. Ci sono, però, due differenze fondamentali:

- in TCP il puntatore nella finestra è al singolo byte, mentre a livello Data Link è al frame;
- in TCP la dimensione della finestra è variabile, mentre a livello Data Link è fissa.

Riprendiamo brevemente il problema che porta a utilizzare un meccanismo sliding windows e supponiamo che:

- il destinatario abbia un buffer di ricezione di 2.000 byte;
- il trasferimento dati tra mittente e destinatario avvenga correttamente: il processo applicativo invia dati al livello Transport che li trasmette in rete e i byte arrivano ordinatamente al livello Transport del destinatario che li riscontra (invio dell'ACK).

A un certo punto della trasmissione il processo destinatario non legge più i byte dal buffer di ricezione (perché si è bloccato oppure è impegnato in un'altra attività); che cosa succede allora quando il buffer è pieno e arriva un nuovo byte?
È necessario che, prima che si verifichi questa situazione, il ricevente possa comunicare al mittente di sospendere momentaneamente la trasmissione, finché non sarà di nuovo pronto ad accettare dati.

A livello Data Link la soluzione a questo problema è solitamente implementata nell'hardware della scheda di rete e si presuppone che i frame siano processati appena sono ricevuti.

In TCP invece la soluzione è a livello software, nel meccanismo di comunicazione tra TCP mittente e TCP destinatario:

- ogni volta che il ricevente invia un ACK indica, nel campo dell'header windows size, il numero di byte che può accettare in quel momento;
- il mittente non invia un numero di byte superiore a quello indicato nell'ultimo ACK ricevuto;
- il ricevente non può rifiutarsi di accettare il numero di byte che aveva indicato nella finestra.

Si consideri l'esempio precedente supponendo che il buffer di ricezione sia occupato per 1.000 byte, quindi il ricevente invia un ACK al mittente con l'indicazione che può accettare 1.000 byte. Il mittente allora invia 700 byte. Il ricevente invia un ACK per confermare la ricezione dei primi 500 byte, l'indicazione della dimensione della finestra non potrà essere inferiore a 200 byte, altrimenti vorrebbe dire che una parte dei 200 byte che sono ancora nella rete non sarà accettata.

In generale la dimensione della finestra viene scelta uguale allo spazio libero del buffer di ricezione del destinatario. Quando quest'ultimo non è in grado di ricevere altri byte perché il buffer è pieno, invia al mittente un messaggio di ACK con dimensione della finestra pari a zero.
Il mittente che lo riceve deve allora sospendere la trasmissione.

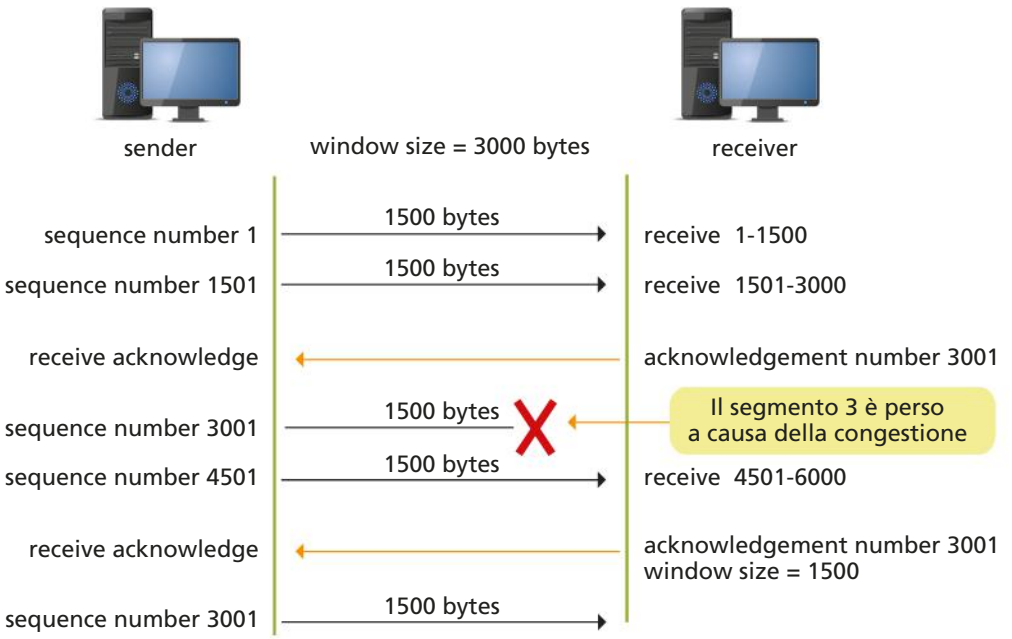
#prendinota

Grazie alla tecnica sliding window TCP è in grado di:

- garantire la consegna affidabile dei dati;
- assicurare che i dati sono consegnati nella giusta sequenza;
- attuare un controllo del flusso dei dati tra mittente e destinatario: chi invia i dati non deve superare la capacità del buffer del ricevente.

La conferma negativa (NACK = Not ACKnowledged, cioè non ricevuto) non esiste in TCP, quindi il protocollo prevede un meccanismo di timeout, che evita l'attesa infinita del riscontro al segmento inviato.
Tale meccanismo a finestre scorrevoli è il **Go-back-N** con timeout (FIGURA 17) che abbiamo visto nel volume di terza.
Allo scadere del timeout il segmento non riscontrato viene nuovamente inviato e sono ritrasmessi tutti i segmenti spediti successivamente, anche se alcuni di questi sono già stati ricevuti correttamente. Il mittente deve quindi mantenere in memoria i segmenti trasmessi, ma non ancora riscontrati.

FIGURA 17 Go-back-N



Con questa tecnica si risolve sia il problema di ricezione di **segmenti danneggiati** (il destinatario capisce che il segmento non è corretto dall'informazione contenuta nel campo checksum e lo scarta), sia quello di ricezione di **segmenti fuori sequenza** (out of order). Infatti in entrambi i casi il destinatario invia nuovamente un ACK per l'ultimo segmento ricevuto correttamente, cioè integro e nel giusto ordine, e quando il mittente riceve un ACK duplicato, lo interpreta come NACK per il segmento seguente, che viene quindi ritrasmesso insieme a tutti i successivi già inviati.

Anche la gestione di eventuali **segmenti duplicati** è risolta in modo semplice in TCP. Quando un mittente non riceve l'ACK prima dello scadere del timer, assume che il segmento non sia giunto a destinazione e lo invia nuovamente. Il destinatario che riceve un segmento con lo stesso numero di sequenza di uno già ricevuto si limita a scartarlo.

Infine, può accadere che un messaggio di ACK non giunga al mittente, ma questi potrebbe non venirne a conoscenza. Infatti in TCP il meccanismo del riscontro è cumulativo, ossia ogni riscontro conferma che il destinatario ha ricevuto correttamente tutti i dati trasmessi fino al byte specificato nel messaggio. Per esempio se il destinatario invia un messaggio con acknowledgment number = 1301, significa che tutti i byte fino al 1300 sono stati ricevuti correttamente, quindi il fatto di aver perso un riscontro con acknowledgment number = 1151 passerebbe del tutto inosservato al mittente.

Quanto sopra descritto è evidenziato nell'esempio di Figura 17 che mostra come nell'ambito di una connessione attiva tra un host mittente (sender) e un host destinatario (receiver), ci sia una stretta relazione tra la dimensione della finestra, la perdita di segmenti e la congestione.

6.3 Abbattimento della sessione TCP: Double Two-Way Handshake

In precedenza si è visto che una connessione TCP è bidirezionale, quindi può essere vista come composta da due flussi di dati indipendenti, uno per ciascuna direzione. Quando un programma applicativo non ha più dati da inviare, comunica al TCP di chiudere la connessione in una direzione e questi, dopo aver trasmesso eventuali dati ancora presenti nel suo buffer e ricevuto il relativo riscontro, inizia la procedura di abbattimento della connessione dal "suo" lato. Il TCP ricevente comunicherà al relativo programma applicativo che non riceverà più dati dall'altro utente. Se l'host che ha ricevuto l'informazione di chiusura della connessione ha ancora dei dati da trasmettere, questi continueranno a essere inviati e l'host che li riceve li riscontrerà (invio dell'ACK), anche se ha già chiuso la connessione dal suo lato. Terminato l'invio dei dati ancora da trasmettere, la connessione verrà chiusa anche nell'altra direzione. Questa doppia chiusura prende il nome di Double Two-Way Handshake (stretta di mano doppia a due vie) ed è mostrata in FIGURA 18.

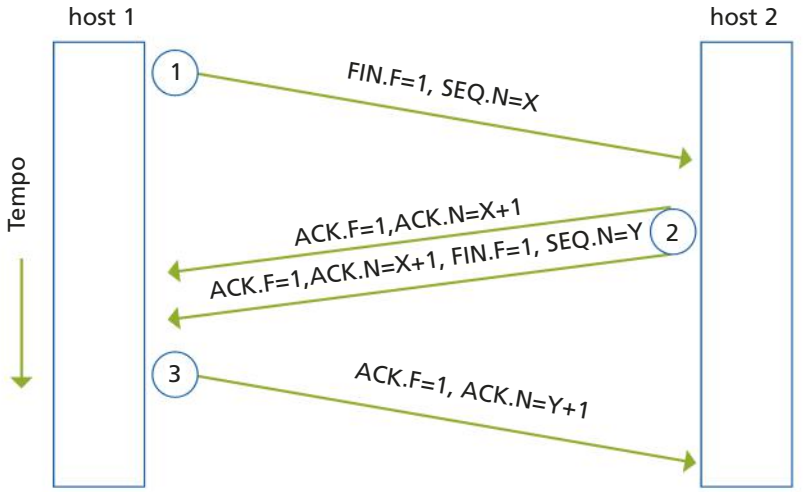


FIGURA 18 Double Two-Way Handshake

Legenda
FIN.F: flag di chiusura
SEQ.N: numero di sequenza
ACK.F: flag di aknowledgment
ACK.N: numero di aknowledgment

Rispetto alla fase di instaurazione si notano due particolarità:

- l'utilizzo del flag FIN (al posto del flag SYN) impostato a **1** dall'host 1 per comunicare l'intenzione di chiudere la sessione;
- la suddivisione in due tempi del passo 2 dovuta alla necessità di confermare immediatamente, tramite ACK.N = **X + 1**, da parte dell'host 2 il ricevimento del SEQ.N = **X**, **indipendentemente dalla richiesta di chiusura**. Mentre è generalmente necessario un po' di tempo per confermare il ricevimento del flag FIN = 1 dovendo essere prima informato il software applicativo dell'host 2 che la connessione è stata chiusa.

Se una connessione non può essere rilasciata secondo la procedura normale sopra descritta a causa, per esempio, di un'anomalia, TCP prevede una procedura di Reset:

si invia un segmento con il bit **RST** impostato a **1**, che comporta la chiusura immediata della connessione, senza ulteriori scambi di segmenti.

6.4 Vulnerabilità

Le connessioni TCP possono essere soggette a diversi tipi di attacchi informatici. Vediamo brevemente i principali.

SYN-FLOODING

È un tipo di attacco DoS (Denial of Service) relativo all'apertura della connessione con la tecnica Three-Way Handshake. Un host invia molte richieste di connessioni al server oggetto dell'attacco (letteralmente "lo inonda di segmenti SYN"). Il server risponde con altrettanti segmenti SYN-ACK a cui l'host mittente ovviamente non risponde con l'ACK. Il server si trova ad aver raggiunto il massimo numero di connessioni che è in grado di gestire e non accetta altre richieste di connessione, rimanendo in attesa degli ACK finali che non arrivano. Allo scadere di un timer il server abbatte le connessioni non completate, ma il fatto di ricevere continuamente delle richieste blocca comunque la sua normale attività.
A seguito della "messa fuori rete" dell'host oggetto del SYN-Flooding, l'attaccante può realizzare altri tipi di attacchi come IP Spoofing.

Contromisure: aumentare la dimensione della coda di connessione (SYN ACK queue) e diminuire il tempo di timeout per il completamento della fase di Three-Way Handshake.

SEQUENCE GUESSING

È un attacco che consiste nel riuscire a indovinare (guess) il numero di sequenza e nel generare, di conseguenza, dei segmenti TCP con mittente falsificato e formalmente corretti. Inoltre, si deve impedire la ricezione dei messaggi di risposta (ACK) da parte del vero mittente.

Contromisure: si configurano i router per non far entrare traffico che provenga dalla rete interna (ingress filtering).

SESSION HIJACKING

Questo attacco consiste nel riuscire a inserirsi in una sessione attiva, sostituendosi a uno dei due host:

- Z spia la connessione tra X e Y e registra i numeri di sequenza dei segmenti;
- Z blocca Y (per esempio con un SYN-Flooding);
- Z invia un segmento con il numero di sequenza corretto, con source Y, in modo che X non si accorga di nulla.

Contromisure: si configurano i router per non far entrare traffico che provenga dalla rete interna (ingress filtering).

Esercizio commentato
I segmenti TCP

FISSA LE CONOSCENZE

- Quali fasi hanno luogo in una comunicazione TCP tra due host?
- Come viene gestito il controllo di flusso nel TCP?
- Come si determina il valore MSS (Maximum Segment Size)?
- Quali vulnerabilità si possono individuare nel protocollo TCP?

7 CONFRONTO TRA I PROTOCOLLI UDP E TCP

Da quanto emerso nelle Lezioni precedenti è chiaro che TCP sia da preferire per il trasferimento dati in cui è importante l'affidabilità della comunicazione, UDP è invece preferibile quando le prestazioni sono più importanti del ricevere i dati in modo perfetto.

UDP si usa:
- su una rete affidabile oppure quando l'affidabilità non è importante, per esempio NFS (Network File System);
- quando l'applicazione mette tutti i dati in un singolo datagram, per esempio DNS (Domain Name System) o NTP (Network Time Protocol);
- quando non è importante che tutti i datagram arrivino a destinazione, ma è necessario non introdurre ritardi; questa è una tipica esigenza delle applicazioni multimediali, infatti i dati di queste applicazioni devono arrivare entro un tempo limite (un pacchetto in ritardo equivale a un pacchetto perso), inoltre non sarebbe accettabile il ritardo che si avrebbe con la ritrasmissione, tipica dei protocolli di tipo connection-oriented;
- quando eventuali meccanismi di ritrasmissione possono essere gestiti direttamente dall'applicazione, come per esempio nel protocollo di gestione SNMP (Simple Network Management Protocol).

TCP si usa:
- quando l'applicazione richiede una comunicazione affidabile, come per esempio la posta elettronica;
- quando è necessario garantire l'integrità dei dati, per esempio nel trasferimento di file o nell'interrogazione a un database;
- quando è necessario garantire che le richieste e le risposte arrivino a destinazione, per esempio la richiesta di una pagina web;
- quando è necessario mantenere il controllo costante della comunicazione (ossia se ne gestisce lo "stato").

La TABELLA 3 presenta le caratteristiche più importanti di un protocollo di livello Transport e ne indica il supporto (sì/no) da parte di TCP e UDP.

TABELLA 3 Confronto tra TCP e UDP

Caratteristiche	TCP	UDP
dimensione header	20-60 byte	8 byte
messaggio trasmesso in rete	segment	datagram
checksum	sì	opzionale
dimensione della checksum (bit)	16	16
connection-oriented	sì	no
full-duplex	sì	sì
trasferimento dati affidabile	sì	no
consegna ordinata dei messaggi	sì	no
controllo di flusso	sì	no

Caratteristiche	TCP	UDP
controllo della congestione	sì	no
supporto ECN	sì	no
path MTU discovery (bit)	16	no
frammentazione del messaggio inviato dall'applicazione in segmenti da inviare in rete	sì	no
assemblaggio dei dati dei segmenti per ricostruire il messaggio originale	sì	no
consentita la chiusura a metà della connessione	sì	non applicabile

Nella Tabella 3 si fa riferimento alla caratteristica **path MTU discovery** (letteralmente "la scoperta dell'MTU del percorso"). Si tratta della possibilità di inviare pacchetti di dimensione tale da non essere frammentati (tale valore è chiamato PMTU = Path MTU). Per scoprire qual è la PMTU, un host inizia a inviare pacchetti con dimensione pari al minore tra i valori di MTU e MSS, questi pacchetti hanno il bit DF (*Don't Fragment*) impostato a 1. Se lungo il cammino si trova un router con MTU più piccola, questi invia al mittente un messaggio ICMP che indica che il pacchetto è troppo grande e non può frammentarlo. Il mittente viene a conoscenza del valore di MTU da usare per quel percorso fino a quel router. L'operazione è quindi ripetuta finché il pacchetto non arriva a destinazione. In questo modo il mittente conosce il valore di MTU per quel cammino.

IN ENGLISH PLEASE

Hello, would you like to hear some funny jokes about TCP and UDP?

"Hi, would you like to hear a UDP joke?."

"Yes, I would like to hear a UDP joke."

"To get to the other side."

...

"Why did the chicken cross the road?"

"Hi, I'd like to hear a TCP joke."
"Hello, would you like to hear a TCP joke?"
"Yes, I'd like to hear a TCP joke."
"OK, I'll tell you a TCP joke."
"Ok, I will hear a TCP joke."
"Are you ready to hear a TCP joke?"
"Yes, I am ready to hear a TCP joke."
"Ok, I am about to send the TCP joke. It will last 10 seconds, it has two characters, it does not have a setting, it ends with a punchline."
"Ok, I am ready to get your TCP joke that will last 10 seconds, has two characters, does not have an explicit setting, and ends with a punchline."
"I'm sorry, your connection has timed out. ... Hello, would you like to hear a TCP joke?"

FISSA LE CONOSCENZE

- Quale protocollo tra UDP e TCP consente di realizzare un collegamento affidabile tra source e destination?
- Quale protocollo tra UDP e TCP effettua la frammentazione dei messaggi che provengono dalle applicazioni?
- Elenca almeno due casi in cui si usa UDP.
- Elenca almeno due casi in cui si usa TCP.

8 IL CONTROLLO DELLE PORTE

8.1 Il comando netstat

Il comando **netstat** permette di determinare le porte aperte da connessioni attive o da eventuali server presenti sull'host locale, consentendo di controllare le connessioni e di rilevare eventuali problemi.
Questo comando può essere eseguito nella finestra del Prompt dei comandi di Windows, ma anche i sistemi Unix/Linux lo utilizzano.

Sintassi:

netstat [-a] [-e] [-n] [-s] [-p proto] [-r] [interval]

-a	visualizza tutte le connessioni e le porte in ascolto (listening).
-e	visualizza le statistiche delle connessioni Ethernet. Può essere usata insieme all'opzione -s.
-n	visualizza gli indirizzi e i numeri di porta in forma numerica.
-p proto	visualizza le connessioni per il protocollo specificato in proto (tipicamente TCP o UDP). Se usato con l'opzione –s per visualizzare le statistiche del protocollo, proto può essere TCP, UDP o IP.
-r	visualizza la routing table.
-s	visualizza le statistiche per protocollo. Di default i dati sono mostrati per TCP, UDP e IP; l'opzione -p può essere usata per specificare i protocolli.
interval	introduce un intervallo di tempo tra visualizzazioni successive delle statistiche di rete. Per fermare questa ripetizione continua è sufficiente digitare CTRL+C.

esempio

La FIGURA 19 e la FIGURA 20 mostrano il risultato dell'applicazione di due opzioni del comando Netstat.
netstat -n visualizza tutte le connessioni attive (*established*) o in fase di chiusura. L'opzione -n serve a visualizzare il risultato in forma numerica e a evitare la risoluzione degli indirizzi IP nei rispettivi nomi.

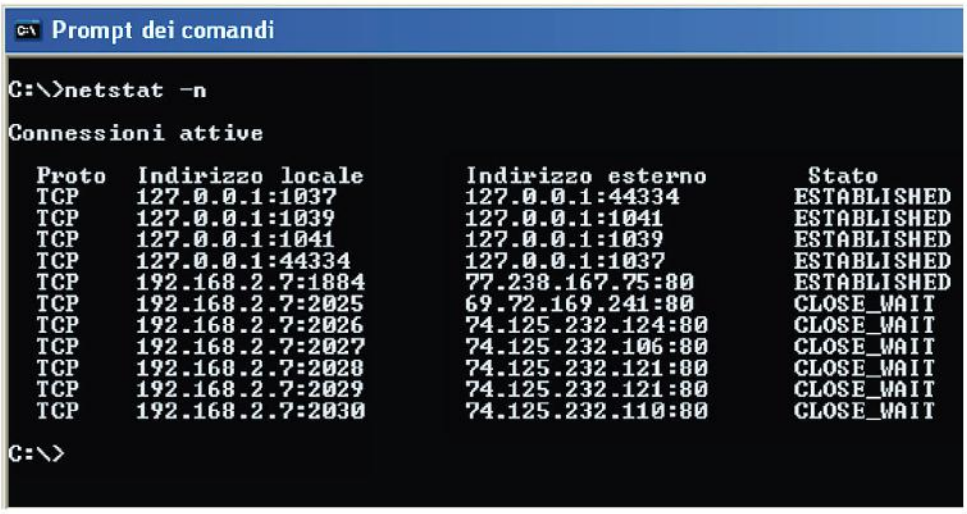


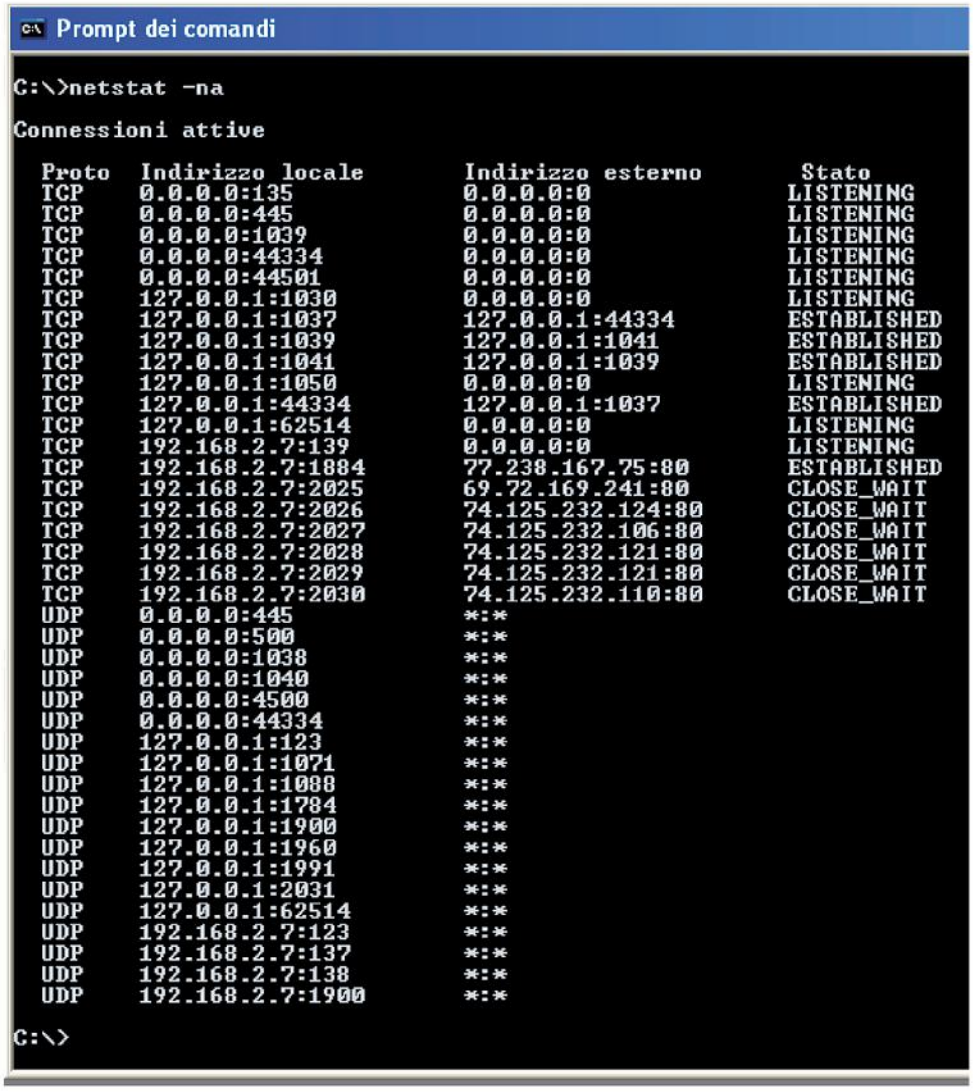
FIGURA 19 Il comando netstat -n.

netstat -na aggiunge alle precedenti informazioni anche quelle sulle porte in ascolto (*listening*) permettendo quindi di individuare tutti i server in esecuzione sul computer in esame.

Lo stato delle porte può essere:

- LISTENING indica quali porte sul computer locale (indirizzo locale) sono "in ascolto";
- ESTABLISHED indica quali porte sul computer remoto (indirizzo esterno) sono connesse;
- CLOSE_WAIT indica che la chiusura non è stata completata, il server ha ricevuto solo il primo FIN dal client.

FIGURA 20 Il comando netstat -na.



Nella Figura 20 si può notare che:

- ogni connessione TCP in uscita causa la creazione di una entry LISTENING sulla stessa porta;
- le porte UDP LISTENING sono il duplicato di una porta TCP LISTENING (si possono quindi ignorare);
- dove compare 0.0.0.0 nella colonna *Indirizzo locale* significa che la porta è in ascolto (LISTENING) su **tutte** le interfacce di rete (per esempio la scheda di rete, il modem, ecc.);

- dove compare 127.0.0.0 nella colonna *Indirizzo locale* significa che la porta è in ascolto (LISTENING) **solo** per connessioni che arrivano dal PC stesso (locale), non da Internet;
- dove compare 77.238.167.75:80 nella colonna *Indirizzo esterno* indica una connessione Internet a un web server (porta 80), attualmente attiva (ESTABLISHED).

Tutte le volte che l'host, o la rete locale, non è direttamente esposta su Internet ma è separata da questa da un firewall (software o hardware), un router o un generico gateway, è utile testare anche la sicurezza dell'host o della rete dall'esterno al fine di verificare quali porte siano aperte o chiuse (in questo caso un'analisi locale avrebbe poco senso).

Non sempre si ha la possibilità di eseguire personalmente questo test da un computer esterno alla propria rete e per questo motivo molti siti, che si interessano di sicurezza, hanno messo a disposizione questo servizio direttamente dal loro sito web.

8.2 Port scanner: Nmap

Nmap (Network Mapper) è un software open source (FIGURA 21) creato per effettuare **port scanning**, utilizzabile sia in ambiente Windows che Unix/Linux. L'obiettivo principale di questo tool è individuare porte aperte su un computer in modo da stabilire quali servizi di rete sono attivi. Si tratta di uno strumento molto utile per avere informazioni sui sistemi e può essere di aiuto a un amministratore per verificare la presenza di eventuali applicazioni server non autorizzate.

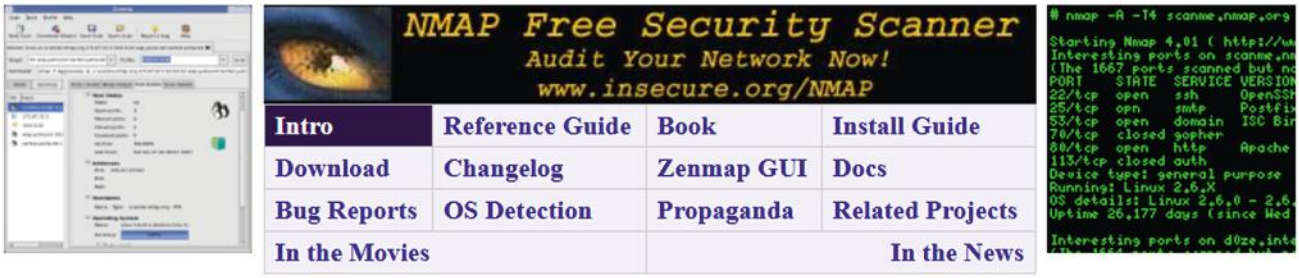


FIGURA 21 *https://nmap.org*

I risultati sono visualizzati all'interno di una tabella delle porte in cui vengono specificati:

- il numero della porta;
- il protocollo;
- il nome e lo stato del servizio.

L'elemento da esaminare con attenzione è lo stato, Nmap ne riconosce 6 tipi diversi:

- **open**: la porta può accettare connessioni TCP o datagram UDP;
- **filtered**: nessuna risposta dalla porta, questo stato forza Nmap a riprovare più volte l'interrogazione della porta;
- **closed**: l'host è accessibile, ma non c'è alcuna applicazione in ascolto su quella porta;
- **unfiltered**: la porta è accessibile ma non si conosce se è open o close;
- **open|filtered**: Nmap non è in grado di stabilire se la porta è open o filtered;
- **close|filtered**: Nmap non è in grado di stabilire se la porta è close o filtered.

La sintassi base del programma è la seguente:

```
nmap –[tipo di scan] –[opzioni]  <host>
```

- **host**: il nome o l'indirizzo IP dell'host su cui si vuol eseguire lo scanning delle porte;
- **tipo di scan**: quando si manda in esecuzione il programma Nmap è necessario indicare quale tecnica di scanning si vuol usare. Infatti Nmap può effettuare diversi tipi di scanning, quali:
  - **sS** (TCP SYN scan): è il tipo di default, prevede che la richiesta di apertura della connessione TCP non venga mai completata (half-open scanning); infatti appena si identifica una porta aperta, la fase di TCP handshake viene annullata, in questo modo non rimane traccia sul file di log delle connessioni sull'host remoto;
  - **sT** (TCP connect() scan): viene eseguita completamente la fase di apertura della connessione TCP (Three-Way Handshake), la connessione sarà quindi registrata sul file di log delle connessioni sull'host remoto;
  - **sU** (UDP scan): si invia alle porte un datagram UDP senza la parte dati, tipicamente si usa per testare servizi come DNS, DHCP, SNMP che usano le porte UDP; dopo l'invio si può avere
    - un datagram UDP di risposta: la porta è **open**
    - nessuna risposta: la porta è **open|filtered**
    - un messaggio ICMP con l'indicazione "port unreachable": la porta è **close**
    - un messaggio ICMP con codice 1, 2, 9, 10 o 13: la porta è **filtered**
  - **sV** (Version scan): fornisce il numero di versione del servizio in esecuzione sulla porta, tale numero è utile se si vuol avere informazioni sul servizio;
- **opzioni**: tra le varie opzioni disponibili troviamo:
  - **O** fornisce informazioni sul Sistema Operativo installato sull'host;
  - **T** imposta un timer, più il valore è alto e più è veloce lo scanning; di default il tempo è fornito in secondi;
  - **A** consente di effettuare più operazioni insieme: avere informazioni sul Sistema Operativo dell'host e sul servizio relativo alla porta, eseguire il comando traceroute, eseguire script;
  - **p** permette di indicare una specifica porta dell'host su cui effettuare lo scanning.

esempio

In FIGURA 22 vediamo un esempio di utilizzo del comando nmap.

FIGURA 22 Esecuzione del comando nmap

```
# nmap -sS -O t
   Starting nmap V. 2.53
   Interesting ports on t (192.168.1.1):
   Port       State       Service
   21/tcp     open        ftp
   22/tcp     open        ssh
   23/tcp     open        telnet
   25/tcp     open        smtp
   37/tcp     open        time
   53/tcp     open        domain
   111/tcp    open        sunrpc
   113/tcp    open        auth
   135/tcp    open        loc-srv
   139/tcp    open        netbios-ssn
   515/tcp    open        printer
   849/tcp    open        unknown
   853/tcp    open        unknown
   7000/tcp   open        afs3-fileserver
   TCP Sequence Prediction: Class=64K rule
                            Difficulty=1 (Trivial joke)
   Remote operating system guess: HP-UX B.10.20 A with tcp_random_seq = 0
```

Il comando nmap richiede di effettuare lo scanning delle porte dell'host "t" e di avere informazioni sul Sistema Operativo installato (opzione -O). Il tipo di scanning richiesto è quello che non completa la fase di Three-Way Handshake (-sS).
Il risultato del comando fornisce informazioni sulle porte (per esempio: 21), sul loro stato (per esempio: open) e sui servizi (per esempio: ftp).
Nell'elenco troviamo due porte aperte (849 e 853) il cui servizio però non è noto, esse dovranno quindi essere oggetto di indagine da parte dell'amministratore (potrebbero essere applicazioni server non autorizzate).
Alla fine del report ci sono le informazioni sul Sistema Operativo: HP-UX B.10.20, cioè il Sistema Operativo Unix di HP, versione B.10.20; questo ci fa anche capire che "t" è una workstation HP, dal momento che quel Sistema Operativo si installa su questo tipo di computer.

IN ENGLISH PLEASE

Nmap is ...

- **Flexible**: supports dozens of advanced techniques for mapping out networks filled with IP filters, firewalls, routers, and other obstacles. This includes many port scanning mechanisms (both TCP & UDP), OS detection, version detection, ping sweeps, and more. See the documentation page.
- **Well documented**: significant effort has been put into comprehensive and up-to-date man pages, whitepapers, tutorials, and even a whole book! Find them in multiple languages here.
- **Portable**: most operating systems are supported, including Linux, Microsoft Windows, FreeBSD, OpenBSD, Solaris, IRIX, Mac OS X, HP-UX, NetBSD, Sun OS, Amiga, and more.
- **Easy**: while Nmap offers a rich set of advanced features for power users, you can start out as simply as "nmap -v -A *targethost*". Both traditional command line and graphical (GUI) versions are available to suit your preference. Binaries are available for those who do not wish to compile Nmap from source.
- **Free**: the primary goals of the Nmap Project is to help make the Internet a little more secure and to provide administrators/auditors/hackers with an advanced tool for exploring their networks. Nmap is available for free download, and also comes with full source code that you may modify and redistribute under the terms of the license.
- **Powerful**: Nmap has been used to scan huge networks of literally hundreds of thousands of machines.
- **Supported**: while Nmap comes with no warranty, it is well supported by a vibrant community of developers and users. Most of this interaction occurs on the Nmap mailing lists. Most bug reports and questions should be sent to the nmap-dev list, but only after you read the guidelines. We recommend that all users subscribe to the low-traffic nmap-hackers announcement list. You can also find Nmap on Facebook and Twitter. For real-time chat, join the #nmap channel on Freenode or EFNet.
- **Acclaimed**: Nmap has won numerous awards, including "Information Security Product of the Year" by Linux Journal, Info World and Codetalker Digest. It has been featured in hundreds of magazine articles, several movies, dozens of books, and one comic book series. Visit the press page for further details.
- **Popular**: thousands of people download Nmap every day, and it is included with many operating systems (Redhat Linux, Debian Linux, Gentoo, FreeBSD, OpenBSD, etc). It is among the top ten (out of 30,000) programs at the Freshmeat.Net repository. This is important because it lends Nmap its vibrant development and user support communities.

by *https://nmap.org*

FISSA LE CONOSCENZE

- Quali informazioni si ottengono con il comando netstat?
- Che significato ha l'indirizzo 0.0.0.0 come indirizzo locale?
- Quali informazioni si ottengono con il software Nmap?
- Quali stati può avere un servizio di rete?

9 WIRESHARK: I PROTOCOLLI UDP E TCP

In questo laboratorio utilizziamo Wireshark per analizzare il datagram del protocollo UDP (**primo esercizio**) e il segment del protocollo TCP (**secondo esercizio**) e verificare quanto studiato in questa unità.

esercizio

➜ PROBLEMA

Dopo aver catturato dei pacchetti DNS relativi alla risoluzione di un nome, visualizzarne il datagram UDP individuando il contenuto di ciascun campo.

➜ ANALISI DEL PROBLEMA

Per poter rispondere alla richiesta dell'esercizio dobbiamo procedere alla cattura dei pacchetti DNS. Il DNS utilizza il protocollo connectionless UDP e quindi incapsula il datagram UDP.
Per catturare un pacchetto DNS dovremo eseguire un'interrogazione verso un name server DNS. Quindi, dopo aver avviato la cattura dei pacchetti con Wireshark, dalla finestra del Prompt dei comandi di Windows (o del terminale per i sistemi Linux) occorre eseguire il comando:

#prendinota

Il comando **nslookup** permette di interrogare un DNS per ricevere l'indirizzo IP corrispondente al nome logico inserito.

```
nslookup www.wireshark.org 8.8.8.8
```

In questo caso si richiede l'indirizzo IP del web server *wireshark.org*, usando un name server di Google avente indirizzo 8.8.8.8.
Infine si interrompe la cattura e si analizzano i pacchetti relativi al DNS.

Per agevolare l'individuazione dei pacchetti da analizzare è utile impostare un filtro che permetta di visualizzare solo i pacchetti UDP relativi al comando dato mediante la procedura spiegata nella Lezione 6 dell'Unità 1. Nel nostro caso impostiamo come filtro (dovrebbero essere solo 2 pacchetti):

```
ip.addr == 8.8.8.8
```

➜ SVOLGIMENTO

Nella FIGURA 23 è stato evidenziato il datagram UDP del pacchetto DNS catturato, da analizzare.

FIGURA 23 L'header del datagram UDP contenuto nel pacchetto DNS catturato

```
▷ Frame 3: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0
▷ Ethernet II, Src: CompalCo_a1:fc:f9 (00:16:d4:a1:fc:f9), Dst: Vodafone_2b:64:e0 (64:59:f8:2b:64:e0)
▷ Internet Protocol Version 4, Src: 192.168.1.10, Dst: 8.8.8.8
◢ User Datagram Protocol, Src Port: 61111 (61111), Dst Port: 53 (53)
    Source Port: 61111
    Destination Port: 53
    Length: 46
  ▷ Checksum: 0xd201 [validation disabled]
    [Stream index: 1]
▷ Domain Name System (query)
```

Ora che abbiamo individuato i pacchetti da analizzare, cerchiamo di trarne le informazioni richieste sul datagram UDP rispondendo alle seguenti domande.

D1. Selezionare un pacchetto UDP del tracciato e indicare quanti campi ci sono nell'header UDP. Qual è la lunghezza (in byte) di ciascuno di questi campi?

R. L'header UDP ha una lunghezza fissa di 8 byte e contiene 4 campi, ciascuno di 2 byte: numero di porta del mittente, numero di porta del destinatario, lunghezza del datagram (header + dati) e checksum.

D2. Qual è il numero massimo di byte che può essere incluso nel payload UDP (cioè nella parte dati)? Nel caso del pacchetto mostrato in Figura 23, qual è la lunghezza del payload?

R. L'header UDP ha una lunghezza fissa di 8 byte, il campo Length occupa 2 byte $\rightarrow 2^{16}-1 = 65.535$ byte.
65.535 – 8 = 65.527 byte è la lunghezza massima del payload UDP.
Nell'header del pacchetto catturato, mostrato in Figura 23, il campo Length indica il numero di byte del datagram UDP (header + dati), quindi sapendo che la lunghezza dell'header è 8 byte, la lunghezza del payload è 38 byte (46 - 8 = 38).

D3. Qual è il numero di protocollo usato per UDP?

R. Per rispondere a questa domanda è necessario esaminare l'header IP, qui troviamo il valore esadecimale 0x11 pari a 17 in base 10.

D4. Analizzando i due pacchetti UDP, uno inviato dal nostro host e l'altro ricevuto in risposta, descrivere la relazione esistente tra i numeri di porta presenti nei due pacchetti.

R. Il numero di porta mittente del pacchetto UDP inviato è uguale al numero di porta destinatario del pacchetto UDP ricevuto; viceversa, il numero di porta destinatario del pacchetto UDP inviato è uguale al numero di porta mittente del pacchetto UDP ricevuto

esercizio

➜ PROBLEMA

Dopo aver catturato dei segment TCP relativi alla richiesta di una pagina web, individuare il contenuto di ciascun campo e analizzare l'handshake a tre vie.

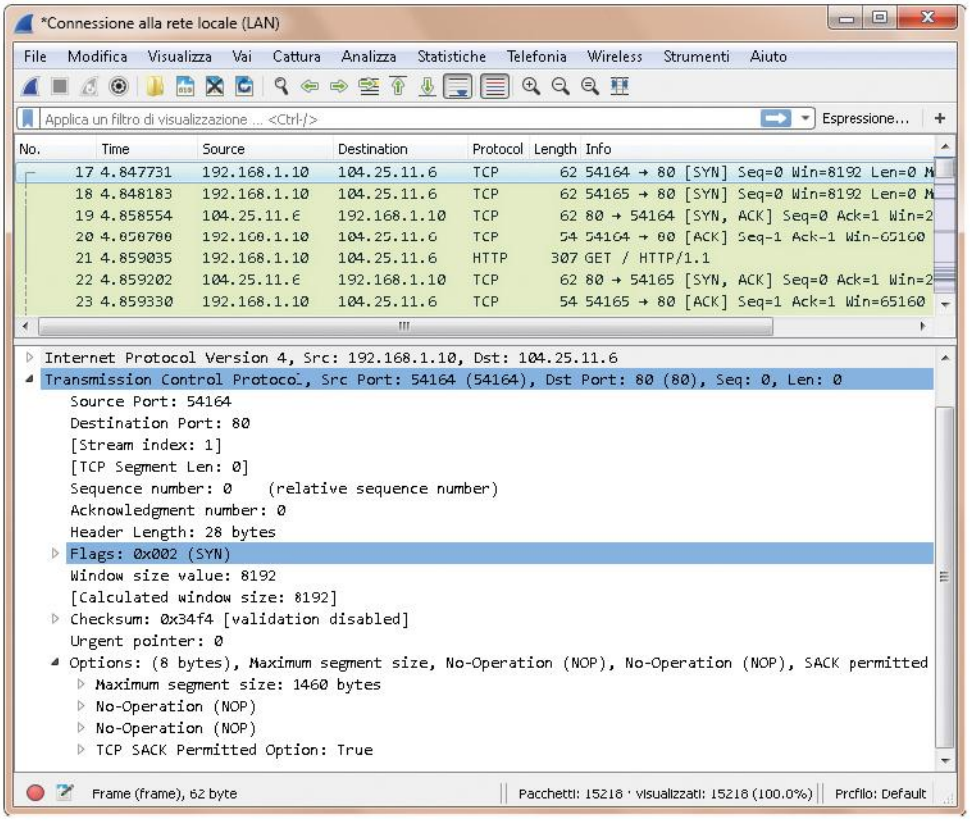
➜ ANALISI DEL PROBLEMA

Per poter rispondere alle richieste dell'esercizio dobbiamo procedere alla cattura dei segment TCP. Effettuando una richiesta di una pagina web (HTTP usa il protocollo connection-oriented TCP) potremo catturare i segment TCP relativi alla fase di avvio della comunicazione (handshake a tre vie).
TCP è un protocollo affidabile e a tale scopo utilizza numeri di sequenza e di acknowledgment per mantenere l'ordine dei pacchetti e tracciarne la ricezione. Di solito questi numeri sono grandi e non è agevole rintracciarli nei pacchetti e seguirne l'utilizzo. Wireshark può convertire e visualizzare questi numeri con un valore relativo che parte dal numero 0 assegnato all'inizio della sessione.
Quindi, dopo aver avviato la cattura dei pacchetti con Wireshark, collegarsi al sito web: *www.wireshark.com*, scaricare la User's Guide dalla sezione Documentation nel menu Get Help, chiudere il browser e poi interrompere la cattura.

➜ SVOLGIMENTO

FIGURA 24 Fase di apertura della connessione TCP tra client e server

La seguente **FIGURA 24** mostra i pacchetti catturati relativi alla fase di apertura della connessione TCP tra web client e web server (handshake a tre vie).



Ora che abbiamo individuato i pacchetti da analizzare, cerchiamo di trarne le informazioni richieste sul segment TCP e sull'handshaking rispondendo alle seguenti domande.

D1. Qual è l'indirizzo IP del client e quale quello del server? Da quale numero di porta il client inizia la connessione e quale numero è usato dal server?

R. Client: indirizzo IP 192.168.1.10 porta TCP 54164
Server: indirizzo IP 104.25.11.6 porta TCP 80

D2. Individuare le fasi dell'**handshake a tre vie**:

- trovare il pacchetto con il flag **SYN** settato: qual è la lunghezza dell'header TCP? Qual è il valore del Sequence number? Qual è la dimensione della finestra (Window size)? Qual è la dimensione massima consentita del segmento (MSS, Maximum segment size)?

R. Durante la fase di handshaking la lunghezza dell'header è un po' più lunga di quella base (20 byte) per via di alcuni campi opzionali.
Sequence number = 0; Acknowledgment number = 0; Window size = 8.192;
MSS = 1.460 byte; Lunghezza dati = 0 (non è un segmento dati)

- trovare il pacchetto di risposta con i flag **SYN** e **ACK** settati: qual è il valore del Sequence number e dell'Acknowledgment number?

R. Sequence number = 0;
Acknowledgment number = 1 (in risposta al Sequence number = 0 ricevuto)

- trovare il pacchetto che conclude l'handshaking con il flag **ACK** settato: qual è il valore del Sequence number e dell'Acknowledgment number?

R. Sequence number = 1 (è il secondo pacchetto che invia al server); Acknowledgment Number = 1 (in risposta al Sequence Number = 0 ricevuto)

D3. Finita la fase di handshaking della connessione, qual è la lunghezza massima del payload TCP?

R. Instaurata la connessione, i segmenti TCP hanno la lunghezza dell'header pari a 32 byte. Quindi la lunghezza massima del payload è di 1.448 byte (1500 – 20 header IP – 32 header TCP).

Una interessante e utile funzionalità di Wireshark, illustrata nella Lezione 6 dell'Unità 1, riguarda la generazione di grafici. Selezionando nel menu Statistiche → Grafici dei flussi TCP → Round Trip Time si ottiene il grafico dei tempi di andata (client → server) e ritorno (server → client) dei segmenti TCP.

L'analisi dei pacchetti TCP è anche interessante per rilevare pacchetti sospetti. La seguente **TABELLA 4** mostra alcuni esempi di filtri da configurare su Wireshark al fine di isolare e ispezionare pacchetti sospetti.

TABELLA 4 Esempi di filtri per analizzare pacchetti TCP sospetti

Descrizione	Filtro da applicare alla visualizzazione
L'analisi dei pacchetti usati nell'handshaking è utile per rilevare attività di scan delle porte TCP, ma anche normali attività di instaurazione e chiusura della connessione.	(tcp.flags&02 && tcp.seq == 0) ‖ (tcp.flags&12 && tcp.seq == 0) ‖ (tcp.flags.ack && tcp.seq == 1 && !tcp.nxtseq > 0 && !tcp.ack > 1)‖ tcp.flags.fin == 1 ‖ tcp.flags.reset ==1
Altre attività sospette riguardano impostazioni della dimensione della finestra inferiori a 1.025, i bit di flag settati o, al contrario, nessuno impostato, MSS inferiore a 1.460.	((tcp.flags == 0x02) && (tcp.window_size < 1025)) ‖ tcp.flags == 0x2b ‖ tcp.flags == 0x00 ‖ tcp.options.mss_val < 1460

Un'attività sospetta è sicuramente quella della scansione delle porte TCP: un host che intende effettuare la scansione delle porte TCP di un altro host, invierà un pacchetto TCP SYN a una porta di questo host che risponderà con SYN o ACK se la porta è aperta o con RST se è chiusa. Come per le attività sospette ARP scan, una scansione TCP si può rilevare da una serie di pacchetti SYN inviati da un solo host verso un determinato IP address su un range di numeri di porta. In tal caso il filtro da applicare è:

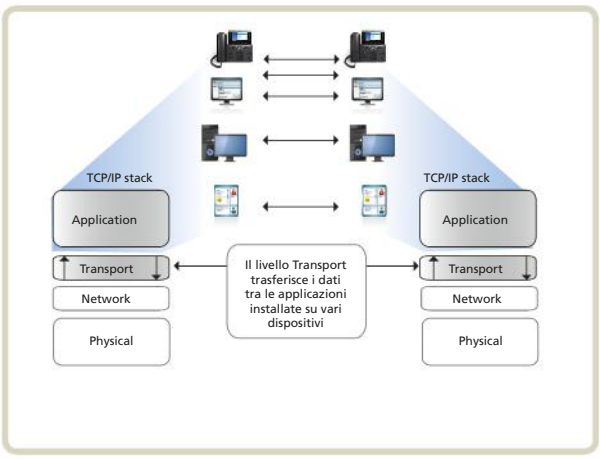
```
ip.dest == <IP Address of target host> && tcp.flags.syn
```

FISSA LE CONOSCENZE

- Perché abbiamo effettuato un'interrogazione al DNS per visualizzare un datagram UDP?
- Perché abbiamo richiesto una pagina web per visualizzare un segment TCP?
- A cosa servono i filtri?

1 Le porte, le socket e i servizi

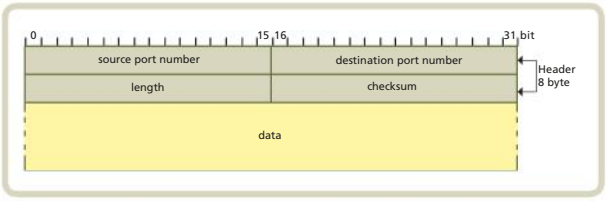
Il livello di trasporto consente la comunicazione tra due processi su host differenti con un controllo a *livello end-to-end*. Un protocollo a livello Transport può gestire più connessioni aperte contemporaneamente verso uno stesso computer, è l'host di destinazione che conosce a quale processo deve consegnare ciascun messaggio ricevuto. Questo meccanismo è realizzato tramite l'uso delle *porte* e delle *socket* con cui configurare i servizi tipici del livello.



2 Le funzionalità di multiplexing e demultiplexing

Tutti i protocolli del livello Transport offrono servizi di *multiplexing* (raccolgono i dati provenienti da diverse applicazioni, vi aggiungono l'header e inoltrano il segmento al livello Network) e *demultiplexing* (ricevono un segmento dal livello Network, esaminano l'header e inoltrano i dati al processo applicativo destinatario).

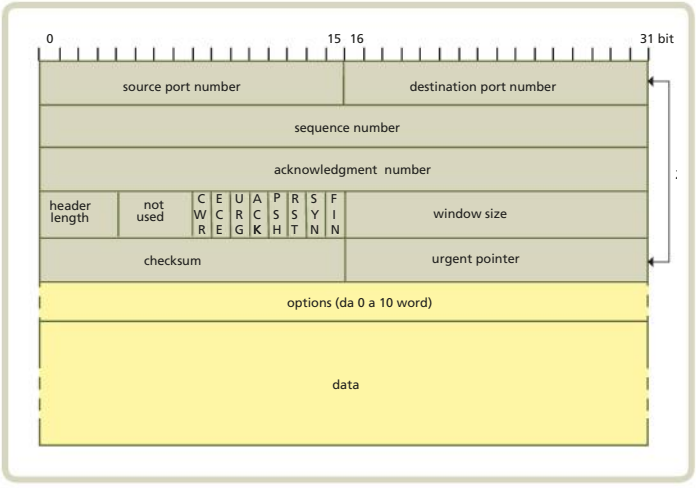
3 Un protocollo di trasporto connectionless: UDP



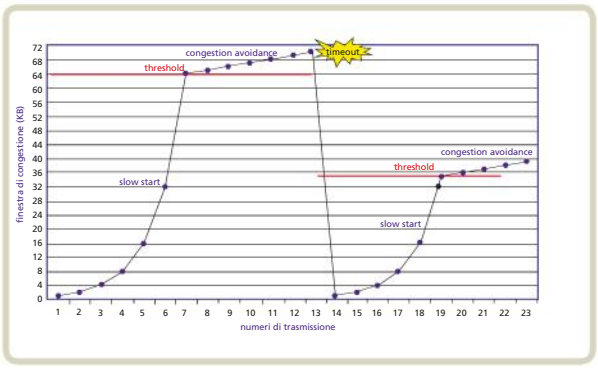
UDP offre solo le funzionalità di multiplexing/demultiplexing.
Il servizio fornito è *connectionless* e *non affidabile*. Nei primi anni Duemila è stata standardizzata una versione *lite* per soddisfare le esigenze delle nuove applicazioni di streaming e di telefonia su IP.

4 Un protocollo di trasporto connection-oriented: TCP

TCP è il protocollo più utilizzato tra quelli che sono stati standardizzati per il livello Transport. Offre un servizio *connection-oriented* e *affidabile*. Un processo, che vuole inviare un messaggio, scrive il testo da spedire in un buffer nel suo spazio in memoria, inserisce le informazioni di controllo (*header*) e passa il controllo a TCP. Analogamente farà il processo che deve ricevere un messaggio.

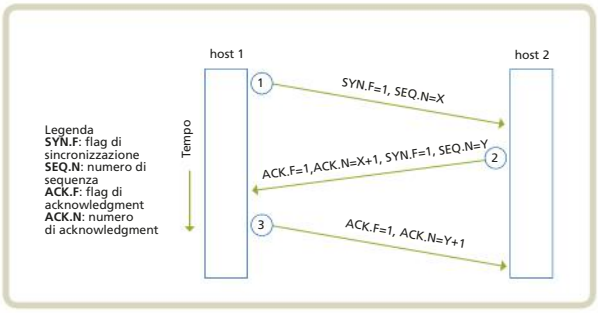


5 La gestione della congestione



La funzione TCP di gestione della congestione è particolarmente critica perché si basa su deduzioni che avvengono agli end system e non su dati precisi prelevati in rete. Non ci sono quindi garanzie che queste deduzioni siano sempre esatte. L'idea di base è di diminuire la finestra di congestione quando un pacchetto è scartato dalla rete e aumentarla quando un pacchetto è riscontrato. Infatti se si sono persi pacchetti per la congestione, essi devono essere ritrasmessi, aumentando così il traffico in una rete che ne ha già troppo. Quindi per non far collassare la rete è importante che il mittente riduca il traffico che genera e sia cauto nell'aumentarlo. TCP realizza tutto questo con gli algoritmi *slow start* e *congestion avoidance*.

6 L'handshaking TCP



La comunicazione tra mittente e destinatario avviene attraverso 3 fasi: instaurazione di una sessione TCP, trasferimento dati e chiusura della sessione. L'apertura della sessione avviene con la tecnica Three-Way Handshake. Durante la fase di trasmissione dei dati, TCP gestisce il controllo di flusso e degli errori (segmenti danneggiati, fuori sequenza o duplicati) con meccanismi di *sliding windows* come il *Go-back-N*. La chiusura della sessione avviene con la tecnica Three-Way Handshake se in contemporanea, con la Double Two-Way Handshake in sequenza.

7 Il confronto tra i protocolli UDP e TCP

TCP e UDP hanno in comune alcune caratteristiche tipiche dei protocolli del livello Transport: le funzionalità di multiplexing/demultiplexing e l'impiego delle porte. TCP è il protocollo da preferire quando è richiesta integrità dei dati e affidabilità. UDP è invece preferibile quando le prestazioni sono più importanti del ricevere i dati in modo perfetto, senza alcuna perdita.

8 LABORATORIO: Il controllo delle porte

I sistemi che implementano il protocollo TCP/IP tipicamente forniscono un'utility, netstat, che può essere usata per avere varie informazioni, tra le quali l'elenco delle porte aperte. Un software interessante per effettuare port scanning è *Nmap*.

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. Tutti i protocolli del livello Transport offrono la funzionalità di multiplexing/demultiplexing. V F
2. Per i protocolli di trasporto connection-oriented, le connessioni sono identificate dalla coppia <Server IP address><Server Port Number>. V F
3. Le porte sono punti di accesso ai servizi di un host. V F
4. Le socket consentono la comunicazione tra le applicazioni. V F
5. Con primitiva si intende una chiamata di sistema a una routine del kernel. V F
6. UDP usa l'handshaking per instaurare una connessione. V F
7. Il Three-Way Handshake è un algoritmo per la gestione della congestione. V F
8. UDP è un protocollo connection-oriented. V F
9. Per un'applicazione, che necessita di garanzie circa la consegna dei dati, si utilizza UDP. V F
10. L'header UDP è di soli 8 byte. V F
11. Il *checksum coverage length* specifica quanti byte del datagram UDP-Lite saranno controllati. V F
12. TCP è un protocollo che prevede la conferma dei dati trasmessi. V F
13. Il campo *sequence number* nell'header TCP è usato per confermare il numero dell'ultimo segmento ricevuto. V F
14. La tecnica Double Two-Way Handhake viene usata da TCP per instaurare la connessione. V F
15. Con dimensione della finestra si intende il numero di byte che si possono trasmettere senza essere riscontrati. V F
16. Lo *slow start* è un algoritmo per abbattere una connessione TCP. V F
17. Entrambi i protocolli UDP e TCP garantiscono la consegna *ordinata* dei dati. V F
18. Il comando netstat e il software Nmap permettono di determinare le porte aperte. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **Il livello Transport gestisce il trasferimento dati tra:**
   - A router
   - B end system
   - C processi
   - D utenti

2. **La funzione di multiplexing (*due scelte*):**
   - A consiste nel determinare a quale socket consegnare i dati
   - B consiste nel ricevere i dati dalle varie socket e aggiungervi un header
   - C è svolta dal livello Transport in trasmissione
   - D è svolta dal livello Transport in ricezione

3. **Quale parametro indica la dimensione massima di un segmento TCP?**
   - A MTU
   - B MSS
   - C MRU
   - D max(MTU, MSS, MRU)

4. **TCP stabilisce un canale logico, tra mittente e destinatario, del tipo:**
   - A full-duplex, multi-point, connection-oriented
   - B full-duplex, point-to-point, connection-oriented
   - C half-duplex, point-to-point, connection-oriented
   - D half-duplex, multi-point, connection-oriented

5. **Quale comando si usa per conoscere quali porte sono aperte su un computer?**
   - A Netstat
   - B Ipconfig
   - C Ping
   - D Traceroute

6. **Gli header di TCP e di UDP hanno in comune i seguenti campi:**
   - A source address e destination address
   - B source port number e destination port number
   - C sequence number e acknowledgment number
   - D sender address e receiver address

7. **Quale dei seguenti non è un algoritmo per il controllo della congestione?**
- A Slow start
- B Slow recovery
- C Congestion avoidance
- D Fast retransmit

8. **Se nell'handshaking a tre vie l'Host1 al primo passo invia SYN.F = 1 e SEQ.N = 30, cosa potrebbe rispondere l'Host2 al secondo passo?**
- A ACK.F = 1 e ACK.N = 31
- B ACK.F = 1 e ACK.N = 29
- C ACK.F = 1 e ACK.N = 0
- D ACK.F = 1 e ACK.N = 30

9. **Se nell'handshaking a tre vie l'Host2 al secondo passo invia ACK.F = 1, ACK.N = 100 e SYN.F = 1, SEQ.N = 30, cosa potrebbe aver ricevuto al primo passo dall'Host1?**
- A ACK.F = 1 e ACK.N = 31
- B ACK.F = 1 e ACK.N = 99
- C SYN.F = 1 e SEQ.N = 31
- D SIN.F = 1 e SEQ.N = 99

10. **Se nell'handshaking a tre vie l'Host2 al secondo passo invia ACK.F = 1, ACK.N = 100 e SYN.F = 1, SEQ.N = 30, cosa potrebbe rispondere l'Host1 al terzo passo?**
- A ACK.F = 1 e ACK.N = 31
- B ACK.F = 1 e ACK.N = 99
- C ACK.F = 1 e ACK.N = 101
- D ACK.F = 1 e ACK.N = 29

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Descrivere le motivazioni che portano alla necessità di un livello Transport.**

2. LEZIONE 1 **Da cosa è composta un'Association?**

3. LEZIONE 1 **Che cosa si intende per trasmissione end-to-end?**

4. LEZIONE 1 **Qual è il vantaggio di usare le porte invece dei process id (identificativi di processo) per indicare una destinazione all'interno di un host?**

5. LEZIONE 1 **Che cosa sono le Well Known Ports? Fai qualche esempio.**

6. LEZIONE 2 **Che cosa si intende per multiplexing e demultiplexing a livello Transport?**

7. LEZIONE 2 **Che cosa si intende per servizio connectionless?**

8. LEZIONE 2 **Che cosa si intende per servizio connection-oriented?**

9. LEZIONE 2 **Può capitare che due client utilizzino lo stesso source number port? È un problema?**

10. LEZIONE 3 **Quale funzionalità introduce UDP rispetto a quanto fornito dal protocollo IP?**

11. LEZIONE 3 **Che tipo di servizio fornisce TCP ai livelli superiori?**

12. LEZIONI 3 E 4 **Quali campi hanno in comune gli header di TCP, UDP e UDP-Lite e che quindi sono indispensabili a livello Transport?**

13. LEZIONE 5 **Quali algoritmi si possono implementare in TCP per la gestione della congestione?**

14. LEZIONE 5 **Su quale idea di base funzionano questi algoritmi?**

15. LEZIONE 6 **Descrivi la fase di creazione di connessione TCP.**

16. LEZIONE 6 **Come può un destinatario TCP indicare al mittente di sospendere momentaneamente la trasmissione dati?**

17. LEZIONE 6 **Descrivi la fase della chiusura di una connessione TCP.**

18. LEZIONE 6 **Descrivi le principali vulnerabilità di TCP.**

19. LEZIONE 7 **In quali contesti si usa il protocollo UDP?**

20. LEZIONE 7 **In quali contesti si usa il protocollo TCP?**

IN ENGLISH, PLEASE

CLIL

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

The Transport Layer of TCP/IP Architecture

The Transport Layer enables the communication between two processes running on different hosts. A protocol belonging to this layer can manage several connections which are simultaneously open on the same computer by means of gateways and sockets. The two main protocols are UDP and TCP. The former provides a connectionless and unreliable service. The latter provides a connection-oriented and reliable service. TCP and UDP share some features typical of Transport Layer protocols, such as multiplexing/demultiplexing and the use of ports. TCP protocol should be preferred in handling data when integrity and reliability are required and critical.
UDP protocol should be preferred when performance is more important than receiving data without any loss.

EXERCISES

Use the appropriate number to match words and meanings.

...	Word	1	The data are received without errors
...	Multiplexing	2	Time expired
...	Timeout	3	From many to one
...	Congestion	4	A unit of 32 bits
...	Acknowledgment	5	Number that defines a limit
...	Setup	6	From one to many
...	Demultiplexing	7	Connection establishment phase
...	Threshold	8	It causes packet loss in the network

Choose the correct answer.

1. Which UDP header field is optional?
- A Source port number
- B Destination port number
- C Length
- D Checksum

2. Which addresses are managed at the Transport Layer?
- A Host name
- B MAC address
- C Port numbers
- D IP address

GLOSSARY

Connection: a TCP connection is initiated with the Three-Way Handshake, that establishes a session between the two hosts. When the hosts have finished sending data, the TCP connection is closed down.
Congestion: the applications send more data than the network devices can manage, so their buffers become full and incoming packets are lost.
Demultiplexing: receiving a Transport Layer segment from the Network Layer and delivering its data to the correct application process.
Handshaking: the process by which two hosts initiate a TCP session.
Multiplexing: the operation of gathering data at the source host from different application processes, enveloping the data with header information, and passing the segments to the Network Layer.
MSS (Maximum Segment Size): it is the maximum size of the user data field that can be managed by TCP.
Out of order: a segment with a different sequence number from what the receiver is expecting.
Port: an internal address that acts as a pathway to control data flow.
Sequence number: number used to ensure correct sequencing of the incoming data.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper scegliere il tipo di protocollo di trasporto in base al grado di affidabilità, alla velocità e alla sicurezza del servizio che si vuole offrire.
- Saper descrivere e comparare il funzionamento dei protocolli di trasporto.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Consultare fonti Internet.
- Esporre i risultati del lavoro svolto alla classe.

tempi

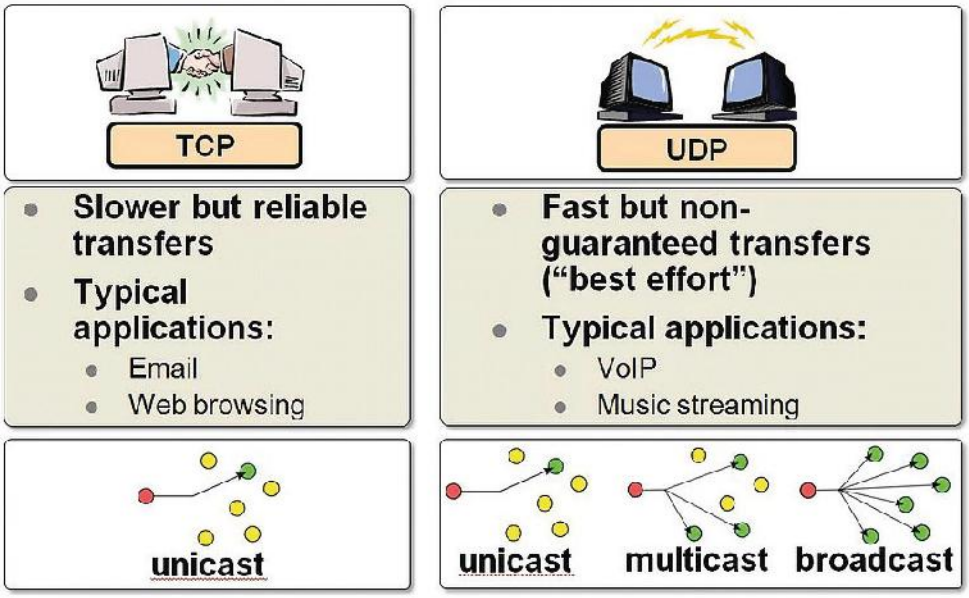
- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Dispositivo connesso a Internet.
- Carta e penna.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Per capire l'affidabilità dei principali servizi messi a disposizione dalla rete Internet e le caratteristiche delle trasmissioni che li realizzano, il primo passo è andare a vedere quale protocollo di trasporto viene utilizzato dalle applicazioni che gestiscono i servizi.



Compila una tabella riportando per ogni applicazione individuata (come la posta elettronica o la navigazione web) il protocollo di livello superiore che la genera e scopri quale header, tra UDP e TCP, viene incapsulato per caratterizzarne il trasporto.

SVOLGIMENTO

Dopo aver individuato alcune delle principali applicazioni di rete e il protocollo di livello Application che ognuna utilizza, cerchiamo, sul libro di testo o su Internet, il protocollo di livello Transport associato.
La tabella seguente mostra il risultato della ricerca fatta.

Applicazione	Protocollo livello Application	Protocollo livello Transport
posta elettronica	SMTP	TCP
accesso a terminale remoto	Telnet	TCP
trasferimento file	FTP	TCP
web	HTTP	TCP
streaming audio/video	RTSP/RTP	TCP (comandi) + UDP (flusso)
server di file remoto	NFS	UDP
telefonia su Internet (VoIP)	SIP, H.323…	UDP
gestione della rete	SNMP	UDP
risoluzione dei nomi	DNS	UDP (richieste/risposte semplici), TCP (per trasferimenti di grandi quantità di dati)
assegnazione di indirizzo IP	DHCP	UDP
routing	RIP	UDP

A CASA

- Ipotizza una tua soluzione al tema proposto.
- Leggi la proposta di SVOLGIMENTO per verificare se le tue ipotesi si adattano al caso preso in esame e se la trattazione proposta risulta completa.
- Cerca su Internet qualche altra applicazione non inclusa nella tabella e aggiungila.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Cconfrontate e discutete insieme i casi che sono stati presentati.
- Stabilite quale caso rappresenta l'esempio migliore per completezza e che meglio si adatta alla soluzione del tema proposto.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso quasi tutte le richieste in autonomia. ☐	Ho identificato tutte le richieste in autonomia. ☐
Ho compilato una tabella esaustiva?	Sono riuscito a trovare solo alcune applicazioni e non sempre ho individuato il protocollo di trasporto utilizzato. ☐	Con la guida del docente e la collaborazione dei compagni sono riuscito a trovare molte applicazioni ma non sempre ho individuato il protocollo di trasporto utilizzato. ☐	Con la guida del docente sono riuscito a compilare una tabella con molte applicazioni e il rispettivo protocollo di trasporto utilizzato. ☐	Sono riuscito a compilare in autonomia una tabella con molte applicazioni e il rispettivo protocollo di trasporto utilizzato. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ

7

LA CONFIGURAZIONE DEL DHCP E DEL DNS

IN QUESTA UNITÀ

Guarda la **presentazione** dell'unità

DDI+ Didattica Digitale Integrata Plus

conoscenze	abilità	competenze
Caratteristiche e funzionamento dei protocolli DHCP e DNS.	Classificare una rete e i servizi offerti con riferimento agli standard tecnologici.	Configurare, installare e gestire sistemi di elaborazione dati e reti.
Modalità di configurazione dei parametri TCP/IP su differenti Sistemi Operativi.	Configurare il software di rete sugli host.	Descrivere e comparare il funzionamento di dispositivi e strumenti elettronici e di comunicazione.
		Scegliere dispositivi e strumenti in base alle loro caratteristiche funzionali.

FLIPPED CLASSROOM

A casa

- Leggi la Lezione 3 di questa Unità;
- supponi di essere un tecnico di rete in uno scenario WLAN con centinaia di host;
- raccogli in una tabella i vantaggi e gli svantaggi dell'assegnazione dinamica degli indirizzi IP mediante il DHCP in un tale scenario.

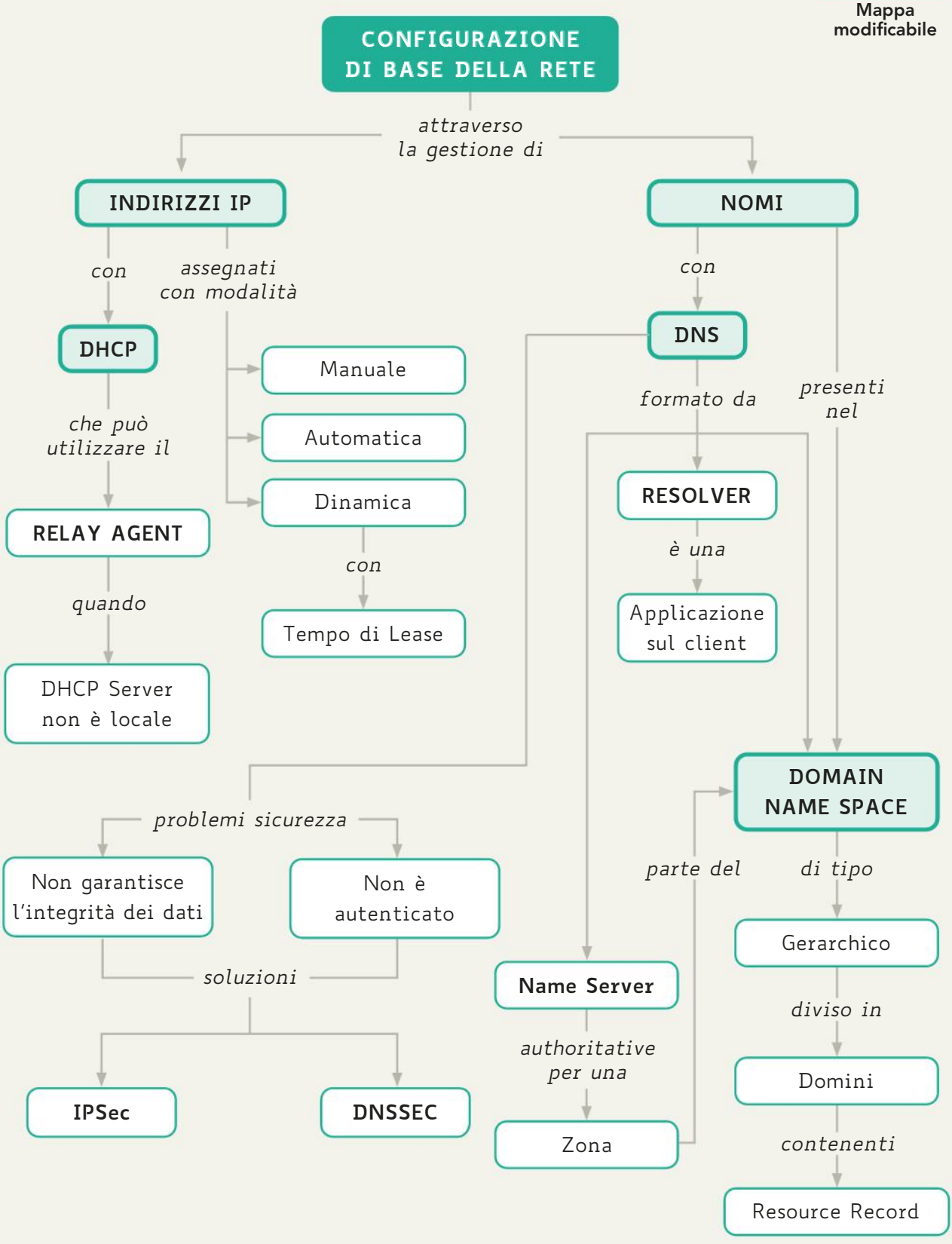
In classe

- Confrontate i risultati raccolti;
- valutate se tra questi vi sono differenze;
- discutete i motivi che spiegano le eventuali differenze.

MAPPA CONCETTUALE

DIDATTICA INCLUSIVA

Mappa modificabile



1 LA CONFIGURAZIONE DEGLI HOST

1.1 Introduzione

Gli amministratori di rete svolgono quotidianamente molte operazioni per mantenere efficiente la rete. Quando questa è di dimensioni ridotte, o localizzata in uno stesso edificio, può ancora essere gestita manualmente, come avveniva in passato. Quando, però, l'amministratore di rete deve occuparsi di reti estese e complesse, queste richiedono strumenti per una gestione automatica e da remoto delle risorse.

> In particolare per i device mobili (notebook, tablet, smartphone, ecc.), l'automatizzazione delle operazioni di configurazione e gestione della rete è diventata indispensabile, poiché questi dispositivi necessitano di un nuovo indirizzo IP ogni volta che cambiano rete.

In generale, possiamo distinguere due tipi di attività che svolge un amministratore di rete:

1. **host configuration**: ogni host deve essere configurato prima del suo utilizzo e della messa in rete; la configurazione permette all'host di svolgere le sue funzioni in modo corretto e appropriato;
2. **host management**: la gestione e il controllo degli host di una rete sono attività che devono poter essere effettuate da remoto, mantenendo sotto controllo alcuni parametri che segnalano all'amministratore anomalie (per esempio se l'interfaccia di un router diventa down) o anche solo un degrado delle prestazioni (per esempio se un elevato numero di pacchetti su un link viene perso).

Il principale protocollo usato per la configurazione degli host è **DHCP** (Dynamic Host Configuration Protocol), mentre per la gestione si usa il protocollo **SNMP** (Simple Network Management Protocol), che sarà trattato nel volume del quinto anno. Entrambi questi protocolli sono stati definiti in ambito IETF e appartengono al livello Application. Questa collocazione può sembrare incoerente (infatti uno dei primi rudimentali protocolli usati per la configurazione degli host è stato il protocollo **RARP** (Reverse Address Resolution Protocol), definito nel livello Network); in realtà, la definizione dei protocolli DHCP e SNMP a livello Application offre diversi vantaggi:

- le operazioni di configurazione non sono legate all'hardware della macchina, ma sono svolte in modo indipendente, risultando valide per qualunque tipo di host;
- i messaggi vengono scambiati tra gli end system (E2E), quindi potrebbe accadere che attraversino più reti; ciò non sarebbe possibile con un protocollo definito nei livelli più bassi dello stack TCP/IP che svolge i suoi compiti in un ambito locale;
- sono utilizzabili le funzionalità offerte dai protocolli del livello Transport.

1.2 Un po' di storia: il protocollo BOOTP

Quando si configura la connessione in rete di una macchina (PC, stampante, ecc.) è necessario assegnarle un indirizzo IP che viene memorizzato sul suo disco interno.

Negli anni Ottanta del secolo scorso ci fu una certa diffusione di workstation Unix che non utilizzavano il disco rigido interno e dovevano caricare il Sistema Operativo da un server. Fu quindi necessario definire un protocollo di comunicazione tra workstation e server che ne permettesse il **bootstrap**, cioè permettesse a un computer senza disco di

comunicare con un host che gli fornisse le informazioni di rete; da qui il nome protocollo **BOOTP**, Bootstrap Protocol (RFC 951).

BOOTP fu definito con alcune caratteristiche che lo resero preferibile al protocollo RARP:

- utilizzo del protocollo di trasporto UDP e indipendenza dall'hardware della macchina;
- invio, in un unico messaggio, dell'indirizzo IP e di altre informazioni di configurazione (l'indirizzo del router o del proxy da usare come gateway, la maschera di sottorete, l'indirizzo di un DNS Server);
- gestione di host situati in subnet diverse. Una rete locale può essere composta da più subnet, ma le richieste RARP sono inviate con broadcast Ethernet e non IP, pertanto sono limitate alla subnet e il router non le inoltra. BOOTP utilizza l'**indirizzo di broadcast IP**, quindi è possibile avere un unico server BOOTP per la gestione di host appartenenti a subnet differenti.

La procedura di bootstrap di una workstation senza disco, utilizzando BOOTP, consiste di due fasi:

1. su richiesta del client viene fornito l'indirizzo IP e alcune informazioni su come ottenere l'immagine della memoria (per esempio il nome del file da scaricare);
2. il client usa il protocollo **TFTP** (Trivial File Transfer Protocol) per scaricare il file con l'immagine di memoria, contenente anche il Sistema Operativo da caricare sulla macchina.

Poiché BOOTP poteva convogliare anche altre informazioni di configurazione, gli amministratori lo usavano anche per inviare un'installazione client preconfigurata ai computer nuovi da inserire nella rete aziendale.

Un server BOOTP usa la porta **UDP 67** per ricevere le richieste inviate dai client, mentre un client BOOTP aspetta le risposte del server sulla porta **UDP 68**.

La FIGURA 1 mostra l'host A (client) che invia una richiesta BOOTP in broadcast sulla rete locale usando la porta 67 come destination address e l'indirizzo IP 255.255.255.255 di broadcast limitato, quindi si mette in ascolto della risposta sulla porta 68. L'host D (server) è in ascolto sulla porta 67 per eventuali richieste inviate dai client della rete. Quando riceve la richiesta dell'host A, risponde con un messaggio di broadcast con indirizzo IP di destinazione 255.255.255.255 e porta 68. La risposta conterrà l'indirizzo IP di A (196.10.1.9) e le informazioni sul file di installazione da scaricare.

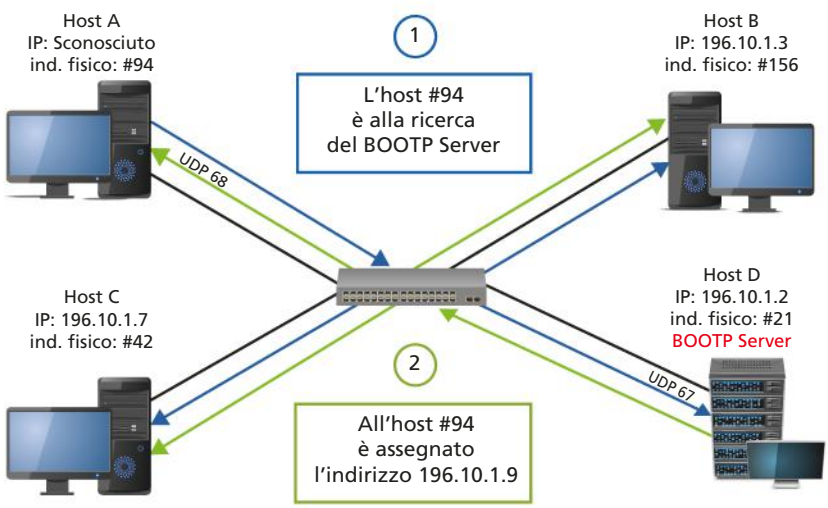


FIGURA 1 Assegnazione dell'indirizzo IP con il protocollo BOOTP

#prendinota

Una macchina che non ha indirizzo IP può inviare una richiesta e ricevere la risposta con le informazioni di configurazione, incluso l'indirizzo IP da usare, grazie alle tecniche di *broadcasting* tipiche delle reti TCP/IP e all'impiego di indirizzi IP speciali di broadcast limitato.

#prendinota

Gli indirizzi di broadcast IP sono quegli indirizzi che hanno tutti 1 nella parte dedicata agli host:

- classe A: X.255.255.255
- classe B: X.Y.255.255
- classe C: X.Y.Z.255

L'indirizzo di broadcast di default è l'indirizzo tutti 1 (255.255.255.255), detto *broadcast limited*, essendo riferito solo alla LAN corrente.

FISSA LE CONOSCENZE

- Spiega la differenza tra host configuration e host management.
- Descrivi l'uso dei protocolli di file transfer previsto in BOOTP.
- Descrivi come avviene la comunicazione tra un client BOOTP e un server BOOTP.

2 IL DHCP (DYNAMIC HOST CONFIGURATION PROTOCOL)

2.1 Introduzione

#prendinota

Mentre BOOTP soppiantò RARP nella procedura di bootstrap introducendo modalità del tutto diverse, DHCP si può considerare un miglioramento di BOOTP e non una rivoluzione.

Quando nelle reti si diffusero la tecnologia **wireless** e l'uso di **computer portatili**, BOOTP risultò troppo lento nell'aggiornare quei dispositivi che si spostavano rapidamente da una rete a un'altra. Fu perciò necessario introdurre una nuova modalità di assegnazione **dinamica**, che consentisse di allocare velocemente gli indirizzi per un periodo di tempo limitato, che fu affiancata alla modalità automatica di assegnazione degli indirizzi IP implementata con BOOTP.
In ambito IETF fu definito così il protocollo **DHCP** (**Dynamic Host Configuration Protocol**), la cui specifica si trova in **RFC 2131**. Successivamente sono stati emessi altri documenti per supportare nuove opzioni e l'introduzione di IPv6, senza però cambiarne le caratteristiche basilari.

IN ENGLISH PLEASE

Network Working Group — R. Droms
Request for Comments: 2131 — Bucknell University
Obsoletes: 1541 — March 1997
Category: Standards Track

Dynamic Host Configuration Protocol

Abstract

The Dynamic Host Configuration Protocol (DHCP) provides a framework for passing configuration information to hosts on a TCP/IP network. DHCP is based on the Bootstrap Protocol (BOOTP), adding the capability of automatic allocation of reusable network addresses and additional configuration options. DHCP captures the behaviour of BOOTP relay agents and DHCP participants can interoperate with BOOTP participants.

Tramite DHCP, oltre all'indirizzo IP, un host può ricevere altri parametri di configurazione. I più importanti sono:

- **subnet mask**, indispensabile per conoscere il prefisso della subnet;
- **default gateway**, per esempio l'indirizzo IP del router che connette la subnet alla rete Internet; a questo sono inviati i pacchetti IP aventi indirizzo di rete del destinatario diverso da quello del mittente;
- **DNS Server preferito**, necessario per la risoluzione dei nomi;
- **DNS server alternativo**.

2.2 Impostazione del DHCP (o di APIPA) su un computer Windows

Per mettere un host (fisso o mobile) con Sistema Operativo Windows 10 nelle condizioni di ricevere i parametri elencati, occorre seguire la seguente procedura:

- aprire il Menu Start e cliccare sulle Impostazioni;

- cliccare su "Rete e Internet", quindi selezionare "Wi-Fi" o "Ethernet", a seconda di quale scheda si vuol configurare;
- cliccare su "Modifica opzioni scheda", e poi sulla scheda che si intende configurare, con il tasto destro, scegliere "Proprietà", comparirà la finestra mostrata in FIGURA 2;
- nel box sotto la voce "La connessione utilizza gli elementi seguenti:" selezionare "Protocollo Internet versione 4 (TCP/IPv4)" oppure "Protocollo Internet versione 6 (TCP/IPv6)", a seconda del protocollo usato, quindi cliccare su "Proprietà";
- per default, la scheda "Generale" è impostata su "Ottieni automaticamente un indirizzo IP" (FIGURA 3), come anche quella relativa al gateway e quella relativa al DNS di cui parleremo nella Lezione 5.

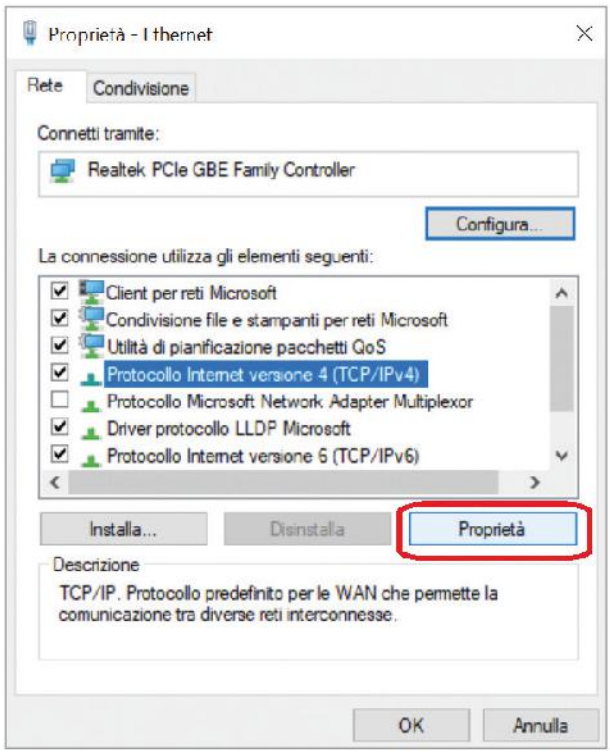


FIGURA 2 Scheda Proprietà - Ethernet

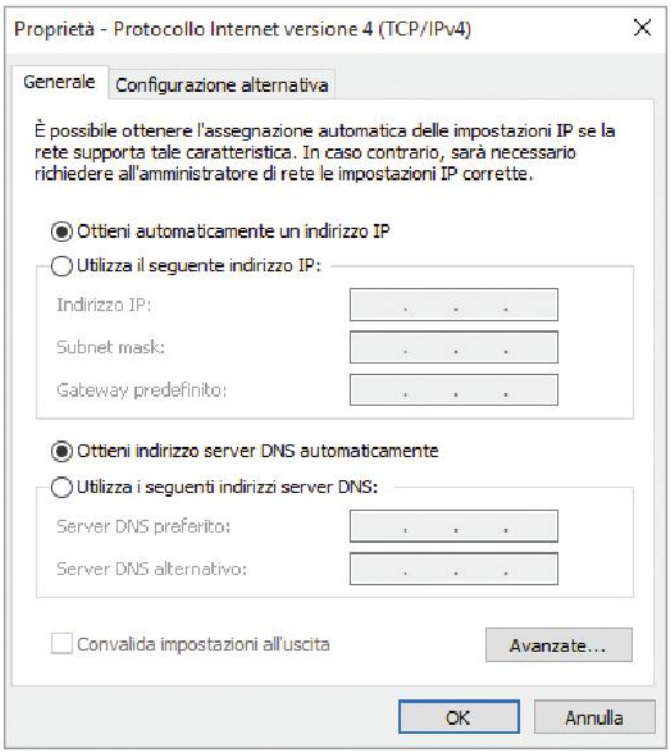


FIGURA 3 Assegnazione automatica dell'indirizzo IP (default)

La versione server del Sistema Operativo Windows di Microsoft include il servizio DHCP Server. Tutti i Sistemi Operativi **Windows Client** di Microsoft installano automaticamente il servizio DHCP Client come parte di TCP/IP.
La configurazione di un computer Windows in LAN è ripresa e approfondita con una esercitazione disponibile sul Laboratorio online di questa Unità.

LABORATORIO ONLINE

CONFIGURAZIONE WINDOWS IN LAN

In questo laboratorio vediamo come configurare un computer Windows Client per la connessione alla rete locale con DHCP.

AUTOMATIC PRIVATE IP ADDRESSING (APIPA)

Il protocollo DHCP richiede la disponibilità di un DHCP Server nella rete; senza di esso un computer configurato per usare DHCP non può ottenere un indirizzo IP e comunicare in rete.

Nel caso in cui un host non trovi un DHCP Server, Windows offre la funzionalità **Automatic Private IP Addressing** (**APIPA**) per fornire all'host in modo automatico un

indirizzo IP e una subnet mask come anticipato nella Lezione 2 dell'Unità 3. Il computer sceglie il proprio indirizzo IP nell'intervallo compreso tra 169.254.0.1 e 169.254.255.254, che IANA (Internet Assigned Numbers Authority) ha riservato a questo scopo.
La Subnet mask viene automaticamente impostata su 255.255.0.0 (maschera di default della Classe B) e l'indirizzo del gateway su 0.0.0.0.
APIPA fa parte del software del Sistema Operativo del computer, quindi l'assegnazione descritta avviene senza bisogno di registrazioni o di verifiche con un'autorithy centrale.
Quando viene usato per sopperire a una momentanea indisponibilità del DHCP Server, nel momento in cui il server è nuovamente accessibile, APIPA rilascia l'indirizzo IP assegnato, in modo che il DHCP Client possa richiedere al server un nuovo indirizzo secondo le regole del protocollo DHCP.
Con l'indirizzo assegnato tramite APIPA, un computer è limitato nella sua connettività, in quanto può comunicare solo con altri host della rete che stanno usando il range di indirizzi di APIPA. Ciò significa, per esempio, che questi computer non possono inviare/ricevere dati verso/da Internet.
APIPA è abilitato di default con l'installazione del Sistema Operativo; per verificare se il computer lo sta usando si può aprire una finestra Prompt dei comandi di Windows e dare il comando:

```
ipconfig /all
```

Se l'opzione "Configurazione automatica abilitata" (Autoconfiguration Enabled) è impostata a Sì il computer sta usando APIPA.
Lasciare APIPA attivato non è un problema, poiché è stato progettato per lavorare con client configurati per usare DHCP, quindi prima di intraprendere una qualunque azione, verifica la presenza di un DHCP Server, lasciando a questi il compito di assegnare l'indirizzo IP al computer.
Inoltre, se a un computer è stato assegnato un indirizzo statico, APIPA non ne riassegnerà un altro.

Si può disabilitare APIPA tramite il Registry Editor (*regedit.exe*): selezionare la voce **Parameters** e con il tasto destro scegliere **Nuovo → Valore DWORD** e chiamare il nuovo valore IPAutoconfigurationEnabled.
Verificare che il valore nella colonna Dati sia 0 e fare clic su **OK** (il sistema dovrà poi essere riavviato).

2.3 Impostazione del DHCP su un computer Linux

Facendo riferimento a un computer con installata la distribuzione **Ubuntu Desktop**, utilizziamo il suo **Network Manager** per gestire le connessioni di rete.
Per mettere un host (fisso o mobile) con Sistema Operativo Linux nelle condizioni di ricevere i parametri elencati, occorre seguire la seguente procedura:

- avviare Network Manager cliccando sull'icona delle connessioni di rete in alto a destra;
- selezionare la voce di interesse nel menu a tendina e cliccare sull'icona a forma di ingranaggio accanto alla connessione di rete (potrebbe essere wired o wireless) che si desidera configurare (FIGURA 4).
- selezionare la scheda IPv4 (FIGURA 5). Per default, questa scheda è impostata su Authomatic (DHCP), come Windows 10, e dà la possibilità di ricevere automaticamente anche il DNS e il gateway (Routers).

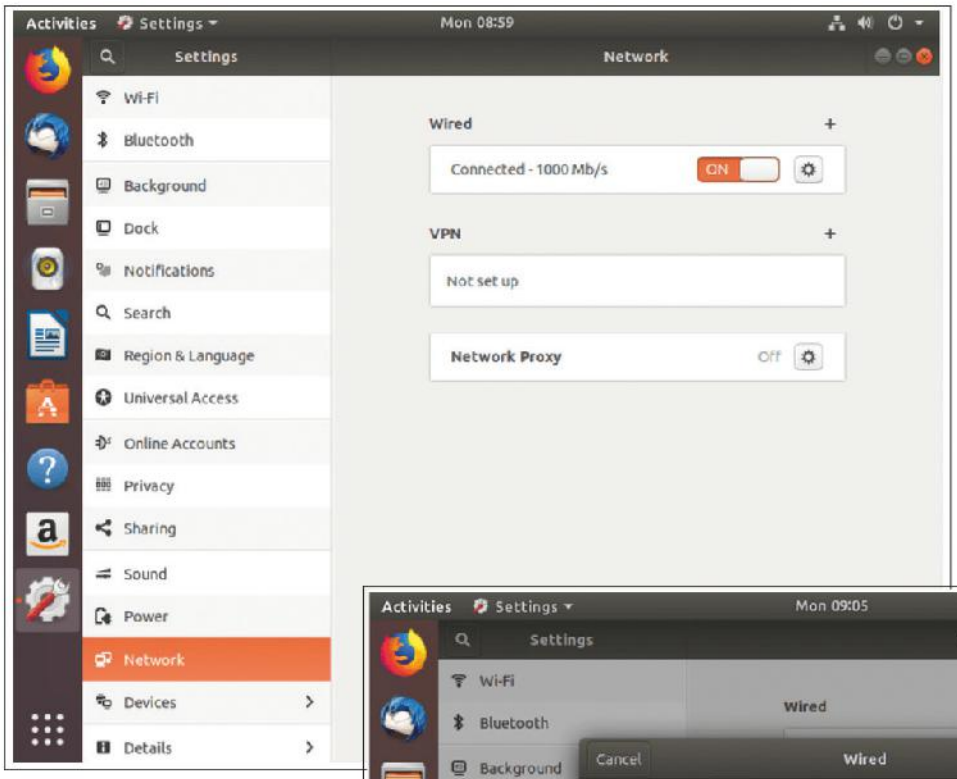


FIGURA 4 Selezione del Network

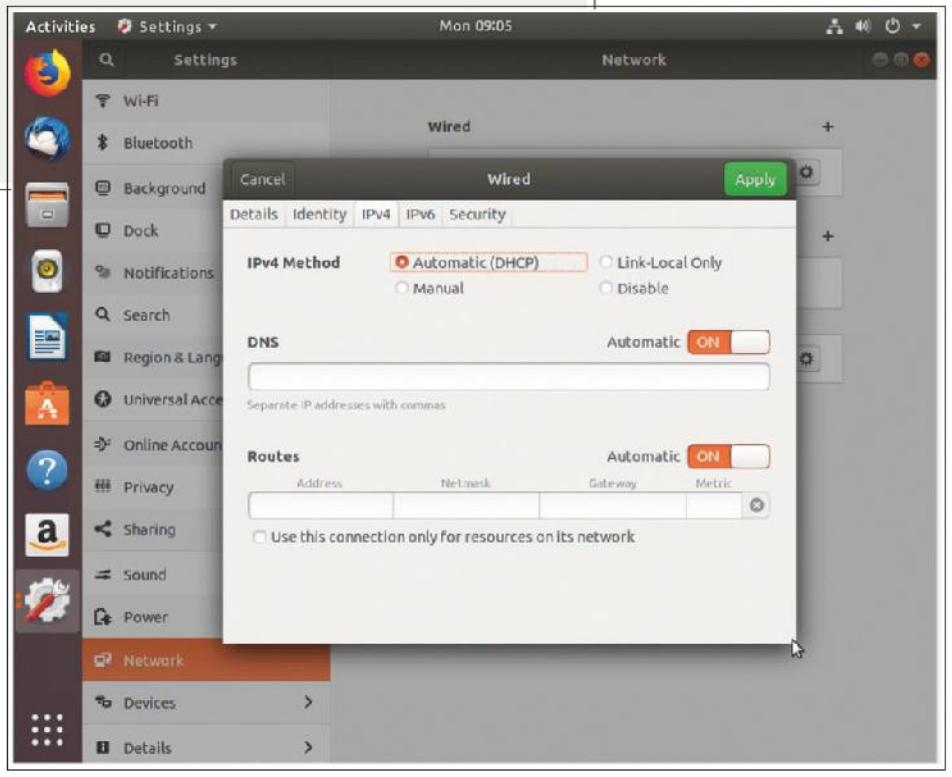


FIGURA 5 Automatic DHCP (default)

La configurazione di un computer Linux in LAN è ripresa e approfondita con una esercitazione disponibile sul Laboratorio online di questa Unità.

LABORATORIO ONLINE

CONFIGURAZIONE DI LINUX IN LAN

In questo laboratorio analizziamo i passaggi per configurare un computer Linux, con Network Manager oppure tramite i comandi da terminale.

FISSA LE CONOSCENZE

- Che cosa ha dato un grosso impulso alla nascita del DHCP?
- Che cos'è APIPA e quando entra in funzione?

3 L'ARCHITETTURA CLIENT/SERVER DHCP

3.1 Metodi di assegnazione degli indirizzi IP con DHCP

La FIGURA 6 mostra un semplice scenario di rete in cui viene usato DHCP per dispositivi fissi e mobili. Il DHCP Server utilizza un database nel quale sono memorizzati gli indirizzi IP a sua disposizione per l'allocazione ai DHCP Client presenti nella rete.

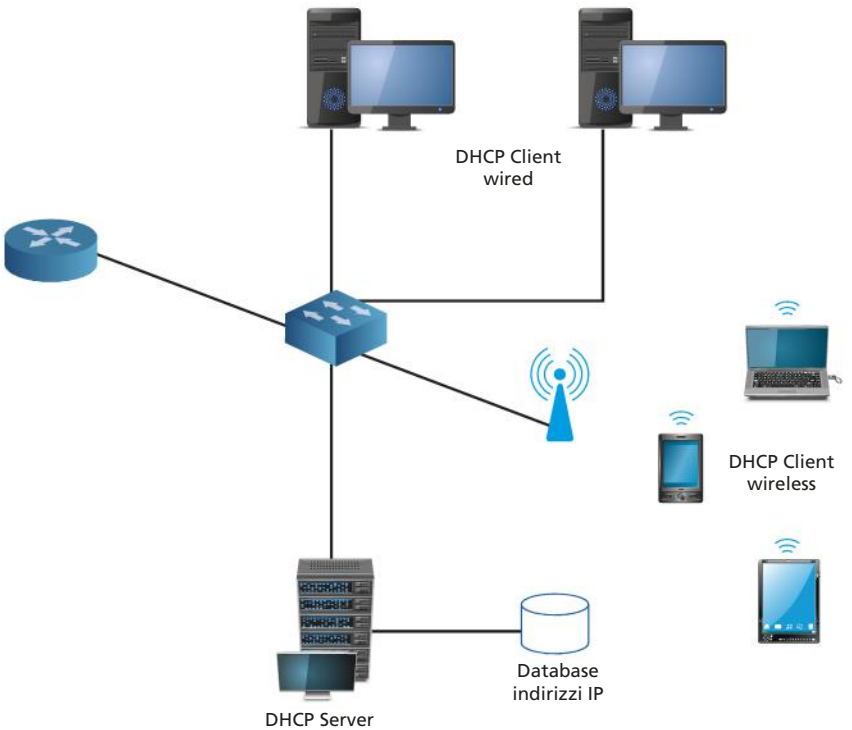


FIGURA 6 Scenario Client/Server DHCP

DHCP (come BOOTP) usa l'indirizzo fisico (MAC address) per identificare l'host. Il server utilizza questo dato e il **network address** della rete a cui l'host è connesso, per consultare il database e stabilire se assegnare all'host un indirizzo IP permanente o temporaneo.

Ogni volta che un host si connette a una rete, il suo DHCP Client richiede un indirizzo IP al DHCP Server, che lo sceglierà, in modo arbitrario, tra quelli disponibili nel suo **#address pool**.

Quando l'host lascerà la rete, il suo indirizzo ritornerà disponibile nel pool.

#techwords

Address pool è l'insieme degli indirizzi IP che il DHCP Server alloca dinamicamente. Solitamente è formato da blocchi contigui di indirizzi detti address range.
Un DHCP Server che gestisce gli indirizzi di più subnet avrà un address range per ciascuna di queste.

L'amministratore di rete può configurare, per ogni subnet e per ogni host, la modalità con cui il DHCP Server risponderà alle richieste dei client scegliendo fra 3 diversi tipi di configurazione.

- **Configurazione manuale**: è possibile assegnare un indirizzo IP specifico a un host, inserendolo manualmente nel DHCP Server; di regola si utilizza per macchine come router e server che si trovano stabilmente in una rete o per host che necessitano di un indirizzo permanente.

- **Configurazione automatica**: il DHCP Server assegna in modo automatico un indirizzo IP **permanente** a ogni host che si collega alla rete.
- **Configurazione dinamica**: il DHCP Server assegna un indirizzo IP a un host per un tempo limitato (**tempo di lease**) in base alla **#lease length policy** stabilita. Allo scadere del tempo di lease il client può richiederne il rinnovo o richiedere l'assegnazione di un nuovo indirizzo.

La configurazione dinamica presenta tutta una serie di **vantaggi**.

- **Automazione**: l'assegnazione dell'indirizzo IP al client avviene in modo automatico senza un intervento manuale dell'amministratore.
- **Gestione centralizzata**: l'amministratore esegue tutte le operazioni lavorando solo sul DHCP Server (aggiornamento degli indirizzi e degli altri dati di configurazione, modifica del tempo di lease, ecc.).
- **Condivisione e riutilizzo degli indirizzi**: gli host di una rete non sono sempre connessi tutti insieme nello stesso tempo, quindi la rete può supportare un numero di host superiore al numero di indirizzi disponibili (*condivisione*); nel momento in cui un host non è più connesso alla rete il suo indirizzo IP torna nel pool a disposizione di altri client (*riutilizzo*).
- **Portabilità**: non essendoci un assegnamento predefinito host-indirizzo IP, qualunque client che si connette alla rete può richiedere un indirizzo, supportando così la mobilità degli host.
- **Assenza di conflitti**: essendo possibile l'allocazione solo prelevando l'indirizzo IP dal database DHCP, non si può avere, per esempio, l'assegnazione di indirizzi duplicati.

L'unica vera **criticità** del protocollo DHCP è il fatto che prevede che un host riceva un indirizzo IP nuovo ogni qual volta effettui un accesso a una rete (o subnet) diversa. Nel caso di un utente mobile, che si sposta con continuità da una rete all'altra, questa modalità di gestione degli indirizzi IP comporta la chiusura di eventuali connessioni TCP aperte con un applicativo remoto.
In scenari di mobilità si usano protocolli come Mobile IP (descritto nel volume per il quinto anno) che permette al dispositivo mobile di usare lo stesso indirizzo IP permanente quando si muove da una rete all'altra.

#techwords

Lease length policy
L'amministratore stabilisce le politiche di durata del lease. Tipicamente, un tempo **lungo** è usato per gli host fissi, come i computer desktop, o che svolgono operazioni che richiedono l'attesa di risposte dalla rete; un tempo **breve** è indicato per un host che rimane collegato alla rete per poco tempo, come il portatile di un ospite.

3.2 DHCP Server di backup e router DHCP Relay Agent

Un solo DHCP Server è solitamente in grado di soddisfare le esigenze operative relative all'assegnazione degli indirizzi IP e al setting dei parametri di configurazione sui client della rete locale. In condizioni normali il carico che deriva da queste attività non è particolarmente pesante.

Un DHCP Server può anche essere configurato per servire più subnet, per due scopi:

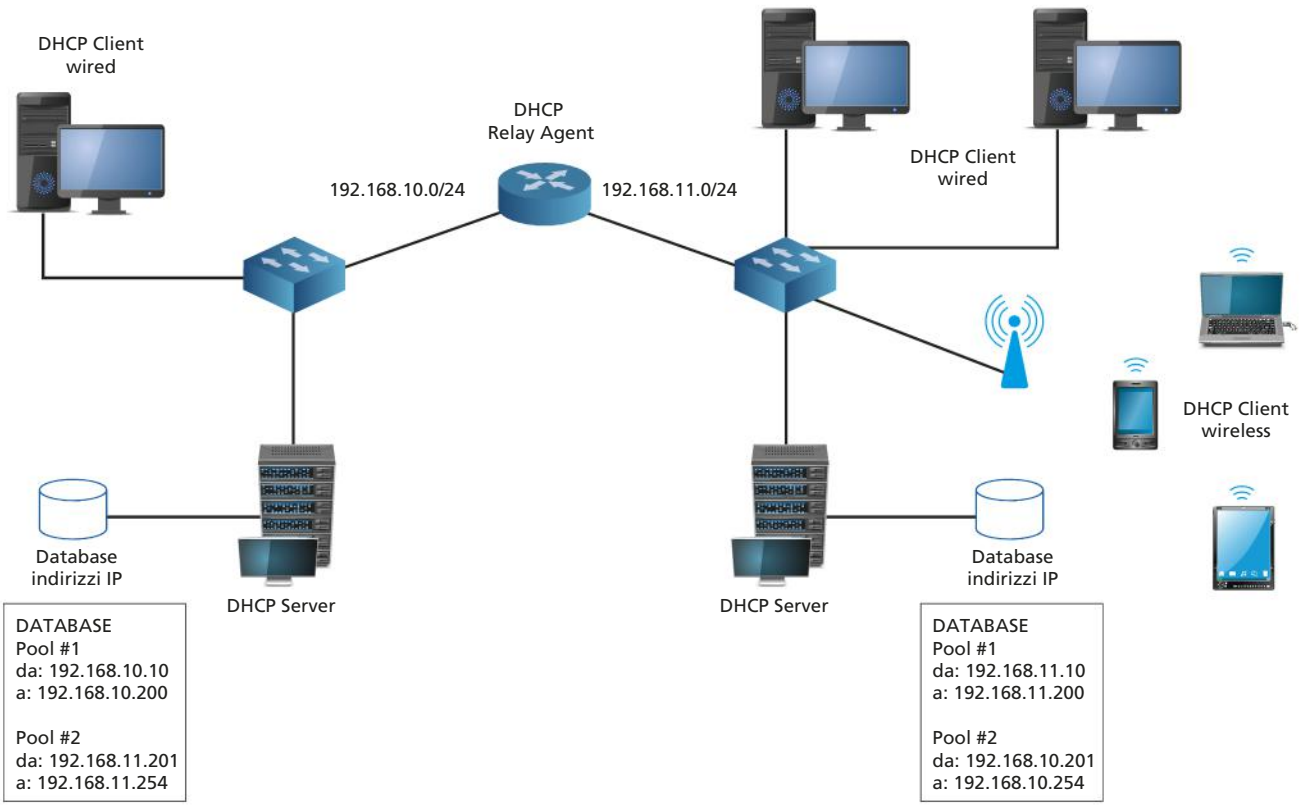
- **fault-tolerance**, per cui in genere è opportuno inserire un secondo DHCP Server come backup, così da mantenere il servizio sempre attivo;
- **bilanciamento del carico di lavoro**, così da diminuire i tempi di risposta del server. In tal caso, alla richiesta di un client per l'assegnazione di un indirizzo IP possono rispondere più di un DHCP Server.

La tipica configurazione con più DHCP Server prevede che il 75% degli IP di una subnet sia su un server primario e il 25% su uno secondario, collocato in una subnet diversa. È poi possibile assegnare al server primario il 25% degli indirizzi del server secondario, in modo che i due server siano di backup uno per l'altro.

Lo standard DHCP non specifica una modalità di colloquio tra i server per evitare che si generino conflitti nell'assegnazione degli indirizzi. La realizzazione di meccanismi che evitino l'assegnazione di indirizzi duplicati viene lasciata alle implementazioni commerciali dei server.

Quando un DHCP Server è responsabile dell'indirizzamento su una subnet diversa dalla propria è necessario introdurre un **relay agent** (agente di ritrasmissione), ossia una macchina che non è né un server né un client, ma svolge un ruolo di intermediario occupandosi di facilitare la comunicazione tra client e server attraverso più reti (FIGURA 7).

FIGURA 7 Scenario con 2 DHCP Server e un router DHCP Relay Agent



Nel corso del quinto anno vedremo come configurare il servizio DCHP su Windows Server.

FISSA LE CONOSCENZE

- Quali sono i 3 modi con cui si può configurare il DHCP e in cosa differiscono?
- Dove sono memorizzati i dati di configurazione che il DHCP Server invia al DHCP Client?
- In cosa consiste il backoff reciproco tra i DHCP Server?
- A cosa serve il router DHCP Relay Agent?

4 LA COMUNICAZIONE TRA DHCP CLIENT E DHCP SERVER

4.1 Il formato dei messaggi DHCP

Come per qualunque applicazione Client/Server, i messaggi scambiati tra DHCP Client e DHCP Server sono di due tipi: richieste (**request**) e risposte (**reply**).
Il protocollo di trasporto usato è **UDP**, come in BOOTP, e utilizza le stesse porte: **67** per il server e **68** per il client.

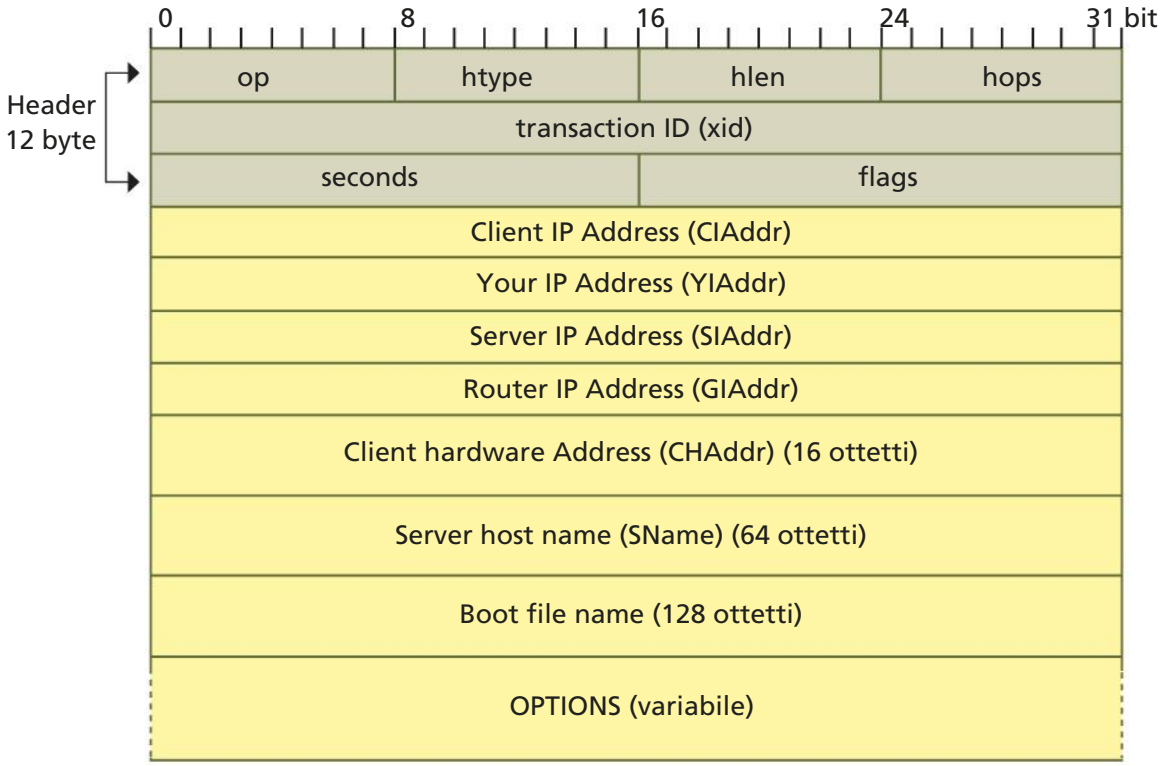


FIGURA 8 Il formato delle PDU DHCP

Vediamo in dettaglio i vari campi del pacchetto DHCP **(FIGURA 8)**:

- **op** (operation code): specifica se il messaggio è request (1) o reply (2);
- **htype** (hardware type): indica il tipo di hardware usato nella rete locale (per esempio 1 per Ethernet 10 Mbps, 2 per le reti IEEE 802, 20 per linea seriale);
- **hlen** (hardware address length): indica la lunghezza (in byte) dell'indirizzo fisico contenuto nel messaggio, per esempio per le reti IEEE 802 che usano il MAC address il valore è 6;
- **hops**: è usato dai DHCP Relay Agent per controllare l'inoltro dei messaggi DHCP, all'inizio della comunicazione è impostato a 0 dal client;
- **transaction ID** (xid): è un numero casuale scelto dal client che permette di associare la risposta alla richiesta;
- **seconds**: indica il numero di secondi trascorsi da quando il client ha iniziato il processo di acquisizione o rinnovo del tempo di lease;

- **flags**: nello standard è stato definito un solo bit di flag, il più significativo, B = broadcast; se impostato a 1 significa che il client desidera ricevere la risposta con l'indirizzo IP di broadcast;
- **Client IP Address** (CIAddr): questo campo contiene un valore, l'indirizzo IP, solo se il client è nello stato di BOUND, RENEW, REBIND, mentre negli altri stati il client sta acquisendo un indirizzo IP, quindi questi bit sono posti a 0;
- **Your IP Address** (YIAddr): contiene l'indirizzo IP che il server ha assegnato al client;
- **Server IP Address** (SIAddr): contiene l'indirizzo IP del server, quello che il client deve usare per il completamento del processo di bootstrap, che potrebbe anche non coincidere con l'indirizzo del server che ha mandato la risposta; in un messaggio di reply l'indirizzo del server è inserito nell'opzione Server Identification (nel campo Option);
- **Router IP Address** (GIAddr, Gateway IP Address): questo campo contiene l'indirizzo del router con funzioni di DHCP Relay Agent, da usare quando client e server si trovano in reti diverse;
- **Client Hardware Address** (CHAddr): è l'indirizzo fisico del client, usato per la sua identificazione;
- **Server host name** (SName): è un campo opzionale, il server che invia un messaggio DHCP reply può inserirvi il suo nome che può anche essere un nome di dominio (per esempio: *dip1.azienda.com*);
- **Boot file name**: è usato dal server per specificare il nome completo del file (directory e nome) che contiene i dati di configurazione utili nella fase di bootstrap del client;
- **Options**: è un campo che può essere usato sia dal client sia dal server e contiene molti parametri necessari allo svolgimento delle operazioni di base del DHCP (tipicamente quelle relative al lease dell'indirizzo IP dinamico).

Ogni parametro specificato nel campo Options è codificato con 3 campi: il primo ottetto contiene il **codice**, il secondo ottetto la **lunghezza** (in termini di ottetti) del successivo campo **valore**, che contiene il parametro specificato.

Vediamo alcuni esempi:

- per specificare il **tipo di messaggio** DHCP si codifica la relativa opzione in 3 ottetti, come segue:

l valori da 1 a 8 indicano il tipo di messaggio inviato:

1	DHCPDISCOVER
2	DHCPOFFER
3	DHCPREQUEST
4	DHCPDECLINE
5	DHCPACK
6	DHCPNACK
7	DHCPRELEASE
8	DHCPINFORM

- per specificare la **subnet mask** si codifica la relativa opzione in sei ottetti, come segue:

codice (1)	lungh.(4)	*mask1*	*mask2*	*mask3*	*mask4*

- per specificare il **nome di dominio** (per esempio *scuola.it*) si codifica la relativa opzione in più ottetti, come segue:

codice (15)	lungh.(N)	*dn1*	*dn2*	...	*dnN*

4.2 Le 4 fasi del DHCP per l'assegnazione dell'indirizzo IP

La comunicazione tra il nuovo host e il server, al fine di ottenere i dati per la configurazione di rete, avviene in 4 fasi.

1. **Ricerca del DHCP Server**: l'host deve per prima cosa trovare un DHCP Server al quale inviare la richiesta, quindi invia in broadcast sulla rete locale (indirizzo IP di destinazione uguale a 255.255.255.255) un messaggio **DHCPDISCOVER** con porta di destinazione UDP 67.
2. **Offerta al DHCP Client**: i DHCP Server presenti nella rete rispondono con un messaggio **DHCPOFFER**, inviato anch'esso in broadcast, con l'indirizzo IP offerto e il tempo di lease.
3. **Richiesta al DHCP Server**: il client sceglie una delle offerte ricevute e invia al relativo server un messaggio **DHCPREQUEST** con i parametri di configurazione presenti nel messaggio DHCPOFFER; gli altri DHCP Server, che avevano inviato un'offerta di lease, rendono nuovamente disponibile l'indirizzo IP nel proprio address pool.
4. **Conferma al DHCP Client**: il server risponde con un messaggio **DHCPACK**, confermando i parametri di configurazione.

Completata questa procedura, l'host può utilizzare l'IP assegnato per tutta la durata del tempo di lease. La FIGURA 9 mostra lo scambio dei messaggi tra client e server relativi all'offerta del tempo di lease.
La FIGURA 10 mostra lo scambio dei messaggi relativi alla selezione del lease e all'accettazione dell'offerta del server.

IN ENGLISH PLEASE

DHCPDISCOVER: sent by a client to start the IP address lease process.

DHCPOFFER: returned by a DHCP Server offering a valid IP address lease to a requesting client.

DHCPREQUEST: sent by a client to accept a DHCP lease offer and decline other server offers.

DHCPACK: returned by a DHCP Server to acknowledge the client's acceptance of an address lease that includes the valid address lease and, possibly, optional TCP/IP configuration settings.

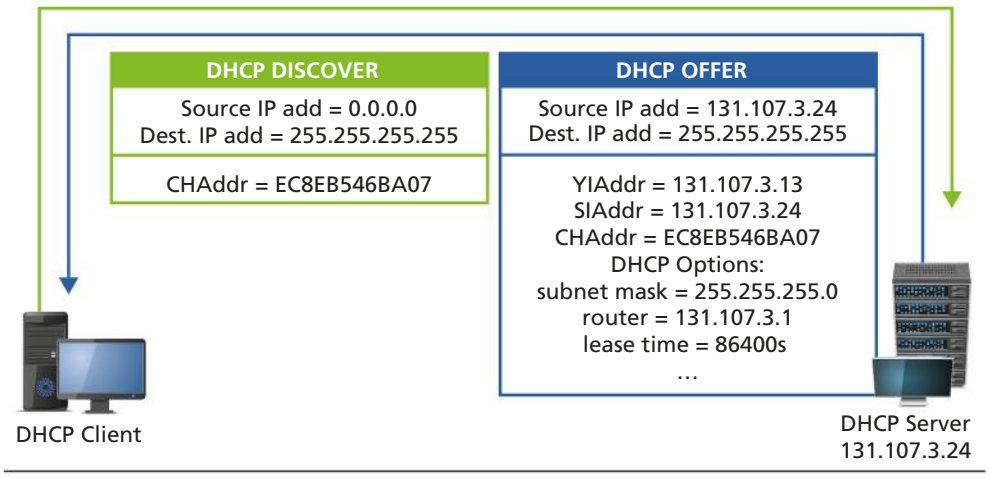


FIGURA 9 DHCP Discover e DHCP Offer

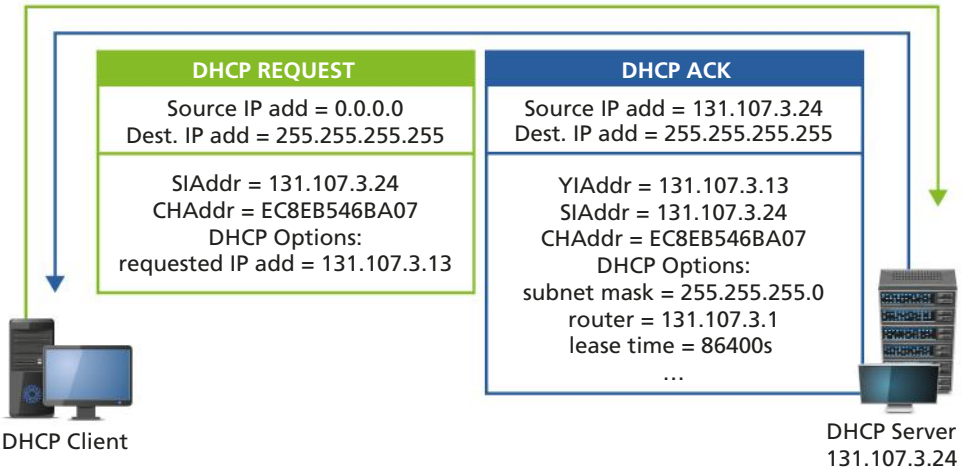


FIGURA 10 DHCP Request e DHCP Ack

#prendinota

DORA = Discovery, Offer, Request, Acknowledgement. È l'acronimo usato per indicare le 4 fasi dell'assegnazione dinamica con DHCP.

4.3 Gli stati del DHCP Client

La specifica di DHCP usa una **macchina a stati finiti** (**FSM**) per descrivere il ciclo di vita del lease dell'indirizzo IP dal punto di vista del client.

La **FIGURA 11** mostra i 6 stati principali di un DHCP Client e le transizioni tra essi. In figura sono indicati i messaggi in ingresso, con l'evento che causa la transizione, e i messaggi in uscita preceduti dal simbolo /.
All'inizio il DHCP Client è nello stato **INITIALIZE**. Dopo l'invio in broadcast del messaggio DHCPDISCOVER, assume lo stato **SELECT**.

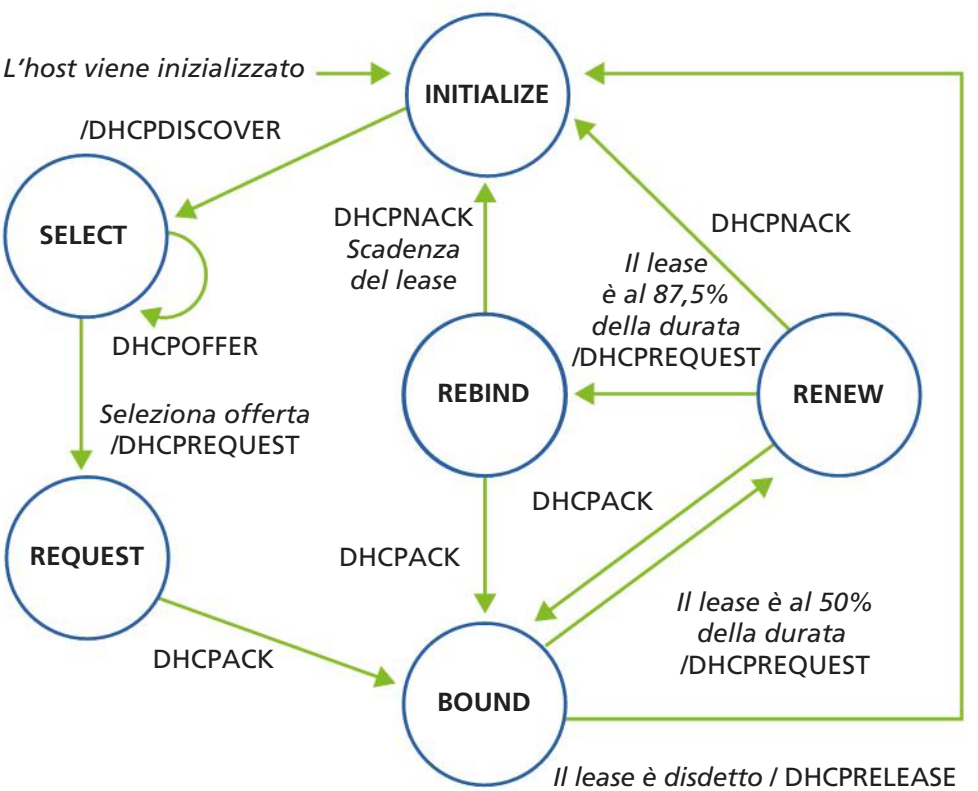


FIGURA 11 La macchina a stati finiti (FSM) del DHCP Client

IN ENGLISH PLEASE

Finite State Machine (FSM). The protocol's specific behavior is illustrated by showing the different states a device can be in, what possible transitions exist from one state to another, what events cause transitions to occur and what actions are performed in response to an event.

Quando è nello stato SELECT, il client raccoglie le risposte DHCPOFFER dai DHCP Server. Il client sceglie una delle offerte ricevute inviando al server un messaggio DHCPREQUEST. A questo punto il client passa allo stato **REQUEST**.

La ricezione del messaggio DHCPACK inviato dal server selezionato, fa in modo che il client passi allo stato **BOUND**, in cui usa l'indirizzo assegnatogli, memorizza i parametri di configurazione che il server gli ha inviato e imposta due timer:

1. **renewal timer** (**T1**): indica al client quando è il momento di effettuare una richiesta di rinnovo dell'indirizzo;
2. **rebinding timer** (**T2**): dice al client che è scaduto il tempo per rinnovare l'indirizzo IP.

Ci sono casi in cui il lease viene disdetto prima dello scadere dei timer, per esempio su richiesta dell'utente dell'host, con un esplicito messaggio DHCPRELEASE. Dallo stato BOUND allora il client ritorna nello stato INITIALIZE.

4.4 Il rinnovo del lease dell'indirizzo IP

Quando il DHCP Client entra nello stato BOUND, imposta i timer che controllano il rinnovo, la conferma e la scadenza del lease. Il valore a cui impostare questi timer può essere specificato dal DHCP Server oppure il client può usare quelli predefiniti.
Il valore predefinito del **renewal timer** è la metà del tempo totale del lease; quando scade, il client passa nello stato RENEW e invia un messaggio DHCPREQUEST al server da cui ha ottenuto il lease.

Il server risponde in due modi:

1. comunicando al client di interrompere l'uso dell'indirizzo, inviando un messaggio DHCPNACK; in questo caso il client ritorna nello stato INITIALIZE;
2. concedendo ancora l'utilizzo del lease, inviando al client un messaggio DHCPACK che lo fa tornare nello stato BOUND reimpostando i timer.

Quando il client passa nello stato RENEW utilizza il **rebinding timer** per gestire il caso in cui il server, al quale ha inviato la richiesta di rinnovo, non possa rispondere, perché è diventato inattivo o non è più raggiungibile. Questo timer è impostato quando il client passa nello stato BOUND e il valore predefinito è l'87,5% del periodo di lease. Quando scade questo timer il client passa nello stato **REBIND** e invia in broadcast sulla rete locale un messaggio DHCPREQUEST.

A questo punto si possono avere 3 casi:

1. un server della rete risponde positivamente con un messaggio DHCPACK e il lease dell'indirizzo è prorogato, il client ritorna nello stato BOUND e imposta nuovamente i timer;
2. un server della rete risponde negativamente con un messaggio DHCPNACK, il client deve smettere di usare l'indirizzo IP e passare nello stato INITIALIZE per richiederne uno nuovo;
3. nessun server risponde alla richiesta del client (caso raro), scade il periodo di lease e torna allo stato INITIALIZE.

Vediamo un esempio di rilascio e rinnovo dell'indirizzo IP con Windows e con Linux.

WINDOWS

Per cancellare le impostazioni TCP/IP configurate sul computer, tra cui l'indirizzo IP in lease, scrivere nella finestra del **Prompt dei comandi**:

```
ipconfig /release
```

dopo l'esecuzione di questo comando sia l'indirizzo IP che la subnet mask saranno: 0.0.0.0.

Per ottenere un nuovo indirizzo, scrivere il comando:

```
ipconfig /renew
```

che dà inizio al processo di acquisizione di un nuovo indirizzo e che porterà alla configurazione dei parametri TCP/IP cancellati prima.

LINUX

Da terminale, si usa il comando: dhclient (package *dhclient*) o il comando: ip (package *iproute2*), da eseguire come **utente root**.
Il comando ip sostituisce il comando ifconfig (si confrontino i vecchi comandi del package *net-tools* e i nuovi comandi della *iproute2* suite).
Nei comandi seguenti [interface] è da sostituire con il nome di un'interfaccia di rete, per esempio Eth0.

Se il client non vuole più usare l'indirizzo, lo segnala al server con il comando:

```
dhclient –r [interface]
```

(dove -r significa *release*) oppure rende inattiva l'interfaccia di rete:

```
ip link set dev [interface] down
```

Per ottenere un nuovo indirizzo per la scheda di rete, si scrive il comando:

```
dhclient [interface]
```

oppure si rende nuovamente attiva l'interfaccia di rete:

```
ip link set dev [interface] up
```

FISSA LE CONOSCENZE

- Descrivi come sono codificati i parametri contenuti nel campo Options della PDU DHCP.
- In quali casi il DHCP Client ritorna allo stato INITIALIZE?
- Spiega il significato dei timer T1 e T2 e il loro utilizzo.
- Descrivi come un client può rinnovare il lease del suo indirizzo IP.

5 IL DHCP PER IPv6

5.1 Il DHCPv6

La definizione della versione 6 del protocollo IP (IPv6, vista nell'Unità 4) ha ovviamente avuto impatto anche sul protocollo DHCP che lavora con le informazioni di configurazione degli host, tra le quali la più importante è proprio l'indirizzo di rete. Di conseguenza, anche di questo protocollo è stata definita una nuova versione denominata DHCP for IPv6 (spesso abbreviata in **DHCPv6**).
Gli host IPv6 possono essere configurati in due modi, a seconda delle caratteristiche della rete in cui si trovano:

1. **stateless autoconfiguration**: l'host si configura autonomamente senza bisogno di aiuto da parte di altre macchine. Questo meccanismo è definito in RFC 4862;
2. **stateful autoconfiguration**: un server fornisce all'host le informazioni di configurazione. Si tratta del tradizionale metodo usato con il protocollo DHCP ed è descritto in RFC 8415 (DHCPv6).

DHCPv6 (modalità *stateful*) è usato quando si vuole avere un controllo centralizzato degli host della rete, quindi abitualmente nelle reti aziendali, mentre il metodo *stateless* è preferibile dove non c'è una gestione centrale, per esempio nelle reti domestiche. Nonostante ciò, anche gli host che usano l'autoconfigurazione stateless possono aver bisogno di parametri, necessari alla loro configurazione, da richiedere tramite DHCPv6.

#prendinota

DHCPv6 opera in modo simile a DHCPv4, ma il protocollo è stato completamente riscritto, abbandonando la dipendenza da BOOTP, di cui conserva solo la terminologia utilizzata.

DHCPv6 usa ancora UDP come protocollo di trasporto, ma con diversi numeri di porta:

- **UDP 546** è il numero di porta per i DHCPv6 Client;
- **UDP 547** è il numero di porta per i DHCPv6 Server.

Questo lo rende non compatibile con gli host che usano IPv4.
In IPv4, DHCP si basava sull'indirizzo MAC, partendo dal presupposto che un host avesse una sola scheda di rete. Ormai la maggior parte dei computer ha più interfacce IP, e per esempio un laptop può usare il protocollo IP allo stesso tempo su Bluetooth, WLAN ed Ethernet. Pertanto in DHCPv6 è cambiata la modalità di identificare un host, sia esso un client o un server.
Si usa cioè un identificatore univoco, denominato **DHCP Unique Identifier** (**DUID**), per l'host e un insieme di identificatori per le sue interfacce di rete, denominati **Identity Association Identifier** (**IAID**).
Il DUID viene generato una volta sola dal Sistema Operativo e memorizzato in modo permanente.

In Windows è possibile leggere il valore del DUID con il comando ipconfig /all. Oltre al DUID è visualizzato anche lo IAID dell'interfaccia di rete.

La FIGURA 12 mostra un esempio di output di tale comando.

```
Scheda LAN wireless Wi-Fi:

   Suffisso DNS specifico per connessione: station
   Descrizione . . . . . . . . . . . . . : Intel(R) Dual Band Wireless-AC 3165
   Indirizzo fisico. . . . . . . . . . . : B8-81-98-36-66-9F
   DHCP abilitato. . . . . . . . . . . . : Sì
   Configurazione automatica abilitata   : Sì
   Indirizzo IPv6 locale rispetto al collegamento . : fe80::6809:4b90:3cdc:246f%4
   Indirizzo IPv4. . . . . . . . . . . . : 192.168.1.9(Preferenziale)
   Subnet mask . . . . . . . . . . . . . : 255.255.255.0
   Lease ottenuto. . . . . . . . . . . . : sabato 25 novembre 2017 20:54:03
   Scadenza lease . . . . . . . . . . . . : domenica 26 novembre 2017 01:54:04
   Gateway predefinito . . . . . . . . . : 192.168.1.1
   Server DHCP . . . . . . . . . . . . . : 192.168.1.1
   IAID DHCPv6 . . . . . . . . . . . . . : 79200664
   DUID Client DHCPv6. . . . . . . . . . : 00-01-00-01-1F-22-F3-00-EC-8E-B5-46-EA-33
   Server DNS . . . . . . . . . . . . . . : 192.168.1.1
   NetBIOS su TCP/IP . . . . . . . . . . : Attivato
```

FIGURA 12 Output del comando ipconfig /all per un'interfaccia wireless

5.2 Comunicazione tra DHCPv6 Client e Server

La comunicazione tra il Client e il Server, con il protocollo DHCPv6, può avvenire con due diverse modalità:

- **Modalità a 2 messaggi**
  Questa modalità si usa quando il client non ha bisogno di aver assegnato l'indirizzo IP, ma necessita di altre informazioni di configurazione (per esempio l'elenco dei DNS Server attivi). Il client invia un messaggio **Information-Request** e il server risponde con un messaggio **Reply** contenente le informazioni richieste.
- **Modalità a 4 messaggi**
  Si usa se il client necessita di uno o più indirizzi IPv6:
  – il client invia un messaggio **Solicit** (corrisponde al messaggio DHCPDISCOVER in DHCPv4) all'indirizzo ff02::1:2 (formato usato per gli indirizzi IPv6 multicast) per trovare i server disponibili;
  – i server che possono soddisfare la sua richiesta rispondono con un messaggio **Advertise** (corrisponde al messaggio DHCPOFFER in DHCPv4);
  – il client sceglie uno di questi server, tipicamente il primo da cui ha ricevuto risposta presupponendo che sia il più vicino, quindi invia al server selezionato un messaggio **Request** (corrisponde al messaggio DHCPREQUEST in DHCPv4) chiedendo conferma dell'assegnazione dell'indirizzo e altre informazioni di configurazione;
  – il server risponde con un messaggio **Reply** (corrisponde al messaggio DHCPACK/DHCPNACK in DHCPv4) contenente indirizzi e informazioni.

  Prima dello scadere del tempo di lease, il client può inviare un messaggio **Renew** (corrisponde al messaggio DHCPREQUEST in DHCPv4) al server per chiedere l'estensione del periodo di lease degli indirizzi.

In FIGURA 13 è mostrato lo scambio di messaggi tra DHCPv6 Server e Client.

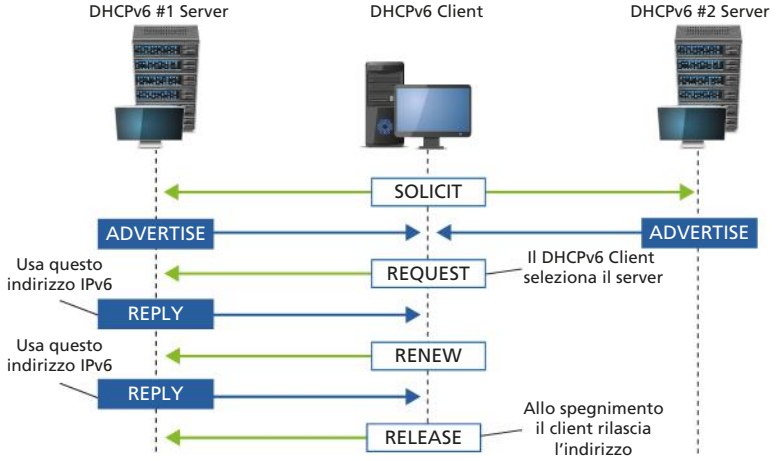


FIGURA 13 Scambio di messaggi tra DHCPv6 Client e Server

FISSA LE CONOSCENZE

- Qual è la differenza tra una autoconfigurazione stateful e una stateless?
- Descrivi la modalità di comunicazione tra DHCPv6 Client e Server a 4 messaggi.
- Per il DHCPv6 Client ricevere più messaggi Advertise è una condizione anomala?
- Che tipo di indirizzo è ff02::1:2?

6 IL DNS (DOMAIN NAME SYSTEM)

6.1 La risoluzione dei nomi

L'applicazione **DNS (Domain Name System)** consente agli utenti della rete di usare dei nomi per identificare un computer con funzioni di server al posto del suo indirizzo IP. Le specifiche del DNS si trovano negli RFC 1034 e RFC 1035 e successivi aggiornamenti. Questo sistema è basato su un **database distribuito**, organizzato gerarchicamente e che segue il modello Client/Server.

Il DNS è formato da 3 componenti principali.

- **Domain Name Space**, specifica la struttura ad albero dei nomi di dominio. Il Name Space risulta diviso in 3 tipi di domini:
  1. **domini radice**: sono i domini di primo livello (Top Level Domain, TLD);
  2. **domini intermedi**: sono domini che hanno a loro volta dei sottodomini;
  3. **domini foglia**: sono domini privi di sottodomini e contengono solo host.
- **Name Server**, è un processo applicativo con il ruolo di server che contiene informazioni su alcune parti del Name Space chiamate **zone**. Il Name Server costituisce una *authority* per tali zone (viene detto *authoritative Name Server*). Un Name Server contiene anche i puntatori ad altri Name Server che possono essere usati per ricavare informazioni su altre zone. La zona dei TLD è detta **root zone** e i Name Server che ne rispondono sono i **root Name Server**. Questi conoscono anche gli indirizzi degli authoritative Name Server per ciascun dominio TLD.
- **Resolver**, è un programma con il ruolo di client che ottiene informazioni dal Name Server. Tipicamente viene realizzato da procedure del Sistema Operativo. Per esempio in Unix per accedere al resolver si richiamano le routine gethostbyname e gethostbyaddr.

#prendinota

La gestione dei nomi di dominio di primo livello è effettuata dall'organizzazione **IANA** (*Internet Assigned Numbers Authority*).
Un elenco aggiornato dei Top Level Domain si trova all'indirizzo: *www.iana.org/domains root/db/*.
IANA collabora con **ICANN** (*Internet Corporation for Assigned Names and Numbers*) l'ente che gestisce il DNS a livello mondiale e coordina le attività dei root Name Server.

Il sistema DNS è usato anche all'interno delle reti locali private per risolvere i nomi dei computer (hostname). Infatti, grazie al DNS, si possono associare alle macchine dei nomi facili da ricordare; i nomi possono rimanere gli stessi anche se cambia l'indirizzo IP; gli utenti possono connettersi ai server locali usando le stesse convenzioni usate su Internet (URL).

Per comporre il nome completo di un dominio si percorre il cammino dalla foglia (che rappresenta l'host) alla radice, rappresentata da un punto (.), separando le varie componenti con un punto.
Per esempio: www.ietf.org è il nome di dominio dell'host che offre il servizio web per l'organizzazione IETF (Internet Engineering Task Force). Infatti www è il nome del server web che si trova nel dominio *ietf* che a sua volta è contenuto nel dominio *org* il quale discende dal dominio radice.
Da notare che per il DNS i nomi "www.ietf.org" e "www.ietf.org." sono uguali in quanto il punto finale (che indica il dominio radice) è implicito in ogni nome di dominio, quindi può essere omesso.

Valgono poi le seguenti regole:

- i nomi delle singole componenti del cammino completo non devono superare i 63 caratteri, inclusi i punti (sono preferibili i nomi facili da ricordare);

- un cammino completo non deve superare i 255 caratteri, quindi meglio limitare il numero di livelli usati (non più di 5) e scegliere dei nomi brevi, per non renderne troppo complessa l'amministrazione;
- i nomi sono *case-unsensitive*, quindi è indifferente scrivere .it oppure .IT, perché entrambi individuano lo stesso dominio;
- ogni dominio controlla i suoi sottodomini, quindi se si vuole creare un nuovo sottodominio è necessario il permesso del dominio *padre*, mentre non si deve chiedere alcun permesso ai livelli superiori dell'albero.

6.2 Il formato dei pacchetti DNS

Il formato dei pacchetti DNS consente a un client di porre più richieste (**query**) in un singolo messaggio (**DNS request**). Ogni query consiste nel nome di dominio del quale il client cerca l'indirizzo IP e il tipo di oggetto desiderato (per esempio l'indirizzo). Il server risponde restituendo un messaggio simile (**DNS reply**) che contiene le risposte (**answer**) alle query. Se non può soddisfare tutte le query, il server indica nel messaggio di risposta altri Name Server che il client può contattare per ottenere le risposte. In FIGURA 14 è mostrato il formato delle PDU DNS, dove RR indica il **Resource Record.**

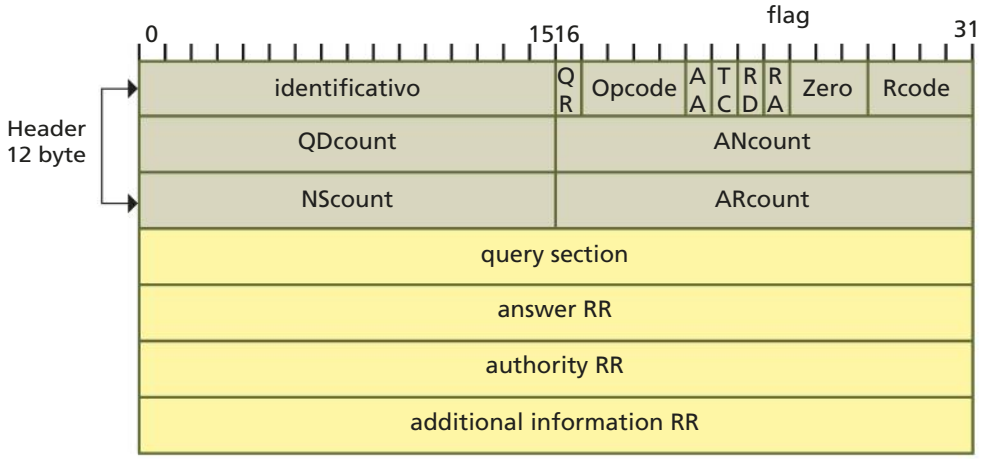


FIGURA 14 Formato delle PDU DNS

Il pacchetto è formato da una parte di header fissa di 12 byte e da 4 campi a lunghezza variabile.

I campi hanno il seguente significato:

- **identificativo**: è deciso dal client e inserito nelle risposte del server, permette di far corrispondere le reply alle request;
- **flag**: è un campo a 16 bit così suddiviso:
  - **QR** indica se il messaggio è una query (0) o una risposta (1);
  - **Opcode** indica il tipo di request:
    0 = Query Standard
    1 = Query Inversa
    2 = Richiesta di Stato del Server
    dal 3 al 15 non sono attualmente utilizzati;
  - **AA** se vale 1 indica una risposta authoritative del server;
  - **TC** se vale 1 indica che la reply eccedeva i 512 byte e il messaggio è stato troncato;

- **RD** se vale 1 indica che si desidera una ricerca ricorsiva, altrimenti la ricerca sarà iterativa, è impostato dal **#local resolver**;
- **RA** Ricorsione Disponibile (Available), impostato dal server;
- **Zero** deve essere 0, è riservato per usi futuri;
- **RCod** Codice di ritorno:
  0 = Nessun errore
  1 = Errore nella costruzione della query
  2 = Errore interno nel server dei nomi
  3 = Ricevuto da un server di autorità, indica che il nome specificato nella query non esiste nel dominio
  4 = Il server dei nomi non implementa quel tipo di query
  5 = Il server si rifiuta di eseguire l'operazione per motivi di impostazioni
  Dal 6 al 15 sono riservati per usi futuri;
- i 4 campi successivi specificano il numero di occorrenze presenti in ciascuno dei 4 campi che si trovano dopo l'header (query, answer, authority, additional information). Per esempio, per una query, **QDcount** è di solito 1 e i contatori degli altri campi sono 0; per una answer, **ANcount** è almeno 1, mentre gli altri contatori possono essere 0 o maggiori;

- **query section**: di solito in questa sezione si trova la domanda; contiene un numero di entry, con nome, tipo e classe della query, pari al valore QDcount specificato;
- le 3 sezioni successive sono tutte costituite nello stesso modo: ogni sezione ha un numero variabile di Resource Record.

#techwords

Local resolver
È l'applicazione client che risiede sull'host e, a fronte di un nome, restituisce un valore, solitamente un indirizzo IP.

6.3 I Resource Record (RR)

Ogni dominio, o meglio, ogni zona mantiene le informazioni in strutture dette **Resource Record** (letteralmente: descrittore di risorsa).
L'uso più frequente di queste strutture è per ottenere un indirizzo IP: dato il nome di un host, il DNS trova il Resource Record che mantiene l'associazione tra quel nome e l'indirizzo IP.
In realtà, questi record, come si vedrà in seguito, possono contenere altri dati oltre all'indirizzo IP.

Name	Type	Class	TTL	RDLength	RData

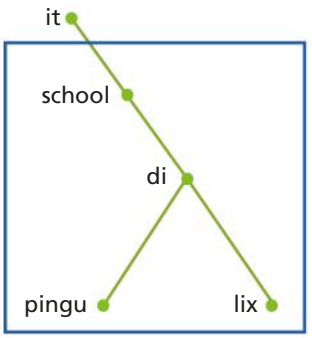
FIGURA 15 Tracciato di un Resource Record

La **FIGURA 15** mostra il formato di un Resource Record, definito in RFC 1035, in cui:

- **Name** (DN, Domain Name) è il nome del dominio a cui il record appartiene;
- **Type** identifica il tipo di informazione contenuta nel campo RData;
- **Class** indica se le informazioni del record fanno riferimento a Internet o ad altro. La classe Internet è indicata con IN;
- **TTL** (Time To Live) indica la stabilità del record; più il valore è alto, più il record è stabile e sarà memorizzato nella cache del DNS. Un valore zero indica che il RR non deve essere memorizzato nella cache (per esempio i RR SOA sono sempre distribuiti con TTL = 0);
- **RDLength** specifica la lunghezza in ottetti del campo RData;
- **RData** è il valore restituito dal DNS, può contenere un numero, una stringa ASCII o un nome di dominio, dipende da quanto scritto nel campo Type.

Alcuni tipi di Resource Record sono diventati obsoleti, elenchiamo quindi di seguito solo quelli attualmente utilizzati.
Per gli esempi riportati in ogni voce dell'elenco, si fa riferimento alla parte di albero evidenziata in FIGURA 16 dove, con nomi del tutto inventati, il Domain Name è una macchina (chiamata pingu) del dipartimento di informatica (di) di una scuola (school) che si trova in Italia (it).

FIGURA 16 Tracciato di un Resource Record



Contenuto del campo Type (identifica il tipo di informazione contenuta nel campo RData):

- **A** (Address): indica che nel campo RData si trova l'indirizzo IPv4 dell'host, quindi un numero binario di 32 bit.
  Per esempio:

  ```
  pingu.di.school.it. A IN 86400 198.45.30.165
  ```

  indica che l'host con DN = pingu.di.school.it. ha indirizzo IP 198.45.30.165 e questa informazione è stabile (TTL = 86.400 secondi è un valore alto).
- **AAAA**: come A ma riferito a indirizzi IPv6; questo tipo è stato introdotto in seguito nell'RFC 1886.
- **CNAME** (Canonical NAME): indica un nome di dominio e viene tipicamente usato per definire degli *alias*.
  Per esempio:

  ```
  www.di.school.it. CNAME IN pingu.di.school.it.
  ```

  definisce www.di.school.it. come un alias per l'host il cui nome canonico (cioè standard) è pingu.di.school.it. Quindi se in futuro l'host pingu verrà cambiato e il nome di dominio del nuovo web server sarà lix.di.school.it., si modificherà l'informazione nel Resource Record, ma non sarà necessario cambiare anche l'indirizzo web usato dagli utenti.
- **MX** (Mail eXchange): specifica una lista di server di posta elettronica ai quali inviare le e-mail destinate a uno specifico nome di dominio; nel campo RData si troverà il nome del dominio che accetta la posta per conto del dominio indicato nel campo Name.
  Per esempio:

  ```
  pingu.di.school.it. MX 1 IN istruzione.it.
  pingu.di.school.it. MX 2 IN education.it.
  ```

  Supponiamo che l'host pingu non sia inserito in Internet. Non può quindi ricevere la posta elettronica, e il Resource Record a lui associato indica che le e-mail destinate a pingu devono essere inviate ad altri domini che poi le ritrasmetteranno a pingu secondo gli accordi presi (per esempio via rete mobile). Nell'esempio, la posta viene inviata prima a istruzione.it. (il numero 1 indica la prima scelta) e nel caso questo server non sia in grado di riceverla, deve essere inviata a education.it. (numero 2).
- **NS** (Name Server): è l'authoritative Name Server per il dominio specificato. I Name Server usano i Resource Record di tipo NS per trovarsi l'un l'altro.
  Per esempio:

  ```
  di.school.it. NS IN name1.di.school.it.
  ```

  indica che il dominio di.school.it. ha come Name Server name1.di.school.it.

Si deve avere un NS record per ogni Name Server (primario o secondario) di un dominio.

- **PTR** (PoinTerR): è un puntatore a un'altra parte dello spazio dei nomi, ed è utilizzato soprattutto per associare un nome di dominio a un indirizzo IP nel dominio in-addr.arpa per la *risoluzione inversa* (trattata nelle pagine successive). Ci deve essere un solo PTR record per ciascun indirizzo IP.
  Per esempio:

```
165.30.45.198.in-addr.arpa. PTR IN pingu.di.school.it.
```

  indica che l'indirizzo IP = 198.45.30.165 appartiene all'host: pingu.di.school.it.

- **TXT** (TeXT): consente di associare un testo a un nome di dominio. Si possono avere più record TXT associati a un singolo Name.
  Per esempio:

```
pingu.di.school.it.
TXT IN "Server Linux del Dipartimento di Informatica"
TXT IN "Amministratore: marcot@di.school.it"
```

- **SOA** (Start Of Authority): fornisce il nome della fonte principale di informazioni sulla zona del Name Server. Esso contiene la versione attuale del database DNS, l'indirizzo di posta elettronica dell'amministratore e altri parametri. Questo record deve essere obbligatoriamente presente nel livello più alto del dominio e deve essere unico per ogni Name Server (o, meglio, per ogni zona a cui appartiene il Name Server).

6.4 Come funziona il DNS?

L'albero gerarchico del DNS è implementato mediante una **base di dati distribuita** in cui sono memorizzati i Resource Record. Il fatto che sia distribuita garantisce il funzionamento continuo della rete: se tutte le informazioni fossero memorizzate su un unico server e questo si guastasse, si fermerebbe tutta la rete Internet. Non solo, questo server sarebbe così sovraccarico per tutte le richieste da soddisfare da non essere in pratica utilizzabile.

> Quindi la soluzione è stata di suddividere lo spazio dei nomi del DNS in zone disgiunte, ognuna con un Name Server **principale** (**DNS primario**) e dei Name Server **secondari** (**DNS secondario**) che attingono al principale per avere le informazioni.

I client che accedono ai Name Server sono i **resolver**: quando un'applicazione necessita di informazioni dal DNS usa questa libreria per effettuare le interrogazioni (query). Se il Resource Record è authoritative per la zona richiesta, il server DNS risponderà direttamente in quanto dispone dell'informazione, mentre in caso contrario effettuerà una ricerca all'interno dello spazio dei nomi per trovare i dati richiesti.

Questo processo si chiama **risoluzione dei nomi**.
Ci sono due tipi di query DNS:

- **iterative**: richiedono a un server DNS la miglior risposta che già conosce;
- **ricorsive**: chiedono al server DNS di rispondere alla query in modo completo.

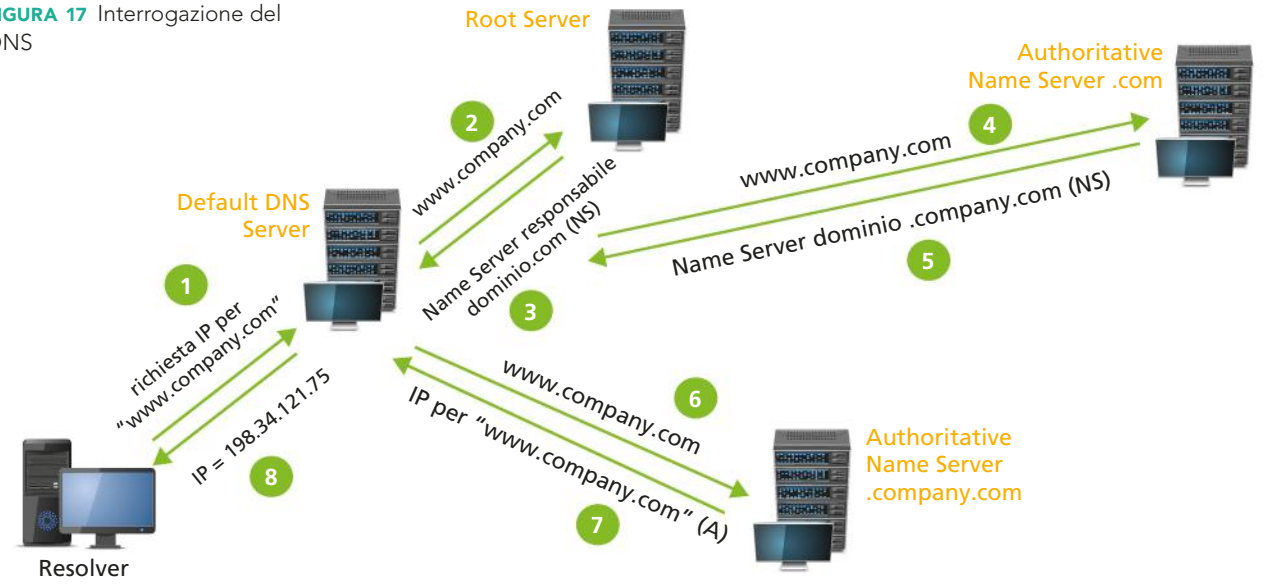
Di regola i resolver effettuano query ricorsive, lasciando così al Name Server il compito di risolvere il nome. I Name Server invece solitamente effettuano query iterative, seguendo via via i rimandi, finché non trovano la risposta.

Infatti un Name Server può inviare una query ricorsiva a un altro Name Server, ma spesso questi la rifiuta per evitare di essere sovraccaricato. Quando si verifica questa situazione, l'interrogazione diventa iterativa senza che l'utente finale se ne accorga.

esempio

In **FIGURA 17** è illustrato il meccanismo di interrogazione del DNS nel caso di un utente che scrive l'indirizzo mnemonico del sito: *www.company.com*, nella barra degli indirizzi del browser.

FIGURA 17 Interrogazione del DNS



1) Il computer Resolver chiede al Default DNS Server di risolvere il nome fornito, inviando un messaggio DNS request con il tipo di richiesta (A) e la classe (IN) per ottenere così il corrispondente indirizzo IP, che successivamente sarà utilizzato per stabilire la connessione con la macchina remota che ospita il web server. Il Default DNS Server solitamente è il server del provider che fornisce la connessione a Internet, ma potrebbe anche essere un server DNS interno alla LAN se il computer è connesso a una rete locale. Il suo indirizzo IP si trova nel file di configurazione del protocollo TCP/IP sul computer.
2) Il Default DNS Server verifica prima di tutto se possiede l'indirizzo IP corrispondente al nome da risolvere. Potrebbe, infatti, essere in grado di risolvere autonomamente quel nome, perché ha nella cache le informazioni relative oppure perché tale nome è già stato risolto in precedenza. In caso contrario, esso interroga uno dei root server, dando inizio al processo di ricerca delle informazioni all'interno della gerarchia dei Domain Name. Le interrogazioni DNS request avvengono utilizzando un datagram UDP.
3) Il Root Server risponde con l'indicazione del server responsabile per lo spazio dei nomi .com.
4) Così il Default DNS Server può inoltrare la stessa richiesta all'authoritative Name Server di .com, il quale non è in grado di risolvere completamente il nome www.company.com, ma conosce il server che lo può fare (ossia il Name Server responsabile per la zona company.com).
5) Il Name Server del dominio .com invia al Default DNS Server l'indirizzo dell'authoritative Name Server della zona .company.com.

6) Il Default DNS Server invia la richiesta di risoluzione del nome all'indirizzo appena trovato del Name Server della zona company.com.
7) Il Name Server riconosce il nome www come facente parte della zona company.com e restituisce al Default DNS Server l'indirizzo IP corrispondente.
8) Con il messaggio DNS reply del Default DNS Server, il computer ottiene l'indirizzo IP.

Una volta ottenuto l'indirizzo IP del web server, il computer può inviargli la richiesta della pagina web con un messaggio di get che verrà instradato nella rete e, arrivato a destinazione, verrà letto dal server web. Questi risponderà inviando la pagina desiderata.

Alcune considerazioni:

- un Name Server intermedio potrebbe anche avere già soddisfatto una simile richiesta e memorizzato nella **cache** una copia della risposta. In questo caso il Name Server fornisce una risposta dichiarandola di tipo *non authoritative*. Questo tipo di risposta non è del tutto affidabile poiché se nel frattempo fossero intervenute delle modifiche, queste non verrebbero propagate agli altri Name Server che quindi manterrebbero nella loro cache il dato precedente, non più corretto. Di qui l'importanza del campo TTL (Time To Live) del Resource Record che sta a indicare la stabilità dell'informazione. Per esempio un hostname con un TTL molto alto indica che quell'host ha lo stesso indirizzo IP da molto tempo e quindi anche se l'informazione non è authoritative ha un'alta probabilità di essere corretta;
- ogni Name Server che ha autorità su un dominio è duplicato per motivi di affidabilità; si ha quindi un Name Server **primario** e un Name Server **secondario** che devono essere periodicamente sincronizzati così da avere le stesse informazioni memorizzate in entrambi. Per effettuare questo allineamento si usa una connessione TCP che consente di trasferire in modo affidabile notevoli quantità di dati (invece di usare UDP come nelle interrogazioni). Quindi di norma il Name Server è in ascolto sulla porta **53 TCP** e sulla porta **53 UDP**.

6.5 La risoluzione inversa

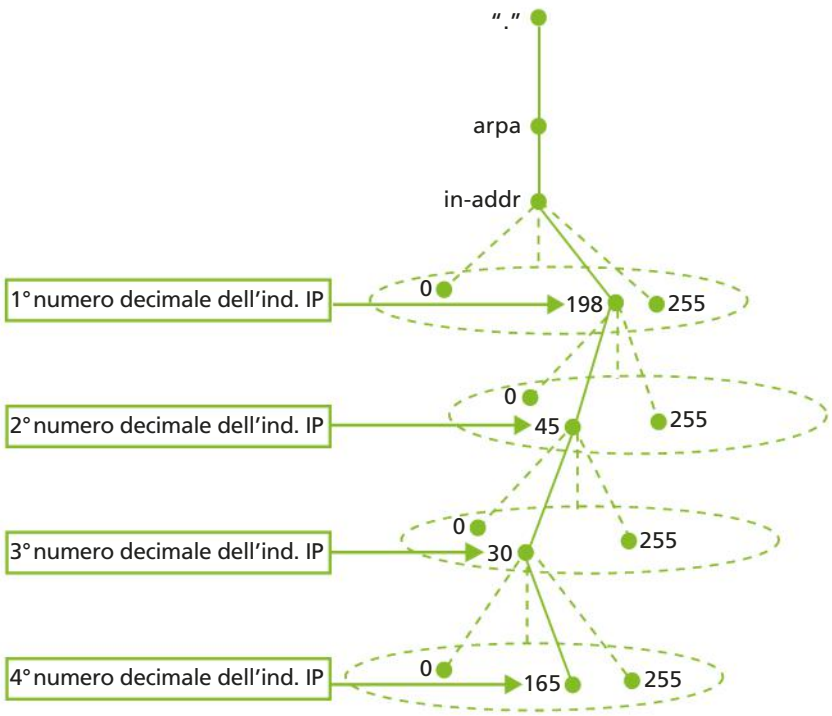
Quanto descritto nell'esempio precedente è il tipico **processo di risoluzione dei nomi**: dato un nome si deve trovare il corrispondente indirizzo IP.

> Esiste anche la possibilità di associare a un indirizzo IP il nome corrispondente e questo processo viene chiamato **risoluzione inversa**.

Di risoluzione inversa abbiamo già scritto a proposito del Resource Record PTR: infatti questo tipo di RR è usato per memorizzare l'associazione tra IP address e name, partendo dalla conoscenza dell'indirizzo IP (al contrario, quindi, degli RR di tipo A usati per la risoluzione, più frequente, dei nomi in indirizzi IP).
Questa ricerca viene resa semplice dal fatto che è stato creato un apposito dominio ***in-addr.arpa*** che usa la rappresentazione numerica degli indirizzi IP come etichette dei nodi (cioè come *name*). Questo speciale dominio può avere fino a 256 sottodomini di terzo livello (numerati da 0 a 255) corrispondenti ai possibili valori del primo

byte di un indirizzo IP; ognuno di questi a sua volta può avere fino a 256 sottodomini di quarto livello, anch'essi numerati da 0 a 255, che corrispondono ai possibili valori del secondo byte, e così via.
Quindi per risolvere un indirizzo IP richiesto, il resolver deve semplicemente chiedere al Name Server il Resource Record di tipo PTR del nodo corrispondente nel dominio *in-addr.arpa*. Per esempio, per ottenere il nome di dominio associato all'indirizzo IP 198.45.30.165, il resolver richiederà al DNS il Resource Record PTR del nome di dominio 165.30.45.198.in-addr.arpa **(FIGURA 18)**.

FIGURA 18 Risoluzione inversa



Come si può notare, nel dominio *in-addr.arpa* gli indirizzi IP sono al contrario, in conseguenza della differente interpretazione della posizione degli elementi negli indirizzi IP e nei nomi di dominio. Infatti gli indirizzi IP diventano più specifici da sinistra verso destra, mentre i nomi di dominio diventano più specifici da destra verso sinistra. Quindi chiamare i nodi del dominio *in-addr.arpa* in questo modo permette agli indirizzi IP di riflettere correttamente la struttura gerarchica del DNS.

FISSA LE CONOSCENZE

- Qual è l'ente preposto alla gestione del DNS?
- Perché il database del DNS è distribuito?
- Descrivi il formato dei messaggi scambiati tra DNS Client e DNS Server.
- In quale struttura vengono mantenute le informazioni sui nomi di dominio?
- Sia l'RR di tipo A sia l'RR di tipo PTR contengono l'associazione tra un indirizzo IP e un nome; in che cosa, però, differiscono?
- Come è da interpretare una risposta di tipo non authoritative?

7 PROBLEMATICHE DI SICUREZZA

7.1 La non sicurezza di DHCP e DNS

Il **DHCP** utilizza i protocolli UDP e IP che sono intrinsecamente insicuri e nelle sue specifiche non si fa riferimento a possibili misure per la sicurezza.

Se ai tempi in cui è stato standardizzato questo protocollo il numero di utenti di Internet era limitato e si trattava per lo più di enti accademici e di ricerca, attualmente le problematiche di sicurezza sono di centrale importanza, soprattutto per un protocollo come DHCP che tratta informazioni di configurazione.

In particolare, ne individuiamo due derivanti da:

- **DHCP Server non autorizzati**: un DHCP Server abusivo potrebbe inserirsi e rispondere alle richieste del client fornendo informazioni false in modo da inibire gli host, oppure configurandoli per azioni fraudolente;
- **DHCP Client non autorizzati**: un host potrebbe ottenere le informazioni di configurazione destinate a un certo client con lo scopo di creare danni alla rete, oppure potrebbe usare un software che genera moltissime richieste DHCP così da esaurire gli indirizzi a disposizione del server e bloccare nuovi accessi alla rete.

Un modo per ovviare a questi inconvenienti è introdurre meccanismi di sicurezza nei livelli più bassi, per esempio evitando che un device non autorizzato possa inserirsi fisicamente nella rete.
Inoltre, si potrebbe usare **IPsec** per rendere sicuro il livello Network.

#prendinota

IP security protocol (IPsec) protegge i pacchetti IP scambiati tra host e router a livello Network, garantendone la confidenzialità.
Inoltre, offre i servizi di autenticazione del mittente e di integrità dei dati.

Il **DNS** è particolarmente critico dal punto di vista della sicurezza per vari motivi:

- non è autenticato: l'informazione richiesta potrebbe arrivare non dal DNS Server corretto ma da un'altra macchina;
- è molto lento, quindi è possibile che qualcuno intercetti la richiesta destinata a un DNS Server e risponda al suo posto (spoofing);
- il protocollo non offre meccanismi per proteggere l'integrità delle informazioni distribuite (basti pensare all'associazione tra hostname e indirizzo IP).

In passato si sono avuti casi di **DNS cache poisoning** volti a manomettere le informazioni contenute nei DNS Server, compromettendo la coerenza e l'integrità dei suoi dati.

Nella precedente Lezione si è visto come un DNS Server mantenga in una memoria cache anche informazioni relative a domini non di sua competenza. Una risposta fornita sulla base di questi dati è detta **non authoritative** e il valore del campo TTL indica quanto sia attendibile (più è alto il TTL più è alta la probabilità che il dato sia corretto). Un attacco di tipo **cache poisoning** a un DNS Server comporta la modifica dei dati della sua cache, inserendovi un valore di TTL molto alto, così da rendere attendibile l'informazione modificata. Tipicamente, l'intervento consiste nell'associare a un nome l'indirizzo IP di un server malevolo. Per esempio, un utente scrive nel browser l'URL di un sito web ma viene poi direzionato, a sua insaputa, verso un sito clone costruito per effettuare furti di identità o di dati bancari. Questo succede perché nella cache del DNS Server l'indirizzo IP originale, associato a quel nome, è stato sostituito con quello del web server malevolo.

IN ENGLISH PLEASE

DNS cache poisoning (or DNS spoofing) is a form of computer security hacking in which corrupt DNS data is introduced into the DNS resolver's cache, causing the Name Server to return an incorrect IP address. This results in traffic being diverted to another computer.

Per rimediare alle mancanze del protocollo originario in termini di sicurezza, in ambito IETF è stato creato un gruppo di lavoro che ha definito un'estensione al DNS denominata **DNSSEC** (Domain Name System Security Extensions).
Il compito di DNSSEC è di garantire all'utente che il sito web che sta visitando è quello originale e non una copia creata per scopi fraudolenti. A tal scopo si usano delle chiavi crittografiche per autenticare i dati nel DNS, a partire dalla root. Le chiavi per la root sono gestite da ICANN, l'ente responsabile dei Domain Name di primo livello (generici e nazionali).
Proprio per il ruolo particolarmente critico che il DNS riveste nell'attuale scenario di Internet, l'ICANN ha evidenziato la necessità di stabilire metriche e modalità per il controllo del DNS, individuando 5 indicatori importanti: **coerenza**, **integrità**, **velocità**, **disponibilità** e **robustezza**.

7.2 La protezione dei client nelle reti Microsoft

In una rete interna Microsoft, un client può ottenere indirizzi IP differenti ogni volta che viene acceso, poiché utilizza il protocollo di configurazione dinamica DHCP. Di conseguenza la configurazione dei **Resource Record DNS** è automatica.
Tipicamente i PC con versioni recenti di Windows sono in grado di gestire l'aggiornamento dinamico del record DNS: quando ottengono un indirizzo IP dal DHCP Server, inviano la richiesta di aggiornamento del record A relativo al client.
La FIGURA 19 descrive le due fasi di aggiornamento dinamico del record DNS:

1. quando il computer viene acceso, lo scambio di messaggi con il DHCP Server fornisce indirizzo IP e altre informazioni di configurazione di TCP/IP (default gateway e indirizzo del DNS Server);
2. il client comunica con il DNS Server per creare un nuovo **record A** relativo all'hostname del computer e al suo indirizzo IP.

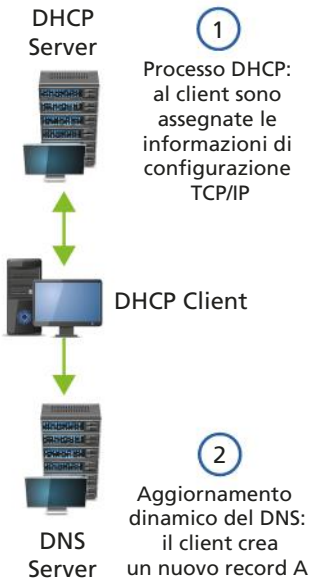


FIGURA 19 Processo di aggiornamento dinamico del record A di un client Windows

Quando il client tenta di registrare un record A e scopre che ne esiste già uno con lo stesso nome, ma indirizzo IP diverso, per default tenta di sostituirlo con quello nuovo. In questo modo, però, un qualunque computer della rete potrebbe modificare un record A sul DNS Server.
Il processo appena descritto può essere reso sicuro con due diversi procedimenti: **autenticazione del client** oppure **assegnazione dei permessi in base alle zone DNS**.
Le zone che sono configurate per aggiornamenti dinamici sicuri permettono solo ai client autorizzati di modificare i Resource Record.
Di norma, sono gestiti nel seguente modo: dapprima il DNS Client cerca di usare la modalità dinamica non sicura e se viene rifiutata dal server passa allora a usare la modalità sicura. Quando poi una zona è integrata in Active Directory, il DNS Server (configurato su Windows Server) per default consente solo aggiornamenti dinamici sicuri. La configurazione degli aggiornamenti dinamici sicuri richiede quindi l'impostazione di alcuni parametri sul DNS Server. In particolare, Microsoft raccomanda che esso venga installato su un **Domain Controller** e che le zone siano integrate in **Active Directory** (servizi che affronteremo nel quinto anno).

#prendinota

In alcuni ambiti di rete che richiedono un livello di sicurezza molto elevato può non essere sufficiente l'autenticazione del DNS Client per creare o modificare un record DNS. In questi casi una soluzione è la definizione di Access Control List (che verranno trattate nel corso del quinto anno) per definire i permessi degli utenti.

FISSA LE CONOSCENZE

- Descrivi come DHCP Server non autorizzati e DHCP Client non autorizzati possono creare problemi di funzionamento della rete.
- Spiega in che cosa consiste il DNS cache poisoning.
- Come può essere reso sicuro l'aggiornamento del DNS?

8 IL COMANDO NSLOOKUP

8.1 Modalità interattiva

Nslookup è un comando del Sistema Operativo usato per interrogare il DNS, presente sia nei sistemi Windows che in quelli Unix/Linux.
Nslookup permette di effettuare delle query a un server DNS per richiedere:

- la risoluzione di un nome o di un indirizzo IP;
- specifici Resource Record;
- di avere visione del contenuto della memoria di un Name Server.

> Nslookup viene solitamente utilizzato per verificare la configurazione delle zone DNS e per individuare e risolvere problemi di risoluzione dei nomi.

Nslookup si può usare nella modalità **interattiva** e in quella **non interattiva**.
Nella modalità interattiva si digita il comando nslookup senza alcuna opzione. Con questa istruzione si possono inviare più query e visualizzare i singoli risultati.
Come risultato si ottiene il Default DNS Server configurato per l'host su cui è stato eseguito il comando, seguito dal prompt > a indicare che l'applicazione è pronta a ricevere i comandi.
In FIGURA 20 si mostra il risultato dell'esecuzione del comando *help*; questo comando può essere eseguito solo nella modalità interattiva.

```
C:\>nslookup
Server predefinito:
Address:  192.168.2.1

> help
Comandi:  (gli identificatori sono indicati in maiuscolo, [] indica opzionale)
NOME            - stampa informazioni sul NOME host/dominio usando il server
                  predefinito
NOME1 NOME2     - come sopra, ma usa NOME2 come server
help o ?        - stampa informazioni su comandi comuni
set OPZIONE     - imposta una opzione
    all                     - stampa opzioni, server corrente e host
    [no]debug               - stampa informazioni di debugging
    [no]d2                  - stampa informazioni dettagliate di debugging
    [no]defname             - aggiungi nome dominio ad ogni query
    [no]recurse             - per risposte ricorsive alla query
    [no]search              - usa elenco di ricerca dominio
    [no]vc                  - usa sempre un circuito virtuale
    domain=NOME             - imposta il NOME predefinito del dominio
    srchlist=N1[/N2/.../N6] - imposta dominio a N1 e lista di ricerca su
                              N1,N2, ecc.
    root=NOME               - imposta server principale a NOME
    retry=X                 - imposta numero tentativi a X
    timeout=X               - imposta l'intervallo iniziale di scadenza a X
                              secondi
    type=X                  - imposta tipo query
                              (es. A,ANY,CNAME,MX,NS,PTR,SOA,SRV)
    querytype=X             - come tipo
    class=X                 - imposta classe query (es. IN (Internet), ANY)
    [no]msxfr               - usa trasferimento di zona rapido MS
    ixfrver=X               - versione corrente da usare nelle richieste di
                              trasferimento IXFR
server NOME     - imposta server predefinito su NOME, usando il server
                  corrente predefinito
lserver NOME    - imposta server predefinito su NOME, usando server iniziale
finger [UTENTE] - usa finger per il NOME opzionale all'host corrente
root            - imposta server corrente predefinito su root
ls [opt] DOMINIO [> FILE] - elenca indirizzi nel DOMINIO (opzionale: output a
                            FILE)
    -a          -  elenca noni canonici e alias
    -d          -  elenca tutti i record
    -t TIPO     -  elenca record del tipo indicato (es. A,CNAME,MX,NS,PTR
                   ecc.)
view FILE       - ordina un file di output 'ls' e visualizzalo con pg
exit            - esci dal programma

>
```

FIGURA 20 Risultato del comando help di nslookup

8.2 Modalità non interattiva

Nella modalità non interattiva si può inviare una sola query e visualizzarne il risultato. Di norma si usa quando si vuol interrogare un solo host e quindi si scrive il nome dell'host dopo aver scritto nslookup.
Nella finestra seguente si mostra il risultato del comando nslookup dato in modalità non interattiva, per richiedere la risoluzione DNS del nome: www.google.com:

```
C:\> nslookup www.google.com
Server: ns1.mydns.com
Address: 204.127.202.4
Name: www.google.com
Address: 66.249.87.99
```

Nella finestra seguente si mostra invece il risultato di una richiesta che nslookup non ha potuto soddisfare in quanto non c'è stata risposta dal DNS server:

```
C:\> nslookup www.google.com
*** Default servers are not available
Server: Unknown
Address: 127.0.0.1
*** Unknown can't find www.google.com: No response from server
```

Spesso nella risposta alla query inoltrata al DNS si trova la frase: «Non authoritative answer» che sui sistemi in italiano è stata tradotta con: «Risposta da un server non di fiducia». Ciò significa che il DNS Server che ha fornito l'informazione non possiede il Domain Name record, ma l'informazione è semplicemente memorizzata nella cache del server.

Nslookup consente di cambiare il server DNS da interrogare sostituendolo, per esempio, con un **DNS pubblico**.
Una volta cambiato il Default DNS Server, le successive richieste saranno inviate al nuovo server. Il comando è mostrato nella finestra seguente:

```
C:\> nslookup
Server: intranetserver.local
Address: 172.17.2.10

>server 194.119.192.34
Server predefinito: nameserver.cnr.it
Address: 194.119.192.34
```

#prendinota

DNS Pubblici
nameserver.cnr.it
194.119.192.34
resolver2.opendns.com
208.67.220.220
resolver1.opendns.com
208.67.222.222
google-public-dns-a.
google.com
8.8.8.8
google-public-dns-b.
google.com
8.8.4.4

FISSA LE CONOSCENZE

- Descrivi alcune caratteristiche del comando nslookup.
- Che differenza c'è tra la modalità interattiva e quella non interattiva?
- Che cosa significa la frase «Non authoritative answer»?
- Qual è il comando per cambiare server DNS?

9 PACKET TRACER: LA CONFIGURAZIONE DEGLI HOST

9.1 Configurazione automatica degli host tramite un router DHCP

In questa esercitazione di laboratorio realizzeremo con il simulatore Packet Tracer quanto appreso sul DHCP.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare due reti LAN configurando il servizio DHCP sul **router** per assegnare automaticamente gli indirizzi IP, la subnet mask e il gateway a tutti gli host delle due LAN.

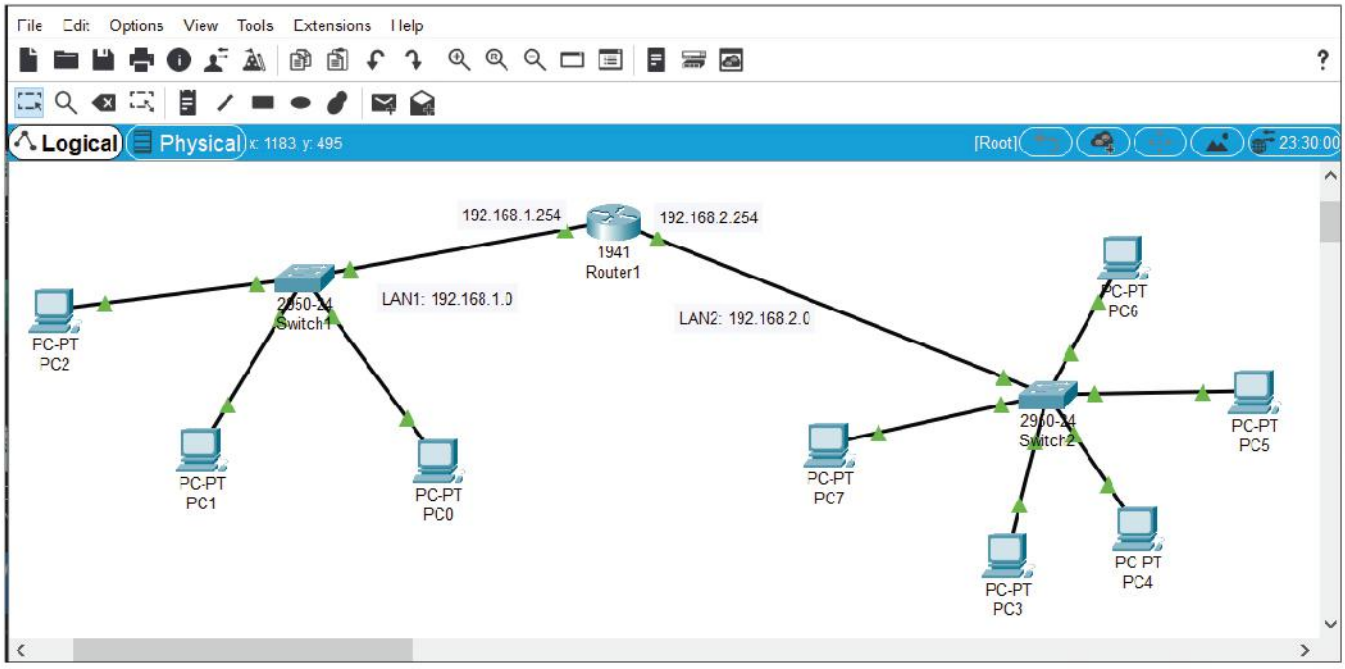
➜ ANALISI DEL PROBLEMA

Il servizio DHCP può essere configurato sul router o su un server. Poiché il router ha anche molti altri compiti e di conseguenza molte elaborazioni da eseguire, è consigliabile affidargli il servizio DHCP solo in presenza di LAN di ridotte dimensioni e fornirsi invece di una macchina server in presenza di LAN con centinaia di host. L'indirizzo IP del router fungerà da gateway per la rete e verrà inviato insieme all'IP e alla subnet mask. Per impostare gli address pool delle due reti useremo la CLI (Command Line Interface) del router.

➜ SVOLGIMENTO

Nella **FIGURA 21** è mostrato un possibile scenario con due LAN collegate da un router.

FIGURA 21 Scenario router DHCP su due LAN



Per le due reti locali utilizziamo i soliti indirizzi privati 192.168.1.0 e 192.168.2.0.
Dopo aver piazzato i dispositivi e averli collegati tra loro, configuriamo manualmente il router assegnando alle sue interfacce GigabitEthernet gli IP 192.168.1.254 per la LAN1 e 192.168.2.254 per la LAN2, entrambe con subnet mask 255.255.255.0.

#prendinota

Le interfacce del router di default sono OFF. Vanno messe ON.

> Questi due indirizzi delle porte del router saranno i gateway per gli host delle due LAN.

Tutti gli altri indirizzi li assegnerà il router automaticamente col servizio DHCP.
Apriamo la CLI del router e scriviamo i comandi per la configurazione della LAN1:

```
Router(config)#ip dhcp pool LAN1
Router(dhcp-config)#network 192.168.1.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.1.254
```

Allo stesso modo configuriamo con i seguenti comandi la LAN2:

```
Router(config)#ip dhcp pool LAN2
Router(dhcp-config)#network 192.168.2.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.2.254
```

La prima istruzione dà un nome al pool di indirizzi della rete in questione, la seconda imposta l'indirizzo generico e la subnet mask di quella rete e la terza imposta il gateway.
A questo punto non resta che controllare se gli indirizzi sugli host sono stati assegnati aprendo le schede IP Configuration di ogni PC.
Nella **FIGURA 22** si possono vedere i valori di IPv4, Subnet Mask e Default Gateway assegnati dal router-DHCP al PC1.

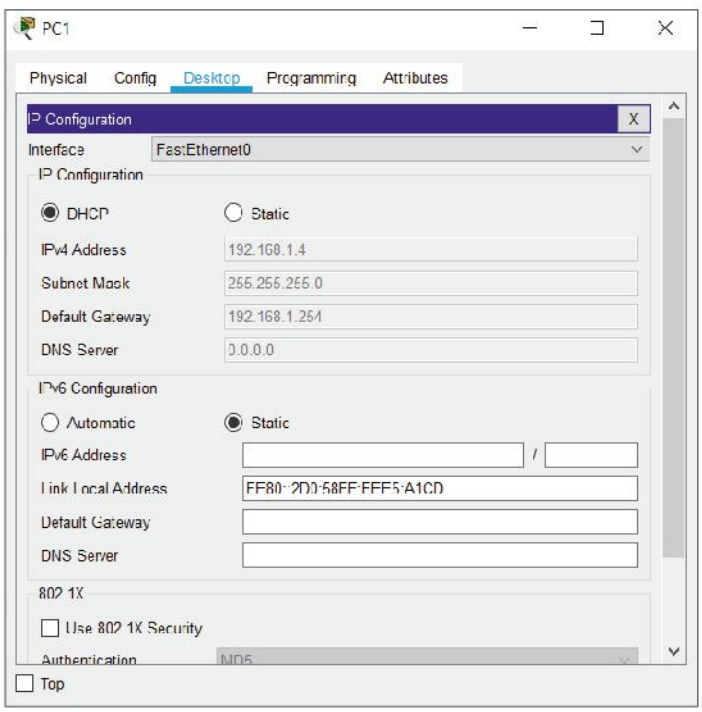


FIGURA 22 Configurazione di rete assegnata automaticamente al PC1

#prendinota

Se il servizio DHCP non è configurato, interverrà l'APIPA (se gli host sono Windows) ad assegnare gli indirizzi IP nell'intervallo 169.254.1.0 - 169.254.254.255. La subnet mask verrà automaticamente impostata su 255.255.0.0 e l'indirizzo del gateway su 0.0.0.0.

A volte è necessario escludere qualche indirizzo IP dal pool di indirizzi affidato al router DHCP perché si vuole utilizzarli come indirizzi statici su host particolari (ser-

ver, stampanti di rete, ecc.). In tali casi si può, sempre dalla CLI del router, dare il comando:

```
Router(dhcp-config)#ip dhcp excluded-address 192.168.1.2
```

Quando il router DHCP ha terminato l'assegnazione di tutti i parametri a tutti gli host, è possibile testare il corretto funzionamento delle due reti con un ping (simple PDU) tra un host della LAN1 e un host della LAN2.

9.2 Configurazione automatica degli host tramite un server DHCP

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare una rete LAN configurando il servizio DHCP su un **server** per assegnare automaticamente gli indirizzi IP, la subnet mask e il gateway a tutti gli host della LAN.

➜ ANALISI DEL PROBLEMA

L'utilizzo di una macchina server posizionata in ogni LAN rende più efficiente il servizio. Le macchine con compiti particolari (router e server) è meglio abbiano un IP scelto dall'amministratore di rete, quindi non casuale.

➜ SVOLGIMENTO

Nella FIGURA 23 è mostrato un possibile scenario LAN con un server DHCP.

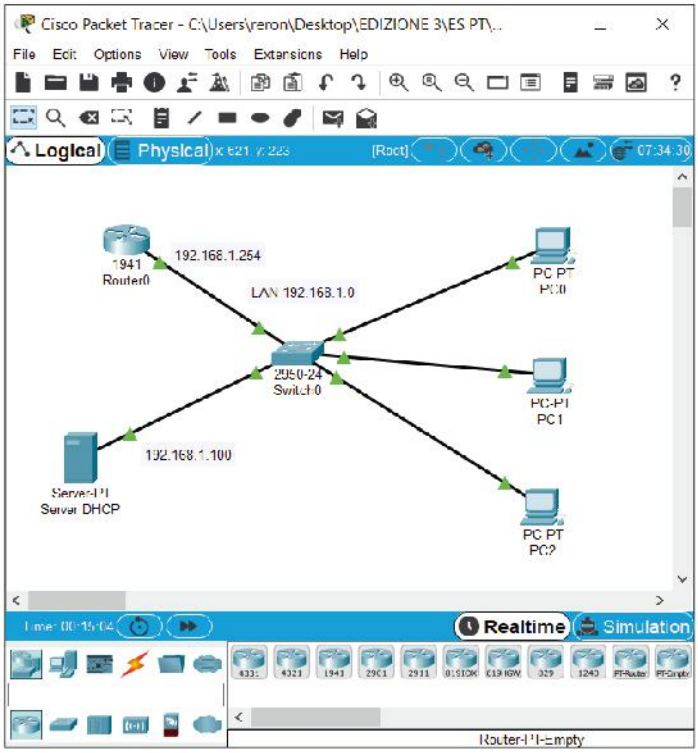


FIGURA 23 Scenario server DHCP in una LAN

Assegniamo manualmente al router e al server rispettivamente gli IP statici 192.168.1.254 e 192.168.1.100. Dopo aver piazzato gli host e averli collegati in rete, configuriamo il

servizio DHCP sul server. Apriamo la scheda **Services** del server (FIGURA 24), selezioniamo DHCP e compiliamo i campi:

- Pool Name
- Default Gateway
- Start IP Address
- Subnet Mask
- Maximum Number of User

#prendinota

Il servizio DHCP sul server, di default è OFF. Va messo ON.

Il Default Gateway sarà, come nell'esercizio precedente, l'IP del router 192.168.1.254. Qualunque altro host aggiunto allo scenario riceverà automaticamente i parametri di configurazione impostati sul server.

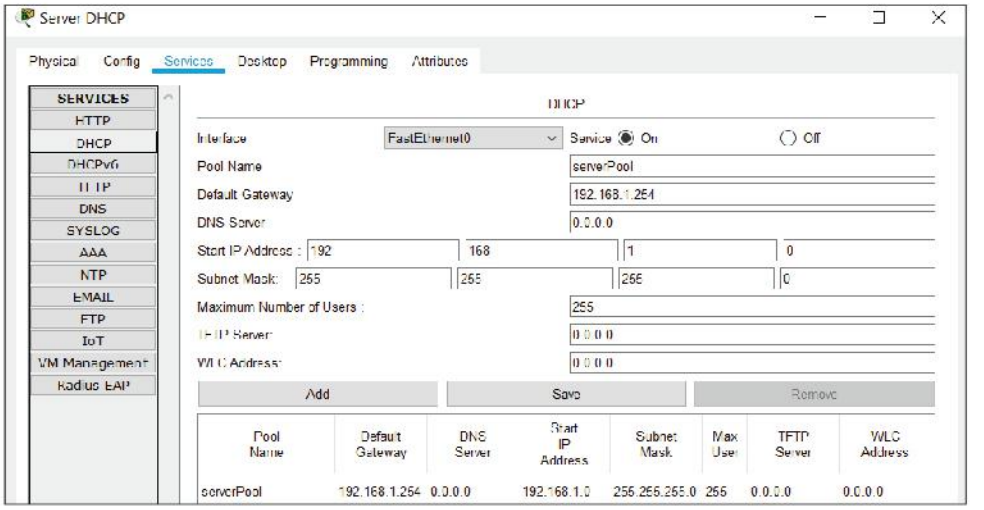


FIGURA 24 Configurazione del servizio DHCP sul server

Nella FIGURA 25 si possono vedere i valori di IPv4, Subnet Mask e Default Gateway assegnati dal server DHCP al PC1, esattamente come aveva fatto il router nell'esercizio precedente (Figura 22).

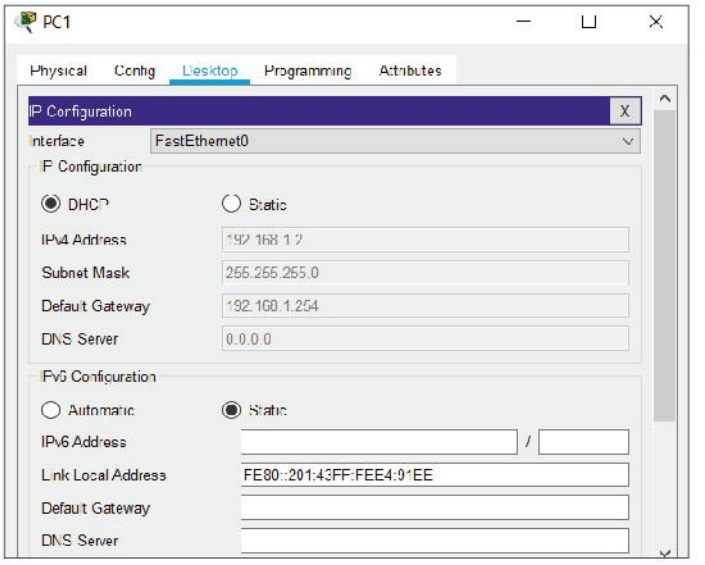


FIGURA 25 Configurazione di rete assegnata automaticamente al PC1

FISSA LE CONOSCENZE

- Come si fa a escludere qualche indirizzo IP dal pool di indirizzi?
- Quale scheda del server si usa per configurare il DHCP?
- Che differenza c'è tra impostare il DHCP sul router o sul server?
- Qual è l'indirizzo del gateway che va impostato per tutta la rete?

10 PACKET TRACER: LA CONFIGURAZIONE DEL SERVER DNS

In questa esercitazione di laboratorio realizzeremo con il simulatore Packet Tracer quanto appreso sul DNS.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare due reti LAN aventi ciascuna una home page accessibile dall'esterno mediante un server DNS pubblico in grado di risolvere i nomi delle pagine. Quindi verificarne il funzionamento chiamando, dal web browser di un qualsiasi PC di una delle due reti, la pagina web dell'altra rete.

➜ ANALISI DEL PROBLEMA

Per poter risolvere i nomi delle due pagine web e trovare l'indirizzo IP del server su cui sono rispettivamente depositate, occorre configurare il servizio DNS su un server dichiarando due Resource Records di tipo A Record che abbinino il nome di ogni pagina web al corrispondente IP.

➜ SVOLGIMENTO

Nella **FIGURA 26** è mostrato un possibile scenario con due LAN e un server DNS pubblico.

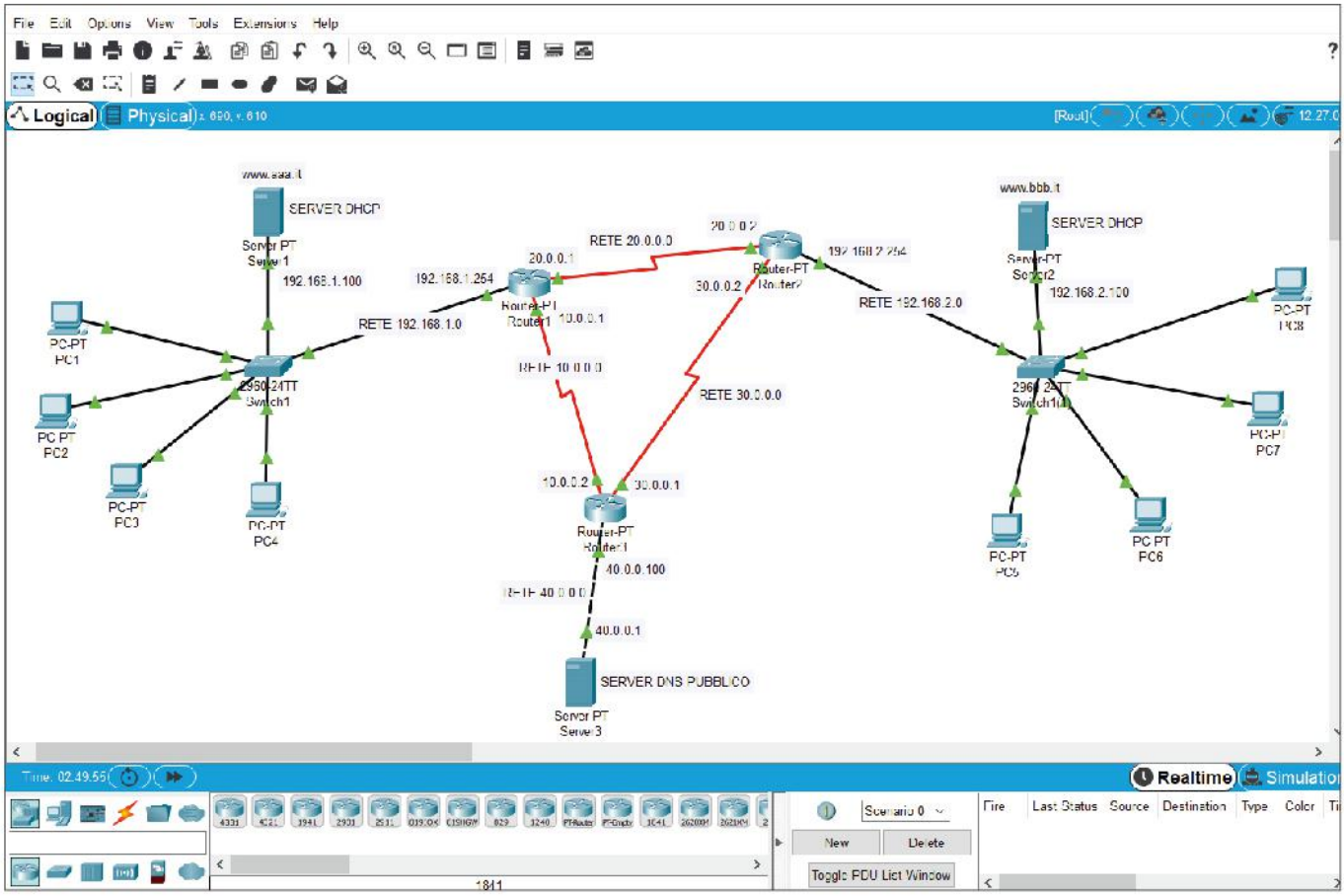
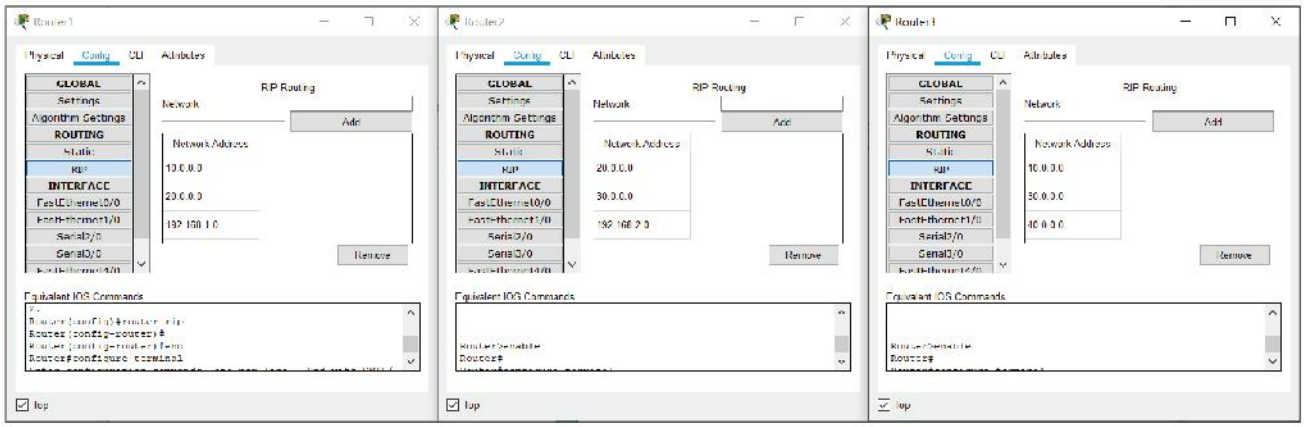


FIGURA 26 Scenario con server DNS pubblico

Per le due reti locali utilizziamo i soliti indirizzi privati 192.168.1.0 e 192.168.2.0 mentre per il DNS pubblico usiamo l'indirizzo 40.0.0.1.
Per le 3 reti costituite dalle coppie di router usiamo gli indirizzi 10.0.0.0, 20.0.0.0 e 30.0.0.0, assegnando manualmente gli indirizzi alle porte Serial2/0 e Serial3/0 di ogni router.
Sempre manualmente assegniamo alle interfacce FastEthernet0/0 dei router delle due LAN gli IP 192.168.1.254 per la LAN1 e 192.168.2.254 per la LAN2, entrambe con subnet mask 255.255.255.0. Tali indirizzi saranno i gateway per gli host delle due LAN.
Sui 3 router configuriamo poi il RIP dalla scheda **Config** (FIGURA 27).

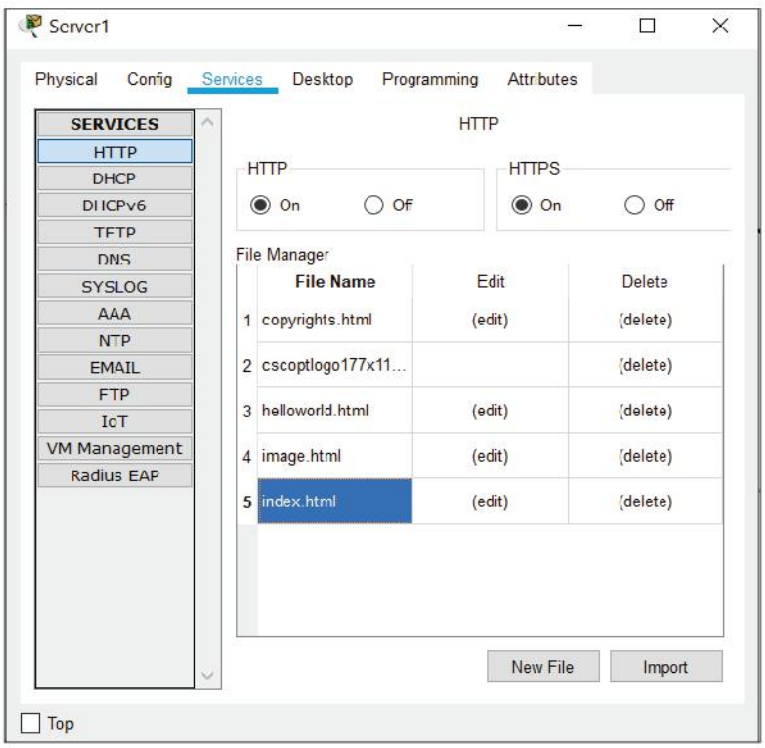
FIGURA 27 Configurazione del RIP sui 3 router



In ogni LAN avremo poi un server DHCP per assegnare automaticamente gli indirizzi IP agli host della rete. I due server DHCP avranno indirizzo rispettivamente 192.168.1.254 e 192.168.2.254.

Aprendo la scheda Services sui server si può notare come il servizio HTTP (**Web Server**) sia preconfigurato, già messo di default a ON e con 5 pagine html precaricate (FIGURA 28).

FIGURA 28 Servizio HTTP sul Server1



Cominciamo dal Server1. Selezioniamo il file **index.html** e clicchiamo sul suo **edit**. Modifichiamo la pagina web di default che si apre **(FIGURA 29)** sostituendo il tag <hr> con **Welcome to www.aaa.it** **(FIGURA 30)** e salvando la modifica con **Save**.

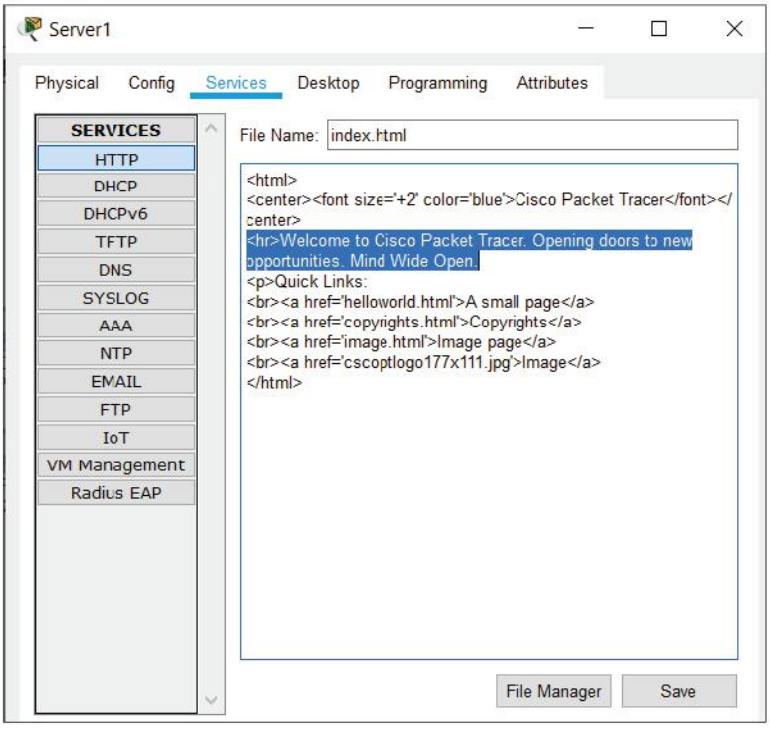


FIGURA 29 Pagina web di default del Server1

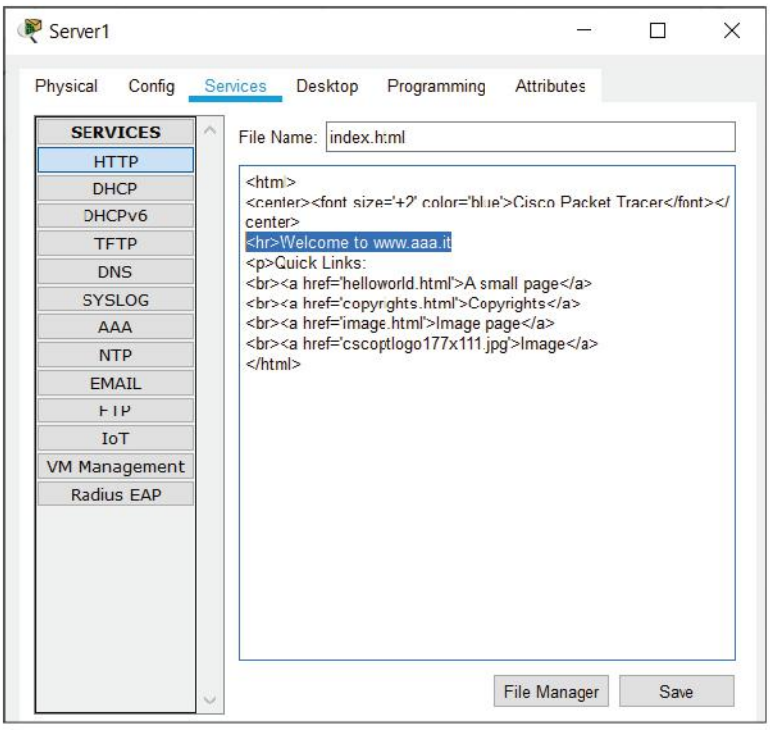


FIGURA 30 Pagina web modificata del Server1

Lo stesso va fatto sul Server2 sostituendo il tag <hr> con **Welcome to www.bbb.it** e salvando la modifica con **Save**.

#prendinota

È possibile caricare qualunque pagina html al posto di quella di default.

A questo punto è tutto pronto per la configurazione del server DNS. Apriamo la scheda Services del Server3 e aggiungiamo 2 entry di Name www.aaa.it e www.bbb.it, di Type **A Record** e con indirizzo IP rispettivamente del Server1 e del Server2 (FIGURA 31). Cliccare poi su **ADD** per aggiungere ogni entry.
Non resta che mettere ON il servizio e provarne il funzionamento.
Clicchiamo per esempio sul PC8 della rete 192.168.2.0, apriamo la scheda **Desktop** e selezioniamo il suo **Web Browser** (FIGURA 32).

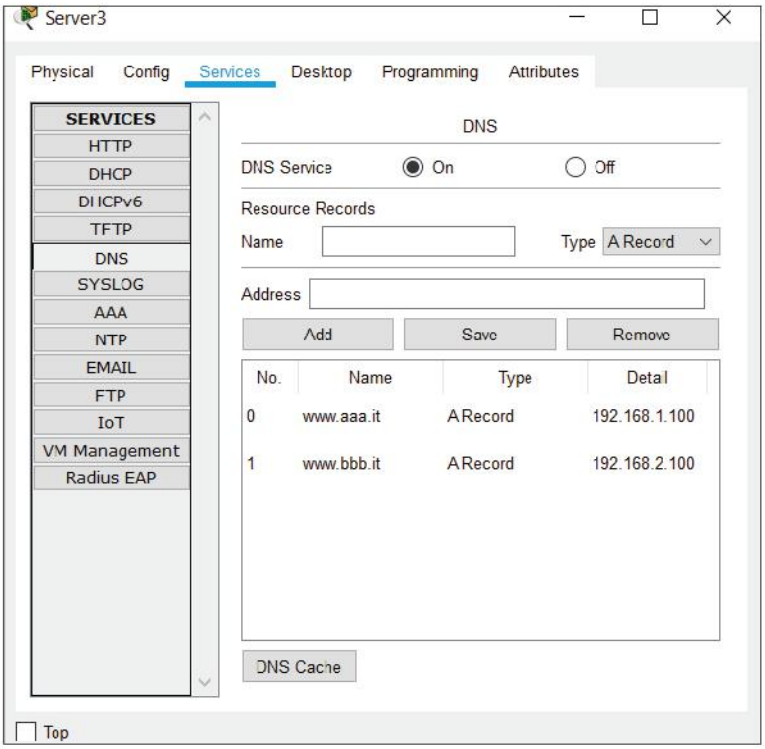


FIGURA 31 Configurazione del server DNS

FIGURA 32 Scheda Desktop di PC8

Nella scheda Desktop che si apre chiamiamo la pagina web dell'altra rete (192.168.1.0) scrivendo www.aaa.it nella barra **dell'URL** (FIGURA 33) e cliccando su **Go**.

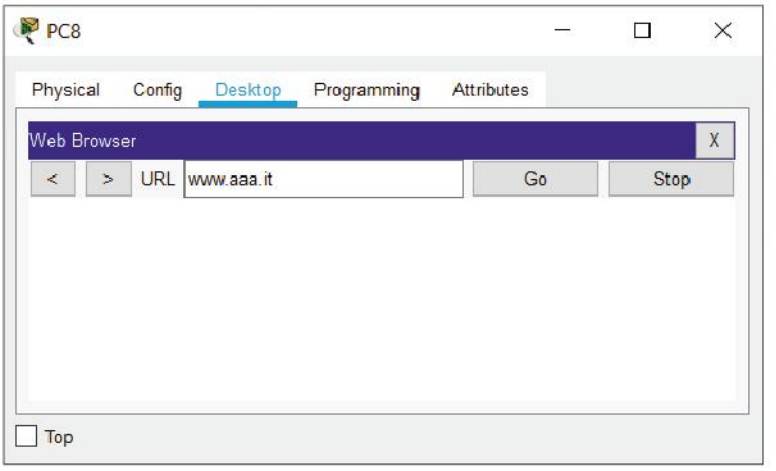
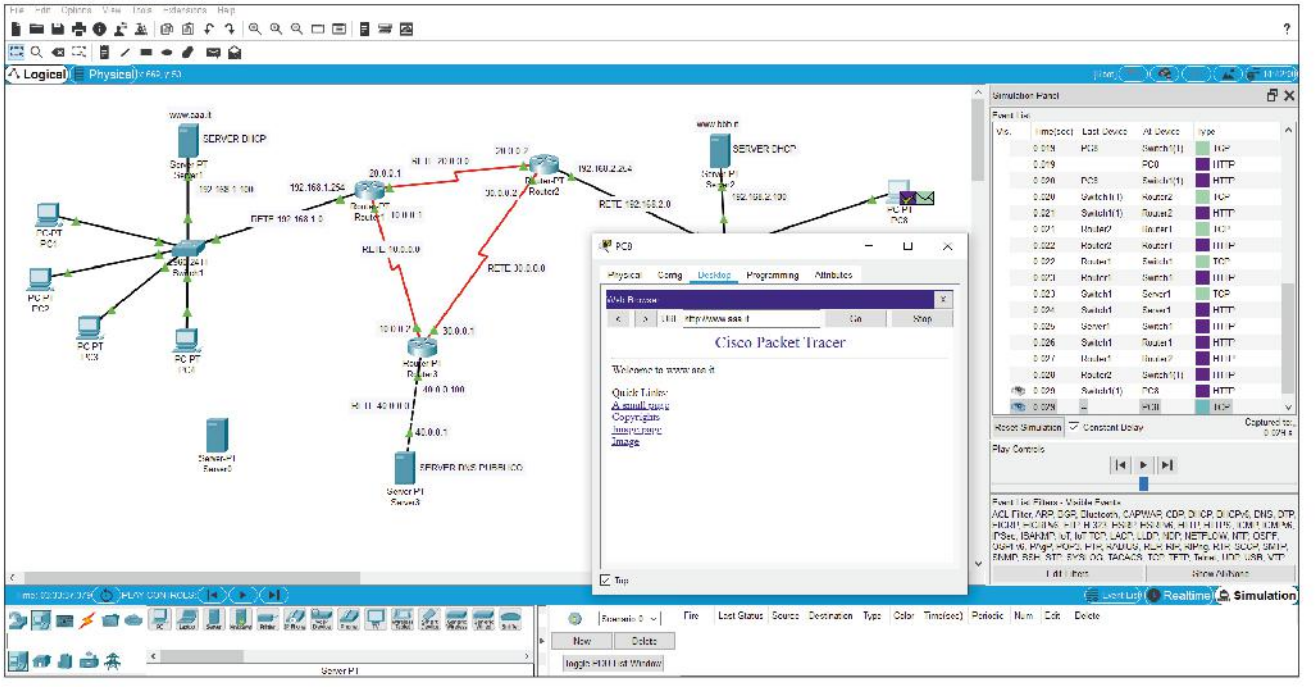


FIGURA 33 Web Browser del PC8

Questo provocherà l'interrogazione del server DNS e il risultato sarà l'apertura della pagina web richiesta (FIGURA 34).

FIGURA 34 Apertura della home page di www.aaa.it richiesta da PC8



Essendo HTTP un protocollo affidabile, userà TCP come protocollo di trasporto come si può vedere dai pacchetti elencati nella **Event List** del **Simulation Panel**.

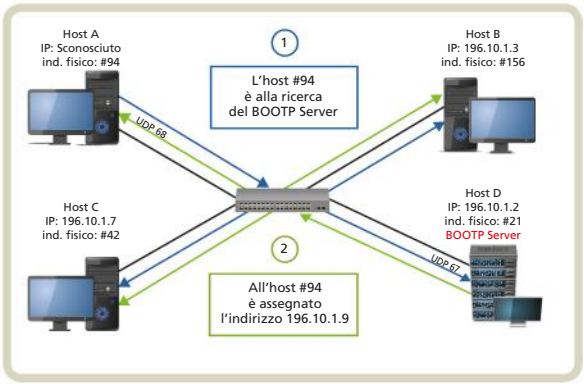
Testare il completo funzionamento del DNS chiamando la pagina www.bbb.it da un qualsiasi PC della rete 192.168.1.0.

FISSA LE CONOSCENZE

- Come si configura il servizio DNS affinché possa risolvere i nomi di una rete?
- Come si può verificare se il DNS è stato correttamente configurato?

1 La configurazione degli host

Per configurare in modo automatico gli host di una rete TCP/IP sono stati definiti in ambito IETF alcuni protocolli. Uno dei primi è stato BOOTP (BOOTstrap Protocol), che mette in comunicazione un computer privo di disco e un secondo host che gli fornisce le informazioni di rete. Poiché BOOTP poteva convogliare altre informazioni di configurazione, gli amministratori lo usavano anche per inviare un'installazione client preconfigurata ai computer nuovi da inserire nella rete aziendale.



2 Il DHCP (Dynamic Host Configuration Protocol)

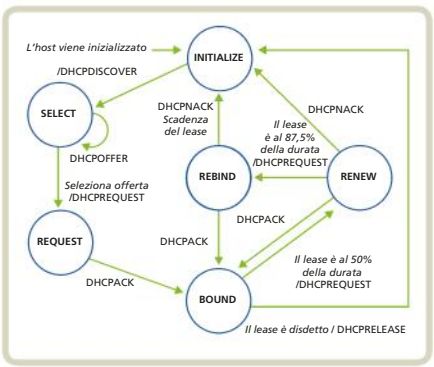
Quando nelle reti si diffusero i dispositivi mobili, BOOTP risultò troppo lento nell'aggiornare quei dispositivi che si spostavano da una rete a un'altra. Fu perciò necessario introdurre una nuova modalità di assegnazione dinamica, che consentisse di allocare velocemente gli indirizzi per un tempo limitato.

Il protocollo DHCP, evoluzione di BOOTP, soddisfa le esigenze di mobilità con l'assegnazione dinamica degli IP. Esso si basa sul concetto di lease, cioè affitto, degli indirizzi per un tempo limitato.

3 L'architettura Client/Server DHCP

Il DHCP Server utilizza un database nel quale sono memorizzati gli indirizzi IP a sua disposizione per l'allocazione ai DHCP Client presenti nella rete. DHCP usa il MAC address e il network address per identificare l'host. Quindi consulta il database e stabilisce se assegnare all'host un indirizzo IP permanente o temporaneo. Ogni volta che un host si connette a una rete, il suo DHCP Client richiede un indirizzo IP al DHCP Server, che lo sceglierà, in modo arbitrario, tra quelli disponibili nel suo address pool. Quando l'host lascerà la rete, il suo indirizzo ritornerà disponibile nel pool.

Quando un DHCP Server è responsabile dell'indirizzamento su una subnet diversa dalla propria è necessario introdurre un relay agent.



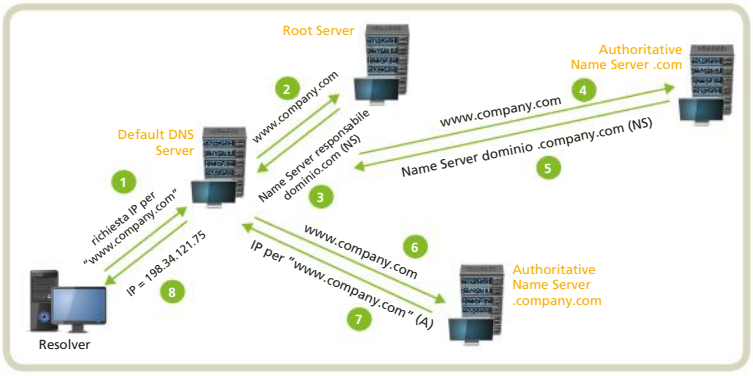
4 La comunicazione tra DHCP Client e DHCP Server

I messaggi scambiati tra DHCP Client e Server sono di tipo request/reply. Il processo di assegnazione di un indirizzo IP avviene in 4 fasi. Il DHCP Client può trovarsi in 6 stati.

5 Il DHCP per IPv6

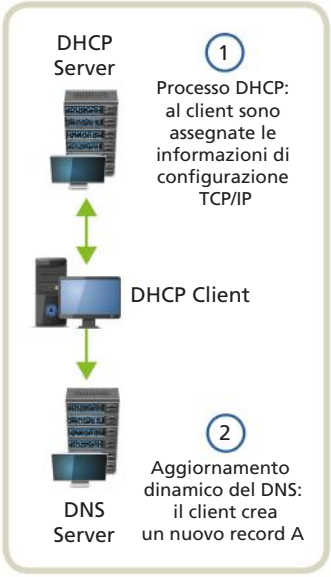
A seguito della release di IPv6, il protocollo DHCP è stato aggiornato nella versione DHCPv6. Le configurazioni degli host IPv6 sono: stateless autoconfiguration e stateful autoconfiguration. DHCPv6 modalità *stateful* è usato per avere un controllo centralizzato degli host della rete, quindi nelle reti aziendali, mentre il metodo *stateless* è preferibile dove non c'è una gestione centrale, per esempio nelle reti domestiche.

6 Il DNS (Domain Name System)



Il Domain Name System (DNS) è un database distribuito usato dagli applicativi TCP/IP per il mapping dei nomi degli host e dei loro indirizzi IP. Gli applicativi accedono al DNS tramite un resolver che è un insieme di funzioni da usare per contattare il Name Server DNS. Il sistema DNS è usato anche all'interno delle reti locali private per risolvere i nomi dei computer (hostname). Infatti, grazie al DNS, si possono associare alle macchine dei nomi facili da ricordare; gli utenti possono connettersi ai server locali usando le stesse convenzioni usate su Internet (URL).

7 Problematiche di sicurezza



I problemi di sicurezza riguardano DHCP Server e Client non autorizzati. La vulnerabilità del DNS è data dal fatto che non è autenticato e che non è garantita l'integrità delle informazioni nei database distribuiti. Per il ruolo critico che il DNS riveste nello scenario di Internet, l'ICANN ha stabilito metriche e modalità per il controllo del DNS, individuando 5 indicatori importanti: coerenza, integrità, velocità, disponibilità e robustezza.

8 Il comando nslookup

Il comando nslookup è usato per interrogare il DNS. Si possono richiedere: la risoluzione di un nome o di un indirizzo IP, specifici Resource Record e di visionare il contenuto della memoria di un Name Server. Nslookup si può usare nella modalità interattiva e in quella non interattiva. Nella modalità interattiva si possono inviare più query e visualizzarne i singoli risultati, digitando solo nslookup senza opzioni. Nella modalità non interattiva si può inviare una sola query e visualizzarne il risultato. Di norma si usa quando si vuol interrogare un solo host e quindi si digita il nome dell'host dopo aver scritto nslookup.

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. BOOTP è un protocollo del livello Application. V F
2. Il DHCP Relay Agent serve a far dialogare, attraverso più reti, il DHCP Client con il DHCP Server. V F
3. L'acronimo DORA è usato per indicare le fasi di rinnovo del lease. V F
4. DHCP for IPv6 può essere configurato solo in modalità stateless. V F
5. Nel DNS i Name Server lavorano solo in modo ricorsivo. V F
6. Il DNS permette di ottenere altre informazioni su una risorsa, oltre al suo indirizzo IP. V F
7. Un Name Server è in ascolto sulla porta 53 sia TCP che UDP. V F
8. Nella risoluzione ricorsiva del DNS un client entra in contatto diretto con più di un server. V F
9. Quando si configura la connessione in rete di una macchina (PC, stampante, ecc.) è necessario assegnarle un indirizzo IP. V F
10. Un server BOOTP usa la porta UDP 67 per ricevere le richieste inviate dai client. V F
11. Un client BOOTP aspetta le risposte del server sulla porta UDP 68. V F
12. Tramite DHCP un host può ricevere solo l'IP. V F
13. Con la funzionalità APIPA il computer sceglie il proprio indirizzo IP nell'intervallo compreso tra 169.254.0.1 e 169.254.255.254. V F
14. Il DHCP in configurazione manuale è usato di solito per macchine server e router. V F
15. Il DHCP in configurazione automatica assegna gli IP in modo non permanente. V F
16. Il DHCP in configurazione dinamica assegna gli IP in modo permanente. V F
17. I messaggi scambiati tra DHCP Client e DHCP Server sono di tipo request e reply. V F
18. DHCP utilizza le stesse porte di BOOTP per il server e per il client. V F
19. Il DHCPv6 utilizza le stesse porte di BOOTP per il server e per il client. V F
20. Il campo htype delle PDU DHCP indica il tipo di hardware usato nella rete locale. V F
21. Con la stateful autoconfiguration l'host si configura autonomamente senza bisogno di aiuto da parte di altre macchine. V F
22. Con la stateless autoconfiguration un server fornisce all'host le informazioni di configurazione. V F
23. Il DNS è molto sicuro. V F
24. Il DNS è molto veloce. V F
25. La risoluzione inversa del DNS consiste nell'associare a un indirizzo IP il nome corrispondente. V F
26. Nslookup è un comando del Sistema Operativo usato per interrogare il DNS. V F
27. Nslookup è un comando presente solo nei sistemi Windows. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **BOOTP si è evoluto, come protocollo di configurazione di un host IP, in:**
   - A RARP
   - B TFTP
   - C HTTP
   - D DHCP

2. **Quale dei seguenti non è un messaggio DHCP?**
   - A DHCPCONFIRM
   - B DHCPREQUEST
   - C DHCPRELEASE
   - D DHCPNACK

3. **Quale tra i seguenti nomi non indica un componente del DNS?**

A Name Server
B Resolver
C Domain Resource Manager
D Domain Name Space

4. **La funzionalità APIPA serve a:**

A fornire all'host Windows in modo automatico un indirizzo IP
B fornire all'host Linux in modo automatico un indirizzo IP
C fornire all'host Windows in modo automatico un indirizzo MAC
D fornire all'host Linux in modo automatico un indirizzo MAC

5. **Il protocollo di trasporto e le porte per il DHCPv6 Client e Server sono rispettivamente:**

A UDP 546 e UDP 547
B UDP 68 e UDP 67
C TCP 546 e TCP 547
D TCP 68 e TCP 67

6. **Quale tra i seguenti campi non fa parte delle PDU DHCP?**

A Client IP Address
B My IP Address
C Server IP Address
D Router IP Address

7. **Il Domain Name Space specifica:**

A il formato delle PDU
B il modo per interrogare il DNS
C la struttura ad albero dei nomi di dominio
D il formato dei Resource Record

8. **Il DHCP può configurare gli host in modo:**

A statico o dinamico
B manuale, statico o dinamico
C manuale, automatico o dinamico
D automatico o dinamico

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Quali sono i vantaggi dei protocolli definiti a livello Application per la configurazione di un host?**

2. LEZIONE 2 **Quali parametri, oltre all'IP, un host può ricevere dal DHCP?**

3. LEZIONE 2 **Descrivi come un computer con Sistema Operativo Windows può ottenere un indirizzo IP in assenza del DHCP Server.**

4. LEZIONE 3 **Quali sono le 3 modalità usate da un amministratore per assegnare l'indirizzo IP a un host?**

5. LEZIONE 3 **Quali sono i vantaggi nell'utilizzo del protocollo DHCP?**

6. LEZIONE 4 **Spiega le 4 fasi dell'assegnazione dinamica dell'indirizzo IP con DHCP.**

7. LEZIONE 4 **Descrivi la macchina a stati finiti usata per rappresentare lo stato e le funzioni svolte dal DHCP Client.**

8. LEZIONE 5 **Com'è cambiato il protocollo DHCP per la gestione degli indirizzi IPv6?**

9. LEZIONE 5 **Descrivi le modalità di comunicazione Client/Server DHCPv6 a 2 e a 4 messaggi.**

10. LEZIONE 6 **Quali sono le principali componenti del DNS?**

11. LEZIONE 6 **Spiega come avviene l'interrogazione del DNS.**

12. LEZIONE 6 **Spiega come avviene il processo di risoluzione inversa dei nomi.**

13. LEZIONE 7 **Descrivi le 2 fasi di aggiornamento dinamico del record DNS.**

14. LEZIONE 7 **Quali sono le principali problematiche di sicurezza del DHCP?**

15. LEZIONE 7 **Quali sono le criticità del DNS?**

16. LEZIONE 8 **Descrivi il funzionamento delle modalità interattiva e non interattiva di nslookup.**

IN ENGLISH, PLEASE

CLIL

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

System and network configuration

Manually managing IP addresses is a complex and tedious task. DHCP simplifies this process by automating the assigning, tracking and reassigning of IP addresses. DHCP is based heavily on BOOTP, and its most significant new feature is dynamic allocation, which changes the way IP addresses are managed. While in traditional IP each device owns a particular IP address, in DHCP the server owns all IP addresses within an address pool, and each client leases an address from the server, usually only for a limited period of time.
DHCP Servers are devices programmed to provide DHCP services to clients. They manage address information and other parameters and respond to client configuration requests. DHCP Clients are TCP/IP devices configured to ask DHCP Server for configuration parameters. They send requests and receive replies, after which they are responsible for managing their own leases, including renewing or rebinding a lease. The client can extend its lease with further requests.
Another part of the network configuration is the DNS. The DNS provides a solution to the problem of translating symbolic names into computer readable IP addresses.

EXERCISES

Use the appropriate number to match words and meanings.

...	UDP	1	The act of terminating a DHCP lease.
...	Interface	2	It is a low-overhead protocol.
...	Lease length policy	3	A set of IP addresses.
...	Release	4	Contiguous blocks of IP addresses.
...	Scope	5	Verifying the user's identity.
...	APIPA	6	A point of interconnection between a computer and a network.
...	Address pool	7	How long the administrator wants client leases to last.
...	Hostname	8	Used by Windows DHCP Clients unable to locate a DHCP Server.
...	Authentication	9	Used to refer to a specific computer.

GLOSSARY

Broadcasting: a mechanism to send packets to all devices in a subnet. Broadcasting is limited to the broadcast domain, which includes only those computers able to talk to one another directly, without going through a router.
Device: a generic equipment connected to a network, such as switch, router, hub, gateway, firewall, access point, server, printer, personal computer, etc.
DHCP Client: a device that obtains its configuration information from a DHCP Server.
DHCP Relay Agent: relay agents are neither clients nor servers, but rather intermediaries that facilitate crossnetwork communication between servers and clients.
DHCP Server: a computer that provides DHCP configuration to multiple clients.
Diskless (or storageless) device: a device that does not include any form of storage (usually does not contain a hard disk), but instead relies on a small amount of read-only memory to connect to a network and to pick up its system files.
Domain: it is any subtree of the Domain Name Space. The Domain Name Space in the Internet is divided into three sections: generic domains, country domains, and inverse domains.
Lease: a limited period of time in which the DHCP Client uses the IP address assigned by DHCP Server.
Registrar (Domain Name Registry): any authorized organization that is responsible for the management of domain names.
Resolution: mapping symbolic names, such as www.mit.edu, to their corresponding IP addresses.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper configurare i servizi principali per le reti LAN.
- Saper verificare il funzionamento dei servizi configurati.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Sapere usare un simulatore di reti.
- Esporre i risultati della ricerca alla classe.

tempi

- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Simulatore di reti Packet Tracer.
- Dispositivo connesso a Internet.
- Carta e penna.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Un'azienda necessita di una rete costituita da 3 dipartimenti: AMMINISTRAZIONE, PRODUZIONE e MARKETING. Ogni dipartimento deve avere il suo server DHCP e il suo sito. Inoltre l'azienda vuole predisporre un server DNS per consentire l'accesso alla home page di ogni dipartimento da qualsiasi host della rete.
Realizzare una rete che soddisfi i requisiti richiesti dall'azienda.

File sorgenti
Scarica il file

SVOLGIMENTO

Progettiamo la rete con indirizzi privati rispettando il seguente **piano di indirizzamento**:

DIPARTIMENTO	RETE	SERVER DHCP e HTTP	ROUTER GATEWAY	DNS
AMMINISTRAZIONE	192.168.1.0/24	192.168.1.100	192.168.1.254	192.168.4.100
PRODUZIONE	192.168.2.0/24	192.168.2.100	192.168.2.254	192.168.4.100
MARKETING	192.168.3.0/24	192.168.3.100	192.168.3.254	192.168.4.100
SERVER DNS	192.168.4.0/24	192.168.4.100	192.168.4.254	

Ogni server DHCP avrà un address pool a partire dall'indirizzo 192.168.x.1.
Un possibile scenario è mostrato in FIGURA 1 dove, per ogni dipartimento, sono stati previsti 3 PC e una stampante oltre al server DHCP.

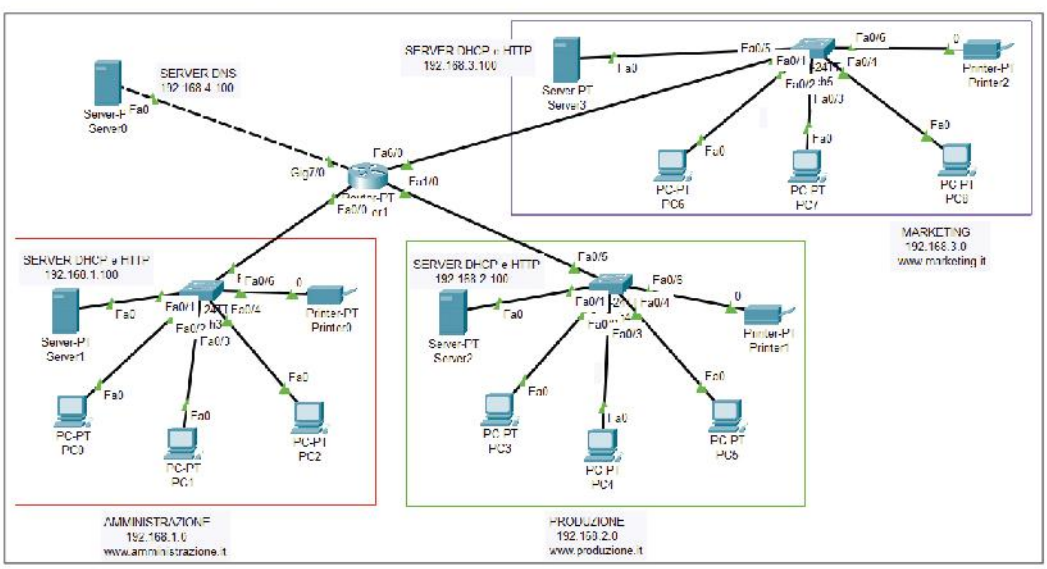


FIGURA 1 Scenario della rete aziendale con 3 dipartimenti e un server DNS

Avendo scelto il Router-PT vanno aggiunte 2 interfacce Ethernet come per esempio una FastEthernet e una GigabitEthernet. Le frecce nella **FIGURA 2** indicano i due slot inseriti.

#prendinota

Ricordarsi che bisogna spegnere il router per poter fare modifiche hardware e riaccenderlo al termine delle modifiche.
Ricordarsi poi di mettere ON tutte le interfacce del router che si devono utilizzare.

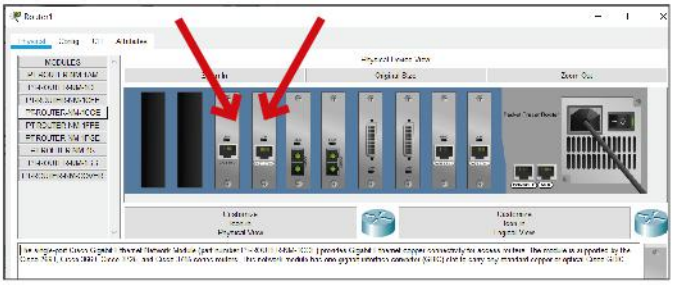


FIGURA 2 Aggiunta di due interfacce Ethernet sul router

Configuriamo il servizio DHCP sui 3 server affinché assegnino gli indirizzi IP, il gateway e il DNS agli host del dipartimento di appartenenza.

La **FIGURA 3** mostra la configurazione del servizio DHCP sul Server1 dell'AMMINISTRAZIONE.

In modo analogo procediamo con gli altri due server DHCP di dipartimento nel rispetto del piano di indirizzamento progettato. Sugli stessi server di dipartimento andiamo a personalizzare il file **index.html** inserendo, per esempio, il nome del dipartimento o modificandolo a piacere.

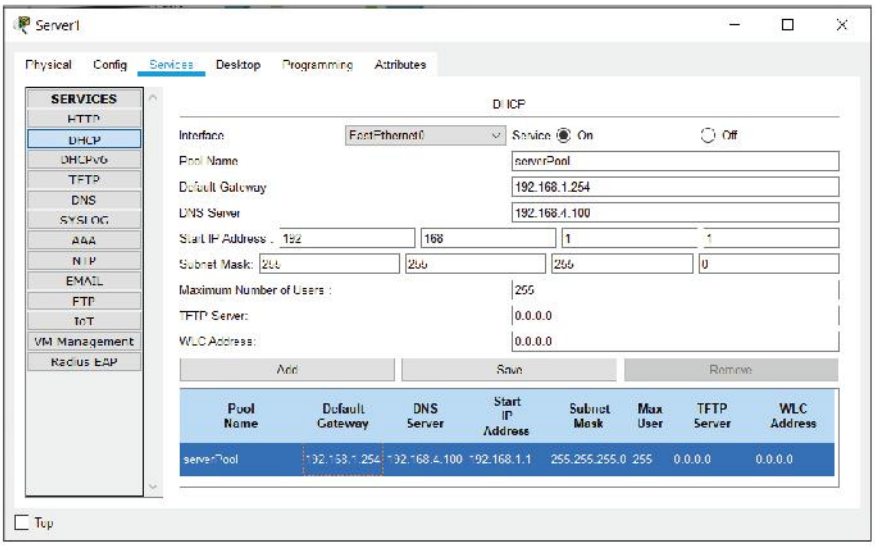


FIGURA 3 Configurazione del servizio DHCP sul Server1 dell'AMMINISTRAZIONE

Terminata la configurazione del servizio DHCP nei 3 dipartimenti, procediamo con la configurazione del servizio DNS sul Server0.

La **FIGURA 4** mostra il risultato della creazione dei 3 Resource Records di tipo A Record, uno per ogni sito web di dipartimento dell'azienda.

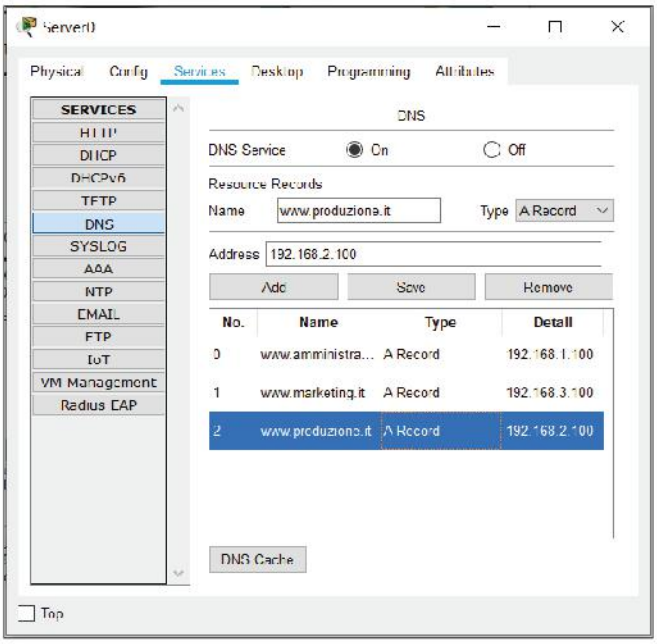


FIGURA 4 Configurazione del servizio DNS sul Server0

Testiamo infine il funzionamento della rete chiamando www.amministrazione.it dal Web Browser del PC8 del dipartimento MARKETING **(FIGURA 5)**.

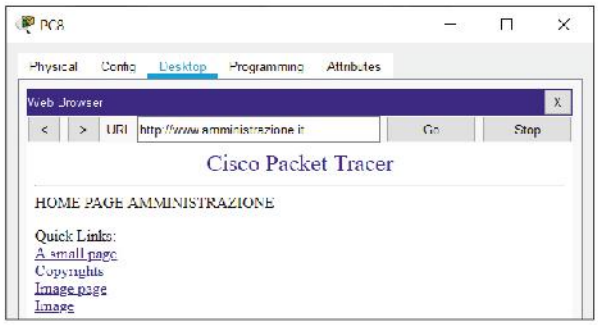


FIGURA 5 Web Browser del PC8

A CASA

- Ipotizza una tua soluzione al tema proposto.
- Leggi la proposta di SVOLGIMENTO per verificare se le tue ipotesi si adattano al caso preso in esame e se la trattazione proposta risulta completa.
- Modifica il file index.html di ogni sito web dipartimentale.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i casi che sono stati presentati.
- Stabilite quale caso rappresenta l'esempio migliore per completezza e che meglio si adatta alla soluzione del tema proposto.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Sono riuscito a configurare i servizi richiesti?	Sono riuscito a configurare la rete ma non i servizi richiesti. ☐	Sono riuscito a configurare solo alcuni dei servizi richiesti. ☐	Con la guida del docente e la collaborazione dei compagni ho configurato tutti i servizi richiesti. ☐	Ho configurato tutti i servizi richiesti autonomamente. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ

8 L'APPLICATION LAYER DEL TCP/IP

Guarda la **presentazione** dell'unità

IN QUESTA UNITÀ

conoscenze	abilità	competenze
Organizzare il software di comunicazione in livelli. Conoscere le principali applicazioni utilizzate nelle reti TCP/IP e i relativi protocolli. Conoscere i principali protocolli per le applicazioni multimediali.	Saper usare i numeri di porta opportuni per le comunicazioni Client-Server tra applicativi. Configurare il software di rete sugli host. Riconoscere le vulnerabilità dei protocolli di livello Application.	Conoscere il funzionamento dei principali protocolli di livello Application. Saper scegliere il tipo di protocollo in base all'applicazione che si vuol utilizzare. Configurare, installare e gestire sistemi di elaborazione dati e reti.

DDI+ Didattica Digitale Integrata Plus

FLIPPED CLASSROOM

A casa

- Leggi la Lezione 2 di questa Unità;
- esegui una ricerca sulle applicazioni attualmente più utilizzate su Internet che usano il paradigma Peer-to-Peer per svolgere attività di: condivisione di file, comunicazione (Instant Messaging), distribuzione di contenuti (Content Delivery Network);
- trasferisci la tua analisi in una tabella o mappa concettuale in cui elenchi le applicazioni che hai trovato, descrivendone caratteristiche, vantaggi e svantaggi.

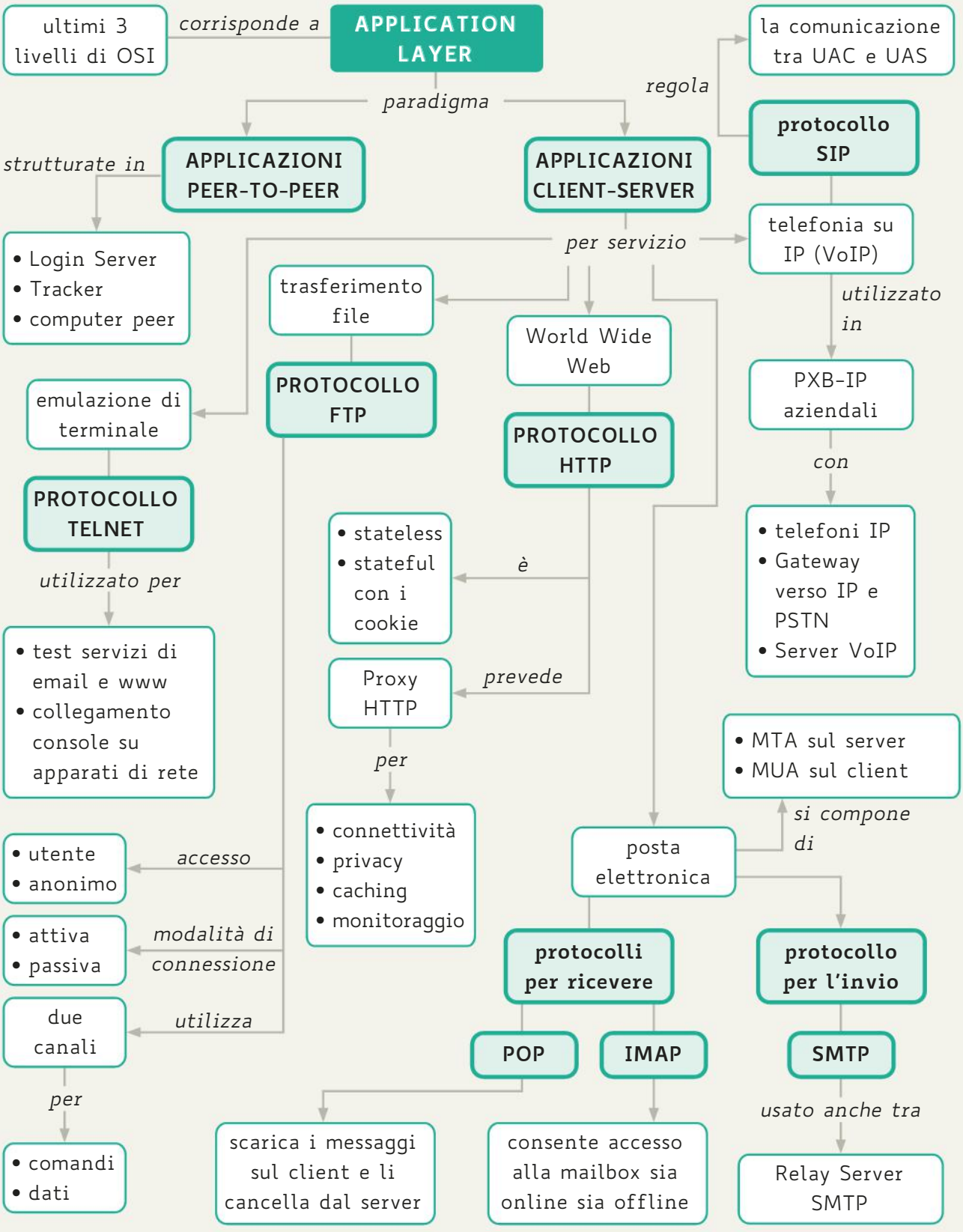
In classe

- Confrontate i risultati descritti nelle tabelle o mappe realizzate;
- discutete i motivi che spiegano le eventuali differenze, al fine di comprendere meglio il funzionamento dell'instradamento nelle reti.

MAPPA CONCETTUALE

DIDATTICA INCLUSIVA

Mappa modificabile



1 UNA VISIONE D'INSIEME DELLA RETE INTERNET

1.1 L'interconnessione delle reti

Nelle precedenti Lezioni e nel volume per il terzo anno, abbiamo descritto le varie componenti di Internet, la rete che interconnette miliardi di dispositivi in tutto il mondo. Più volte abbiamo ripreso e approfondito le varie parti, partendo dall'hardware, gli apparati di rete e le modalità di collegamento, per risalire lo stack TCP/IP fino ad arrivare alle applicazioni software, che offrono i servizi di rete agli utenti finali e sono descritte in questa Unità. Il viaggio nel mondo di Internet però non finisce qui, nel volume per il quinto anno riprenderemo nuovamente alcuni elementi per approfondire ulteriori caratteristiche e funzionalità, soprattutto dal punto di vista della sicurezza e della gestione. Inoltre, si descriveranno le modalità di accesso a Internet da rete mobile, con i nuovi protocolli e standard.

La **FIGURA 1** raffigura le varie componenti di Internet, evidenziando la suddivisione che abbiamo utilizzato più volte tra end system e intermediate system.

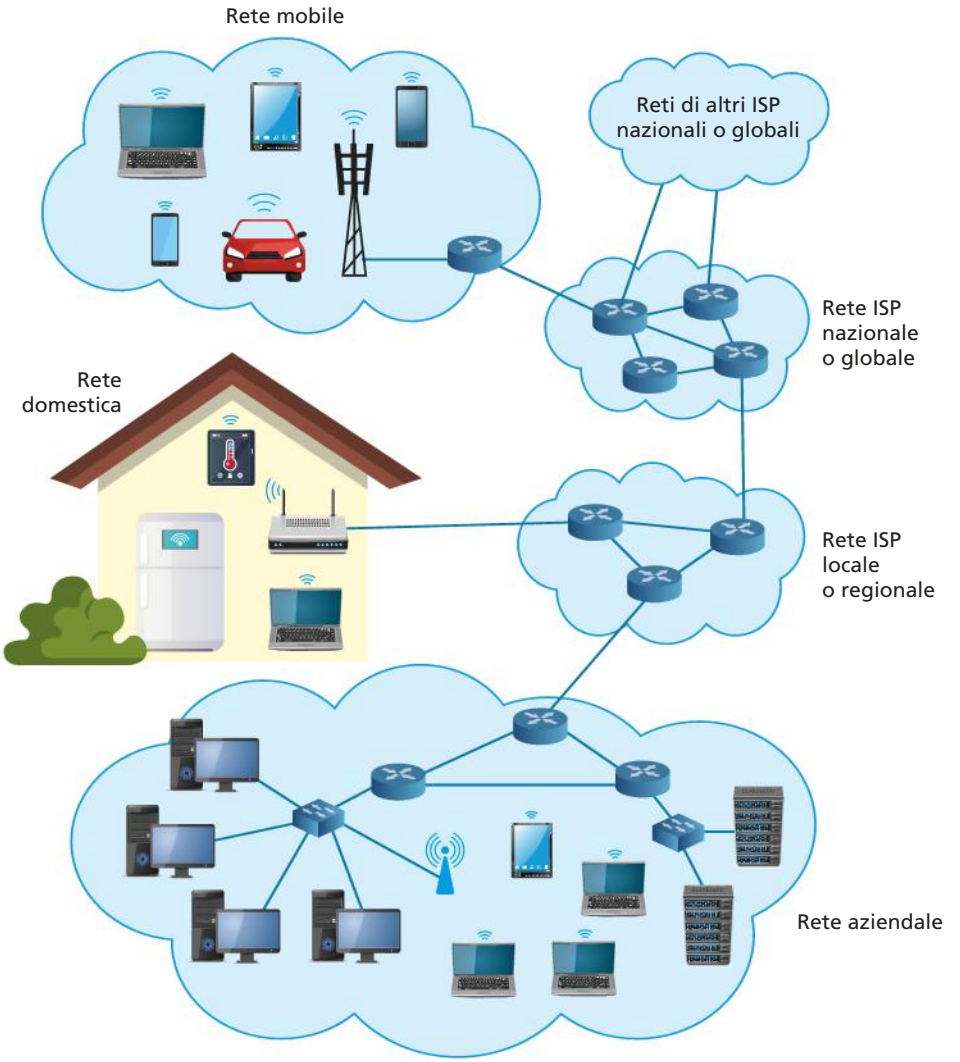


FIGURA 1 Le reti interconnesse con Internet

In passato gli end system erano soprattutto computer, ancora al giorno d'oggi definiamo Internet come una "computer network", ma l'evoluzione della tecnologia e il numero sempre più elevato di connessioni ha ampliato la tipologia di end system, includendo dispositivi di varia natura. Nelle reti domestiche, oltre a PC, laptop, tablet e smartphone, si connettono a Internet TV, console per il gioco, ma anche elettrodomestici e termostati che possono così essere gestiti da remoto. Analogamente la rete aziendale connette tra loro dispositivi wired e wireless come PC, smartphone, tablet e stampanti multifunzione, dai quali, sulla base delle politiche aziendali, si accede a Internet.
Nella Figura 1 si mostra la connessione delle reti periferiche, dove sono collocati gli end system, con le reti degli Internet Service Provider, formate dagli intermediate system, che inoltrano i pacchetti dati verso la destinazione, come visto nell'Unità 5.

1.2 I protocolli per la comunicazione su Internet

Nell'Unità 1 abbiamo descritto i modelli usati per organizzare la comunicazione in rete, spiegando come il modello ISO/OSI sia ormai da considerare un modello di riferimento, mentre TCP/IP è l'architettura a livelli implementata su Internet. IETF (Internet Engineering Task Force) è l'ente internazionale di standardizzazione che si occupa delle specifiche dei protocolli di Internet. I documenti pubblicati da IETF sono chiamati RFC (Request for Comments), spesso in questo volume abbiamo riportato l'abstract degli RFC per i protocolli più importanti. Nell'Unità 2 si è presentato un altro importante ente di standardizzazione, IEEE (Institute of Electrical and Electronics Engineers), che gestisce il progetto 802 pubblicando standard per reti PAN, LAN e MAN.
Nelle successive Unità 3, 4, 5 e 6 sono stati descritti i protocolli di comunicazione e gli standard utilizzati nei livelli Network e Transport.
Con l'Unità 7 siamo ancora saliti nello stack TCP/IP, prendendo in esame due servizi e protocolli di livello Application: il DHCP e il DNS. Si collocano al livello più alto in quanto sono applicazioni Client-Server, che offrono servizi fondamentali per la comunicazione su Internet, strettamente legati ai protocolli dei livello inferiore.
In questa Unità si prendono in esame i protocolli che permettono agli utenti di un'applicazione di comunicare. Per esempio il protocollo HTTP è utilizzato per la comunicazione tra le applicazioni web client (i browser sul dispositivo dell'utente) e le applicazioni web server (sui server del provider), realizzando così il servizio noto come WWW (World Wide Web).
Le applicazioni che usano Internet sono applicazioni distribuite, che coinvolgono molti sistemi che scambiano dati tra loro. Per questo motivo il software sviluppato usufruisce dei servizi offerti dal livello Transport tramite le socket, le interfacce di rete viste nell'Unità 6.

A livello più basso Internet è una rete formata da hardware e software che permette di interconnettere due dispositivi che necessitano di comunicare. A livello più alto, Internet è un'infrastruttura che fornisce servizi alle applicazioni distribuite su diversi sistemi.

FISSA LE CONOSCENZE

- Descrivi le tipologie di end system che si connettono alla rete Internet.
- Con intermediate system quali tipologie di apparati si identificano?
- Spiega che cosa significa che Internet è un'infrastruttura che fornisce servizi alle applicazioni distribuite.

2 IL LIVELLO APPLICATION E I SUOI PROTOCOLLI

2.1 I protocolli del livello Application

Il livello Application dell'architettura TCP/IP ingloba le funzionalità svolte dai 3 livelli più alti del modello OSI: Session, Presentation e Application. Questa scelta ha permesso ai programmatori di avere un buon grado di flessibilità nello sviluppo delle applicazioni.

Con l'analisi delle funzionalità tipiche del livello Application si è giunti all'origine dei dati che attraversano la rete: a questo livello si implementa l'interfaccia tra l'utente e la rete, la comunicazione viene convertita in dati che possono essere trasferiti attraverso una rete (FIGURA 2).

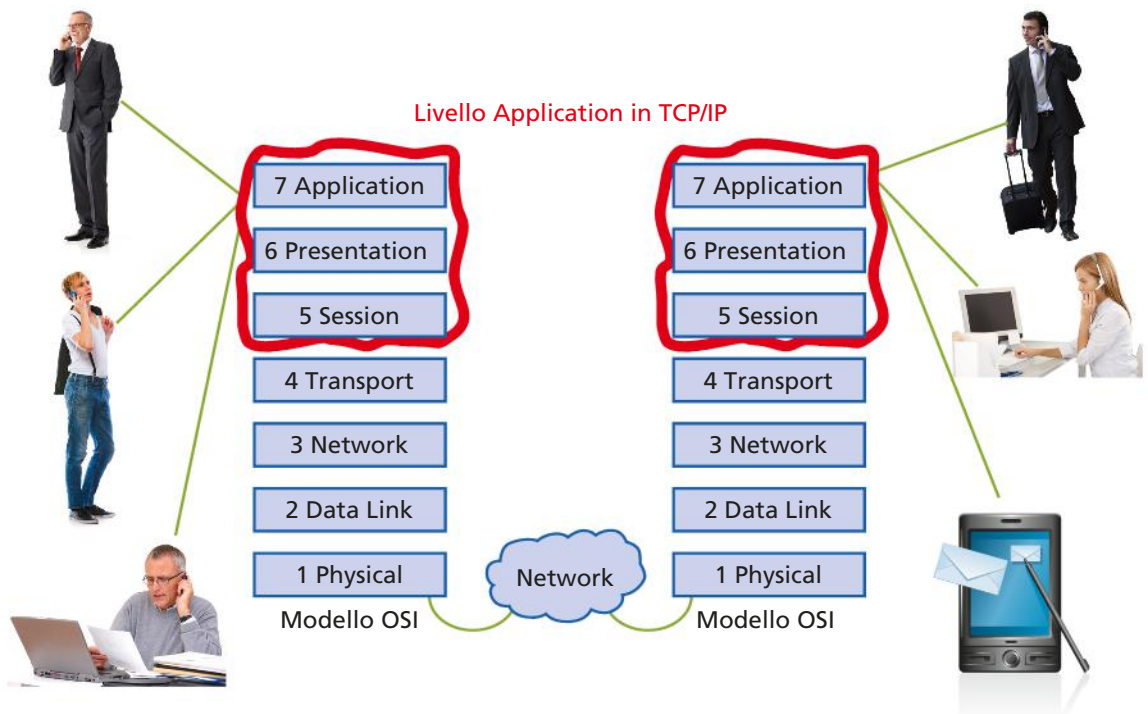


FIGURA 2 Il livello Application di TCP/IP corrisponde agli ultimi 3 livelli del modello OSI

Un protocollo di livello Application definisce:

- i **tipi di messaggi** scambiati, per esempio: richiesta e risposta;
- la **sintassi** di ciascun tipo di messaggio, per esempio quanti sono i campi presenti e quanto spazio occupano;
- la **semantica** dei vari campi, qual è il contenuto informativo che trasportano;
- le **regole** che sottendono al dialogo, quando e come l'applicazione invia un messaggio o risponde a uno ricevuto.

Nelle Lezioni seguenti vedremo queste definizioni applicate ai protocolli relativi alle applicazioni più diffuse.

I protocolli del livello Application supportano la comunicazione tra i processi client e server. Nelle Lezioni seguenti esaminiamo i più diffusi: dagli storici protocolli Telnet

ed FTP all'HTTP per il web e SMTP per la posta elettronica. Un importante protocollo di livello Application comunemente usato per la gestione delle reti IP è il protocollo SNMP (Simple Network Management Protocol) che sarà analizzato nel volume per il quinto anno dove si affronta la tematica della gestione della rete.
Nell'Unità 5 del volume del terzo anno, avevamo descritto i due modelli Client-Server e Peer-to-Peer applicati alle reti, li ritroviamo nei protocolli del livello Application:

- **Client-Server (C/S)**: è l'architettura software tra le più diffuse; fin dalle origini di Internet, infatti, la utilizzano applicazioni come il WWW, la posta elettronica e il file transfer; ogni servizio applicativo ha una componente client e una server:
  - il server è sempre attivo in attesa di ricevere le richieste dai molti client, ha un indirizzo IP assegnato staticamente e una porta TCP o UDP, di tipo Well Known nel caso delle applicazioni più diffuse;
  - un client si connette solo nel momento in cui deve comunicare con il server, sovente ha un indirizzo IP assegnato dinamicamente; da notare che i client non comunicano direttamente tra loro;
- **Peer-to-Peer (P2P)**: è una comunicazione tra pari, gli utenti scambiano informazioni tra loro in modo cooperativo, mediante specifici protocolli; i peer non sono sempre connessi come i server e cambiano spesso l'indirizzo IP, quindi la gestione risulta più complessa. Le applicazioni Peer-to-Peer possono essere di file sharing, per esempio BitTorrent, di videoconferenza o telefonia su Internet come Skype e altre ancora.

In generale, per usufruire di un servizio applicativo è necessario averne l'autorizzazione, quindi gli utenti devono disporre di un account che viene loro concesso dall'amministratore del server remoto e che useranno ogniqualvolta vorranno inviare delle richieste al server.

2.2 Applicazioni Peer-to-Peer

Nelle reti denominate Peer-to-Peer non c'è distinzione tra computer server e computer client. Infatti in questo modello ogni computer è considerato alla pari degli altri e sono i singoli utenti a decidere quali risorse del proprio computer condividere.
Quando si passa da una rete P2P alle applicazioni P2P, lo scenario cambia, infatti un'applicazione P2P permette al computer di agire sia come client sia come server all'interno di una stessa sessione di comunicazione. Ciò non è realizzabile a livello di rete P2P dove è consentito che un computer svolga sia il ruolo di client sia di server, ma su due distinte sessioni di comunicazione.

> Un'applicazione Peer-to-Peer non deve utilizzare una rete Peer-to-Peer necessariamente, può anche funzionare con reti Client-Server.

La FIGURA 3 mostra come è strutturata una generica applicazione P2P, sono presenti:

- un portale web, denominato **Login Server**, a cui si connette un peer per verificare la disponibilità dei servizi desiderati;
- un server, denominato **Tracker**, che fornisce al nuovo utente l'elenco dei peer disponibili;
- i computer **peer**, che sono macchine anonime che si collegano al sistema quando ne hanno necessità.

#prendinota

Una peculiarità delle architetture P2P è l'**auto-scalabilità**, per esempio: in un'applicazione di file sharing, alcuni peer scaricano un file generando un certo traffico in rete. Nel momento in cui mettono a loro volta a disposizione il file per altri peer, aumentano automaticamente la capacità del sistema.

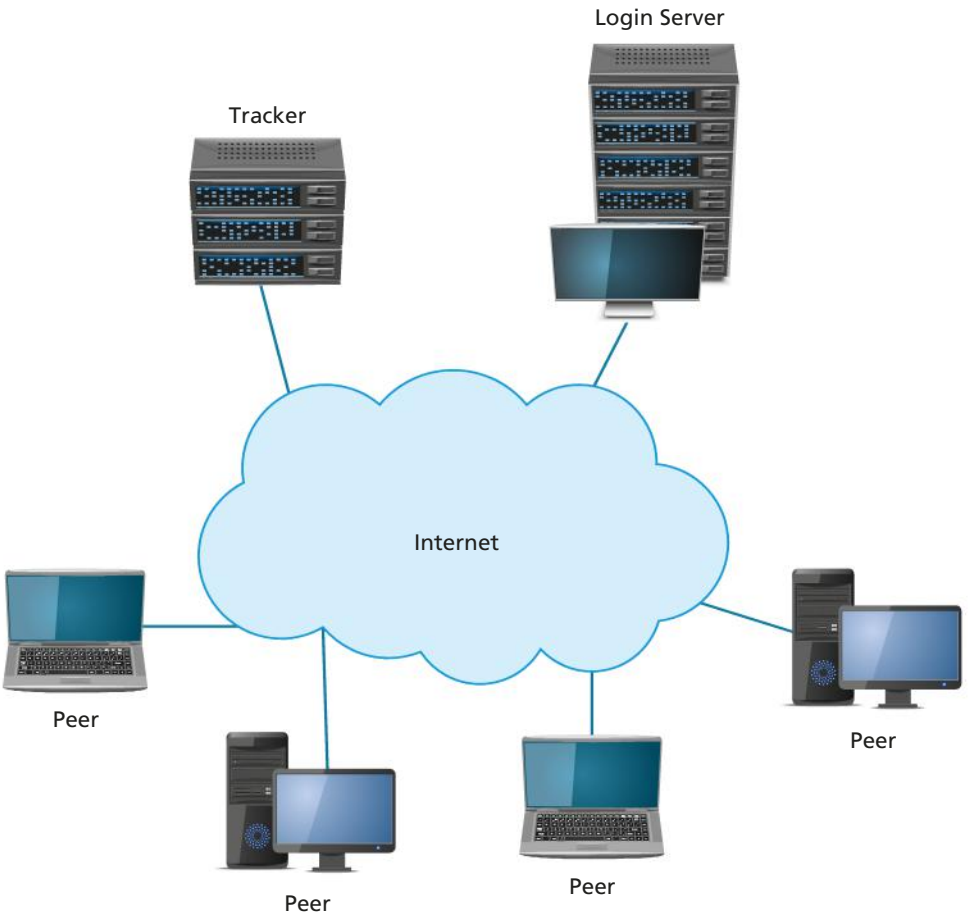


FIGURA 3 Architettura di una generica applicazione P2P

Un'applicazione P2P è caratterizzata da 3 aspetti fondamentali:

1. **ricerca**: quando un nuovo peer utilizza l'applicazione P2P per prima cosa ricerca quali servizi, dati e peer sono disponibili;
2. **locazione**: il nuovo peer necessita di alcune informazioni utili a trovare il tracker dell'applicazione, per esempio il suo indirizzo IP, e i peer che hanno i dati che desidera. Inoltre, anche il nuovo peer deve fornire al tracker informazioni utili per la sua localizzazione e sui dati che possiede e può rendere disponibili agli altri peer;
3. **trasferimento dei dati**: le applicazioni P2P utilizzano approcci diversi per realizzare le operazioni di upload e download dei dati desiderati dal peer. I più frequenti sono il metodo **push**, in cui è il peer che carica i dati a stabilire i peer destinatari degli stessi, e il metodo **pull** in cui è il peer che vuole scaricare i dati a inviare la richiesta a un insieme di potenziali peer da cui effettuare il download.

FISSA LE CONOSCENZE

- Qual è la differenza tra il livello Application del modello OSI e quello dell'architettura TCP/IP?
- Quali modelli di comunicazione si possono implementare a livello Application?
- Descrivi le caratteristiche delle applicazioni Peer-to-Peer.

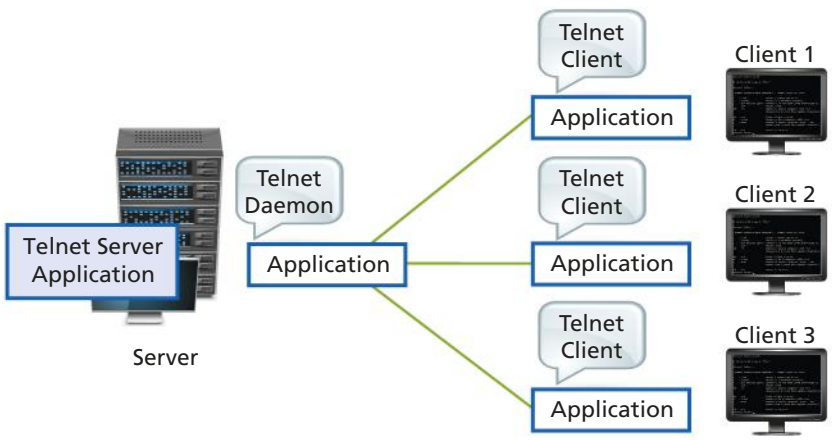
3 TELNET: IL PROTOCOLLO PER L'EMULAZIONE DI TERMINALE

3.1 La sessione Telnet

Telnet è un protocollo Client-Server: la componente client di Telnet è un'applicazione di emulazione di terminale (testuale) che permette agli utenti di un sistema di accedere ad applicazioni che si trovano su host remoti, come se fossero direttamente connessi a tali sistemi. L'host di destinazione deve avere la componente server di Telnet. Il procedimento può essere ripetuto in quanto dall'host remoto ci si può connettere a un altro host e così via.

Nella FIGURA 4 si mostra un tipico scenario Client-Server in cui viene utilizzato il protocollo Telnet: più client Telnet richiedono la connessione a un server Telnet. Sul server è in esecuzione un servizio chiamato Telnet Daemon. Le richieste dei client devono essere gestite sul computer server contemporaneamente e in modo separato. Per far ciò il protocollo Telnet si affida alle funzionalità offerte dai livelli sottostanti.

FIGURA 4 Tipico scenario Client-Server dell'applicazione Telnet



Vediamo un semplice esempio di utilizzo di questa applicazione: un client Telnet viene avviato su un computer Windows per connettersi a un computer remoto Linux. Sul computer Windows si apre una finestra che consente all'utente di lavorare direttamente sul computer Linux. L'utente deve possedere un account che gli consenta l'accesso al computer remoto, quindi prima di poter inviare comandi ed eseguire applicazioni, l'utente deve essere autenticato.

In generale, non si usa Telnet per connettere un computer Windows con un altro computer Windows, perché Windows ha una propria modalità per connettere i suoi computer attraverso una rete: la funzionalità Connessione desktop remoto. Inoltre, dalla versione Windows Vista non è più disponibile il comando Telnet dal Prompt dei comandi, in modalità predefinita, però è possibile ripristinarlo dal Pannello di controllo.

esercizio

➜ PROBLEMA

Abilitare Telnet su Windows 10.

➜ SVOLGIMENTO

Su Windows 10 il client Telnet è disabilitato, ma può essere abilitato dal Pannello di controllo: in Programmi selezionare la voce Attiva o disattiva funzionalità di Windows e spuntare la casella relativa a Telnet.

Per attivare Telnet è poi necessario eseguire l'applicazione Prompt dei comandi come amministratore (tasto destro) e digitare:

```
C:\>dism /online /Enable-Feature /FeatureName:TelnetClient
```

3.2 Lo standard del protocollo Telnet

Le specifiche di Telnet furono definite agli inizi degli anni Ottanta e sono contenute negli RFC 854 e RFC 855.

IN ENGLISH PLEASE

Network Working Group — J. Postel
Request for Comments: 854 — J. Reynolds
ISI
Obsoletes: NIC 18639 — May 1983

TELNET PROTOCOL SPECIFICATION

1. INTRODUCTION

The purpose of the TELNET Protocol is to provide a fairly general, bi-directional, eight-bit byte oriented communications facility. Its primary goal is to allow a standard method of interfacing terminal devices and terminal-oriented processes to each other. It is envisioned that the protocol may also be used for terminal-terminal communication (*linking*) and process-process communication (distributed computation).

#prendinota

Il protocollo Telnet è uno dei primi protocolli applicativi creato per la suite TCP/IP. Come altri servizi applicativi nati insieme a Internet, Telnet è ormai poco usato su reti pubbliche, nella sua forma originale, a causa della scarsa sicurezza che offre.

Telnet utilizza **TCP** come protocollo di trasporto e la porta **23**. I dati e i comandi sono trasmessi in formato ASCII a 8 bit.

Alcuni dei principali comandi di Telnet sono:

- **open host [port]** apre una sessione Telnet su host usando port;
- **close** chiude la sessione Telnet;
- **display** mostra i parametri relativi alla sessione;
- **send code** invia caratteri speciali al server;
- **status** visualizza lo stato attuale della sessione.

I comandi che invia il client al server devono essere preceduti da un **carattere di escape** per far sì che il server interpreti le informazioni ricevute come comandi e non come dati.

#prendinota

Telnet è ancora utilizzato per il test dei servizi di rete presenti sui web server e di posta elettronica, in quanto permette di inviare, in modo semplice, i comandi e di esaminarne le risposte. Telnet è anche utilizzato per collegarsi come console ad apparati di rete, per esempio per accedere a un router remoto.

La versione originale di Telnet offre un livello minimo di sicurezza per controllare l'accesso al computer remoto, realizzato con username e password, che, però, viene a cadere dal momento che i dati viaggiano in chiaro sulla rete.
L'impiego di SSH (Secure SHell) rende il protocollo più sicuro grazie all'uso della crittografia.

Molte sono le implementazioni del protocollo Telnet che si possono scaricare gratuitamente da Internet. Un esempio, valido sia in ambiente Unix che Windows, è il software **PuTTY** che offre un client Telnet con crittografia, utilizza infatti SSH-2.
Il sito web è: *http://www.chiark.greenend.org.uk/~sgtatham/putty*.

FISSA LE CONOSCENZE

- Qual è lo scopo originario del protocollo applicativo Telnet?
- In quali altri casi può essere utilmente impiegato Telnet?
- Telnet è l'unica modalità per potersi connettere a un computer remoto?
- Descrivi alcuni comandi tipici di Telnet.

4 FTP: IL PROTOCOLLO PER IL TRASFERIMENTO DI FILE

4.1 Gli standard del protocollo FTP

Il **File Transfer Protocol** (**FTP**) è un protocollo per il trasferimento di file tra un computer client e un server. FTP è stato standardizzato negli anni Ottanta e le sue specifiche sono descritte nell'RFC 959.

IN ENGLISH PLEASE

Network Working Group — J. Postel
Request for Comments: 959 — J. Reynolds
ISI
Obsoletes RFC: 765 (IEN 149) — October 1985

FILE TRANSFER PROTOCOL (FTP)

1. INTRODUCTION

The objectives of FTP are 1) to promote sharing of files (computer programs and/or data), 2) to encourage indirect or implicit (via programs) use of remote computers, 3) to shield a user from variations in file storage systems among hosts, and 4) to transfer data reliably and efficiently. FTP, though usable directly by a user at a terminal, is designed mainly for use by programs.

A distanza di pochi anni dalla specifica di FTP venne definita una versione più leggera denominata **Trivial File Transfer Protocol** (**TFTP**), specificata nell'RFC 1350. La novità fu la sostituzione del protocollo di trasporto TCP, utilizzato in FTP, con **UDP**, **port 69**. TFTP è ancora usato per trasferire file all'interno di una rete locale, per via della maggior sicurezza che offre la LAN (TFTP non prevede né autenticazione né cifratura) e della bassissima percentuale di pacchetti errati o persi.

IN ENGLISH PLEASE

Network Working Group — K. Sollins
Request For Comments: 1350 — MIT
STD: 33 — July 1992
Obsoletes: RFC 783

THE TFTP PROTOCOL (REVISION 2)

[...]

1. Purpose

TFTP is a simple protocol to transfer files, and therefore was named the Trivial File Transfer Protocol or TFTP. It has been implemented on top of the Internet User Datagram protocol (UDP or Datagram) [2] so it may be used to move files between machines on different networks implementing UDP. (This should not exclude the possibility of implementing TFTP on top of other datagram protocols.) It is designed to be small and easy to implement. Therefore, it lacks most of the features of a regular FTP. The only thing it can do is read and write files (or email) from/to a remote server. It cannot list directories, and currently has no provisions for user authentication.

#prendinota

Un esempio di applicazione TFTP, gratuita, è **Solarwinds TFTP Server**, utilizzata soprattutto per lavorare sugli apparati di rete per operazioni di upload, backup o di configurazione.

Attualmente ci sono molti modi per trasferire file attraverso una rete dati (come Internet) che utilizzano tecnologie non specificatamente pensate a questo scopo, per esempio email, instant messaging, chat e web server. Tutte queste applicazioni offrono il vantaggio di un'interfaccia familiare a chi le usa quotidianamente, ma mancano della robustezza che offre un'applicazione di file transfer creata a questo scopo.

4.2 La connessione tra client e server FTP

A differenza di altri protocolli, FTP utilizza **due canali** per la comunicazione tra client e server:

- un canale viene utilizzato per l'invio di **comandi**, e relative risposte, tra client e server; questo canale viene sempre aperto in direzione client → server e utilizza la **porta 21**, chiamata porta di controllo;
- l'altro canale è utilizzato per l'invio dei **dati**, viene quindi aperto in direzione server → client e utilizza la **porta 20** , chiamata porta dati.

La connessione tra client e server può avvenire secondo due modalità: FTP active mode e FTP passive mode.

FTP ACTIVE MODE

La FIGURA 5 mostra lo scambio di messaggi tra client e server FTP nella modalità attiva: il client si connette da una porta qualsiasi **N** (con N > 1.023) alla porta di controllo del server, la porta 21, e si mette in ascolto sulla porta dati N + 1 inviando al server il comando **Port N+1**. Il server si connette alla porta dati specificata dal client (N + 1) utilizzando la propria porta dati, la porta 20.

1. Il client apre la porta comandi (5030) per il colloquio con il server e gli comunica il numero di porta da usare per i dati (5031).
2. Il server, usando la porta 21, conferma la connessione con il client.
3. Il server apre la porta dati 20 per la comunicazione con la porta 5031 del client.
4. Il client invia un riscontro (ACK) di conferma ricezione. Da questo punto in poi inizia il trasferimento dei dati dalla porta 20 alla porta 5031 (Data Channel).

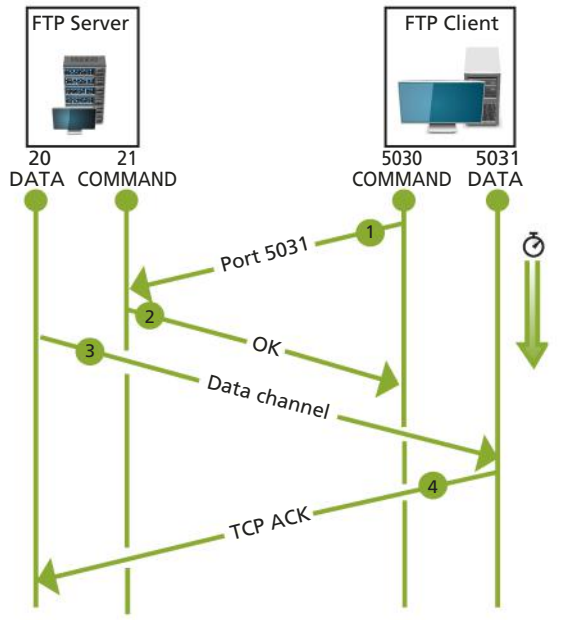


FIGURA 5 Modalità attiva di FTP

Questa modalità comporta problemi di sicurezza nel lato client: infatti non è il client FTP che si connette alla porta dati del server, esso segnala solo al server su quale porta è in ascolto. Sarà il server che aprirà un canale verso questa porta per l'invio dei dati. Quindi se nel lato client è presente un firewall questi rileva un tentativo di intrusione dall'esterno e lo blocca.

Per questo motivo è attualmente sconsigliato l'uso di FTP active mode, per poter garantire la sicurezza della rete locale (intranet).

FTP PASSIVE MODE

La FIGURA 6 mostra lo scambio di messaggi tra client e server FTP nella modalità passiva: il client FTP inizia entrambe le connessioni con il server, sia comandi che dati, risolvendo in questo modo il problema del filtraggio della connessione da parte del firewall lato client. Il client apre localmente due porte **N** e **N+1** (con N > 1.023) e con la porta N, la porta comandi, contatta il server sulla porta 21. A questo punto invia un comando **PASV** che permette al server di aprire una porta casuale **P** (con P > 1.023) e inviare il comando **Port P** al client sulla sua porta comandi N. Il client inizia allora la connessione dalla sua porta dati N + 1 alla porta P sul server per ricevere i dati.

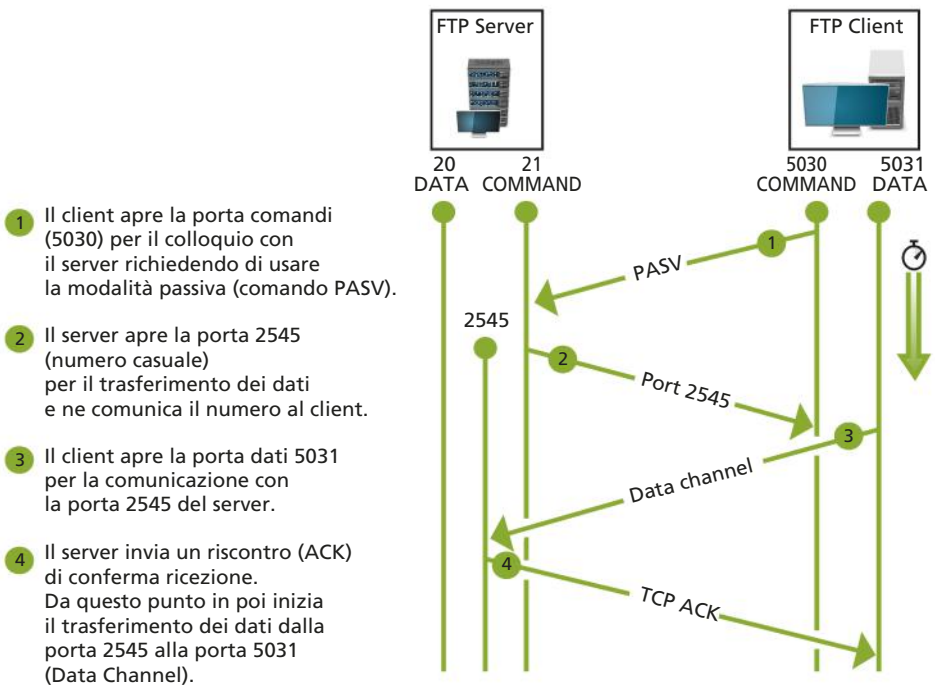


FIGURA 6 Modalità passiva di FTP

4.3 Le modalità di accesso al server FTP

FTP ha due modalità predefinite di accesso: **utente** e **anonima**.
La modalità utente prevede un accesso al server FTP con username e password, mentre nella modalità anonima si ha accesso come utente **anonymous**. Quest'ultima è molto utilizzata per scambio di dati pubblici, come modulistica, codice, ecc.
La modalità di accesso anonima presenta due limitazioni:

- è necessario limitare l'accesso alle sole informazioni che si vogliono diffondere;
- non deve essere permesso l'uso di FTP server per la distribuzione di materiale di terzi (per esempio si può rendere la cartella di upload accessibile solo in scrittura e non in lettura).

Se possibile, per la sicurezza del sistema, è meglio evitare di usare l'accesso anonimo. Nel caso in cui sia proprio necessario, esso deve essere configurato correttamente

e amministrato con attenzione, soprattutto se si vuole rendere accessibili in upload, quindi scrivibili, delle directory (o cartelle) nelle aree FTP anonymous.

CLIENT E SERVER FTP

Sono disponibili vari software FTP, a pagamento o distribuiti con licenza open source. Solitamente i software client FTP sono gratuiti e permettono di collegarsi a un server FTP per caricare un file trasferendolo dal proprio computer o per scaricare un file dal server. L'operazione di upload di un file è tipicamente svolta quando si utilizza un servizio di hosting di un sito web e si necessita di caricare le pagine web sul server per pubblicarle. Uno dei software client FTP più diffusi è **Filezilla**, gratuito e disponibile per sistemi Windows, Mac e Linux. Scaricando questo software sul proprio computer è possibile in modo semplice e sicuro caricare i file delle nostre pagine web sul web server del provider. La sicurezza è data dall'utilizzo della crittografia nel trasferimento dei dati, tecnica non prevista nella specifica originale di FTP, come descritto nel paragrafo successivo. Filezilla offre anche il software per il server FTP, ma solo per sistemi Windows. Il sito del progetto da cui scaricare il software e la documentazione è: *https://filezilla-project.org*.

4.4 Le vulnerabilità di FTP

I maggiori problemi di sicurezza di FTP sono riconducibili al fatto che le specifiche non prevedono la cifratura delle informazioni scambiate tra client e server:

- **password in chiaro:** le password viaggiano in chiaro attraverso la rete e sono facilmente intercettabili con strumenti come gli sniffer che consentono di analizzare il traffico tra client e server;
- **dati in chiaro**: anche i dati vengono trasferiti senza essere crittografati, anch'essi sono dunque intercettabili.

La soluzione a questi problemi è stata una nuova specifica di FTP denominata **FTP over TLS** (**FTPS**, RFC 4217) che aggiunge un livello tra Transport (TCP) e Application (FTP), per la gestione della crittografia, utilizzando il protocollo **Transport Layer Security** (TLS). TLS è una versione più recente del protocollo Secure Sockets Layer (SSL).

Altri problemi di sicurezza sono legati a:

- **sessione in due processi**: la necessità di avere due processi per ogni connessione rende più semplice effettuare manovre malevole;
- **permessi utente**: i permessi di accesso FTP vanno incrociati con i permessi utente sul server in modo da limitare lo spazio su disco e le operazioni sui file.

Con il diffondersi del World Wide Web, molti utenti preferiscono usare il browser come FTP client. La maggior parte dei browser supporta solo la modalità passiva quando si accede con **ftp://URL**.

Esercizio commentato
Trasferimento di una cartella con FTP

FISSA LE CONOSCENZE

- Qual è lo scopo dei protocolli applicativi FTP e TFTP?
- Quali modalità di colloquio tra un client FTP e un server FTP possono essere implementate?
- Quali sono le porte Well Known utilizzate per FTP?
- Quali modalità di accesso al server sono previste in FTP?
- Quali sono le maggiori vulnerabilità del protocollo FTP?

5 HTTP: IL PROTOCOLLO PER LE APPLICAZIONI WEB

5.1 HTTP e WWW

HTTP (**HyperText Transfer Protocol**) è il protocollo di livello Application usato nell'applicazione Client-Server **WWW (World Wide Web)**, la parte di Internet più usata e cresciuta più velocemente. Il protocollo HTTP regola lo scambio di messaggi tra il web server e il web client (si parla anche di HTTP server e HTTP client o anche di WWW server e WWW client). Nell'uso comune il client corrisponde al browser e il server al sito web.

La FIGURA 7 mostra un semplice esempio di comunicazione nel WWW che utilizza il protocollo HTTP:

- il **browser** è il programma client dell'applicazione, esempi sono Edge, Firefox, Chrome, Opera. Il browser svolge due funzioni fondamentali: inoltra la richiesta di una pagina web al server (HTTP request) e presenta i dati ricevuti (HTTP response) all'utente. Le pagine web sono create con un linguaggio chiamato #HTML (HyperText Markup Language);
- il **web server** contiene le pagine web del sito e risponde alle richieste che riceve dai web client, i più diffusi web server sono Apache, open source e multi-piattaforma, e Internet Information Services (IIS) per i sistemi Windows.

#techwords

HTML
È il linguaggio nato per realizzare i siti web, utilizzato anche per la creazione di contenuti e di applicazioni mobile.
Non è un linguaggio di programmazione ma di markup (contrassegno), che permette di indicare come disporre i vari elementi all'interno di una pagina web.

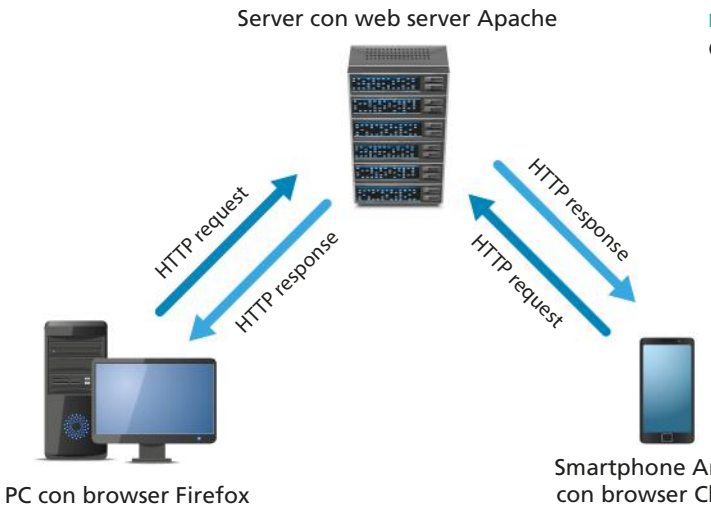


FIGURA 7 Comunicazione Client-Server con HTTP

Il WWW e i suoi protocolli sono nel tempo diventati la piattaforma di comunicazione per applicazioni quali la posta elettronica, per esempio Gmail, per distribuzione di video, per esempio YouTube, e per la maggior parte delle applicazioni mobile che usano Internet.

GLI STANDARD HTTP/1.0, HTTP/1.1, HTTP/2 E HTTP/3

La prima versione del protocollo HTTP (HTTP/1.0) è stata standardizzata in **RFC 1945** in cui sono definite le modalità di scambio dei messaggi tra client e server e la struttu-

#prendinota

W3C è la più importante organizzazione internazionale per il WWW, formata da aziende informatiche, operatori telefonici, organizzazioni no-profit, università e centri di ricerca. Oltre alla definizione degli standard, sviluppa software open source per il WWW.

ra dei messaggi HTTP. Lo standard fu il risultato del lavoro congiunto di IETF e di **W3C (World Wide Web Consortium).**

IN ENGLISH PLEASE

Network Working Group
Request for Comments: 1945
Category: Informational

T. Berners-Lee
MIT/LCS
R. Fielding
UC Irvine
H. Frystyk
MIT/LCS
May 1996

Hypertext Transfer Protocol -- HTTP/1.0

Abstract

The Hypertext Transfer Protocol (HTTP) is an application-level protocol with the lightness and speed necessary for distributed, collaborative, hypermedia information systems. It is a generic, stateless, object-oriented protocol which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods (commands). A feature of HTTP is the typing of data representation, allowing systems to be built independently of the data being transferred.
HTTP has been in use by the World-Wide Web global information initiative since 1990. This specification reflects common usage of the protocol referred to as "HTTP/1.0".

In pochi anni, grazie alla diffusione del browser grafico Mosaic, il WWW crebbe enormemente e divennero evidenti alcuni limiti della prima versione di HTTP:

- la mancanza di meccanismi di sicurezza, perché non erano previste l'autenticazione e la crittografia dei dati;
- non era possibile ospitare più siti web sullo stesso server;
- per ogni richiesta era necessario creare una connessione separata con il server; per esempio se nella pagina web erano presenti delle immagini, il client doveva inoltrare ulteriori richieste al server per scaricarle.

Il protocollo fu ampliato e venne specificata una nuova versione **HTTP/1.1** pubblicata in **RFC 2616** nel 1999. Queste specifiche sono state completamente riviste nel 2014 e descritte nei nuovi **RFC 7230, 7231, 7232, 7233, 7234 e 7235**.
Nel 2015 esce una nuova versione denominata **HTTP/2** e descritta in RFC 7540 e in RFC 8740 che esprime la semantica del web in modo più efficiente e utilizza TLS (Transport Layer Security).
In via di standardizzazione è la nuova versione **HTTP/3** che utilizza UDP in sostituzione di TCP.
È già supportata da browser come Chrome e Firefox dal 2019.

Le nuove versioni HTTP/2 e HTTP/3 non rendono obsolete le precedenti versioni di HTTP.

5.2 Le modalità di lavoro di HTTP

Gli **#hyperlink** (collegamenti ipertestuali) permettono una facile navigazione: quando si fa clic su un hyperlink si dirige il browser su una nuova pagina.
Ogni pagina web ha un suo indirizzo simbolico detto **URL** (Uniform Resource Locator), per esempio *http://www.azienda.com/news/* (TABELLA 1), dove la parte iniziale **http://** indica al browser il protocollo da usare e la seconda parte **www** indica il servizio o la macchina in rete. Per conoscere l'indirizzo IP corrispondente al nome del computer si utilizza il DNS.
HTTP è avviato da TCP/IP ogni qualvolta l'URL contiene nel primo campo la parola http.

TABELLA 1 HTTP identifica le risorse del WWW mediante un indirizzo simbolico: URL

http://	www.	azienda.com	/news/
indica al browser quale protocollo deve essere usato	identifica il nome di una specifica macchina (il web server)	rappresenta l'entità di dominio (domain entity) del sito web	identifica la cartella dove si trova la pagina web sul server. Se non viene specificato nulla, il browser carica la pagina web di default presente sul server

Quando si vuole leggere una pagina, i livelli superiori del client iniziano una sessione col web server. Il client fa la richiesta della pagina desiderata, il server risponde inviando la risorsa richiesta: il testo, l'audio, il video, i file grafici contenuti in quella pagina. Il client riassembla il tutto e chiude la sessione.

L'HTTP è un protocollo **stateless** (senza memoria) che permette sia la ricerca che il recupero dell'informazione in maniera veloce, seguendo gli hyperlink. La scelta di un protocollo stateless, cioè di un protocollo che non conserva memoria della connessione fatta, è stata necessaria affinché fosse possibile saltare velocemente da un web server a un altro attraverso i link ipertestuali.
HTTP a ogni richiesta di un web client effettua una nuova connessione al web server che viene chiusa al termine del trasferimento dell'oggetto richiesto (pagina HTML, immagine, ecc.).
Il server resta in attesa di una richiesta di connessione sulla sua socket, la porta assegnata di default per HTTP è la 80, salvo che nell'URL sia specificata una porta diversa, per esempio *http://www.azienda.com/news/:8080*.
La caratteristica stateless di HTTP limita l'interazione con l'utente, per esempio se effettuiamo il login su una pagina, nel momento in cui ci spostiamo su un'altra dobbiamo nuovamente inserire le nostre credenziali. La soluzione a questo problema è stata l'introduzione dei **#cookie**, piccoli blocchi di dati memorizzati nel browser che permettono di:

- implementare metodi di autenticazione, usati per esempio per i login;
- memorizzare dati utili alla sessione di navigazione, come le preferenze sull'aspetto grafico o linguistico del sito;
- tracciare la navigazione dell'utente, per esempio per fini statistici o pubblicitari.

#techwords

Hyperlink
Gli ipertesti (hypertext), grazie agli hyperlink, abbandonano la secolare abitudine alla lettura lineare, sequenziale, stabilita dall'autore di un testo, per passare a una lettura che vede il lettore come protagonista, non più come fruitore passivo. Ciò che consente questo processo sono gli hyperlink o "parole calde": ad alcuni termini di un ipertesto viene associata la possibilità di collegarsi, col semplice clic del mouse, ad altre parti dell'ipertesto. Il percorso di lettura che ne consegue, quindi, è deciso dal lettore. Si può pensare la lettura di un ipertesto come una forma di lettura ramificata, che, di link in link, porta il lettore direttamente al cuore di ciò che gli interessa leggere.

#techwords

Cookie (biscotto)
Sono file di testo di piccola dimensione inviati da un web server a un web client e poi rimandati indietro dal client al server, senza subire modifiche, ogni volta che il client accede allo stesso server. Poiché possono essere usati per monitorare la navigazione su Internet, i cookie sono oggetto di discussioni concernenti il diritto alla privacy.

5.3 I metodi e i messaggi di HTTP

L'acquisizione di una risorsa da parte del client può essere schematizzata in 4 fasi:

- **connessione**: il client crea una connessione TCP/IP con il server usando il suo nome di dominio (o l'indirizzo IP) ed eventualmente il numero della porta di trasmissione; come detto, se non viene fornito il numero di porta, il protocollo assume per default che il numero sia 80;
- **richiesta**: il client invia la richiesta di una risorsa (pagina HTML, immagine, ecc.) mediante una riga di caratteri ASCII che termina con una coppia di caratteri CR-LF (Carriage Return, Line Feed);
- **risposta**: la risposta inviata dal server è un messaggio in linguaggio HTML nel quale è contenuta la risorsa richiesta o una segnalazione d'errore;
- **disconnessione**: il server subito dopo aver spedito la risorsa richiesta si disconnette. Anche il client può interrompere la connessione in ogni momento; in questo caso il server non registrerà nessuna condizione d'errore.

Il protocollo HTTP mette a disposizione del client una serie di metodi. Un **metodo** HTTP può considerarsi un comando, proprio del protocollo HTTP, che il client invia come richiesta al server.
La versione HTTP/1.0 ha 3 metodi obbligatori: GET, HEAD, POST. Alcune implementazioni di HTTP/1.0 ne aggiungono altri due: PUT e DELETE. In HTTP/1.1 sono stati aggiunti altri 3 metodi: OPTIONS, TRACE e CONNECT.
Nel dettaglio:

- GET: richiede una risorsa (pagina HTML, immagine, ecc.) al server; quando un utente fa clic su un hyperlink il client invia una GET al server;
- HEAD: richiede solo l'header senza la risorsa, di fatto viene usato soprattutto per la diagnostica;
- POST: invia informazioni al server, cioè all'URL specificato;
- PUT: richiede l'upload di un file sul server, creandolo o riscrivendolo (se autorizzato);
- DELETE: richiede la cancellazione di un file sul server (se autorizzato);
- OPTIONS: richiede l'elenco dei metodi permessi dal server;
- TRACE: traccia una richiesta, visualizzando come viene trattata dal server;
- CONNECT: richiede una connessione mediante proxy, utilizzata, per esempio, per la creazione di un tunnel.

Vi sono due tipi di messaggi HTTP: messaggi richiesta (request) da parte del client e messaggi risposta (response) da parte del server.

IL MESSAGGIO REQUEST

È composto dalle seguenti 3 parti:

1. riga di richiesta (request line);
2. sezione header (informazioni aggiuntive);
3. body (contenuto della richiesta).

La riga di richiesta è composta da metodo, URI e versione del protocollo. **URI** sta per Uniform Resource Identifier e indica l'oggetto della richiesta.
Per esempio per ottenere una pagina web la richiesta è: **GET /info.html HTTP/1.1**.

Gli header di richiesta più comuni sono:

- **Host**: nome del server a cui si riferisce l'URI;
- **User-Agent**: identificazione del tipo di client: browser, produttore, versione, ecc.

IL MESSAGGIO RESPONSE

È composto dalle seguenti 3 parti:

1. **riga di stato**: contiene un codice di risposta a 3 cifre in cui la prima cifra specifica il tipo di stato:
   - **1xx**: Informational (messaggi informativi);
   - **2xx**: Success (la richiesta è stata soddisfatta);
   - **3xx**: Redirection (non c'è risposta diretta, ma la richiesta è ritenuta corretta e viene detto come ottenere la risposta);
   - **4xx**: Client error (la richiesta non può essere soddisfatta perché sbagliata);
   - **5xx**: Server error (la richiesta non può essere soddisfatta per un problema interno del server).
2. **header**: contengono informazioni aggiuntive. Quelli più comuni sono:
   - **Server**: indica il tipo e la versione del server. Può essere visto come l'equivalente dell'header di richiesta User-Agent;
   - **Content-Type**: indica il tipo di contenuto restituito. Essi sono detti tipi MIME (Multimedia Internet Message Extensions, presenti anche nella posta elettronica, come descritto nella Lezione successiva). Esempi di tipi MIME sono:
     – text/html (documento HTML);
     – text/plain (documento di testo non formattato);
     – text/xml (documento XML);
     – image/jpeg (immagine in formato JPEG).
3. **body**: è la parte in cui si trova il contenuto della risposta. I codici di risposta più comuni sono:
   - **200 OK**: il server ha fornito correttamente il contenuto nella sezione body;
   - **400 Bad Request**: la richiesta non è comprensibile al server;
   - **403 Forbidden**: il client non è autorizzato a ricevere i dati richiesti;
   - **404 Not Found**: la risorsa richiesta non è stata trovata e non se ne conosce l'ubicazione;
   - **500 Internal Server Error**: il server non è in grado di rispondere alla richiesta per un suo problema interno;
   - **505 HTTP Version Not Supported**: la versione di HTTP non è supportata.

Un server HTTP ha il compito (che può risultare computazionalmente dispendioso) di rispondere a tutte le richieste che giungono dalla rete. Si pensi che WWW server di siti professionali raggiungono facilmente le 300.000 richieste al giorno.
La versione HTTP/1.1 ha permesso di aumentare l'efficienza consentendo di utilizzare la stessa connessione TCP/IP per effettuare operazioni multiple.

5.4 I proxy HTTP

Un web server e un web client possono utilizzare un **#proxy HTTP** (detto anche *proxy server*) per gestire lo scambio di messaggi. La presenza di un proxy server fa sì che le richieste HTTP dei client vengano automaticamente indirizzate al proxy.

#techwords

Proxy
È un programma che si interpone tra un client e un server facendo da tramite o interfaccia. Il client si collega al proxy, invece che al server, e gli invia delle richieste. Il proxy, a sua volta, si collega al server e inoltra la richiesta del client, poi, ricevuta la risposta, la inoltra al client.
I proxy nella maggior parte dei casi lavorano a livello Application.

Un proxy HTTP può essere usato per diversi motivi:

- **connettività**: un proxy server può essere configurato per permettere a una rete privata di accedere a Internet con un unico computer, cioè un computer fa da proxy tra gli altri computer e Internet;
- **privacy:** un proxy server può garantire un maggiore livello di privacy mascherando il vero indirizzo IP del client in modo che il server remoto non venga a conoscenza di chi ha effettuato la richiesta;
- **caching:** un proxy server può immagazzinare per un certo tempo i risultati delle richieste di un client e se un altro client effettua le stesse richieste, può rispondere senza dover consultare il server originale. Collocando il proxy in una posizione "vicina" (prossima) ai client, questo permette un miglioramento delle prestazioni e una riduzione del consumo di ampiezza di banda;
- **monitoraggio**: un proxy server può permettere di tenere traccia di tutte le operazioni effettuate (per esempio, tutte le pagine web visitate), consentendo statistiche e osservazioni dell'utilizzo della rete che possono anche violare la privacy degli utenti;
- **amministrazione**: un proxy server può applicare regole definite dall'amministratore di sistema per determinare quali richieste inoltrare e quali rifiutare, può limitare l'ampiezza di banda utilizzata dai client oppure filtrare le pagine web in transito, per esempio bloccando quelle il cui contenuto è ritenuto offensivo in base a determinate regole.

I server esterni a cui si collega il client quando si utilizza un proxy vedranno generalmente l'indirizzo IP del proxy (e non quello del client). Se l'uso di un proxy garantisce una relativa privacy del client (il server esterno, o chi analizzi il traffico diretto a esso, non potrà infatti conoscere l'indirizzo IP del client), può impedire la connessione a quei siti che utilizzino l'indirizzo IP del client per scopi di autenticazione o di riconoscimento delle sessioni (come per esempio nei collegamenti agli sportelli bancari online).
Il protocollo HTTP prevede però che un proxy server possa inserire nelle richieste che inoltra al server degli header standardizzati, che permettono di riconoscere che la richiesta è stata inoltrata da un proxy e possono contenere anche l'indirizzo IP del client, che in questo modo può essere noto a un server remoto opportunamente configurato. Quando viene usata questa funzionalità, il web server remoto si fida dell'indirizzo del client inviatogli dal proxy server non potendo in alcun modo verificare questa informazione. L'amministratore di un proxy server può decidere se inviare o meno questi header determinando quindi il livello di anonimato del proxy.

I proxy HTTP, a seconda dell'anonimato che riescono a fornire, possono essere suddivisi in:

- **NOA** (NOn Anonymous Proxy Server) proxy non anonimi (o trasparenti): modificano alcuni header trasmessi dal browser e ne aggiungono altri, mostrano anche l'indirizzo IP reale del richiedente. Sono facili da riconoscere da parte del web server;
- **ANM** (Anonymous Proxy Server) proxy anonimi: non trasmettono l'IP del richiedente, ma modificano o aggiungono alcuni header. Sono pertanto facilmente riconoscibili;

- **HIA** (HIgh Anonymous Proxy) proxy altamente anonimi (o élite): non trasmettono l'IP del richiedente e non modificano gli header della richiesta. Sono difficili da riconoscere attraverso i normali controlli;
- **proxy distorcenti**: trasmettono un IP casuale, diverso da quello del richiedente e modificano o aggiungono alcuni header. Solitamente vengono scambiati per proxy anonimi, ma offrono una protezione maggiore, in quanto il web server remoto vede le richieste di un utente provenienti da indirizzi IP diversi.

Per vedere se il proxy server consente una navigazione anonima, ossia se non rivela l'IP del client a nessun altro server della rete, è bene effettuare un **whois** (Lezione 7 dell'Unità 5). Il server del sito per il whois deve restituire l'IP del proxy server; se invece rende visibile un IP diverso, presumibilmente si tratta di quello del client e il test è fallito.

5.5 La sicurezza con HTTPS

Per garantire la sicurezza nelle transazioni commerciali o in generale nel trasferimento di dati sensibili, si usa il protocollo **HTTPS** (HyperText Transfer Protocol over Secure Sockets Layer).

Le differenze tra HTTPS e HTTP sono sostanzialmente due:

- l'utilizzo della porta 443 al posto della 80;
- l'applicazione del protocollo TLS/SSL.

In pratica tra il protocollo TCP e il protocollo HTTP si interpone un livello di crittografia/autenticazione come il Secure Sockets Layer (SSL) o il Transport Layer Security (TLS), in modo da impedire intercettazioni dei contenuti.

Infatti, viene implementata una tecnica di crittografia asimmetrica che utilizza chiavi private e pubbliche a lungo termine, per generare chiavi di sessione a breve termine. Queste chiavi sono utilizzate successivamente per cifrare il flusso dei dati scambiati tra client e server.
Un sito web non può avere dei contenuti accessibili con HTTPS e altri con HTTP, per esempio la pagina di login su HTTPS e le altre pagine su HTTP: ciò implicherebbe una vulnerabilità a possibili attacchi.
Inoltre, se il sito è su HTTPS, anche i cookie devono essere trasmessi in modo sicuro. È quindi necessario impostare un parametro, chiamato **Secure attribute**, che segnala al browser di inviare il cookie solo su HTTPS e mai su HTTP.

FISSA LE CONOSCENZE

- Qual è il compito del protocollo HTTP?
- Che cos'è un hyperlink?
- Che cos'è un metodo HTTP?
- Quali sono gli 8 metodi dell'HTTP/1.1?
- Quali sono i 2 tipi di messaggi HTTP?
- Qual è il ruolo del proxy server in una comunicazione HTTP?

6 SMTP, POP E IMAP: I PROTOCOLLI PER LA POSTA ELETTRONICA

6.1 Invio e ricezione di email

La posta elettronica (electronic-mail o email) è una delle prime applicazioni nate con Internet e continua a essere una delle più importanti e utilizzate.
La FIGURA 8 mostra le fasi di invio, trasmissione in rete e ricezione di una email.

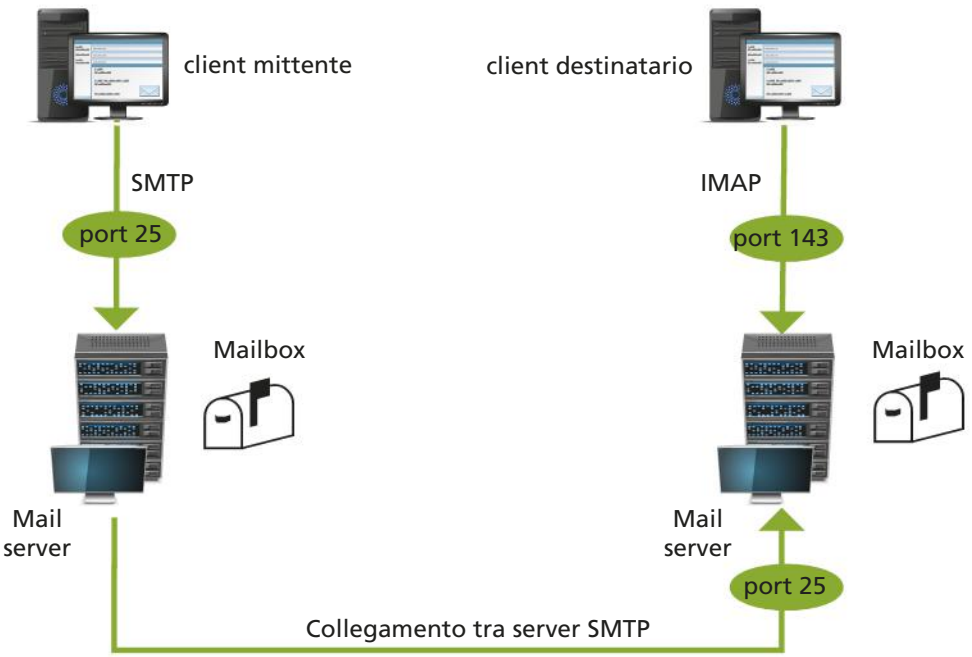


FIGURA 8 Invio e ricezione di una email con i protocolli SMTP e IMAP

Sono evidenziate le componenti principali di un sistema di posta elettronica:

- **Mail client**, è l'applicazione di email utilizzata dall'utente per inviare/ricevere email, per esempio Outlook o Thunderbird;
- **Mail server**, è l'applicazione di email che risiede sui server: riceve e inoltra i messaggi, gestisce le caselle di posta, **mailbox**, degli utenti; l'insieme dei server costituisce l'infrastruttura del sistema di posta elettronica;
- i **protocolli**, sono stati definiti più protocolli per la posta elettronica: **SMTP** (Simple Mail Transfer Protocol) per l'invio delle email e per la comunicazione tra i mail server, **POP3** (Post Office Protocol, version 3) e **IMAP4** (Internet Message Access Protocol version 4) per la ricezione delle email.

LA POSTA ELETTRONICA SUL WEB

Un'alternativa allo scenario presentato in Figura 8 è il sistema **web-based email**, o **webmail**, nel quale l'utente utilizza il browser per inviare e ricevere le email. Il primo a offrire un servizio di webmail fu Hotmail, a metà degli anni Novanta, seguito poi da altri, come Google Gmail e Yahoo! Mail.
In questo scenario cambia il client di email: non è più un programma, ma un'interfaccia utente fornita tramite pagine web. Quando l'utente accede alle pagine web del

servizio di email gli viene chiesto di autenticarsi con login e password. Queste credenziali sono inviate al server che le valida, costruisce sul momento una pagina web con il contenuto della mailbox e la invia all'utente.
L'invio e la ricezione dei messaggi avviene quindi con il protocollo HTTP, essendo una comunicazione tra web client e web server. Il web server si occuperà poi di introdurre i messaggi nel tradizionale sistema di posta elettronica basato sul protocollo SMTP.

GLI INDIRIZZI DELLA POSTA ELETTRONICA

Ogni utente (client) è individuato da un indirizzo di posta composto dal user ID dell'utente seguito dal simbolo @ e dal dominio del gestore del servizio di posta elettronica:

nomeutente@dominiogestoreservizio

Per esempio: bianchi@azienda.com.
Se il client mittente e il client destinatario usufruiscono dello stesso fornitore del servizio email (quindi hanno lo stesso dominio, per esempio azienda.com) allora il server SMTP ha il compito semplificato perché con un semplice programma, chiamato delivery agent, può direttamente depositare la email nella mailbox. Se invece i fornitori sono diversi, e dunque sono diversi i domini, il server SMTP del mittente deve interrogare il **DNS** (Domain Name System) per risalire dal nome di dominio all'indirizzo IP del server SMTP del destinatario. Per associare il server SMTP a un dato nome di dominio si usa un Resource Record di tipo MX (Mail eXchange), come visto nella Lezione 6 dell'Unità 7.

6.2 Il protocollo SMTP

Il protocollo SMTP gestisce il trasferimento del messaggio di posta elettronica dal mittente al destinatario.
La prima versione del protocollo SMTP è del 1982 contenuta nell'RFC 821, ma era già utilizzato da molti anni dagli utenti di Internet. SMTP è stato revisionato nel 2008, **RFC 5321**, con successivi aggiornamenti riguardanti l'uso dei codici di risposta.

IN ENGLISH PLEASE

Network Working Group — J. Klensin
Request for Comments: 5321 — October 2008
Obsoletes: 2821
Updates: 1123
Category: Standards Track

Simple Mail Transfer Protocol

Abstract

This document is a specification of the basic protocol for Internet electronic mail transport. It consolidates, updates, and clarifies several previous documents, making all or parts of most of them obsolete. It covers the SMTP extension mechanisms and best practices for the contemporary Internet, but does not provide details about particular extensions. Although SMTP was designed as a mail transport and delivery protocol, this specification also contains information that is important to its use as a "mail submission" protocol for "split-UA" (User Agent) mail reading systems and mobile environments.

#prendinota

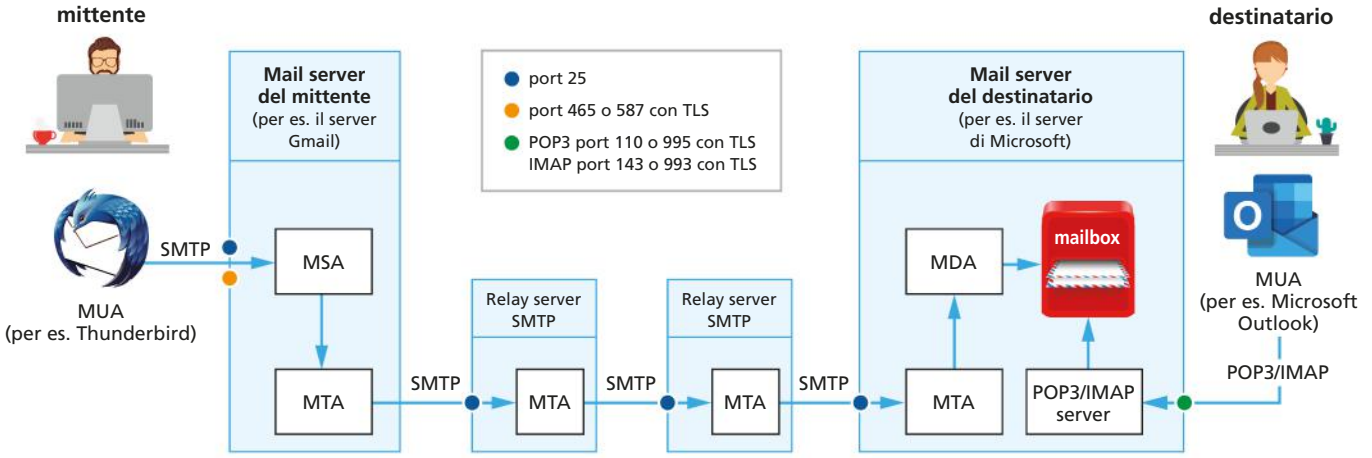
L'ingegnere informatico americano Ray Tomlinson nel 1971 inventò la posta elettronica elaborando un programma che permetteva a tutti coloro che frequentavano le università americane, collegate tra loro tramite la rete ARPANET, di potersi scambiare messaggi scritti. Lo stesso Tomlinson nel 1972 usò il simbolo @ (at, cioè "presso" in inglese, *chiocciola* in italiano) come separatore tra il nome del destinatario e il server che svolgeva le funzioni di cassetta della posta. Nel marzo del 2010, Paola Antonelli, Senior Curator del Department of Architecture and Design del MoMA di New York, ha reso noto che la chiocciola è stata inserita nella collezione, perché non è soltanto uno strumento utilizzato in informatica, ma è un mezzo di comunicazione e una forma della nostra identità.

La FIGURA 9 mostra il sistema di email su Internet. Come descritto nelle specifiche di SMTP si usano i seguenti termini per riferirsi al software utilizzato su client e server:

- **Mail User Agent** (MUA) è il software client sul computer dell'utente, il mittente è chiamato **sender** e il destinatario **recipient**; questo programma offre un'interfaccia grafica con funzionalità per la composizione, lettura e organizzazione delle email;
- **Mail Submission Agent** (MSA) è il software che risiede sul mail server al quale l'utente invia una email tramite il MUA;
- **Mail Delivery Agent** (MDA) è il software che risiede sul mail server e gestisce la mailbox dalla quale l'utente legge la email ricevuta;
- **Mail Transfer Agent** (MTA) è il software che risiede sui mail server che si occupano dell'inoltro della email all'interno del sistema di posta elettronica.

I distinti ruoli che ciascun software MTA, MSA e MDA assume nella trasmissione di un messaggio di posta elettronica possono essere svolti da uno stesso programma che risiede nel server e implementa più funzionalità.
Nei sistemi di posta webmail il MUA è rappresentato dall'interfaccia web eseguita nel browser del computer dell'utente.

FIGURA 9 Il sistema di posta elettronica



Analizziamo le fasi di invio e ricezione di una email mostrate nella Figura 9 entrando nel dettaglio di ciò che succede sui server:

- **invio**: il mittente (sender) invia al server una email utilizzando il software MUA presente sul suo dispositivo, questa viene presa in carico dal mail server che gestisce la sua mailbox, in particolare dal software MSA, con una connessione TCP alla porta 25 del server. MSA invia il messaggio al software MTA, presente sullo stesso mail server, affinché lo inoltri al mail server che gestisce la mailbox del destinatario (recipient);
- **transito**: l'MTA del mittente apre una connessione TCP verso l'MTA del mail server di destinazione, sempre tramite la porta 25, sulla quale trasmette il messaggio di posta elettronica. Nel mail server di destinazione, il messaggio è inviato da MTA al software MDA che lo memorizza nella mailbox del destinatario. A volte la connessione tra i due mail server non è diretta, ma il messaggio passa attraverso più MTA presenti su server intermedi, chiamati **relay server**;
- **ricezione**: il client destinatario utilizza il software MUA per connettersi alla porta 110 del server POP3 o alla porta 143 per IMAP4 e accedere alla sua mailbox per leggere il messaggio ricevuto.

#prendinota

Se confrontiamo la posta elettronica con il servizio di posta tradizionale, il software MTA svolge un ruolo analogo all'ufficio postale locale e MSA è l'analogo dell'impiegato che smista la posta ricevuta. Se il destinatario vive nello stesso comune, la lettera è data direttamente al postino che la recapita nella casella di posta del ricevente, in caso contrario viene trasmessa all'ufficio postale (MTA, per email) del comune dove abita il destinatario.

Qualora il server SMTP destinatario non risulti raggiungibile, il server SMTP mittente prova a rispedire la email per un certo periodo di tempo. Se l'invio continua a fallire, il server mittente invia al client mittente una segnalazione di mancato recapito (badmail o delivery failure).

LA SICUREZZA DI SMTP

Il protocollo SMTP non garantisce né l'autenticazione del mittente né l'autorizzazione all'invio, chiunque può spedire una email a chiunque.
Per questi motivi lo **#spamming** non può essere in alcun modo limitato ed è inoltre possibile inviare email facendo apparire come mittente l'indirizzo corrispondente a un altro account.

Inoltre, nelle specifiche originarie di SMTP, non è previsto l'impiego della crittografia e i messaggi sono trasmessi in chiaro nella rete.
Questo problema è stato risolto con l'utilizzo di connessioni TLS (Transport Layer Security) e la registrazione presso IANA di un'altra porta, la 465, da utilizzare in sostituzione della porta 25.

> La maggior parte dei provider non utilizza più la porta 25 per la connessione tra client e server SMTP sia perché affetta dal traffico di spam e malware, sia perché la richiesta di un servizio di confidenzialità implica l'utilizzo di connessioni TLS che usano la porta **587 (RFC 6409)** o la porta **465 (RFC 8314)**. La porta 25 continua a essere usata per le connessioni tra i relay server SMTP.

La porta 587 è stata definita per sostituire la porta 25, introducendo un nuovo comando STARTTLS da utilizzare quando il client vuole stabilire una connessione sicura con TLS. Infatti, mentre con la porta 465 è implicito l'uso di TLS, con la porta 587 deve essere esplicitamente richiesto con il comando STARTTLS.

IL FORMATO DELLE EMAIL

Il formato dei messaggi di posta elettronica è definito nell'**RFC 5322**.

> Una email è formata da un header (envelope) e il testo del messaggio (content o body).

L'**envelope** contiene le informazioni utili per la spedizione del messaggio. Lo standard specifica le parole chiave da usare in ogni linea, separate da una virgola, alcune sono obbligatorie, per esempio From e To, altre opzionali, per esempio Subject.
Un tipico header di una email è:

From: bianchi@azienda.com
To: rossi@academy.edu
Subject: Richiesta articolo

Dopo l'envelope si inserisce una riga vuota, da lì in avanti c'è il testo del messaggio in ASCII (American Standard Code for Information Interchange).

> SMTP è stato progettato per incidere il meno possibile sull'occupazione del canale. Per questo motivo utilizza il codice **ASCII** a 7 bit per codificare i caratteri del messaggio.

#techwords

Spamming
È l'invio indiscriminato di messaggi di posta elettronica. La mailbox dell'utente viene inondata di email spesso pubblicitarie, ma anche contenenti link dannosi e allegati con virus o malware. Il nome SPAM deriva da **S**houlder of **P**ork **A**nd ha**M** (spalla di maiale e prosciutto), carne in scatola immessa sul mercato nel periodo della Seconda guerra mondiale, diventata sinonimo di fastidiosa e non richiesta valanga di materiale.

Per inviare dei caratteri accentati, o comunque non compresi nei primi 128 caratteri codificati dall'ASCII a 7 bit, bisogna ricorrere ad algoritmi che integrino le specifiche **#MIME** (Multipurpose Internet Mail Extensions):

- **base64** per i file in allegato;
- **quoted-printable** (abbreviazione QP) per i caratteri speciali contenuti nel corpo del messaggio.

esercizio

➜ PROBLEMA

Nella propria casella di posta elettronica, selezionare un messaggio con un file pdf allegato ed esaminarne il contenuto codificato in #MIME. Ripetere l'analisi su messaggi con altri tipi di allegati, immagini, video, ecc.

#techwords

MIME e MIME type

Lo standard MIME fu proposto da Bell Communication nel 1991 per sopperire alle limitazioni di SMTP e consentire agli utenti di inviare email con caratteri non-ASCII7 e file contenenti immagini, audio, video e programmi (file binari).
MIME è utilizzato anche dal web server per comunicare al browser il tipo di dati che gli viene inviato nella risposta (MIME type).

➜ SVOLGIMENTO

Nel servizio di posta **Gmail** per vedere il contenuto del messaggio nel formato MIME si deve aprire il messaggio e selezionare la voce Mostra originale nel menu con i tre puntini. La stessa procedura si può seguire con il servizio di posta elettronica **Yahoo! Mail** selezionando la voce Visualizza messaggio in formato Raw.
Il testo che compare è il seguente, relativo a una email con un file pdf allegato.

```
MIME-Version: 1.0
Date: Sun, 24 Gen 2021 09:49:26 +0100
Message-ID: <CAAO7sbcbZDryMw4-ZWHj5PaLs6eTVJOur8SrcyjEHpzNhrv87g@mail.
gmail.com>
Subject: PROVA
From: user1@gmail.com
To: user2@yahoo.com
Content-Type: multipart/mixed; boundary="0000000000003dba8005b4a92079"

--0000000000003dba8005b4a92079
Content-Type: multipart/alternative; boundary="0000000000003dba7d05b4a92077"

--0000000000003dba7d05b4a92077
Content-Type: text/plain; charset="UTF-8"

Messaggio di prova con allegato

--0000000000003dba7d05b4a92077
Content-Type: text/html; charset="UTF-8"

<div dir="ltr">Messaggio di prova con allegato<br></div>

--0000000000003dba7d05b4a92077—
--0000000000003dba8005b4a92079
Content-Type: application/pdf; name="ocument.pdf"
Content-Disposition: attachment; filename="ocument.pdf"
Content-Transfer-Encoding: base64
X-Attachment-Id: f_khsiuwid0
Content-ID: <f_khsiuwid0>

--0000000000003dba8005b4a92079—
```

Vediamo i vari campi presenti:

- MIME-Version:1.0, indica che il messaggio è nel formato MIME;
- Content-Type: multipart/mixed, indica che sono presenti parti in formato testo e altre parti non testuali;
- Content-Type: multipart/alternative, indica che il messaggio è inviato in più formati, nel nostro esempio il contenuto del messaggio si presenta sia come normale testo sia in html;
- boundary="…", è una stringa che separa le parti del messaggio MIME, nel nostro esempio sono definite due stringhe, una delimita le due parti del contenuto, plain e html, l'altra delimita la parte che contiene l'allegato;
- Content-Type: text/plain, specifica il tipo di formato, text, e il sottotipo, plain, contenuto in quella parte del messaggio; text/plain è il formato predefinito di Content-Type;
- Content-Type: text/html, specifica il tipo di formato, text, e il sottotipo, html, contenuto in quella parte del messaggio;
- Content-Type: application/pdf, specifica il tipo di formato, application, e il sottotipo, pdf, del file allegato, il cui nome è specificato nel campo name. Altri formati sono previsti per i vari tipi di allegati, per esempio: image/jpeg, audio/mp3, video/mp4;
- Content-Disposition: attachment, indica di presentare il file come un allegato con il nome specificato nel campo filename; l'alternativa ad attachment è inline, che indica di visualizzare in automatico il contenuto del file quando la email viene aperta (si usa soprattutto per le immagini);
- Content-Transfer-Encoding: base64, indica che il contenuto del messaggio è stato codificato secondo lo schema base64 e quindi il client deve operare la necessaria decodifica per consentire all'utente di leggere il messaggio nella sua codifica originale, per esempio UTF-8.

I PRINCIPALI COMANDI DI SMTP

I comandi SMTP hanno il seguente formato:

```
keywords : parametri
```

Non tutti i comandi prevedono dei parametri.

La TABELLA 2 elenca i comandi principali definiti per la comunicazione dal client verso il server SMTP.

TABELLA 2 I principali comandi SMTP

Comando	Esempio	Descrizione
EHLO (sostituisce HELO)	EHLO 193.56.47.125	Identifica il computer mittente attraverso l'indirizzo IP o il nome del dominio.
EMAIL FROM:	MAIL FROM: mittente@dominio1.com	Specifica il mittente del messaggio.
RCPT TO:	RCPT TO: destinatario@dominio2.com	Specifica il destinatario del messaggio.
DATA	DATA messaggio	Indica l'inizio del contenuto del messaggio, che sarà inviato linea per linea.
QUIT	QUIT	Chiude la connessione TCP con il server SMTP.

Il server SMTP a sua volta spedisce dei messaggi di risposta al client che gli ha inviato i comandi. Ogni risposta inizia con un codice identificativo di 3 cifre, opzionalmente seguito da un testo informativo. Alcuni codici e il relativo testo sono elencati nella TABELLA 3.

TABELLA 3 Codici di alcune risposte inviate dal server SMTP al client

Categoria	Codici	Descrizione
Positive Completition Reply	2xx	Informa che l'azione richiesta dal client è stata portata a termine con successo.
Positive Intermediate Reply	3xx	Informa il client che il comando è stato accettato, ma l'azione richiesta è in sospeso.
Transient Negative Completition Reply	4xx	Informa il client che il comando non è stato accettato per una situazione di errore temporanea, il client può provare a inviare di nuovo il comando.
Permanent Negative Completition Reply	5xx	Informa il client che il comando non è stato accettato e che il server non è in grado di eseguire l'azione richiesta.

6.3 Il protocollo POP

#prendinota

SMTP è un protocollo **push** (spingi): il mail server del mittente invia il file al mail server del destinatario. Il destinatario ottiene il messaggio con un'operazione **pull** (estrai), quindi sono stati definiti i protocolli di accesso POP3 e IMAP4, o HTTP per la webmail.

Il protocollo POP è il primo a essere stato definito per l'accesso al server di posta per scaricare i messaggi dalla mailbox. Ha subito varie modifiche dalla sua prima versione, l'attuale versione è **POP3** definita nell'**RFC 1939**.
Con POP3 i messaggi di posta elettronica, per essere letti, vengono scaricati in locale sul computer e cancellati dal server. Questo risulta particolarmente utile qualora il client abbia convenienza a leggere le email offline, ma se si usa una webmail non sarà più possibile leggere le email dopo averle scaricate.
È comunque sempre possibile configurare il client per lasciare una copia del messaggio nella mailbox del server.
Le porte definite per POP3 sono la **110** e la **995** per le connessioni TLS.
Questo protocollo gestisce l'autenticazione attraverso **username** e **password**. Quest'ultima, come le email, non è cifrata. Per poter codificare la password e beneficiare di un'autenticazione sicura è possibile selezionare un servizio opzionale che solo pochi server implementano.
Il protocollo POP3 blocca la casella postale durante la consultazione al fine di evitare una consultazione simultanea da due utenti.
I principali comandi POP3 sono riportati nella TABELLA 4.

TABELLA 4 I principali comandi POP3

Comando	Descrizione
USER identificativo	Questo comando permette di autenticarsi. Esso deve essere seguito dal nome dell'utente, cioè da una stringa di caratteri che identificano l'utente sul server. Il comando USER deve precedere il comando PASS.
PASS password	Il comando PASS permette di indicare la password dell'utente il cui nome è specificato nel comando USER precedente.
STAT	Informazione sui messaggi contenuti sul server.
RETR	Numero di messaggi da recuperare.
DELE	Numero di messaggi da cancellare.
LIST [msg]	Numero di messaggi da visualizzare.
NOOP	Permette di mantenere le connessioni aperte in caso di inattività.
TOP <messageID> <n>	Comando che visualizza n linee di messaggio, dove n è dato in argomento. In caso di risposta positiva da parte del server, questo rinvia le intestazioni del messaggio, poi una linea vuota e infine le n prime linee del messaggio, indipendentemente dalla sessione.
QUIT	Chiede l'uscita del server POP3. Esso implica la cancellazione di tutti i messaggi segnati come eliminati e rinvia lo stato di questa azione.

6.4 Il protocollo IMAP4

IMAP4 è un protocollo di accesso per leggere i messaggi ricevuti nella mailbox, come POP3. Questo protocollo è particolarmente indicato per i client in grado di mantenere una connessione continua a un server (online), infatti permette di sincronizzare il client con il server.

IMPA4 è stato definito nell'**RFC 3501** per superare le limitazioni di POP3, infatti offre molte più possibilità:

- accesso alla posta sia online sia offline: il client rimane connesso e risponde alle richieste che l'utente fa attraverso l'interfaccia; questo permette di risparmiare tempo se ci sono messaggi di grandi dimensioni;
- più utenti possono utilizzare la stessa casella di posta: permette connessioni simultanee alla stessa mailbox, fornendo meccanismi per controllare i cambiamenti apportati da ogni utente;
- accesso a singole parti di un messaggio: la maggior parte delle email sono trasmesse nel formato MIME, che permette una struttura ad albero del messaggio, dove ogni ramo è un contenuto diverso (intestazioni, allegati o parti di esso, messaggio in un dato formato, ecc.). Il protocollo IMAP4 permette di scaricare una singola parte MIME o addirittura sezioni delle parti, per avere un'anteprima del messaggio o per scaricare una email senza i file allegati;
- informazioni sui messaggi presenti nella mailbox: ogni singolo client può tenere traccia di ogni messaggio, per esempio per sapere se è già stato letto o se ha avuto una risposta;
- organizzazione in cartelle (folder) delle email ricevute: si possono creare, modificare o cancellare cartelle sul server.

Con IMAP4 i messaggi, sia della cartella Posta in arrivo sia delle altre cartelle, rimarranno sempre sul server e sul computer client ne sarà scaricata soltanto una copia. Si potrà quindi accedere alla propria casella da più dispositivi e ritrovare tutte le email, purché tutti gli accessi avvengano via IMAP4 o webmail e non con POP3.

Le porte definite per IMAP4 sono la **143** e la **993** per le connessioni TLS.

LABORATORIO ONLINE

TELNET E LA POSTA ELETTRONICA

L'attività di laboratorio consiste nell'inviare una email utilizzando una sessione Telnet tra il nostro computer client e il server SMTP, tramite la porta 25, e poi, tramite la porta 110, collegarsi al server POP3 per ricevere la stessa email.

LABORATORIO ONLINE

WIRESHARK: ANALISI DI HTTP, SMTP, POP3

L'attività di laboratorio consiste nell'utilizzo di Wireshark per esaminare i dati che vengono scambiati tra un web client e un web server con HTTP e i dati scambiati tra client e server di posta elettronica con SMTP e POP3.

FISSA LE CONOSCENZE

- Descrivi le fasi di invio e ricezione di una email.
- Da quali parti è composto un indirizzo di posta elettronica?
- Quale codice utilizza il protocollo SMTP e perché?
- Il protocollo SMTP non offre garanzie di sicurezza. Perché?
- In che modo il protocollo POP3 gestisce l'autenticazione dell'utente?
- Perché è stato introdotto il protocollo IMAP4?

7 I PROTOCOLLI PER LE APPLICAZIONI MULTIMEDIALI

7.1 La classificazione delle applicazioni multimediali

Le applicazioni multimediali in rete si occupano del trasferimento di dati di tipo **audio** e **video** attraverso la rete.
La possibilità di trasmettere segnali audio e video in rete consente di avere forme più avanzate di comunicazione, ma richiede un elevato impiego di risorse, decisamente maggiore rispetto ad altri tipi di trasmissione. Infatti, prima di essere trasmessi in rete i segnali audio/video devono essere digitalizzati e compressi.
Tipicamente, le applicazioni multimediali sono classificate in 3 categorie: le applicazioni memorizzate su un server di streaming, le applicazioni live e le applicazioni interattive.

#techwords

Jitter indica una variabilità nel tempo di arrivo dei pacchetti. Se il ritardo è costante il destinatario riceverà il messaggio in modo comprensibile. Se invece i pacchetti subiscono ritardi variabili, l'effetto sarà di una comunicazione a scatti.
Questo concetto è stato introdotto nell'Unità 8 del volume 2 a proposito della QoS (Quality of Service) applicata alle comunicazioni di tipo interattivo.

STORED STREAMING APPLICATION

I dati multimediali sono memorizzati su un server e trasmessi al client su richiesta (**on demand**). Si dice che sono trasmessi in streaming, perché il client è in grado di visualizzarli subito, prima che il trasferimento sia completato. Questa caratteristica comporta stringenti vincoli temporali, per la consegna dei dati ancora da inviare, affinché la fruizione da parte del client sia adeguata (un ritardo di 5-10 secondi è ancora accettabile). La trasmissione avviene in modalità **unicast** e si adatta al dispositivo e alla disponibilità di banda dell'utente. Lo streaming sul client è visualizzato tramite un media player che cerca di rimuovere i #jitter, decomprimere i dati, visualizzare i controlli per un uso interattivo da parte dell'utente.
Alcuni esempi sono: visione di film, serie TV, ascolto di brani musicali.

#prendinota

Streaming Application
Esempi di fornitori di contenuti in streaming on demand sono Amazon Prime video, Hulu, Netflix, Spotify e Youtube.

Facebook Live e Periscope sono esempi di fornitori di live streaming.

LIVE STREAMING APPLICATION

La trasmissione dei dati di queste applicazioni è simile alla diffusione dei programmi radio e televisivi; la differenza è che la trasmissione avviene attraverso la rete Internet. Anche in questo caso, come nel precedente, si tratta di traffico sensibile al ritardo e non è possibile ritrasmettere i pacchetti. La differenza è che le applicazioni di tipo stored streaming prevedono una trasmissione dei contenuti on demand, mentre in quelle live streaming la trasmissione è in **diretta** e può anche essere in multicast per servizi come IPTV (la trasmissione on demand è invece sempre unicast). In una trasmissione live i contenuti vengono inseriti dal fornitore del servizio, man mano che si rendono disponibili. Ciò comporta un processo di compressione dei dati più veloce e meno ottimizzato, che può tradursi in una maggiore quantità di dati da trasmettere in rete.
Alcuni esempi sono: IPTV, Internet radio, videogiochi online, concerti online.

INTERACTIVE APPLICATION

Si tratta di applicazioni di tipo **interattivo**, con esigenze di trasmissione in tempo reale: bassissimo jitter e nessuna ritrasmissione. Tali requisiti sono in genere soddisfatti attraverso il sovradimensionamento della rete o la definizione di classi di priorità nell'assegnazione della banda (rimane però il problema se il carico della rete aumenta considerevolmente).
Alcuni esempi sono: telefonia via Internet, audio/video conferenza.

7.2 Real Time Streaming Protocol

#prendinota

Un'applicazione che tratta streaming audio/video deve poter fornire all'utente i comandi che normalmente si usano sui player, ossia: rewind, fast forward, pause, resume, repositioning, ecc.

Per soddisfare le esigenze delle applicazioni di streaming audio e video è stato standardizzato il protocollo **RTSP** (Real Time Streaming Protocol), **RFC 7826**, che si colloca al livello Application dello stack TCP/IP e segue il paradigma Client-Server tipico di Internet.

RTSP offre all'utente quei comandi, tipici dei player, che HTTP non è in grado di fornire (play, pause, ecc.), inoltre tiene traccia dello stato del client in ogni sessione (esempi di stato del client sono: riproduzione, fermo immagine, ecc.). A questo scopo, il protocollo RTSP numera le sessioni e questi valori sono usati come identificatori nelle richieste e risposte RTSP, per aiutare il server a mantenere lo stato delle sessioni aperte con i vari client.

RTSP non definisce come i dati audio/video devono essere incapsulati per realizzare lo streaming sulla rete, né specifica come devono essere trasportati.

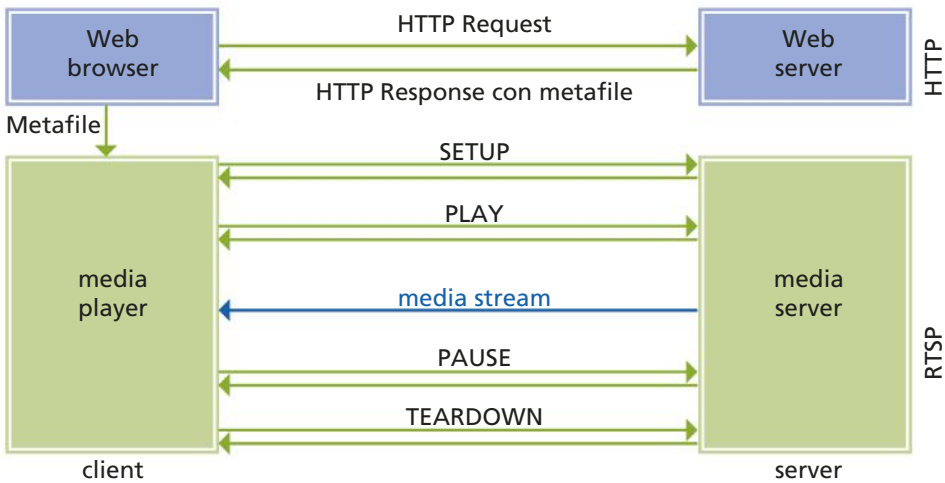


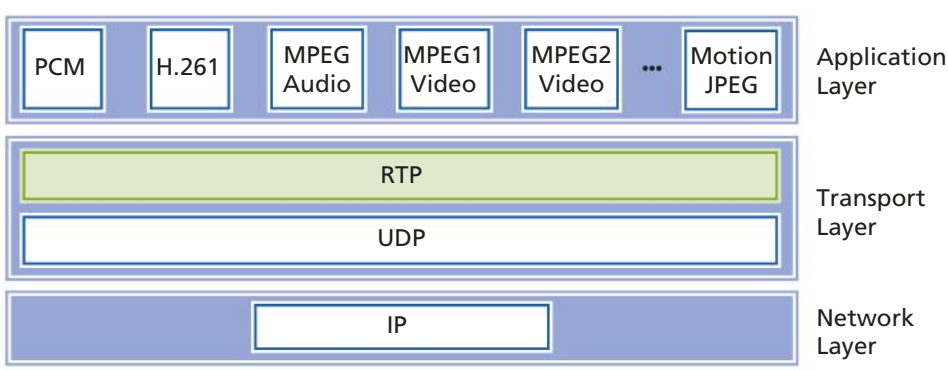
FIGURA 10 Le operazioni previste dal protocollo RTSP

La FIGURA 10 mostra uno scambio di messaggi RTSP tra client e server: il browser del client contatta il web server (HTTP Request), questi invia come risposta un metafile (HTTP response con metafile) contenente le informazioni necessarie per avviare il download dei dati in streaming (URL, tipo di codifica dei dati, ecc.), il browser avvia il player il quale contatta il server per instaurare una sessione RTSP (Setup). Da qui in poi avviene la riproduzione, fino alla chiusura (Teardown).

7.3 Real Time Transport Protocol

I protocolli di livello Transport tipici di Internet (TCP e UDP) non sono adatti per il traffico generato dalle applicazioni real time interattive. Quindi è stato standardizzato un protocollo ad hoc, per gestire questo tipo di dati: il protocollo **RTP** (Real Time Transport Protocol), **RFC 3550**, che si colloca tra il livello Transport e il livello Application. In particolare si interfaccia con il protocollo UDP dello strato Transport (FIGURA 11).

FIGURA 11 La collocazione di RTP nello stack TCP/IP



RTP definisce un formato standard per i pacchetti multimediali e deve essere integrato all'interno dell'applicazione:

- RTP è un processo attivo a livello di end system e specifica la struttura che devono avere i pacchetti che trasportano dati audio/video;

- i pacchetti RTP sono incapsulati all'interno di una socket UDP;
- in ricezione, i dati applicativi devono essere estratti dai pacchetti RTP e passati al player per la riproduzione.

RTP viene usato insieme a un altro protocollo: **RTCP** (Real Time Transport Control Protocol), che svolge il compito di raccogliere statistiche al fine di ottimizzare le prestazioni. Tutti i partecipanti a una sessione (relativa, per esempio, a una partita online con più giocatori) inviano pacchetti RTCP, siano essi mittenti o destinatari.
I rapporti statistici contengono dati sul numero di pacchetti inviati, persi, jitter, ecc. e sono usati dall'applicazione per modificare la velocità di trasmissione della sorgente.
Per evitare che l'invio di pacchetti RTCP da parte di tutti i partecipanti alla sessione crei congestione, è stata definita una semplice regola: la banda totale usata per i pacchetti RTCP deve essere il 5% della banda utilizzata per la sessione RTP e di questa ne viene riservato il 25% al mittente e il 75% ai destinatari.

esempio

Supponiamo di avere la trasmissione di un video con una banda di 2 Mbps con il 5% della banda riservato ai pacchetti RTCP.
Trasformando 2 Mbps in 2.000 Kbps e applicando il 5% otteniamo che la banda riservata è pari a 100 Kbps.
Dei 100 Kbps viene usato il 75% dai destinatari (75 Kbps).

7.4 Le reti CDN per le applicazioni multimediali

Una rete **CDN** (Content Distribution Networks) risponde all'esigenza dei fornitori di servizi di streaming di distribuire i contenuti su più server collocati in varie aree geografiche. La CDN si occupa della gestione di questi server: memorizza una copia dei video da trasmettere e indirizza le richieste degli utenti al server che meglio potrà soddisfarle. Uno stesso video non viene memorizzato su tutti i server della CDN; se un client lo richiede a un server in cui non è presente, questi lo recupererà da un altro server della CDN e mentre lo trasmette in streaming al client, lo memorizza nel suo repository.
Una CDN può essere privata, è il caso di Netflix e di Google per i video di YouTube, oppure di terze parti, un esempio è la CDN di Akamai.

FISSA LE CONOSCENZE

- Quali sono le problematiche del trasporto su Internet di dati generati da applicazioni multimediali?
- Come si classificano le applicazioni multimediali?
- Quale protocollo è stato standardizzato per il trasporto dei dati delle applicazioni stored streaming?
- Come avviene lo scambio dei messaggi con il protocollo RTSP?
- Spiega com'è organizzata la distribuzione dei contenuti su una rete CDN.

8 VoIP: LA TECNOLOGIA PER LA VOCE

8.1 L'applicazione Voice over IP

VoIP (Voice over IP), chiamata anche **Internet Telephony**, è un'applicazione real time che utilizza i protocolli della rete IP e le relative tecniche di routing per implementare una rete telefonica distribuita e flessibile.

Le problematiche che questo approccio comporta sono quelle descritte nell'Unità 8 del volume del terzo anno a proposito della **QoS**: ottimizzare le prestazioni della rete in termini di banda, tasso d'errore e di pacchetti persi, latenza e jitter.
Per evitare il meccanismo di acknowledgment e delle eventuali ritrasmissioni del protocollo TCP, le applicazioni VoIP utilizzano **UDP** come protocollo di livello Transport, scelta comune a molte applicazioni real time. Infatti, la perdita di alcuni pacchetti durante una conversazione audio non ne compromette la comprensione da parte degli interlocutori.
Nel caso di trasporto di dati audio e video, si usa il protocollo **RTP** (Real Time Transport Protocol), descritto nella Lezione precedente.

All'inizio VoIP fu presentata come un'applicazione che consentiva di effettuare telefonate gratuitamente, ma il suo successo è dovuto anche ad altre caratteristiche che migliorano il servizio rispetto alla telefonia tradizionale:

- **realizzazione più semplice**: molte funzioni che prima erano distribuite in vari punti di accesso alla rete, ora sono centralizzate, ne consegue una più veloce installazione e riduzione delle attività di amministrazione;
- **rete di trasporto IP**: non è più necessario riservare linee dedicate per il traffico telefonico, la voce viene trasportata nelle reti IP come gli altri dati, è però necessaria una configurazione iniziale;
- **riduzione dei costi**: è significativa per tutti, ma soprattutto per le aziende dove si fanno quotidianamente molte telefonate e spesso internazionali;
- **offerta di servizi a valore aggiunto**: l'infrastruttura VoIP si presta bene a realizzare vari servizi per gli utenti, quali trasferimento di chiamata, richiamo automatico, messaggistica e video-conversazione;
- **anytime-anywhere**: l'utente può telefonare in qualunque momento e ovunque si trovi effettuando un accesso a Internet e usando un account registrato (FIGURA 12).

FIGURA 12 La comunicazione tra dispositivi mobili e fissi

8.2 Il telefono IP

La telefonia su IP richiede l'impiego di tecniche di digitalizzazione della voce, come mostrato nella FIGURA 13.

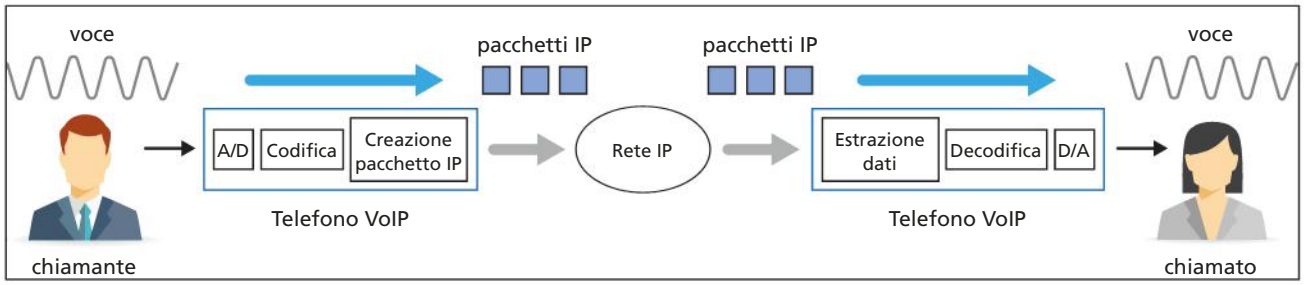


FIGURA 13 Come funziona un telefono VoIP

Il segnale audio che si ottiene dal microfono viene campionato con una velocità elevata e quantizzato per trasformarlo da segnale analogico a digitale (conversione A/D). È successivamente codificato per adattarlo al tipo di canale trasmissivo (riduzione del bit rate, compressione, regolazione della #dinamica).

Dopo la codifica la voce viene inserita nei pacchetti IP da trasmettere sulla rete. In genere, un pacchetto contiene 20 o 30 ms di audio. Questo processo avviene nel dispositivo chiamato **telefono VoIP** utilizzato dalla persona che inizia la chiamata. Quando il pacchetto che contiene la voce arriva a destinazione, viene consegnato al telefono VoIP della persona chiamata e qui subisce il processo inverso di estrazione dei dati dal pacchetto IP, di decodifica e conversione in segnale analogico (conversione D/A).

L'evoluzione delle tecniche di codifica della voce e, più in generale, dei segnali audio, hanno portato alla specifica di vari tipi di codici (abbreviati in **#codec**). Attualmente i codec impiegati nel VoIP comprimono l'audio di un fattore 8 o 10 rispetto alla telefonia tradizionale.

In fase di codifica, i codec applicano la tecnica di **soppressione dei silenzi**, interrompendo la trasmissione durante i periodi di inattività. Infatti, durante una chiamata la percentuale di silenzi è piuttosto elevata e queste tecniche consentono un notevole risparmio di bandwidth.

Nella fase di decodifica, i codec possono compensare eventuali **pacchetti persi** durante la trasmissione così da rendere l'audio accettabile dall'orecchio umano.

Una tecnica utilizzata è la **FEC (Forward Error Correction)** che consiste nell'inserire informazioni ridondanti nei pacchetti unitamente ai dati originali, così da poterle utilizzare in fase di ricezione per recuperare le informazioni contenute nei pacchetti persi. Questa tecnica è valida, però, se la percentuale di pacchetti persi è inferiore al 20%.

Un'altra tecnica utilizzata dal ricevitore è la **ripetizione dei pacchetti**: il contenuto dei pacchetti persi si sostituisce con quello dei pacchetti che li hanno appena preceduti e sono arrivati integri a destinazione.

Spesso, però, quando il numero di pacchetti persi è basso, si sfrutta la capacità di recupero dell'orecchio umano, che è in grado di tollerare bene fino al 5% di pacchetti persi.

Sulla qualità della comunicazione incide maggiormente la variabilità nel tempo di arrivo dei pacchetti (**jitter**) che può essere contenuta con l'impiego di buffer, lato ricezione, prestando però attenzione affinché non introducano un ritardo nella consegna. Per questo motivo si utilizzano dei buffer dinamici, in grado di cambiare dimensione in funzione dello stato della rete.

#techwords

Dinamica

Nel linguaggio musicale la dinamica si occupa dell'**intensità dei suoni** e della loro gradazione da adottare nell'esecuzione di una composizione.

#techwords

Codec

Software o hardware che si occupa della codifica digitale, e decodifica, di un segnale audio o video, utilizzando tecniche di compressione dei dati. Nei sistemi di telecomunicazioni, il codec attua anche la codifica di canale sui dati da trasmettere. Esempi di codec audio sono wav e mp3. Codec spesso utilizzati in VoIP sono G.711 e G.729.

HARDPHONE E SOFTPHONE

Un telefono VoIP è utilizzato tipicamente nelle reti telefoniche aziendali con centralini PBX. Viene chiamato **hardphone**, per distinguerlo dalle applicazioni software installate sui dispositivi che svolgono funzioni analoghe e vengono chiamate **softphone**. Skype è un esempio di applicazione softphone.
Un hardphone VoIP dispone di un display, anche touch, e pulsanti con cui interagire con le varie funzionalità del telefono.
A differenza dei tradizionali telefoni, essi sono dei computer con un Sistema Operativo che permette loro di svolgere compiti avanzati come, per esempio, una videochiamata grazie alla videocamera integrata.
Esistono anche hardphone con un aspetto diverso da quello tipico del telefono, come quelli utilizzati nelle audio conferenze (FIGURA 14).

FIGURA 14 Telefono VoIP per audio conferenze

FIGURA 15
Esempi di softphone

Un softphone è un'applicazione software che si installa su computer, tablet o smartphone fornendo a questi dispositivi le funzionalità tipiche di un telefono VoIP (FIGURA 15). L'interazione con l'utente avviene tramite un'interfaccia grafica.
La prima evidente differenza tra hardphone e softphone è nella mobilità: il primo è un telefono da scrivania, può essere cordless, ma ha una mobilità limitata, il secondo si installa su un dispositivo mobile e segue la persona ovunque si muova.
Un hardphone offre una qualità migliore delle chiamate rispetto a un softphone, in quanto dispone di hardware e software dedicato e quindi non subisce le interferenze delle altre applicazioni che lavorano in background sul computer. Inoltre il funzionamento del softphone è soggetto allo stato, acceso o spento, del dispositivo che lo ospita.
Per contro un hardphone è molto più costoso e meno personalizzabile di un softphone. Infatti, molti softphone offrono le funzionalità di base gratuitamente, richiedono di disporre di hardware come microfono, altoparlanti e videocamera che spesso sono già integrati sui laptop più moderni, oltre che, ovviamente, su tablet e smartphone.

8.3 I centralini telefonici su IP

Le aziende di una certa dimensione installano un centralino telefonico per lo smistamento delle chiamate negli uffici.
La rete telefonica privata di un'azienda viene chiamata **PBX (Private Branch Exchange)**.

PBX-IP è il sistema basato su IP che supera le limitazioni in numero di linee telefoniche e di dispositivi telefonici interni che erano presenti nei PBX tradizionali.

#prendinota

Asterisk è uno dei più diffusi software PBX-IP, è gratuito e si installa su server Linux. Supporta i protocolli standard SIP, MGCP e H.323, lavorando come gateway tra telefoni IP e la rete PSTN. Utilizza un protocollo interno (IAX, Inter-Asterisk eXchange) per la comunicazione tra i PBX. Asterisk si ritrova come componente centrale in molti prodotti commerciali e progetti open source.

La FIGURA 16 mostra gli elementi principali che costituiscono una rete VoIP con PBX-IP:

- **dispositivi IP**: sono quelli coinvolti nella chiamata VoIP e devono essere quindi dei dispositivi IP che dialogano con i protocolli VoIP, hardphone e softphone;
- **VoIP server**: è l'elemento centrale che stabilisce una comunicazione tra chiamante e chiamato, la gestisce e la termina; implementa i protocolli di segnalazione (spiegati più avanti) e assicura il routing corretto dei pacchetti IP che trasportano la voce;
- **Gateway**: assicura la comunicazione tra dispositivi VoIP presenti su reti con caratteristiche diverse (Gateway VoIP) e tra dispositivi VoIP e telefoni connessi alla rete tradizionale PSTN, Public Switched Telephone Network (Gateway PSTN);
- **Gatekeeper**: è un software gestionale che può essere installato sul server o su un hardware a parte e che mantiene i dati per la tariffazione insieme a quelli delle varie chiamate effettuate.

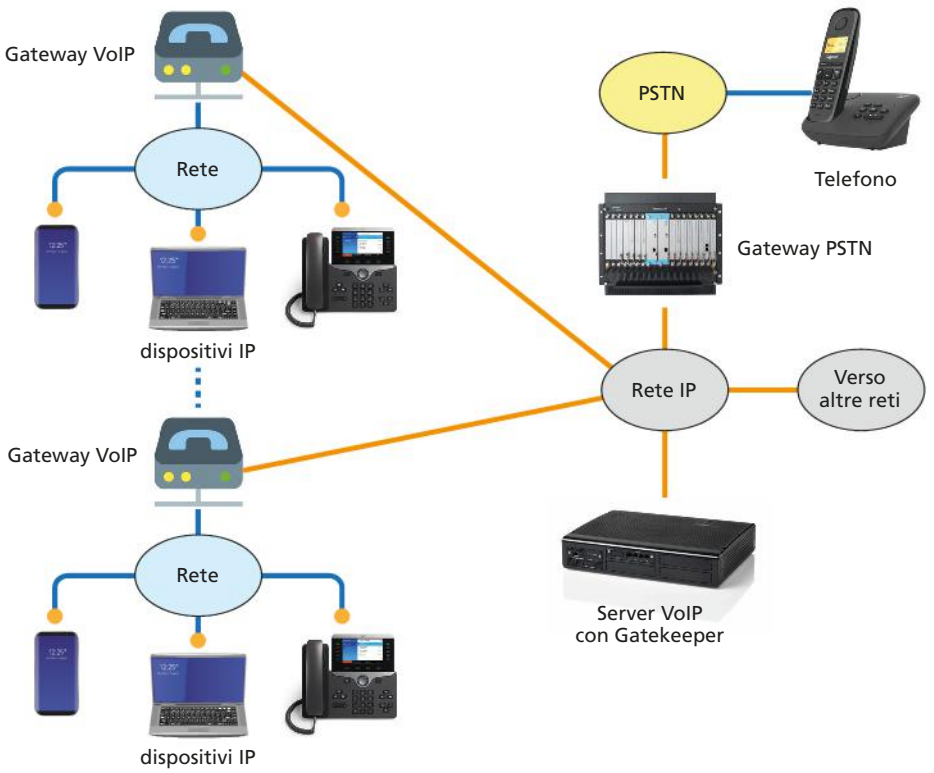


FIGURA 16 Un esempio di sistema PBX-IP

I CENTRALINI VIRTUALI O IN CLOUD

Il centralino virtuale è un sistema PBX installato in remoto e fornito come servizio attraverso la rete.

Le aziende che lo utilizzano possono usufruire dell'eliminazione dei costi di installazione, operativi e di gestione del centralino in quanto è il provider VoIP a farsene carico. È una soluzione altamente scalabile, infatti è possibile aggiungere o rimuovere numeri interni o linee telefoniche in base alle necessità del momento e il personale dell'azienda può lavorare anche all'esterno utilizzando lo stesso numero interno. Un centralino in cloud funziona allo stesso modo di un PBX installato in un'azienda, ma è ospitato su un server cloud.

8.4 Il protocollo SIP

Nei primi anni in cui ha iniziato a svilupparsi la telefonia su Internet, si utilizzavano spesso tecnologie proprietarie, ma anche successivamente, quando furono emessi i primi standard definiti dagli enti di standardizzazione, le modalità di realizzazione e i protocolli specificati erano svariati, sia proprietari sia standard sviluppati in ambito ITU-T.
Tutti, però, mantenevano la distinzione, derivata dalle reti telefoniche, tra:

- **protocollo di segnalazione**: si occupa delle fasi di instaurazione e disconnessione della chiamata e dei servizi aggiuntivi, come quelli di #directory per la gestione dei contatti;
- **protocollo di trasporto**: quando la chiamata è stabilita, gestisce il trasferimento dei pacchetti che trasportano la voce tra i telefoni VoIP.

#techwords

Directory
È un elenco di nomi, nella telefonia è l'elenco telefonico (telephone directory).

Come molti altri servizi che abbiamo visto in questa Unità, anche VoIP è implementato secondo il paradigma Client-Server, quindi il protocollo di segnalazione regola la comunicazione tra client VoIP e server VoIP: il client invia al server la richiesta di chiamare un certo numero e il server lo contatterà per instaurare la comunicazione tra i due telefoni.
Una volta instaurata la comunicazione, si userà un protocollo di trasporto per la trasmissione dei pacchetti che contengono la voce.

Verso la fine degli anni Novanta si diffuse il protocollo di segnalazione **H.323** definito in ambito ITU-T. H.323 era una suite di protocolli e utilizzava molti principi della telefonia tradizionale. La sua complessità di realizzazione e la diffusione sempre maggiore del VoIP generò l'esigenza di protocolli più snelli e semplici da implementare. Nacque un gruppo di lavoro in ambito IETF per la definizione di un nuovo protocollo di segnalazione per il VoIP che portò alla specifica di **SIP (Session Initiation Protocol)**.
IETF definì anche un nuovo protocollo di trasporto, **RTP**, che si inserisce tra il protocollo di livello Transport UDP e i protocolli VoIP di livello applicativo, come visto nella precedente Lezione.
Le specifiche di SIP sono state pubblicate nell'**RFC 3261** nel 2002; pur non essendo mai stato reso obsoleto, questo RFC è stato più volte aggiornato negli anni successivi, con oltre venti nuovi RFC.
Il protocollo SIP offre meccanismi per:

- la notifica della chiamata al chiamato, equivalente allo squillo del telefono nella telefonia tradizionale;
- la negoziazione tra il chiamante e il chiamato sui dispositivi da usare e sulla codifica;
- la chiusura della chiamata, equivalente al riaggancio del telefono nella telefonia tradizionale;
- permettere al chiamante di conoscere l'indirizzo del chiamato, infatti potrebbe aver ottenuto l'indirizzo IP da un DHCP, quindi non fisso;
- la gestione della chiamata, per esempio è possibile allargare la conversazione ad altri interlocutori, cambiare codifica durante la chiamata, ecc.

Il protocollo SIP può essere usato insieme a RTP, ma non è un obbligo, infatti le specifiche prevedono che possa lavorare anche con altri protocolli e servizi.

Su ogni dispositivo SIP è installata un'applicazione di tipo Client-Server che si chiama **UA** (**User Agent**). Un UA può operare come client, **UAC** (User Agent Client), e inviare una richiesta al server, oppure come server, **UAS** (User Agent Server), che elabora la richiesta del client e invia la risposta.

> Un UA può svolgere entrambi i ruoli, quindi essere sia un client che inizia una conversazione inviando a un server un messaggio di request, sia un server che risponde alla richiesta di un client.

Si possono avere diversi tipi di UAS:

- **Proxy server**: riceve le richieste SIP da un client e le inoltra in rete verso gli UAS, riceve le risposte e le inoltra verso gli UAC;
- **Redirect server**: invece di inoltrare la chiamata al server di destinazione, restituisce al client il suo indirizzo, così questi potrà contattare il nuovo server direttamente;
- **Registrar server**: gli utenti SIP devono registrare la loro posizione in un Registrar server, che di solito si trova all'interno di un Proxy o Redirect server e memorizza le informazioni ricevute in un Location database.

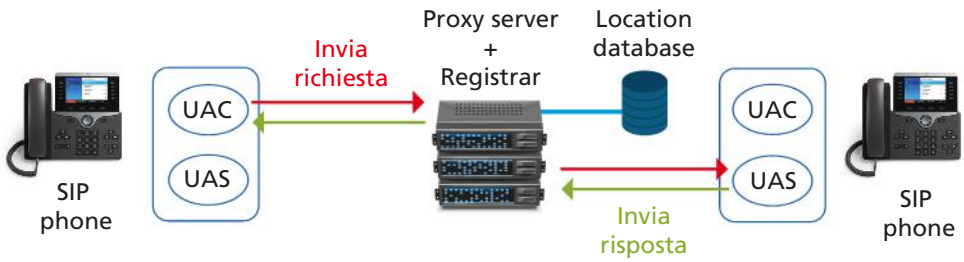


FIGURA 17 Componenti Client-Server di SIP con Proxy server

La **FIGURA 17** mostra un esempio di comunicazione tra due telefoni SIP tramite un Proxy server che svolge anche le funzioni di Registrar server.

GLI INDIRIZZI SIP

Nel protocollo SIP gli indirizzi sono definiti nel formato URI (Uniform Resource Identifier). Esempi di indirizzi SIP:

- sip:bianchi@azienda.com
- sip:bianchi@178.25.49.161
- sips:+00393351069482@azienda.com:5062

dove "sips" indica che il trasporto dei dati avviene con TLS.

Gli indirizzi SIP possono essere inseriti in una pagina web come URL: quando il visitatore clicca sull'indirizzo, l'applicazione VoIP del suo dispositivo si attiva e invia la richiesta di chiamata.

I COMANDI DI SIP

Il protocollo SIP è stato progettato in origine per essere molto semplice, con un numero limitato di comandi:

- INVITE: richiesta per stabilire una connessione, *si invita* un utente a ricevere una chiamata;
- ACK: conferma della ricezione della richiesta INVITE;

- BYE: termina la connessione tra gli utenti;
- CANCEL: cancella ogni azione in sospeso, generalmente una richiesta INVITE;
- OPTIONS: interroga il server per avere un elenco delle sue funzionalità e stato;
- REGISTER: comunica a un SIP registrar uno o più indirizzi di contatto dell'utente (UA).

I numerosi RFC di aggiornamento hanno introdotto nuovi comandi: SUBSCRIBE, NOTIFY, INFO, UPDATE, ecc.

Per i messaggi di **response** (le risposte alle richieste) SIP usa dei codici simili a quelli usati da HTTP, per esempio:

- 100 (Trying);
- 200 (OK);
- 404 (Not found);
- 500 (Server internal failure).

Nella FIGURA 18 è mostrata una chiamata tra due telefoni SIP, il successivo scambio di informazioni in streaming tramite i protocolli RTP e RTCP e la chiusura della connessione.

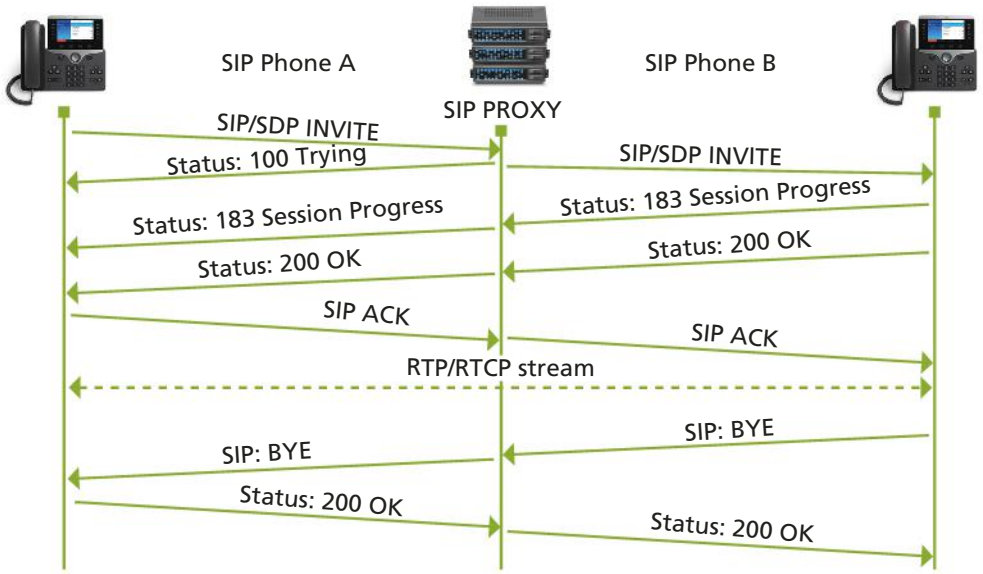


FIGURA 18 Esempio di colloquio tra telefoni SIP tramite Proxy server

#prendinota

WhatsApp utilizza SIP?
La famosa piattaforma di messaggistica offre servizi di telefonia su IP, infatti permette di effettuare audio e video chiamate dall'applicazione client installata su dispositivi mobili o sul computer.
WhatsApp non utilizza SIP, bensì un protocollo di segnalazione proprietario.

FISSA LE CONOSCENZE

- Descrivi le principali caratteristiche del VoIP.
- Qual è il ruolo dei codec nei servizi VoIP?
- Che cosa sono i PBX-IP?
- Quale protocollo è stato standardizzato da IETF per la telefonia su IP?
- Spiega il ruolo client e server dello User Agent presente sui dispositivi SIP.

9 PACKET TRACER: SERVER SMTP E POP3

In questa esercitazione di laboratorio realizzeremo, con il simulatore Packet Tracer, un server di posta elettronica (EMAIL server) in ogni LAN, configurando i protocolli SMTP e POP3, visti nella Lezione 6, che governano il servizio.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare 3 reti LAN con un server di posta elettronica configurato per il servizio EMAIL in ciascuna rete. Di seguito, su ogni PC, creare un'utenza. Quindi verificare il funzionamento del servizio simulando l'invio delle email tra utenti di qualsiasi rete.

➜ ANALISI DEL PROBLEMA

Per poter utilizzare la posta elettronica dobbiamo innanzitutto configurare l'SMTP Service e il POP3 Service su un server di ogni LAN. Per fare questo bisogna creare un dominio (Domain Name) di posta da associare al dominio della rete. La scelta tipica è di chiamare **www.azienda.it** il dominio del sito aziendale e chiamare **azienda.it** il dominio di posta avendo così utenze email del tipo **utente@azienda.it**. Come abbiamo visto nell'Unità 7, i nomi di dominio vanno configurati sul server DNS. In questo caso utilizzando dei Resource Records di tipo **CNAME** che creino un alias di posta.

➜ SVOLGIMENTO

Nella FIGURA 19 è mostrato un possibile scenario con 3 LAN aziendali collegate in WAN da una rete di router.

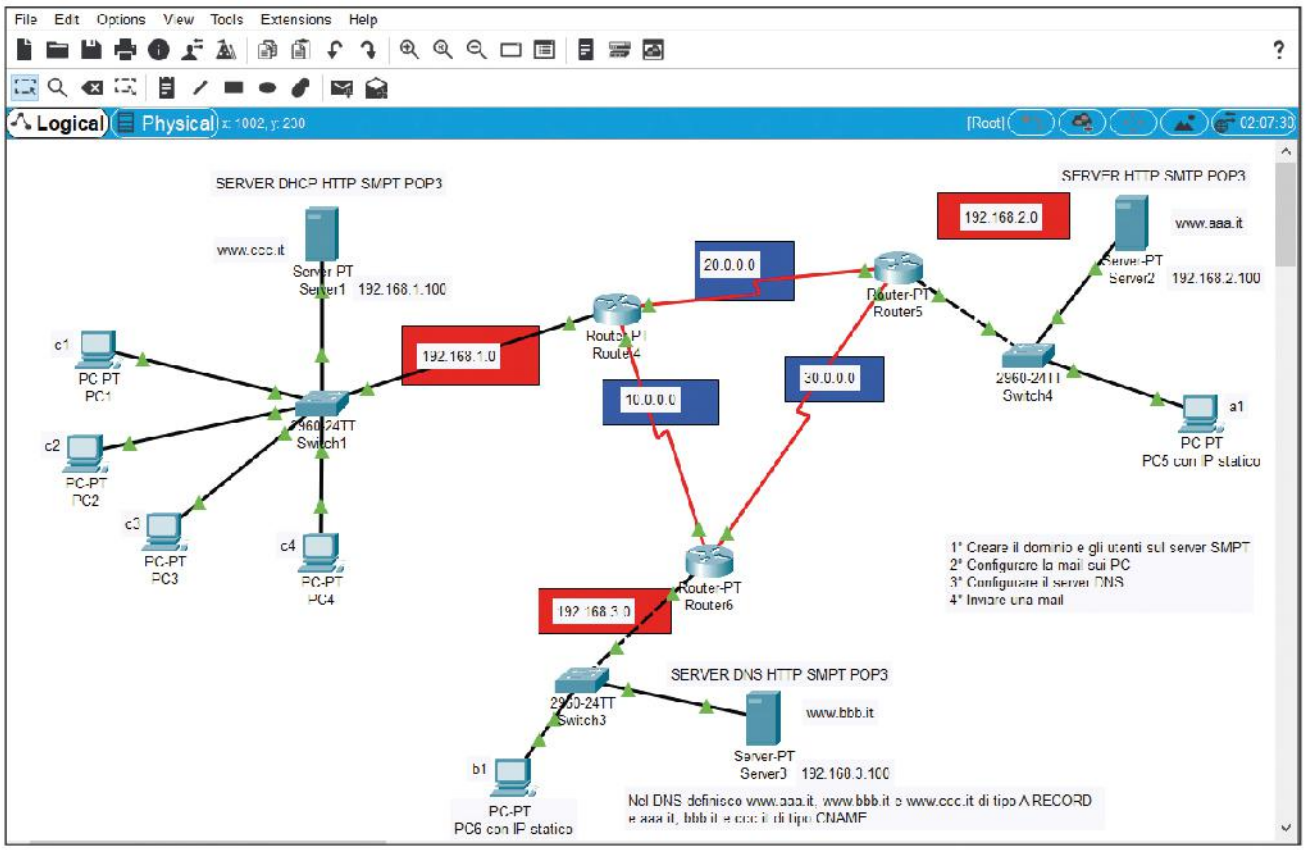


FIGURA 19 Scenario con server di posta elettronica

Per le 3 reti locali utilizziamo i soliti indirizzi privati 192.168.1.0, 192.168.2.0 e 192.168.3.0. Assegniamo manualmente alle interfacce FastEthernet0/0 dei router delle 3 LAN gli IP 192.168.1.254, 192.168.2.254 e 192.168.3.254 tutte con subnet mask 255.255.255.0. Tali indirizzi saranno i gateway per gli host di ciascuna LAN.
Per le 3 reti costituite dalle coppie di router usiamo gli indirizzi 10.0.0.0, 20.0.0.0 e 30.0.0.0, assegnando manualmente gli indirizzi alle porte Serial2/0 e Serial3/0 di ogni router. Sui 3 router configuriamo poi il RIP dalla scheda Config come fatto nelle precedenti esercitazioni.
Ai 3 server su cui configureremo tutti i servizi assegniamo manualmente gli IP 192.168.1.100, 192.168.2.100 e 192.168.3.100.
I servizi da configurare su ogni server di LAN sono i seguenti.

- **Server1** 192.168.1.100 della rete 192.168.1.0:
  – DHCP (essendoci tanti PC, conviene)
  – HTTP (con dominio www.ccc.it)
  – SMTP e POP3 (con dominio ccc.it)
- **Server2** 192.168.2.100 della rete 192.168.2.0:
  – HTTP (con dominio www.aaa.it)
  – SMTP e POP3 (con dominio aaa.it)
- **Server3** 192.168.3.100 della rete 192.168.3.0:
  – DNS (unico per tutto lo scenario)
  – HTTP (con dominio www.bbb.it)
  – SMTP e POP3 (con dominio bbb.it)

Dei servizi DHCP, HTTP e DNS abbiamo già parlato diffusamente nelle precedenti esercitazioni. Occupiamoci dettagliatamente del servizio di posta elettronica coi protocolli SMTP e POP3.
Per configurare il servizio di posta sui server clicchiamo su ogni server e selezioniamo **Services**. Tra i servizi proposti selezioniamo **EMAIL** e settiamo un **Domain Name**. Di seguito configuriamo tanti **User** quanti i PC della rete e li aggiungiamo col pulsante +. A ogni utente diamo una **Password** (per comodità diamo a tutti la stessa password, per esempio 123).
La **FIGURA 20** riassume la configurazione delle utenze sui 3 server in base allo scenario proposto.

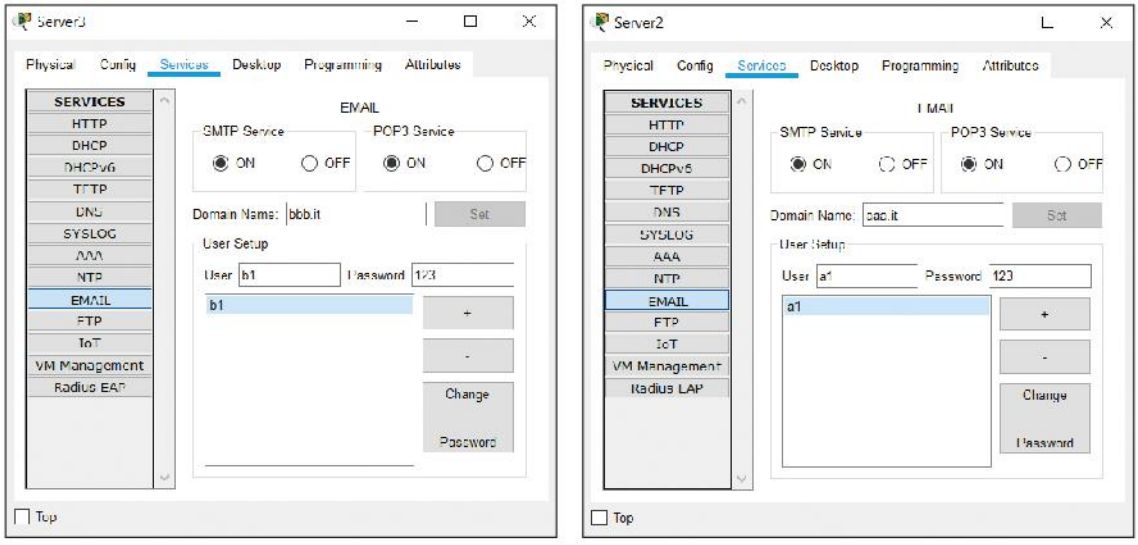


FIGURA 20
Configurazione del servizio EMAIL sui 3 server

Il passo successivo consiste nel creare le utenze sui singoli PC. Prendiamo per esempio il PC1 della rete 192.168.1.0, su cui creiamo l'utenza con indirizzo email **c1@ccc.it** con gli stessi parametri di configurazione impostati sul Server1. Clicchiamo su PC1, selezioniamo la scheda **Desktop**, quindi l'icona **Email** che apre il MAIL BROWSER e poi **Configure Mail**. La FIGURA 21 mostra i 3 passaggi.

Nell'ultima scheda si devono inserire anche gli IP dei server EMAIL della rete di appartenenza che svolgono la funzione di ricezione (*Incoming Mail Server*) e inoltro (*Outgoing Mail Server*) della posta elettronica.

Identica operazione va ripetuta su tutti i PC di ogni rete dello scenario. Il pulsante **Save** memorizza le impostazioni scritte su ogni client di posta configurato.

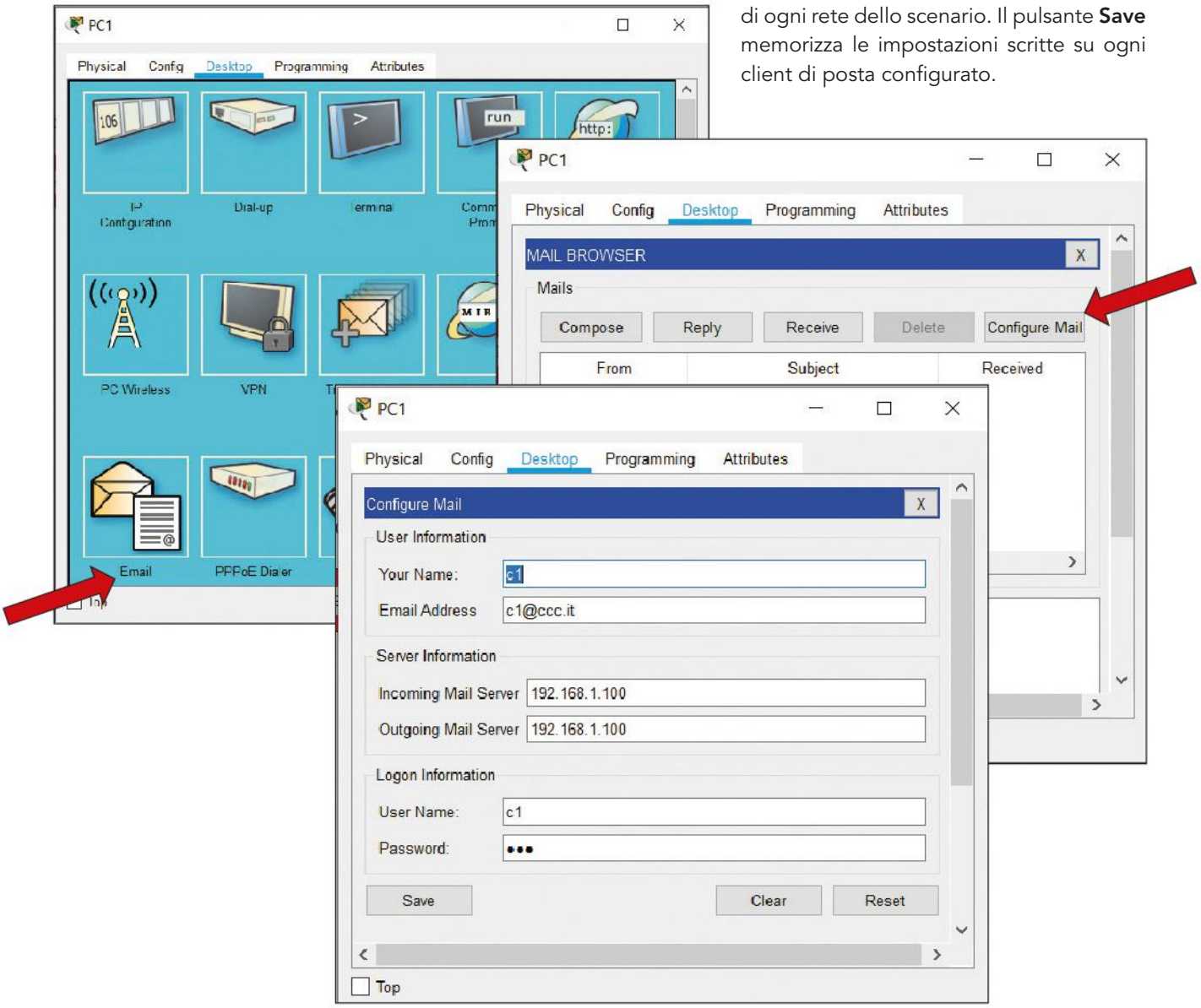


FIGURA 21 Configurazione dell'Email Address su PC1

Resta un ultimo fondamentale passo da fare prima di poter inviare una email: configurare sul server DNS (unico per tutto lo scenario) gli **alias**.
Dovremo quindi dichiarare 3 Resource Records di tipo **CNAME** associandoli ai 3 Resource Records di tipo **A Record** che definiscono l'Host Name del server di ogni LAN.

Dobbiamo pertanto associare *azienda.it* a *www.azienda.it* in modo tale che le email di *nome@azienda.it* possano essere recapitate al server giusto e ricevute dal server giusto.
Apriamo quindi la scheda Services di Server3, selezioniamo il servizio DNS e aggiungiamo 3 entry di tipo CNAME (aaa.it, bbb.it e ccc.it) mediante il pulsante **ADD** come mostrato in FIGURA 22.
Siamo ora pronti a inviare una email tra due host qualsiasi. Da PC1 dove c'è l'account c1@ccc.it inviamo una email a PC5 dove c'è l'account a1@aaa.it:

- clicchiamo su PC1, selezioniamo la scheda **Desktop**, quindi l'icona **Email** che apre il MAIL BROWSER e poi **Compose**. Compiliamo i campi **To** e **Subject**, inseriamo un testo qualsiasi e clicchiamo su **Send**;
- clicchiamo su PC5 selezioniamo la scheda **Desktop**, quindi l'icona **Email** che apre il MAIL BROWSER e poi **Receive**.

La FIGURA 23 mostra la email inviata da PC1 e ricevuta da PC5.

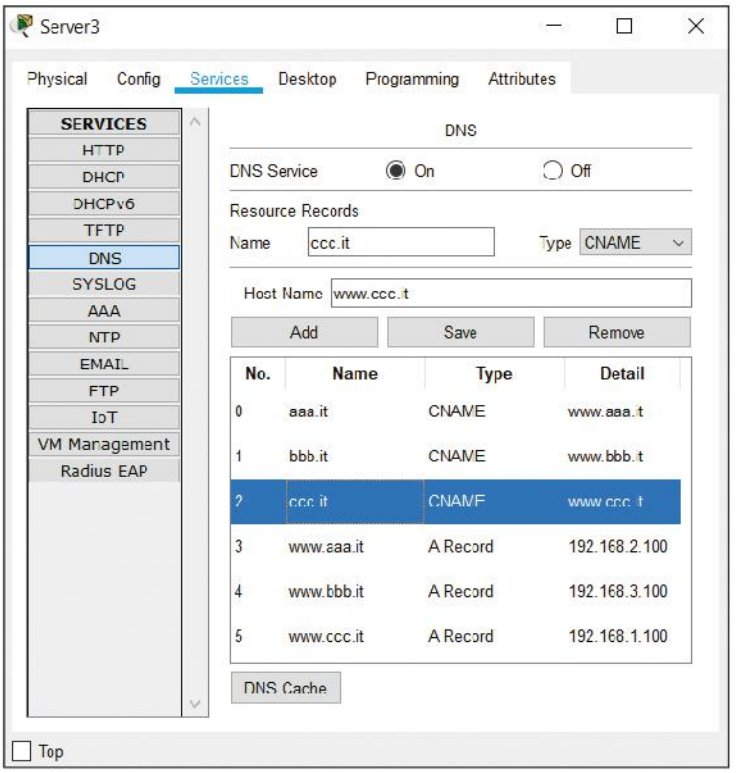
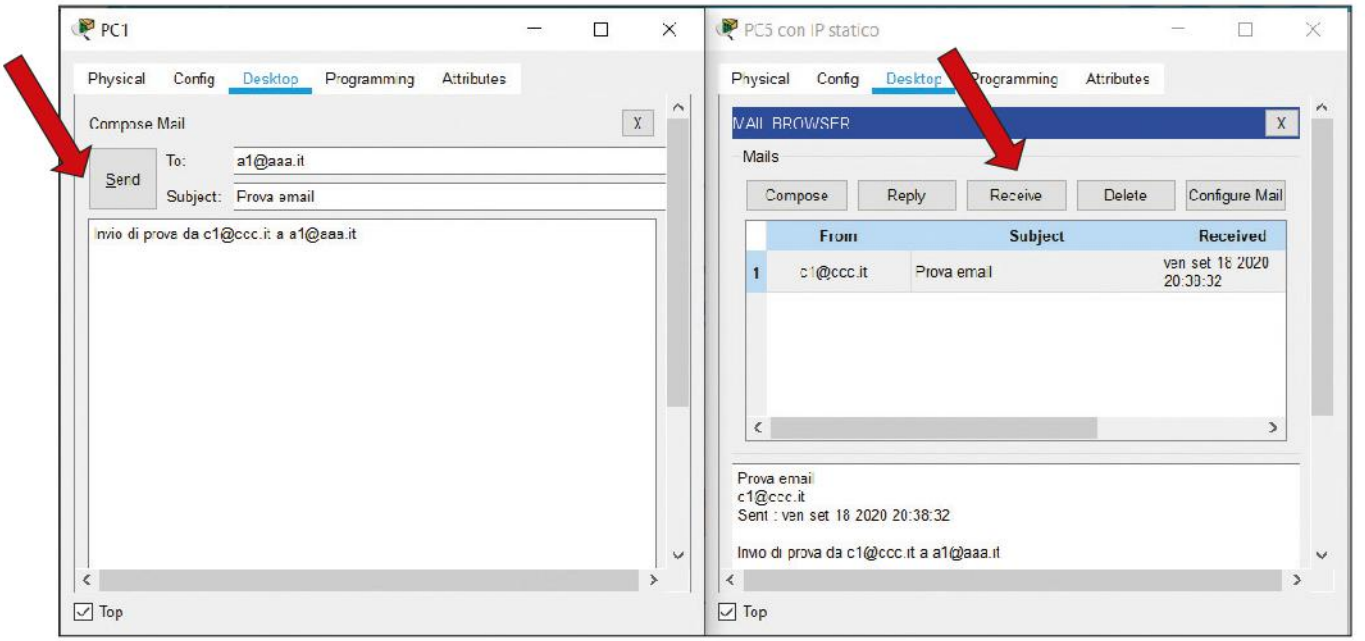


FIGURA 22 ▲ Configurazione degli alias sul server DNS (Server3)

FIGURA 23 ▼ Invio e ricezione di una email da PC1 a PC5



FISSA LE CONOSCENZE

- Come si configurano le utenze di posta sul server?
- Come si configurano le utenze di posta sui PC?
- Perché bisogna aggiungere delle entry al server DNS?

10 PACKET TRACER: SERVER FTP

In questa esercitazione di laboratorio realizzeremo, con il simulatore Packet Tracer, un server per la condivisione dei file, configurando il protocollo FTP, visto nella Lezione 4, che governa il servizio.

esercizio

File sorgenti
Scarica il file

➜ PROBLEMA

Realizzare 3 reti LAN con un unico server FTP, dargli un nome e mapparlo sul server DNS. Quindi verificare il funzionamento del servizio simulando il trasferimento di un file tra utenti di qualsiasi rete.

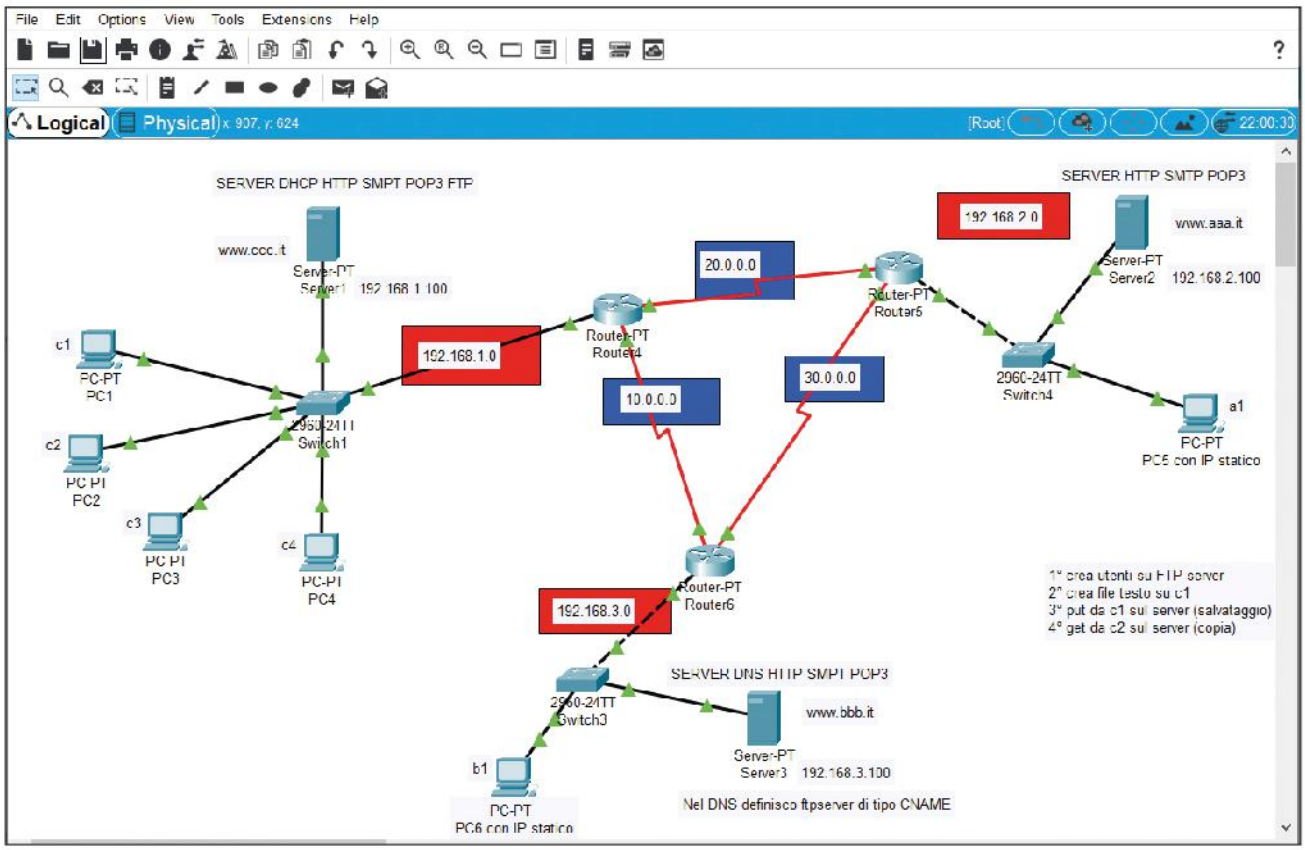
➜ ANALISI DEL PROBLEMA

Per poter utilizzare la condivisione dei file su un server unico dobbiamo innanzitutto configurare l'FTP Service sul server scelto. La rete che ha "in casa" quel server può immediatamente usufruire del servizio utilizzando l'indirizzo IP del server stesso. Poiché è richiesto di rendere operativo il servizio su tutte le reti dello scenario, occorre dare un nome al server FTP e inserirlo nel server DNS.
Come abbiamo visto nell'Unità 7, i nomi vanno configurati sul server DNS. In questo caso utilizzando un Resource Records di tipo **CNAME** per creare un alias per il server FTP.

➜ SVOLGIMENTO

Riproponiamo lo scenario dell'esercitazione precedente (FIGURA 24) in cui configurare il servizio FTP su Server1.

FIGURA 24 Scenario con server FTP su Server1



Per configurare il servizio FTP clicchiamo su **Server1** e selezioniamo **Services**. Tra i servizi proposti selezioniamo **FTP** e configuriamo tanti **User** quanti i PC della rete e a ogni utente diamo una **Password** (per comodità diamo a tutti sempre la stessa password, per esempio 123).
La **FIGURA 25** riassume la configurazione delle utenze su Server1 con privilegi diversi assegnati ai vari client (Write, Read, Delete, Rename, List).

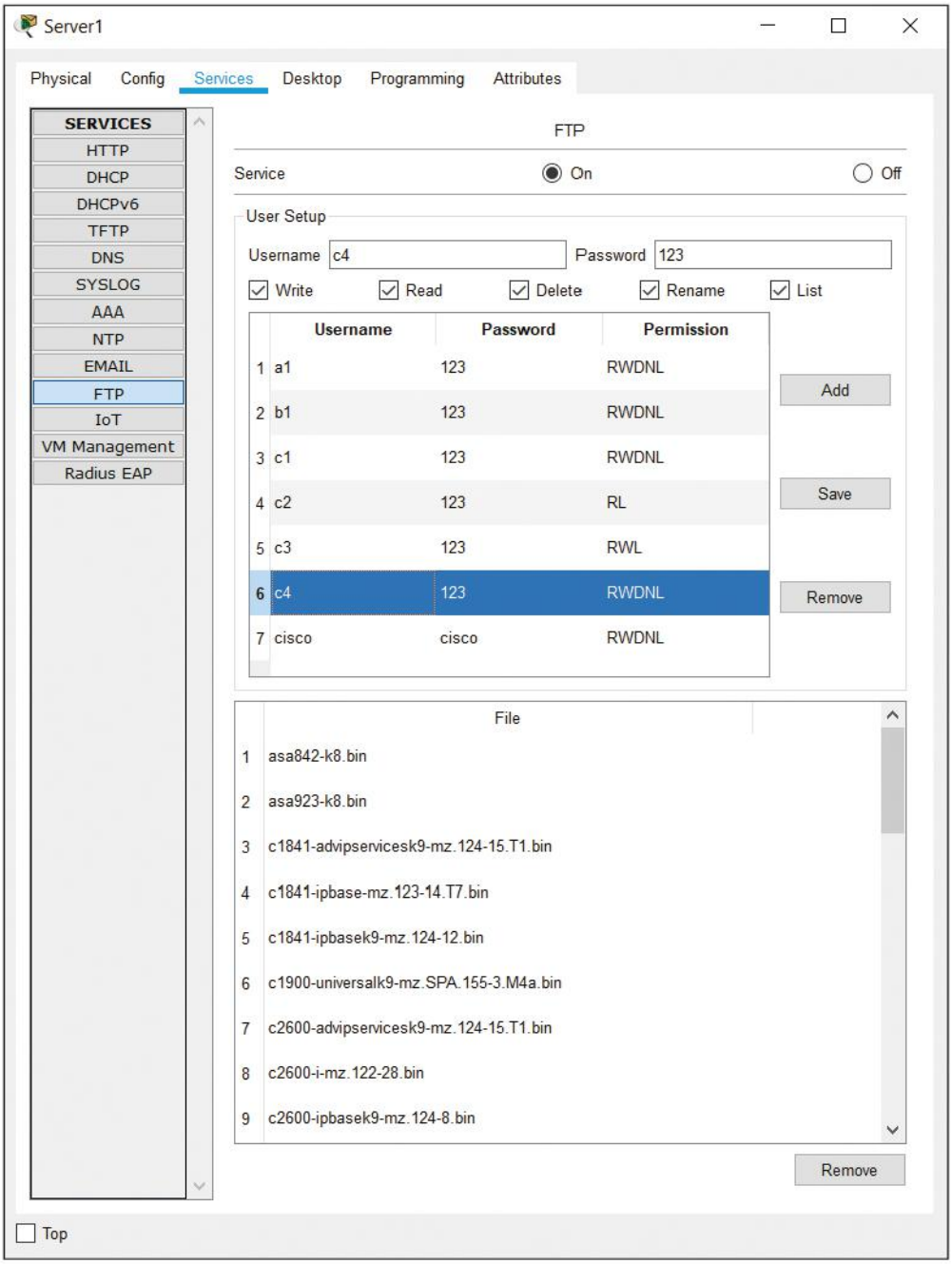


FIGURA 25 Configurazione delle utenze FTP su Server1

A questo punto sarebbe già possibile eseguire dei trasferimenti di file **tra host della rete** 192.168.1.0 avviando il server FTP da qualsiasi PC col comando:

```
C:\>ftp 192.168.1.100
```

#prendinota

Ogni rete può configurare un server DNS locale con mappati i nomi delle proprie risorse per evitare agli utenti di ricordarsi gli indirizzi IP.

Poiché invece vogliamo che il server fornisca il servizio anche alle altre reti, andiamo su **Server3** (il server DNS, unico per tutto lo scenario) e aggiungiamo una entry con Name **ftp_server** e Type **CNAME**, creando così un altro **alias** di 192.168.1.100 (o www.ccc.it) come mostrato in **FIGURA 26**.

Questo consentirà a tutti gli host delle 3 reti di avviare il servizio col comando:

```
C:\>ftp ftp_server
```

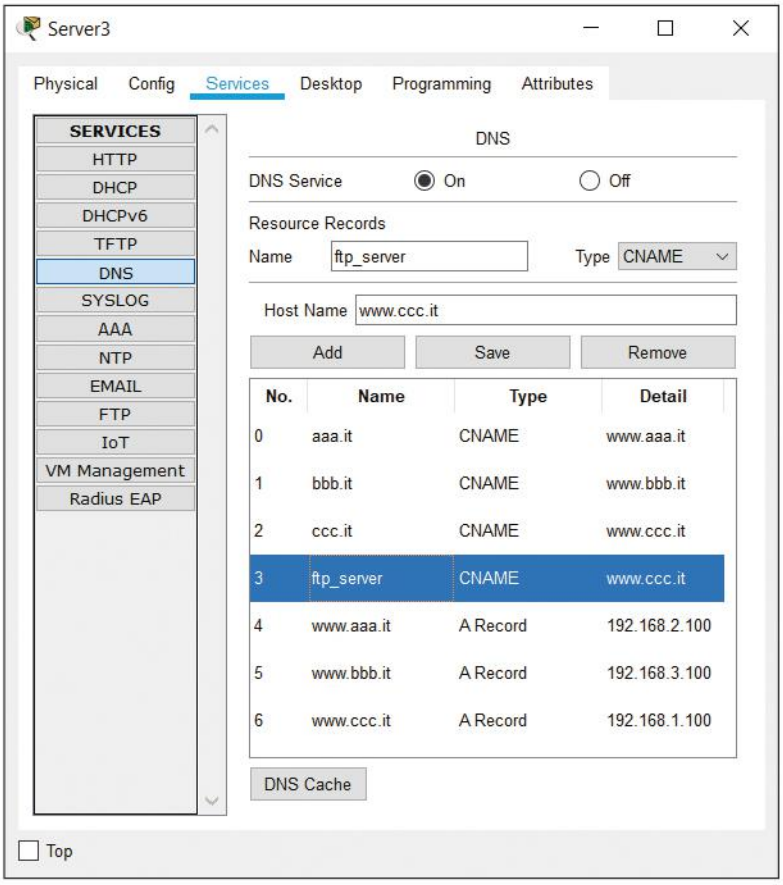


FIGURA 26 Inserimento del CNAME ftp_server nel DNS

Terminata la configurazione del server FTP e di quello DNS, siamo pronti a testare il funzionamento del servizio tra, per esempio, c4 e a1 nel seguente modo:

1) il client c4 crea il file da trasferire mediante il **Text Editor** dalla scheda **Desktop** e lo salva in locale col nome **File_c4-a1.txt** **(FIGURA 27)**, quindi mediante il **Command Prompt** col comando **dir** verifica che il file sia stato salvato **(FIGURA 28)**;
2) il client c4 inoltra richiesta di connessione al server FTP Server1 (Figura 28) mediante il comando C:\>ftp ftp_server a cui fa seguito la richiesta Username (c4) e Password (123);
3) il client c4 inoltra richiesta a Server1 di salvataggio di una copia del file (Figura 28) mediante il comando ftp>put File_c4-a1.txt;
4) il server Server1 accetta la richiesta e il trasferimento ha luogo (Figura 28), il client c4 può procedere ad altri trasferimenti o chiudere la connessione con comando **quit**;

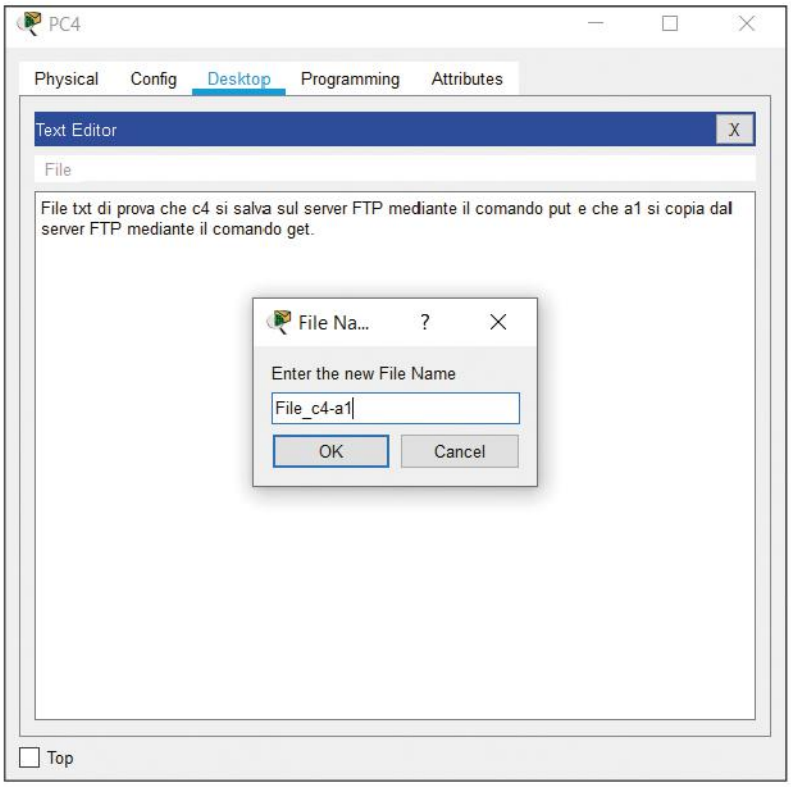


FIGURA 27 Creazione e salvataggio su c4 in locale del file txt

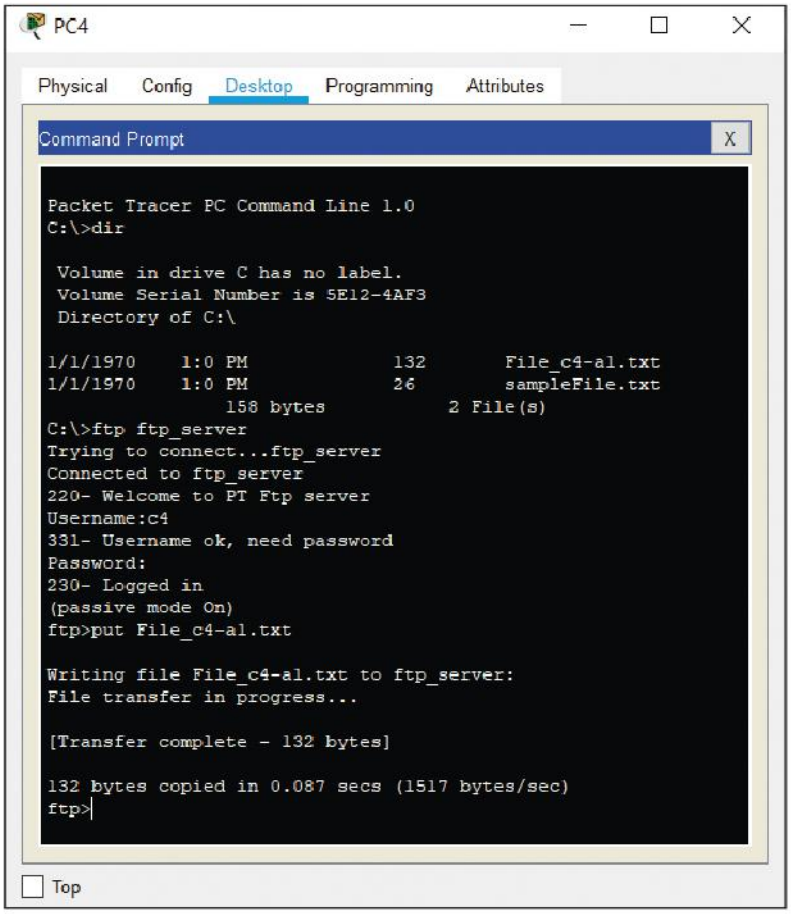
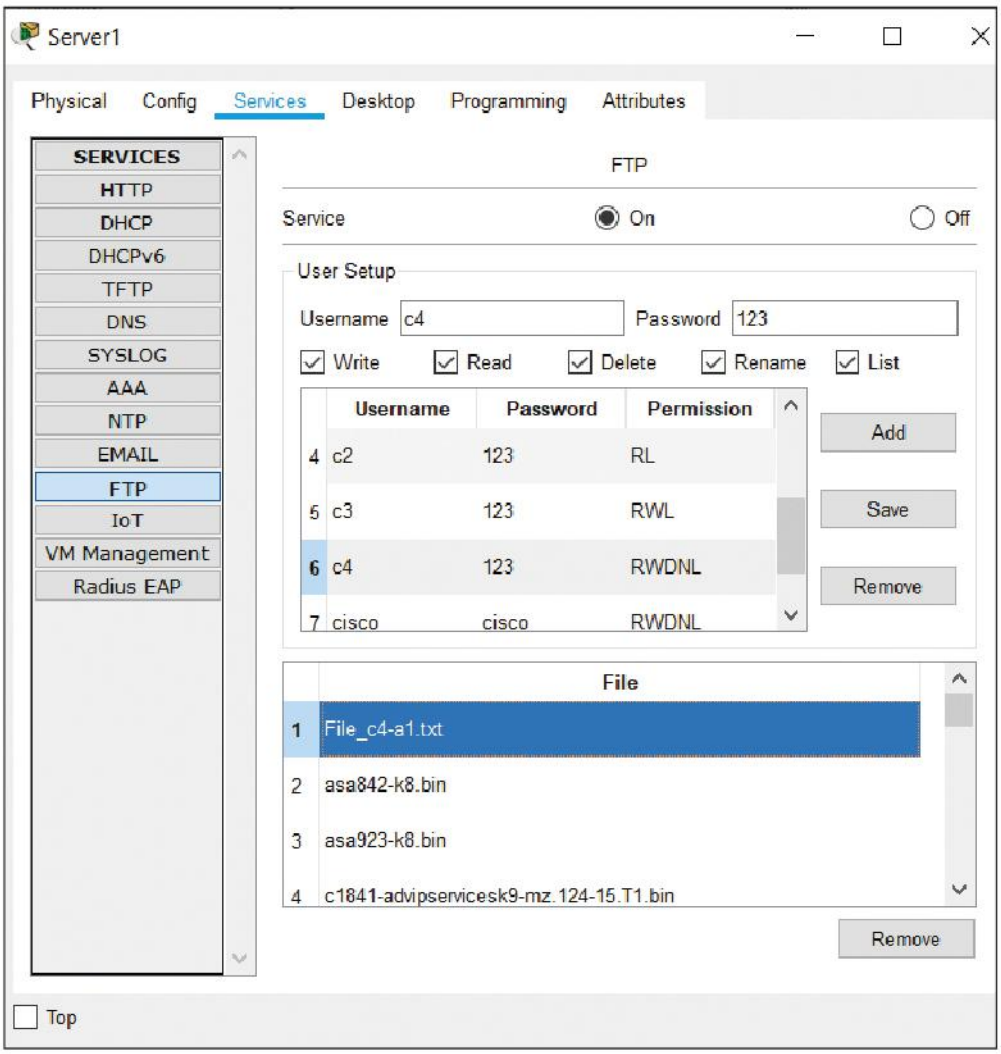


FIGURA 28 Connessione di c4 al server FTP e trasferimento file su Server1

5) terminato il trasferimento, Server1 verifica che sia avvenuto cliccando sulla scheda **Services**, selezionando **FTP** e verificando la presenza del file (FIGURA 29);

FIGURA 29 Salvataggio del file su Server1



6) il client a1 inoltra richiesta di connessione al server FTP Server1 (FIGURA 30) mediante il comando

```
C:\>ftp ftp_server
```

a cui fa seguito la richiesta Username (a1) e Password (123);

7) il client a1 inoltra richiesta a Server1 di trasferimento di una copia del file (Figura 30) mediante il comando

```
ftp>get File_c4-a1.txt
```

8) il server Server1 accetta la richiesta e il trasferimento ha luogo (Figura 30), il client a1 può procedere ad altri trasferimenti o chiudere la connessione con comando **quit**;

9) il client a1 mediante il **Command Prompt** col comando **dir** verifica che il file sia stato salvato (Figura 30).

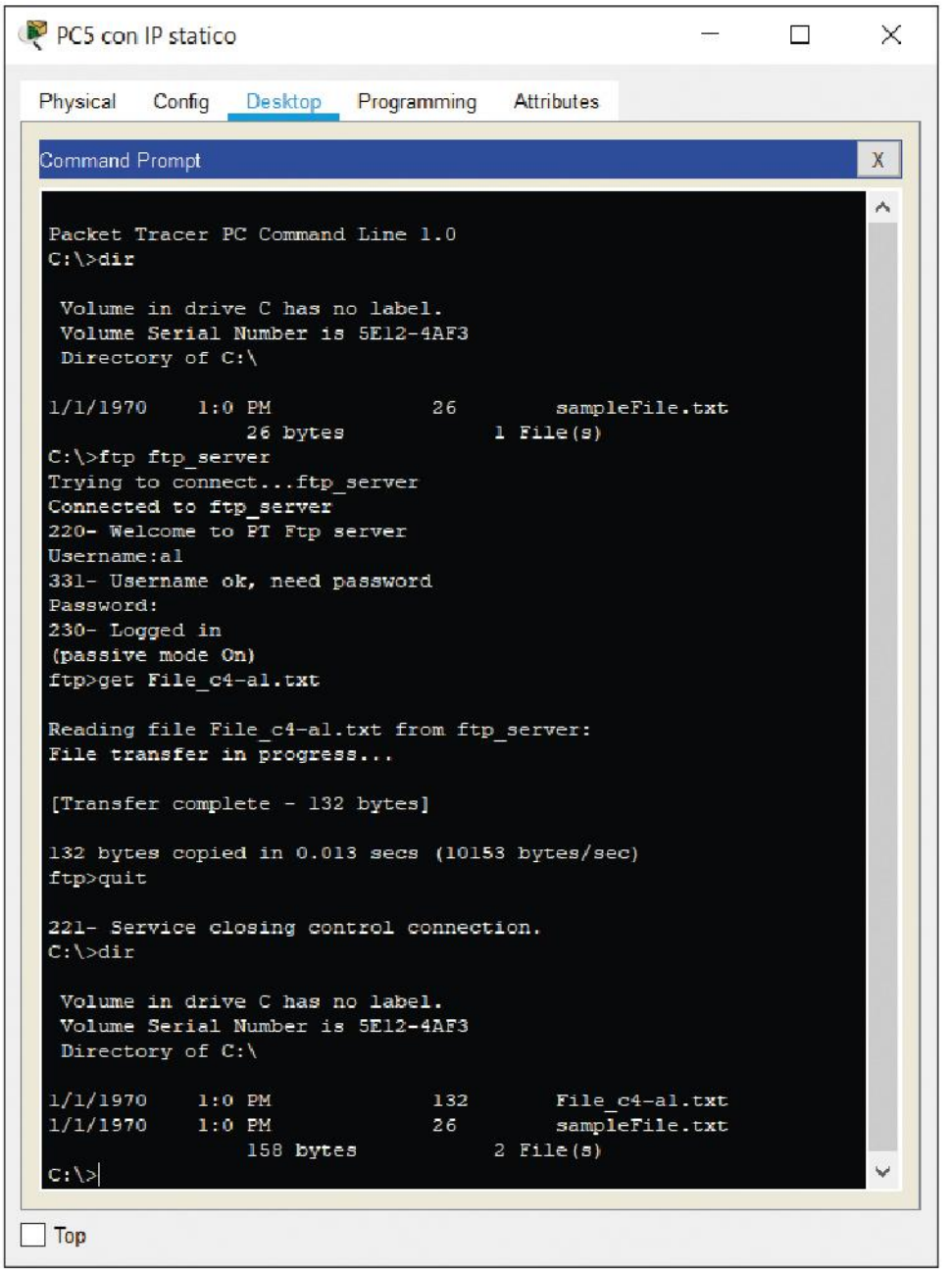


FIGURA 30 Connessione di a1 al server FTP e trasferimento file su a1

FISSA LE CONOSCENZE

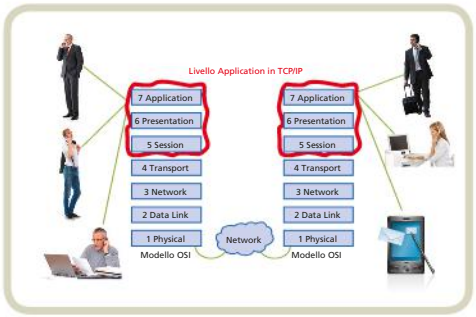
- Come si configura il servizio FTP su un server?
- Perché bisogna aggiungere una entry al server DNS?
- Come possiamo creare in Packet Tracer un file di testo da trasferire?
- Come si può verificare che il trasferimento del file sia avvenuto correttamente?

RIPASSIAMO INSIEME

1 Una visione d'insieme della rete Internet

Le componenti di Internet partono dall'hardware e risalendo lo stack TCP/IP arrivano alle applicazioni software, che offrono i servizi di rete agli utenti finali. A livello più basso Internet è una rete formata da hardware e software che permette a due dispositivi di comunicare. A livello più alto, Internet è un'infrastruttura che fornisce servizi alle applicazioni distribuite su diversi sistemi.

2 Il livello Application e i suoi protocolli

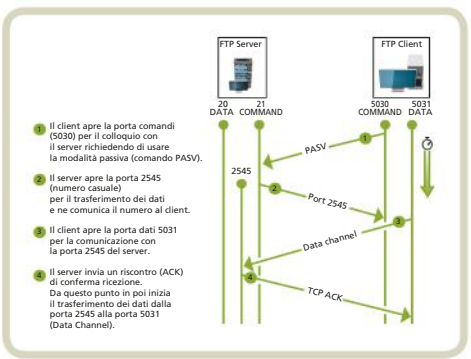


Il livello Application è l'ultimo dell'architettura TCP/IP e corrisponde ai 3 livelli più alti del modello OSI. I protocolli a questo livello possono seguire un modello Client-Server oppure uno Peer-to-Peer.

3 Telnet: il protocollo per l'emulazione di terminale

Il client Telnet è un'applicazione di emulazione di terminale (testuale) che permette di accedere ad applicazioni che si trovano su host remoti, come se l'utente fosse direttamente connesso a tali sistemi. L'host di destinazione deve avere un server di Telnet. Oltre a connettere computer, anche con diversi Sistemi Operativi, Telnet consente di instaurare sessioni FTP, SMTP, POP3 e IMAP4.

4 FTP: il protocollo per il trasferimento di file



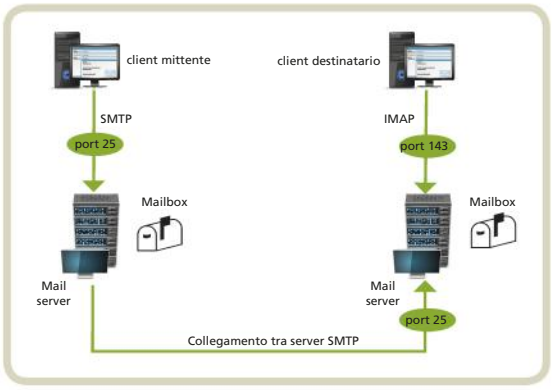
FTP è il protocollo standard per il trasferimento di file tra un client (client FTP) e un server (server FTP). Utilizza TCP come protocollo di trasporto, mentre la sua versione più semplice, denominata Trivial FTP, usa UDP. A differenza di altri protocolli, FTP utilizza due canali per la comunicazione tra client e server: uno per l'invio di comandi e l'altro per i dati. Inoltre sono previste due modalità predefinite di accesso: *utente* e *anonima*; quest'ultima deve essere utilizzata con attenzione in quanto può causare problemi di sicurezza.

5 HTTP: il protocollo per le applicazioni web

Il protocollo HTTP regola lo scambio di messaggi tra il web server e il web client (browser). Ogni pagina web ha un suo indirizzo simbolico detto URL, HTTP è invocato da TCP/IP ogni volta che l'URL contiene nel primo campo la parola http. HTTP è un protocollo stateless che permette sia la ricerca che il recupero

dell'informazione in maniera veloce, seguendo gli hyperlink. L'utilizzo dei cookie rende il protocollo HTTP stateful. Il protocollo HTTP mette a disposizione del client una serie di metodi per inviare le richieste al server. Lo scambio di messaggi tra web server e web client può anche avvenire tramite un proxy server.

6 SMTP, POP3 e IMAP4: i protocolli per la posta elettronica



Per poter inviare e ricevere posta elettronica (email) un client deve connettersi a un server in grado di gestire il servizio. I tipi di server cui connettersi sono 2: SMTP per la posta in uscita o in trasferimento (fase di invio) e POP3 o IMAP4 per la posta in ingresso (fase di ricezione). SMTP è un protocollo che permette soltanto di inviare messaggi di posta, ma non di richiederli a un server, per fare questo il client di posta deve usare i protocolli POP3 o IMAP4. Con POP3 i messaggi di posta elettronica, per essere letti, vengono scaricati in locale sul computer e cancellati dal server. In alternativa a POP3 si usa IMAP4 che è un protocollo particolarmente indicato per i client in grado di mantenere una connessione continua a un server (online). POP3 e IMAP4 gestiscono l'autenticazione tramite username e password.

7 I protocolli per le applicazioni multimediali

I dati di tipo audio e video richiedono un elevato impiego di risorse. Per soddisfare i requisiti di questo tipo di traffico è stato necessario definire nuovi standard. In ambito IETF sono così nati: il protocollo RTSP, per le applicazioni di tipo streaming e il protocollo RTP per il traffico multimediale di tipo real time interattivo.

8 VoIP: la tecnologia per la voce

VoIP (Voice over IP), o Internet Telephony, è un'applicazioni real time che utilizza i protocolli della rete IP per implementare una rete telefonica distribuita e flessibile. Le applicazioni VoIP seguono il modello Client-Server e si appoggiano a UDP per il trasposto delle informazioni. Nel caso di trasporto di dati audio e video, si usa il protocollo RTP. La telefonia su IP richiede tecniche di digitalizzazione della voce negli apparati telefonici, distinti in hardphone e softphone. Anche il tradizionale sistema dei centralini PBX delle aziende è stato rivisto per supportare VoIP (PBX-IP). Uno degli standard più diffusi per realizzare il VoIP è il protocollo SIP, definito da IETF.

VERIFICA DI FINE UNITÀ

Quali delle seguenti affermazioni sono vere (V) e quali false (F)?

1. I servizi offerti dal livello Application consentono all'utente di interfacciarsi alla rete. V F
2. Il livello Application si occupa del controllo di flusso dei dati a livello end-to-end. V F
3. Le applicazioni Peer-to-Peer possono utilizzare reti Client-Server. V F
4. Il protocollo Telnet usa TCP come protocollo di trasporto. V F
5. Il protocollo FTP usa due connessioni a due porte diverse. V F
6. Il protocollo HTTP è usato per scrivere le pagine web. V F
7. SMTP è usato per realizzare il file sharing. V F
8. POP3 è il protocollo usato per scaricare la posta ricevuta dal server. V F
9. Telnet è un'applicazione di emulazione di terminale. V F
10. HTTP consente la connessione tra client e server in modalità passiva e attiva. V F
11. FTP ha due modalità predefinite di accesso: utente e anonima. V F
12. Un proxy HTTP può essere usato per il caching. V F
13. POP3 e IMAP4 gestiscono l'autenticazione tramite username e password. V F
14. Nelle applicazioni stored streaming i file audio/video sono compressi. V F
15. Nelle applicazioni live streaming la comunicazione è on demand. V F
16. SIP è il protocollo usato per le applicazioni di tipo streaming. V F
17. Il protocollo SIP implementa il modello Client-Server. V F

Domande a scelta multipla (una sola è la risposta esatta)

1. **Come fa il livello Application di un server a elaborare le richieste provenienti da più client per uno stesso servizio?**
   - A Disconnette tutte le connessioni al servizio
   - B Rifiuta le connessioni multiple verso lo stesso processo
   - C Sospende una connessione in corso per accettare una nuova connessione
   - D Utilizza le funzioni del livello Transport per distinguere le varie connessioni al servizio

2. **Come si chiama il server che contiene l'elenco dei peer di un'applicazione P2P?**
   - A Login Server
   - B Peer Server
   - C Tracker
   - D P2P Application Server

3. **Il protocollo FTP utilizza le porte:**
   - A 20 e 21
   - B 22 e 23
   - C 24 e 25
   - D 26-27

4. **Telnet è un protocollo che:**
   - A non necessita di stabilire una connessione tra client e server
   - B necessita di stabilire una connessione tra client e server
   - C non è in grado di gestire più sessioni contemporaneamente
   - D non può connettere computer con Sistemi Operativi differenti

5. **Qual è la destination port di una richiesta inviata da un client al web server?**
   - A 81
   - B 80
   - C 8181
   - D 8080

6. **I cookie (due scelte):**
   - A sono file di piccola dimensione
   - B sono usati per tracciare le sessioni HTTP
   - C rispettano il diritto alla privacy
   - D vengono inviati al server su iniziativa del client

7. **Il simbolo @ nell'indirizzo di posta elettronica:**
- A è usato anche per indirizzare i trasferimenti di file
- B viene usato anche per gli indirizzi web
- C separa il nome del dominio del gestore del servizio di posta elettronica (a sinistra) dall'identificativo dell'utente (a destra)
- D separa l'identificativo dell'utente (a sinistra) dal nome del dominio del gestore del servizio di posta elettronica (a destra)

8. **Quale protocollo di posta elettronica si usa quando più utenti accedono alla stessa casella di posta?**
- A SMTP
- B POP3
- C IMAP4
- D Nessuno dei tre

9. **Il livello Application di TCP/IP a quale tra i seguenti gruppi di 3 livelli ISO/OSI corrisponde?**
- A Network, Transport, Application
- B Session, Presentation, Application
- C Physical, Data Link, Application
- D Physical, Session, Application

10. **Quale delle seguenti applicazioni si riferisce ai dati memorizzati su un server e trasmessi on demand?**
- A Interactive Application
- B Multimedia Application
- C Live Streaming Application
- D Stored Streaming Application

11. **Quale dei seguenti servizi non è offerto da un proxy HTTP?**
- A Connettività
- B Monitoraggio
- C Privacy
- D Routing

12. **I server SMTP e POP3 utilizzano rispettivamente le porte:**
- A 25 e 110
- B 20 e 21
- C 80 e 8080
- D qualsiasi porta

13. **Quale delle seguenti applicazioni si riferisce alla diffusione di programmi radio e TV attraverso Internet?**
- A Interactive Application
- B Multimedia Application
- C Live Streaming Application
- D Stored Streaming Application

14. **Quale protocollo gestisce il traffico real time su Internet?**
- A RTP
- B TCP
- C UDP
- D HTTP

15. **Qual è il protocollo che controlla il flusso dei dati e la qualità della trasmissione?**
- A RTP
- B RTCP
- C SIP

Ascolta le **risposte**

PREPARATI PER IL COLLOQUIO ORALE

1. LEZIONE 1 **Descrivi uno scenario di interconnessione di reti tramite Internet.**

2. LEZIONE 2 **Spiega il ruolo del livello Application nello stack TCP/IP.**

3. LEZIONE 2 **Descrivi la generica architettura di un'applicazione P2P.**

4. LEZIONE 3 **Descrivi l'utilità dell'applicazione Telnet per l'amministratore di rete.**

5. LEZIONE 4 **A quale scopo è stato creato il protocollo FTP?**

6. LEZIONE 4 **Descrivi la modalità di lavoro FTP di tipo active mode.**

7. LEZIONE 4 **Descrivi la modalità di lavoro FTP di tipo passive mode.**

8. LEZIONE 5 **Descrivi il protocollo usato per la navigazione sul WWW.**

9. LEZIONE 5 **L'acquisizione di una risorsa da parte del web client può essere schematizzata in 4 fasi. Quali?**

10. LEZIONE 5 **Che cosa sono i proxy distorcenti?**

11. LEZIONE 6 **Descrivi i protocolli definiti per il trasferimento elettronico dei messaggi.**

12. LEZIONE 6 **Descrivi le principali differenze tra POP3 e IMAP4.**

13. LEZIONE 7 **Quali tipologie di traffico generano le applicazioni multimediali?**

14. LEZIONE 8 **Descrivi l'architettura di un sistema PBX-IP.**

15. LEZIONE 8 **Descrivi le componenti client e server del protocollo SIP.**

IN ENGLISH, PLEASE

Ascolta la **pronuncia del testo** e verifica la **traduzione**

ABSTRACT

The Application Layer of TCP/IP

The Application Layer is the top layer in the TCP/IP stack and includes the three highest layers of the OSI model.
At this layer several protocols run applications for end users: Telnet as terminal emulator, FTP for file transfer, HTTP for web browsing, SMTP for email, VoIP for telephony.
Application Layer protocols manage host-to-host data transfer in a Client-Server networking model, but also in a Peer-to-Peer model.
In a P2P application your computer can act as both a client and a server within a single communication session.
New RFC standard have been defined to satisfy the traffic requirements of multimedia applications: RTSP, RTP and SIP.
SIP is a signaling protocol specified for Voice over IP.
The VoIP technology merges data services and voice, reducing the cost of communications.
In VoIP the analog voice is digitized and transmitted over the Internet network in the form of IP packets.

EXERCISES

Use the appropriate number to match words and meanings.

...	Envelope	1	It interprets and displays a web document
...	Anonymous	2	The third address field on a message, unknown to the main recipient
...	Browser	3	Type of FTP that does not require a password
...	URL	4	It consists of the sender's address and the recipient's address
...	Bcc	5	Protocol used to retrieve emails from a mail server
...	Authentication	6	It is a standard for specifying any kind of resource in Internet
...	POP	7	Making some resources available to other users
...	Sharing	8	Verifying a user's identity

GLOSSARY

Application Layer: the TCP/IP layer that implements the functions performed by users to accomplish various tasks over the network. Since they are at the top of the stack, application protocols are the only ones that do not provide services to a higher layer; they make use of services provided by the layers below.

MIME (Multipurpose Internet Mail Extension): a standard that extends the format of email for sending various kinds of information (for example: images, sounds, movies and computer programs).

PBX (Private Branch eXchange): a phone switch located at the customer's premise.

Recipient: an object that can receive email messages.

Softphone: a VoIP telephone in software, it can be installed on a computer and works as a VoIP phone.

Stateful protocol: a protocol which requires the server to retain session information or status about each communications partner for the duration of multiple requests. FTP is an example of stateful protocol.

Stateless protocol: a protocol that treats each request as an independent transaction that is unrelated to any previous request so that the communication consists of independent pairs of request and response. IP and HTTP are examples of stateless protocols.

Streaming: the audio/video content is sent in compressed form over the Internet and the user can listen (or watch) a file after downloading has started.

VoIP phone: a hard telephone with an Ethernet port through which it communicates with a VoIP server, VoIP gateway or another VoIP phone.

COMPETENZE IN GIOCO

Competenze disciplinari

- Saper configurare i servizi principali per le reti LAN.
- Saper verificare il funzionamento dei servizi configurati.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Sapere usare un simulatore di reti.
- Esporre i risultati della ricerca alla classe.

tempi

- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Simulatore di reti Packet Tracer.
- Dispositivo connesso a Internet.
- Carta e penna.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

Partendo dalla rete costituita da 3 dipartimenti (AMMINISTRAZIONE, PRODUZIONE e MARKETING) dell'Unità 7, aggiungere i server di posta e configurare 3 client per 3 nuovi dipendenti: Giovanni Rossi dell'amministrazione, Antonio Verdi della produzione e Francesco Blu del marketing. Verificare poi il funzionamento del servizio inviando una email tra due qualsiasi dei nuovi dipendenti.

File sorgenti
Scarica il file

SVOLGIMENTO

Riprendiamo lo scenario (FIGURA 1) del "Lavorare per competenze" dell'Unità 7, a cui andremo ad aggiungere:

1. il servizio SMTP-POP3 in ogni server di dipartimento;
2. i client di posta dei nuovi dipendenti su un PC del dipartimento che lo ha assunto;
3. i record di tipo CNAME sul server DNS dello scenario.

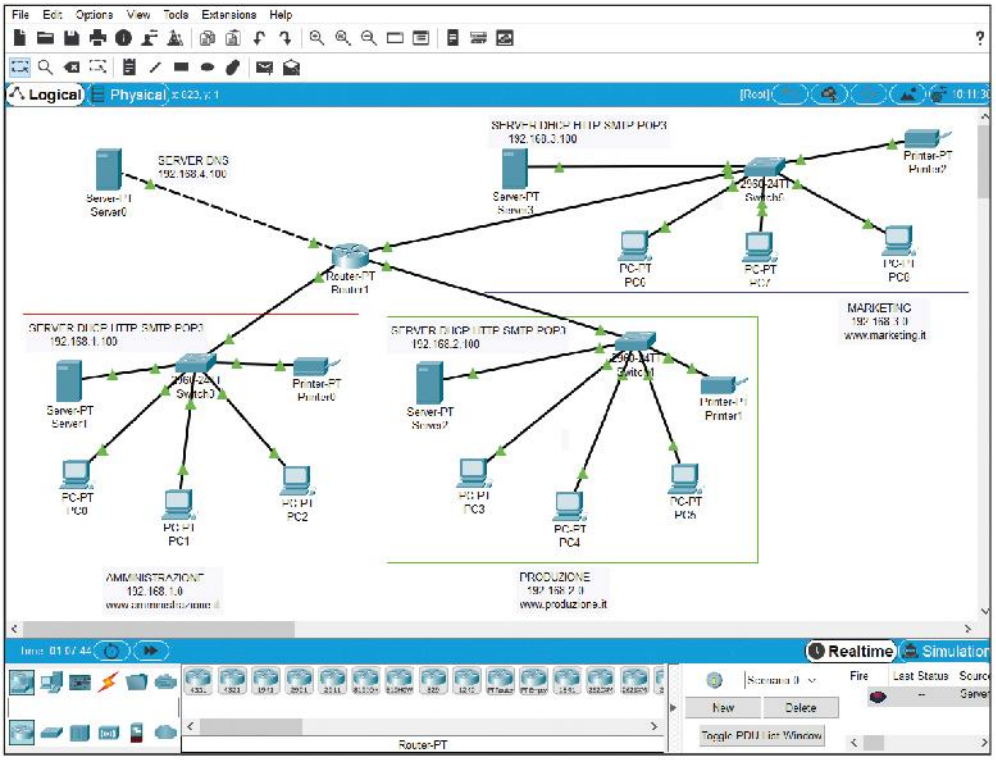
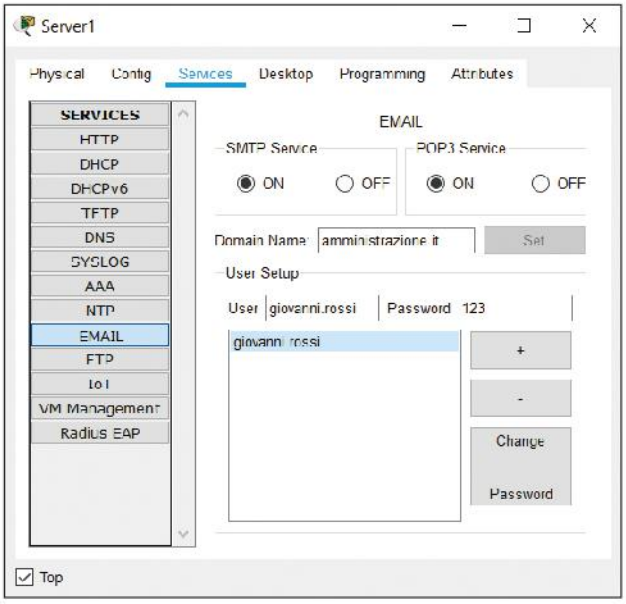
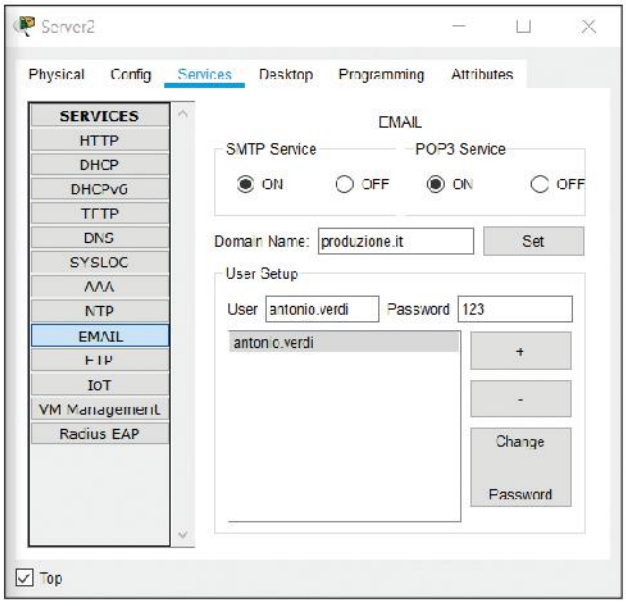


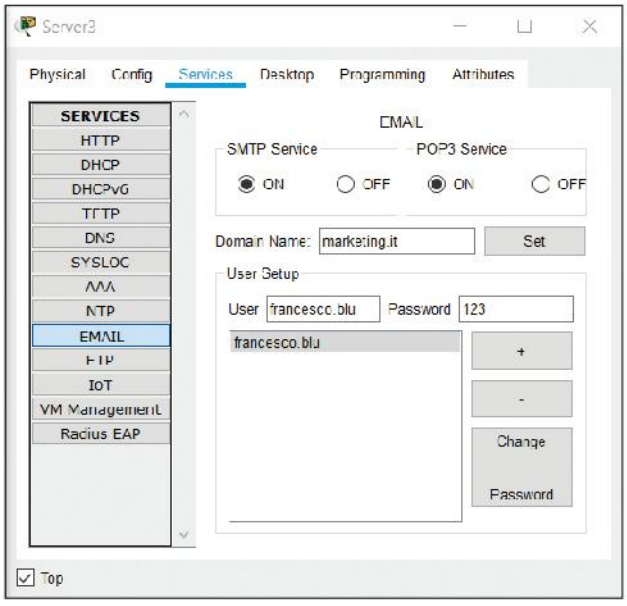
FIGURA 1 Scenario della rete aziendale con 3 dipartimenti

1. Cominciamo col configurare il servizio di posta sul server dell'AMMINISTRAZIONE. Clicchiamo su Server1, selezioniamo la scheda Services e il servizio EMAIL in cui creiamo lo User **giovanni.rossi** sotto il dominio **amministrazione.it**. Compilati tutti i campi, cliccare sul **+** per aggiungere il client creato. Identica operazione va ripetuta su Server2 per **antonio.verdi** sotto il dominio **produzione.it** e su Server3 per **francesco.blu** sotto il dominio **marketing.it** (FIGURA 2).

FIGURA 2 Configurazione del servizio EMAIL su Server1, Server2 e Server3

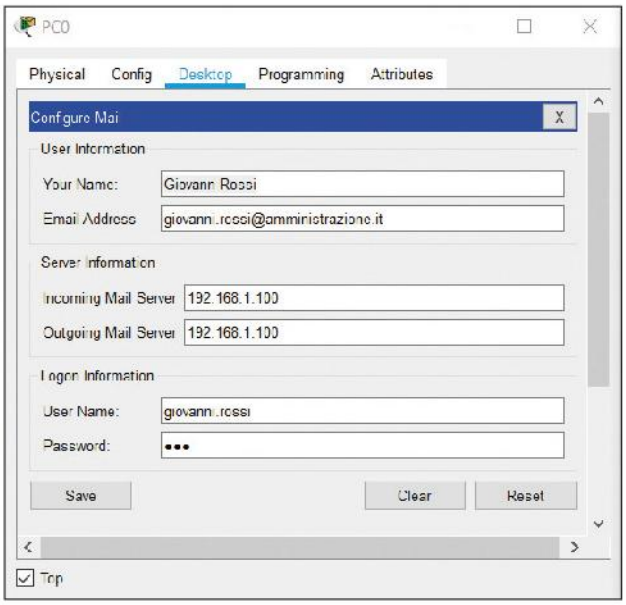




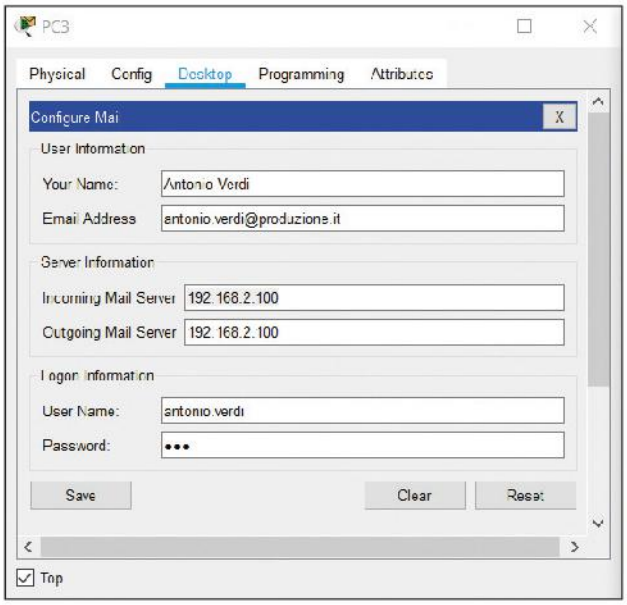


2. Configuriamo ora i client di posta dei nuovi dipendenti. Clicchiamo per esempio su **PC0** per il dipartimento AMMINISTRAZIONE, dalla scheda Desktop selezioniamo Email e quindi Configure Mail. Compiliamo tutti i campi specificando l'indirizzo IP del server di dipartimento nell'Incoming Mail Server e nell'Outgoing Mail Server. Identica operazione va ripetuta per esempio su **PC3** del dipartimento PRODUZIONE e su **PC6** del dipartimento MARKETING. La FIGURA 3 mostra la configurazione dei 3 PC. Cliccare su **Save** al termine di ogni configurazione effettuata.

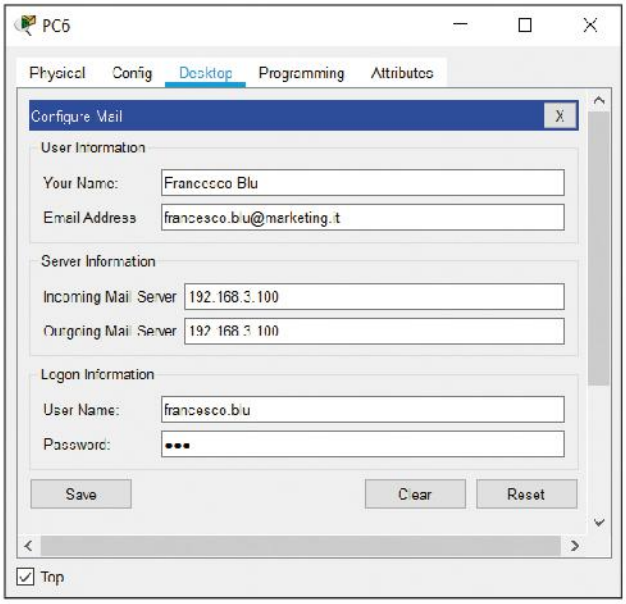
FIGURA 3 Configurazione dell'Email Address su PC0, PC3 e PC6



Giovanni Rossi lavora nel dipartimento AMMINISTRAZIONE (LAN 192.168.1.0) e ha configurato l'account di posta su PC0.



Antonio Verdi lavora nel dipartimento PRODUZIONE (LAN 192.168.2.0) e ha configurato l'account di posta su PC3.



Francesco Blu lavora nel dipartimento MARKETING (LAN 192.168.3.0) e ha configurato l'account di posta su PC6.

3. L'ultimo passo consiste nel dichiarare i 3 alias attraverso i record CNAME sul server DNS **(FIGURA 4)**. Cliccare su **Add** al termine di ogni entry compilata.

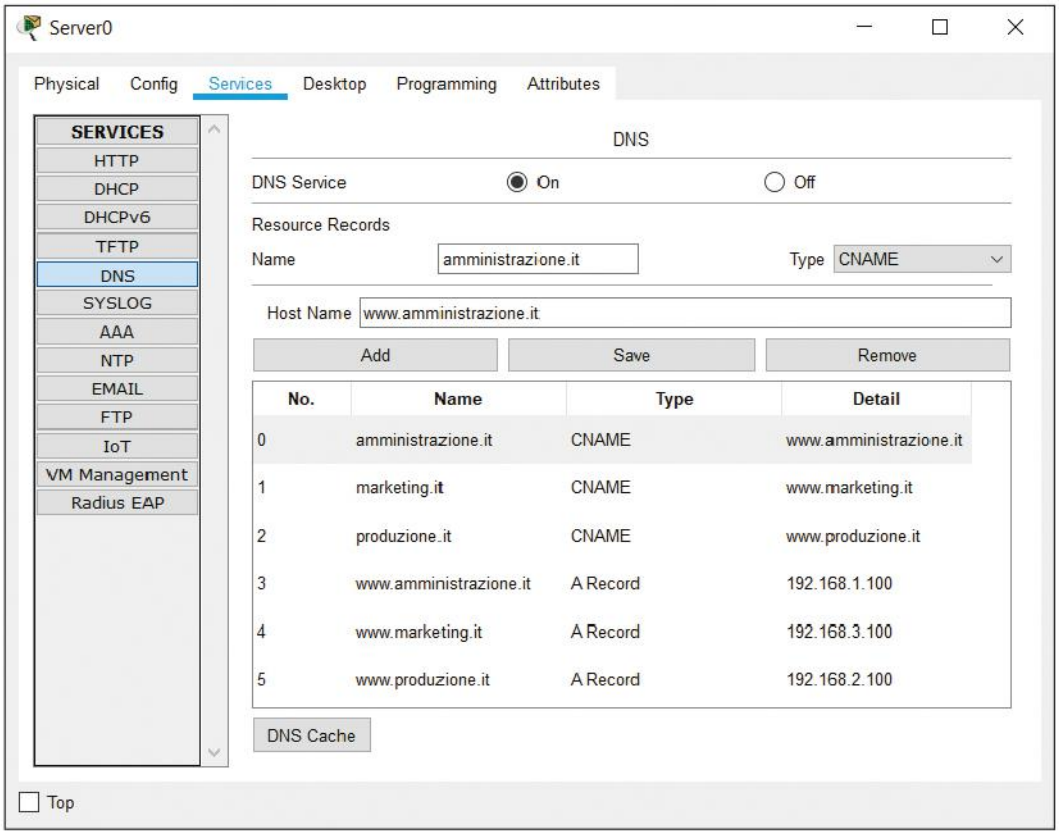


FIGURA 4 Configurazione degli alias su server DNS (Server0)

Possiamo ora verificare il funzionamento del servizio di posta simulando l'invio di una email da Giovanni Rossi ad Antonio Verdi **(FIGURA 5)**.

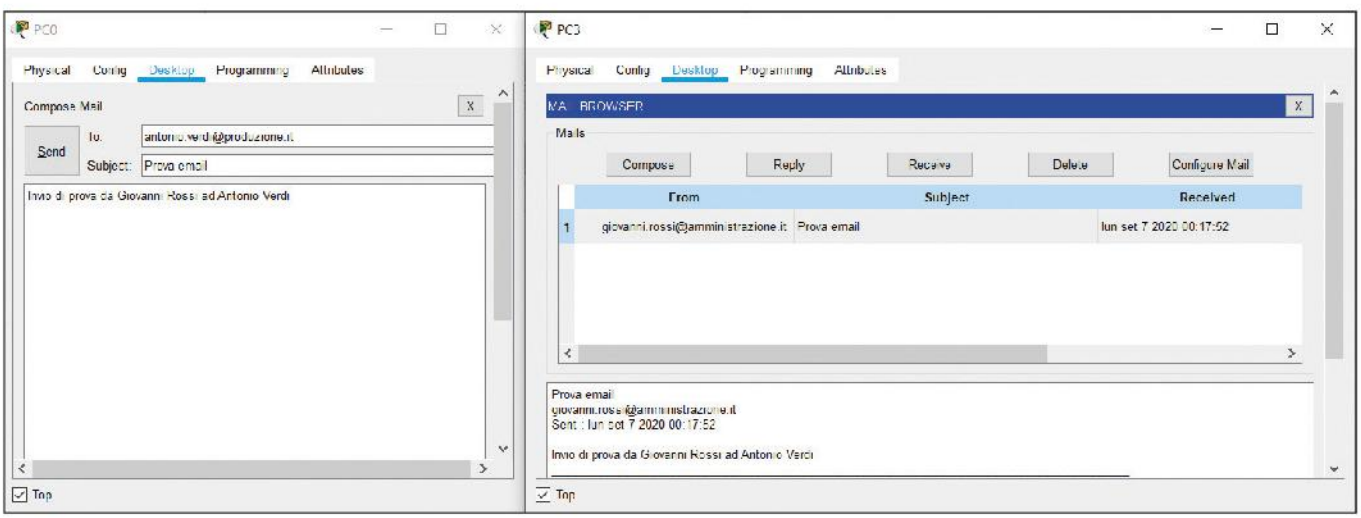


FIGURA 5 Invio e ricezione di una email da Giovanni Rossi ad Antonio Verdi

A CASA

- Ipotizza una tua soluzione al tema proposto.
- Leggi la proposta di SVOLGIMENTO per verificare se le tue ipotesi si adattano al caso preso in esame e se la trattazione proposta risulta completa.
- Verifica il funzionamento del servizio mandando email tra diversi client di posta.
- Raccogli i tuoi risultati in una presentazione (massimo 5 slide).

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i casi che sono stati presentati.
- Stabilite quale caso rappresenta l'esempio migliore per completezza e che meglio si adatta alla soluzione del tema proposto.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Sono riuscito a configurare il servizio richiesto?	Sono riuscito a configurare la rete ma non il servizio richiesto. ☐	Sono riuscito a configurare il servizio richiesto solo su alcuni client. ☐	Con la guida del docente e la collaborazione dei compagni ho configurato il servizio richiesto sui client. ☐	Ho configurato il servizio richiesto autonomamente. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

UNITÀ

9

ARDUINO E RASPBERRY Pi PER LE RETI

Guarda
la **presentazione**
dell'unità

IN QUESTA UNITÀ

conoscenze	abilità	competenze
Conoscere le caratteristiche delle schede di rete. Conoscere le prestazioni delle varie schede.	Saper scegliere le schede di rete più adatte. Saper configurare le schede in base alle specifiche del progetto. Saper configurare le schede in base ai dispositivi esterni.	Configurare le schede di rete per Arduino in base alle specifiche richieste. Configurare una rete con la scheda Raspberry Pi. Valutare le prestazioni e la funzionalità della rete realizzata.

DDI+
Didattica Digitale Integrata Plus

FLIPPED CLASSROOM

A casa

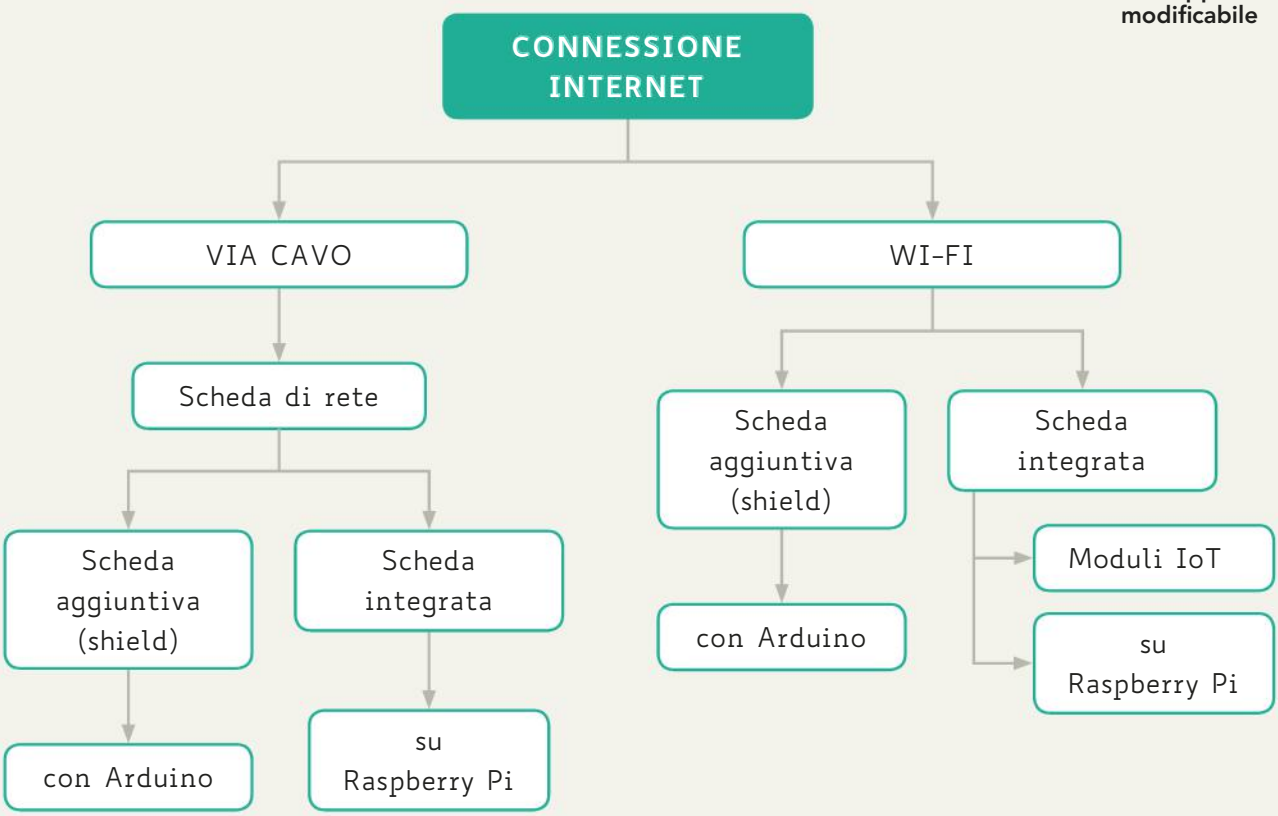
- Leggi la Lezione 1 e rivedi quanto studiato sui parametri di rete;
- valuta la configurazione necessaria per leggere un parametro fisico e trasmetterlo a distanza.

In classe

- Confrontate le soluzioni tecniche trovate;
- discutete quali possono essere le soluzioni tecniche migliori;
- valutate i limiti delle soluzioni tecniche scartate.

MAPPA CONCETTUALE

Mappa modificabile



1 ARDUINO E LA RETE

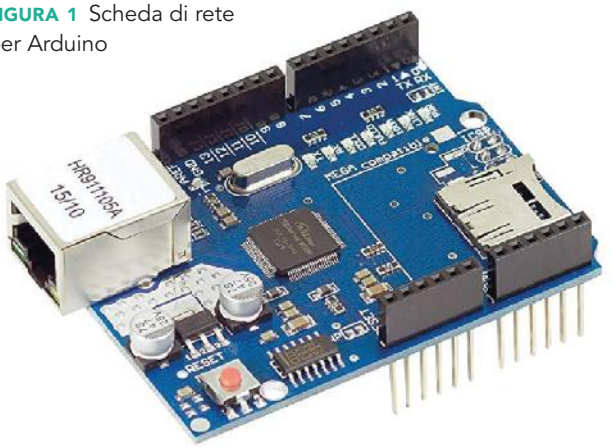
1.1 Introduzione

#techwords

Il nome **Arduino** deriva dal nome del locale di Ivrea in cui erano soliti riunirsi gli ideatori di Arduino. Arduino è anche il nome di un antico re di Ivrea, nominato re d'Italia nel 1002.

In questa Lezione ci occuperemo delle caratteristiche principali della **scheda di rete** (Ethernet shield) per #Arduino UNO R3 (FIGURA 1). Questa scheda nasce dall'esigenza di collegare dispositivi come sensori e attuatori alla rete Internet in modo da ricevere informazioni (parametri ambientali, stato di funzionamento di macchinari, ecc.) da una postazione distante dal luogo in cui si trova la scheda Arduino e di inviare dei comandi a dispositivi (accensione e spegnimento di apparecchiature, invio di comandi, ecc.) che per loro natura non hanno la possibilità di essere collegati a Internet o, a causa del loro posizionamento, non sono facilmente accessibili. L'utilizzo di una scheda con connessione via cavo si utilizza in quegli ambienti in cui la trasmissione wireless è difficoltosa, impossibile o proibita per motivi fisici o di sicurezza.

FIGURA 1 Scheda di rete per Arduino



Caratteristiche della scheda

Alimentazione: dalla scheda Arduino UNO R3
Controller Internet: W5100 con buffer 16 kB
Velocità 10/100 Mb/s
Comunicazione: SPI con scheda Arduino
Slot per scheda SD (capacità massima 4 MB)
Connettore RJ45
Pulsante di reset
Collegamento alla scheda Arduino compatibile pin to pin

LED

PWR: la scheda ARDUINO e la scheda di rete sono alimentate
LINK: presenza di un collegamento di rete (lampeggiante quando la scheda trasmette o riceve dati)
FULLD: connessione di rete full-duplex
100M: connessione di rete 100 Mb/s (invece di 10 Mb/s)
RX: lampeggiante quando la scheda riceve i dati
TX: lampeggiante quando la scheda invia i dati
COLL: lampeggiante quando vengono rilevate le collisioni di rete

Configurazione

L'indirizzo IP viene deciso dal programmatore in fase di configurazione della rete. Per il MAC address possono presentarsi due situazioni in base al produttore della scheda.

- Caso 1: il MAC address è stampato sulla scheda stessa. In tal caso non è modificabile e deve essere riportato nei listati, nel caso sia necessario.
  Nel caso non fosse leggibile, è possibile ricevere il dato con un semplice sketch che interroga la scheda di rete (vedi esempi nel Paragrafo 1.2).
- Caso 2: il MAC address non è stampigliato. In tal caso il programmatore può configurarlo liberamente facendo attenzione all'eventuale presenza di un altro dispositivo con lo stesso MAC address nella stessa rete locale.

Le schede di rete senza MAC address preconfigurato sono in genere meno recenti.

Lo slot per la scheda SD consente di salvare nella scheda SD alcuni parametri di configurazione della scheda di rete e dei dati da inviare successivamente al server a cui la scheda è collegata.

La gestione della scheda, per la complessità delle funzioni, è realizzata tramite la libreria Ethernet.h. In tutti i listati è presente sempre la libreria SPI.h, che gestisce la comunicazione Serial Peripheral Interface tra la scheda di rete e la scheda Arduino.

Alcuni modelli di schede Arduino più recenti possiedono delle schede di rete incorporate; però, tranne un modello (Arduino MKR ETH Shield), prevedono solo la connessione Wi-Fi.

1.2 Laboratorio Arduino per la rete

Qui di seguito sono riportati e commentati (oltre ai commenti originali) alcuni esempi prelevabili e utilizzabili direttamente dal programma *arduino.exe*; tali programmi, inoltre, sono disponibili sul sito *www.arduino.cc*. Nell'ultimo esempio viene utilizzata la modalità connected, utilizzando Processing.

Interrogazione di un web server

esempio

Materiale necessario:
– scheda Arduino;
– cavo USB.

File sorgenti
Scarica il file

```
/*
 Web client
 This sketch connects to a website (http://www.google.com)
 using an Arduino Ethernet shield.
 Circuit:
 * Ethernet shield attached to pins 10, 11, 12, 13
 created 18 Dec 2009
 by David A. Mellis
 modified 9 Apr 2012
 by Tom Igoe, based on work by Adrian McEwen
 */
#include <SPI.h>
#include <Ethernet.h>
// Enter a MAC address for your controller below.
// Newer Ethernet shields have a MAC address printed on a sticker on the shield
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
// if you don't want to use DNS (and reduce your sketch size)
// use the numeric IP instead of the name for the server:
//IPAddress server(74,125,232,128);  // numeric IP for Google (no DNS)
char server[] = "www.google.com";    // name address for Google (using DNS)
// Set the static IP address to use if the DHCP fails to assign
IPAddress ip(192,168,0,177);
// Initialize the Ethernet client library
// with the IP address and port of the server
```

Sezione commenti
Scopo del programma è di effettuare una richiesta HTTP al server di Google (google.com). Il risultato della richiesta viene mostrato nel seriale di Arduino.

Chiamata alle librerie Ethernet.h SPI.h.

Configurazione MAC address.

Inserimento dell'IP del sito a cui collegarsi nel caso ci siano problemi con il nome del sito.

Inserimento del DHCP statico (192,168,0,177 è solo un esempio).

→

Inizializzazione della scheda - verifica del MAC address ed eventuale configurazione.

```
// that you want to connect to (port 80 is default for HTTP):
EthernetClient client;
void setup()
{
 // Open serial communications and wait for port to open:
  Serial.begin(9600);
   while (!Serial)
     {
        ; // wait for serial port to connect. Needed for Leonardo only
     }
  // start the Ethernet connection:
  if (Ethernet.begin(mac) == 0)
     {
     Serial.println("Failed to configure Ethernet using DHCP");
     // no point in carrying on, so do nothing forevermore:
     // try to congifure using IP address instead of DHCP:
     Ethernet.begin(mac, ip);
     }
  // give the Ethernet shield a second to initialize:
  delay(1000);
Serial.println("connecting...");
```

Verifica della connessione sulla porta 80.

```
  // if you get a connection, report back via serial:
  if (client.connect(server, 80))
     {
     Serial.println("connected");
```

Avvio della ricerca sul sito google.com.

```
     // Make a HTTP request:
     client.println("GET /search?q=arduino HTTP/1.1");
     client.println("Host: www.google.com");
     client.println("Connection: close");
     client.println();
  }
  else
  {
    // if you didn't get a connection to the server:
    Serial.println("connection failed");
  }
}
```

Sezione LOOP

```
void loop()
{
```

Se il collegamento è avviato, nella variabile c viene caricato il testo ricevuto come risposta della ricerca e viene visualizzato sulla finestra del monitor seriale.

```
  // if there are incoming bytes available
  // from the server, read them and print them:
  if (client.available())
   {
      char c = client.read();
      Serial.print(c);
    }
  // if the server's disconnected, stop the client:
```

```
  if (!client.connected())
   {
     Serial.println();
     Serial.println("disconnecting.");
     client.stop();
     // do nothing forevermore:
     while(true);
    }
}
```

Non appena viene rilevata la disconnessione viene visualizzato il messaggio di disconnessione in corso, si ferma la ricezione e il loop non esegue più nulla.

Ottenere l'indirizzo IP tramite DHCP

esempio

Materiale necessario:
– scheda Arduino;
– cavo USB;
– scheda di rete per Arduino.

File sorgenti
Scarica il file

```
/*
 DHCP-based IP printer
 This sketch uses the DHCP extensions to the Ethernet library
 to get an IP address via DHCP and print the address obtained.
 using an Arduino Wiznet Ethernet shield.
 Circuit:
 * Ethernet shield attached to pins 10, 11, 12, 13
 created 12 April 2011
 modified 9 Apr 2012
 by Tom Igoe
 */
#include <SPI.h>
#include <Ethernet.h>
// Enter a MAC address for your controller below.
// Newer Ethernet shields have a MAC address printed on a sticker on the shield
byte mac[] = { 0x00, 0xAA, 0xBB, 0xCC, 0xDE, 0x02 };
// Initialize the Ethernet client library
// with the IP address and port of the server
// that you want to connect to (port 80 is default for HTTP):
EthernetClient client;

void setup()
{
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  // this check is only needed on the Leonardo:
   while (!Serial)
```

Dichiarazione delle librerie.

Configurazione MAC address.

Sezione Setup
- inizializzazione della connessione seriale;
- verifica del MAC address ed eventuale inserimento.

→

```
  {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
 // start the Ethernet connection:
 if (Ethernet.begin(mac) == 0)
  {
    Serial.println("Failed to configure Ethernet using DHCP");
    // no point in carrying on, so do nothing forevermore:
    for(;;);
  }
 // print your local IP address:
  Serial.print("My IP address: ");
  for (byte thisByte = 0; thisByte < 4; thisByte++)
  {
  // print the value of each byte of the IP address:
   Serial.print(Ethernet.localIP()[thisByte], DEC);
   Serial.print(".");
  }
 Serial.println();
}

void loop()
{
}
```

Effettua la lettura dei campi dell'IP convertendoli in decimale e visualizzando i dati sul monitor seriale.

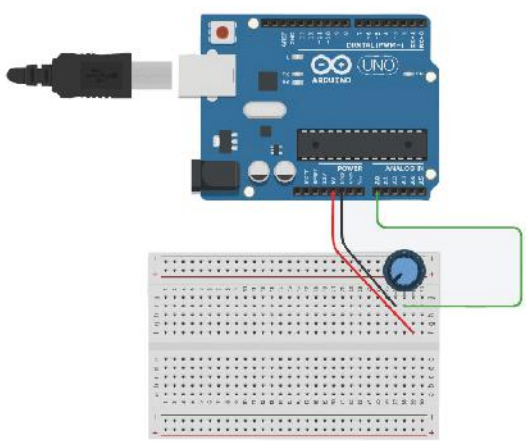
La sezione Loop non deve effettuare nulla.

Lettura del valore dei PIN analogici in una pagina web

esempio

File sorgenti
Scarica il file

Materiale necessario:
– scheda Arduino;
– cavo USB;
– scheda di rete per Arduino;
– potenziometro da almeno 1 kohm;
– fili rigidi;
– basetta millefori.



```
/*
 Web server
 A simple web server that shows the value of the analog input pins A0
 using an Arduino Ethernet shield.
 Circuit:
 * Ethernet shield attached to pins 10, 11, 12, 13
 * Analog input attached to pins A0
 * Other analog pins can be connected  to ground
 created 18 Dec 2009
 by David A. Mellis
 modified 9 Apr 2012
 by Tom Igoe
 */

#include <SPI.h>
#include <Ethernet.h>
// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network:
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192,168,1,177);
// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
EthernetServer server(80);

void setup()
{
 // Open serial communications and wait for port to open:
  Serial.begin(9600);
   while (!Serial)
   {
    ; // wait for serial port to connect. Needed for Leonardo only
   }
  // start the Ethernet connection and the server:
  Ethernet.begin(mac, ip);
  server.begin();
  Serial.print("server is at ");
  Serial.println(Ethernet.localIP());

  // start the Ethernet connection and the server:
  Ethernet.begin(mac, ip);
  server.begin();
  Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
}
 void loop()
 {
   // listen for incoming clients
   EthernetClient client = server.available();
   if (client)
      {
        Serial.println("new client");  // an http request ends with a blank line
```

Dichiarazione delle librerie.

Sezione SETUP
Configurazione e verifica della connessione.

Sezione LOOP
Inizia verificando che il server sia disponibile. La richiesta termina con una linea vuota (blank line).

Se il server ha risposto, comincia la trasmissione del codice HTML della pagina da visualizzare.

Aggiornamento della pagina ogni 5 secondi.

I 6 ingressi analogici (da A0 ad A5) vengono letti e il valore è caricato nella variabile sensorReading che viene poi inviata al server.

Chiusura del codice della pagina HTML.

```
        boolean currentLineIsBlank = true;
        while (client.connected())
          {
           if (client.available())
            {
            char c = client.read();
            Serial.write(c);
            // if you've gotten to the end of the line (received a newline
            // character) and the line is blank, the http request has ended,
            // so you can send a reply
            if (c == '\n' && currentLineIsBlank)
              {
              // send a standard http response header
              client.println("HTTP/1.1 200 OK");
              client.println("Content-Type: text/html");
              client.println("Connection: close");
              // the connection will be closed after completion of the response
              client.println("Refresh: 5");  // refresh the page automatically every 5 sec
              client.println();
              client.println("<!DOCTYPE HTML>");
              client.println("<html>");
              // output the value of each analog input pin
              for (int analogChannel = 0; analogChannel < 6; analogChannel++)
                {
                  int sensorReading = analogRead(analogChannel);
                  client.print("analog input ");
                  client.print(analogChannel);
                  client.print(" is ");
                  client.print(sensorReading);
                  client.println("<br />");
                  }
              client.println("</html>");
              break;
            }
            if (c == '\n')
            {
            // you're starting a new line
            currentLineIsBlank = true;
            }
           else if (c != '\r')
                  {
                    // you've gotten a character on the current line
                   currentLineIsBlank = false;
                  }
          }
       }
      // give the web browser time to receive the data
       delay(1);
      // close the connection:
      client.stop();
      Serial.println("client disconnected");
    }
}
```

Codice HTML all'interno del web server

esempio

Materiale necessario:
– scheda Arduino;
– cavo USB;
– scheda di rete per Arduino.

File sorgenti
Scarica il file

```
/*
Questo programma mostra come sia possibile inserire del codice HTML all'interno della
pagina web del web server di Arduino*/
#include <SPI.h>
#include <Ethernet.h>
// Mac address di Arduino
byte mac[] = {  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
// Viene inizializzata la libreria Ethernet di Arduino e il web server gira sulla porta 80
EthernetServer server(80);
void setup()
 {
  Serial.begin(9600);
  // Viene inilizzato il web server e la connessione di rete
  Ethernet.begin(mac);
  server.begin();
  Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
}

void loop()
 {
  // Vengono ascoltati nuovi client
  EthernetClient client = server.available();
  if (client)
   {
    Serial.println("new client");
    // Finisce una richiesta HTTP
    boolean currentLineIsBlank = true;
    while (client.connected())
          {
           if (client.available())
            {
             char c = client.read();
             Serial.write(c);
            // Se viene completato l'invio della richiesta HTTP, allora il server
            // invia la risposta
            if (c == '\n' && currentLineIsBlank)
              {
                // Viene creata una risposta HTTP, in pratica
                // viene creata una pagina web in HTML
                client.println("HTTP/1.1 200 OK");
                client.println("Content-Type: text/html");
                client.println("Connection: close");
                // Dopo la risposta
```

→

```
                // la connessione si interrompe
                // ogni 5 secondi
                client.println("Refresh: 5");
                // in automatico si aggiorna la pagina web
                client.println();

                client.println("<meta charset=UTF-8>");
                // serve per inserire i caretteri speciali
                client.println("<!DOCTYPE HTML>");
                client.println("<html>");
                client.println("<head> <TITLE>Arduino</TITLE> </head>");
                // Viene creato il titolo
                client.println("<body> <h1> Benvenuto nel Webserver Arduino </h1>");
                // Viene inserito del testo
                client.println("<h3> In questa pagina è possibile inserire il codice HTML che vuoi </h3>");
                // Viene inserita un'immagine, presente in un determinato server
                client.println("<img src = \"http://miosito.com/immagine_mia.jpg\"alt = \"Arduino\"");
                client.println("</body>");
                client.println("</html>");
                break;
              }
            if (c == '\n')
             {
              currentLineIsBlank = true;
             }
            else if (c != '\r')
                   {
                    currentLineIsBlank = false;
                   }
          }
        }
        delay(1);
        // Viene chiusta la connessione
        client.stop();
        Serial.println("client disconnected");
      }
}
```

Chiusura pagina HTML.

Ricevere pacchetti UDP con Arduino

esempio

File sorgenti
Scarica il file

Materiale necessario:
– scheda Arduino;
– cavo USB;
– scheda di rete per Arduino.

Software: Processing.

Per generare il codice UDP occorre utilizzare uno sketch in modalità connected (vedi Volume 1, Unità 9, Lezione 1).

```
/*
This sketch receives UDP message strings, prints them to the serial port
and sends an “acknowledge” string back to the sender.
A Processing sketch is included at the end of file that can be used to send
and received messages for testing with a computer.
created 21 Aug 2010
by Michael Margolis
This code is in the public domain.
*/
#include <SPI.h>
#include <Ethernet.h>
#include <EthernetUdp.h>
// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network:
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
IPAddress ip(192, 168, 1, 177);
unsigned int localPort = 8888; // local port to listen on
// buffers for receiving and sending data
char packetBuffer[UDP_TX_PACKET_MAX_SIZE]; //buffer to hold incoming packet,
char ReplyBuffer[] = “acknowledged”; // a string to send back
// An EthernetUDP instance to let us send and receive packets over UDP
EthernetUDP Udp;

void setup() {
// start the Ethernet and UDP:
Ethernet.begin(mac,ip);
Udp.begin(localPort);
Serial.begin(9600);
}

void loop()
{
 // if there’s data available, read a packet
 int packetSize = Udp.parsePacket();
 if(packetSize)
  {
  Serial.print(“Received packet of size “);
  Serial.println(packetSize);
  Serial.print(“From “);
  IPAddress remote = Udp.remoteIP();
  for (int i =0; i < 4; i++)
  {
  Serial.print(remote[i], DEC);
  if (i < 3)   Serial.print(“.”);
  } //for
  Serial.print(“, port “);
  Serial.println(Udp.remotePort());
  // read the packet into packetBufffer
  Udp.read(packetBuffer,UDP_TX_PACKET_MAX_SIZE);
  Serial.println(“Contents:”);
  Serial.println(packetBuffer);
  // send a reply, to the IP address and port that sent us the packet we received
```

Libreria UDP.

Viene impostata la porta di comunicazione.

Crea buffer per stringa UDP.

Viene inizializzata la porta di comunicazione.

Ricezione della dimensione della stringa UDP.

Visualizza la porta di comunicazione.

```
  Udp.beginPacket(Udp.remoteIP(), Udp.remotePort());
  Udp.write(ReplyBuffer);
  Udp.endPacket();
  }
 delay(10);
}
```

Programma da avviare utilizzando l'applicazione Processing

```
// Processing sketch to run with this example
// Processing UDP example to send and receive string data from Arduino
// press any key to send the "Hello Arduino" message

import hypermedia.net.*;
UDP udp;                              // define the UDP object
void setup()
 {
  udp = new UDP( this, 6000 ); // create a new datagram connection on port 6000
  udp.log( true );                    // <-- printout the connection activity
  udp.listen( true );                 // and wait for incoming message
 }

void draw() { }
```

Funzione che invia il messaggio quando viene premuto un tasto qualunque della tastiera.

```
void keyPressed()
 {
  String ip = "192.168.1.177";          // the remote IP address
  int port = 8888;                      // the destination port
  udp.send("Hello World", ip, port ); // the message to send
 }
```

Funzione che riceve i messaggi dalla scheda Arduino.

```
void receive( byte[] data )   // <-- default handler
 {
    //void receive( byte[] data, String ip, int port )  // <-- extended handler
    for (int i=0; i < data.length; i++)
    {
    print(char(data[i]));
    println();
    }
}
```

FISSA LE CONOSCENZE

- Quale funzione svolge la Ethernet shield di Arduino?
- Qual è la velocità massima di connessione?
- Che funzione svolge la scheda di memoria SD?
- Come è possibile conoscere il MAC address della scheda?
- Come si capisce che la scheda di rete è connessa alla rete Internet?
- Quali modelli di Arduino prevedono la connessione via cavo?
- È possibile leggere dati a distanza utilizzando Arduino e la scheda di rete?

2 RASPBERRY Pi E LA RETE

2.1 Introduzione

La scheda Raspberry Pi (FIGURA 2), come visto nell'Unità 9 del Volume 1, ha il vantaggio di essere una scheda a microprocessore facilmente configurabile. In particolare può essere configurata come server per schede Arduino che hanno comunque dei limiti di funzionamento dovuti alle caratteristiche limitate dei microcontrollori utilizzati.
Per l'installazione del sistema operativo si fa riferimento a quanto indicato nel Volume 1, Unità 9, Lezione 2.

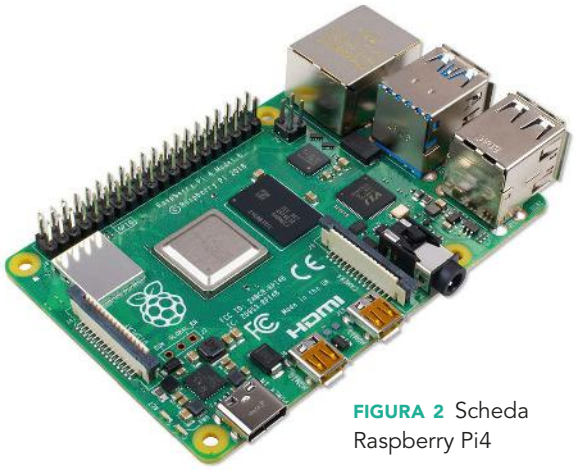


FIGURA 2 Scheda Raspberry Pi4

Web server con NGINX e PHP

esempio

Software necessario:
#NGINX
PHP

Installazione
Per installare NGINX nella versione open source, si può utilizzare il pacchetto disponibile sul repository di Raspbian.

- Aggiornare il packet manager con lo stato del repository

```
sudo apt-get update
```

- Scaricare e installare NGINX

```
sudo apt-get install nginx
```

- Verificare l'installazione e leggere la versione

```
sudo nginx -v
```

La cartella di default dove sono ubicate le pagine html di NGINX è */var/www/html*

- Testare il web server
  a) da un PC collegato in rete locale, aprire il browser;
  b) nella barra di navigazione digitare l'indirizzo IP del Raspberry Pi;
  c) si apre la pagina:

#techwords

NGINX (pronuncia engine x) consente di realizzare un web server ad alte prestazioni progettato per casi d'uso con traffico elevato, ma con risorse ridotte.

2.2 Installare PHP di programmazione

PHP (Hypertext Preprocessor) è un linguaggio di scripting molto utilizzato per la programmazione di pagine dinamiche e lo sviluppo di applicazioni web interpretate lato server.

Installazione

NGINX richiede PHP in implementazione FPM (Fast-CGI Process Manager).

- Scaricare e installare PHP (nell'esempio si fa riferimento alla versione 7.3):

```
sudo apt-get install php7.3-fpm
```

- Editare il file di configurazione di nginx per abilitare le pagine php

```
cd /etc/nginx
sudo nano sites-enabled/default
```

- Trovare la linea *index*.
  Aggiungere *index.php* come prima voce

```
index index.php index.html index.htm;
```

  Scorrere il file fino alla sezione relativa al PHP, togliere il simbolo di commento # in modo che risulti come di seguito indicato (nell'esempio si fa riferimento alla versione 7.3)

```
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/run/php/php7.3-fpm.sock;
}
```

- Salvare con Ctrl+O
- Chiudere il file con Ctrl+X
- Ricaricare il web server

```
sudo /etc/init.d/nginx reload
```

Testare PHP

- Creare un file di prova PHP

```
cd /var/www/html/
sudo nano index.php
```

- Salvare con Ctrl+O
- Chiudere il file con Ctrl+X
- Da un PC collegato in rete locale:
  – aprire il browser
  – nella barra di navigazione digitare l'indirizzo IP del Raspberry

Si apre la pagina di test di PHP con le informazioni sulla versione installata e la configurazione (FIGURA 3).

PHP Version 7.3.3-1

System	Linux deb10 4.19.0-2-amd64 #1 SMP Debian 4.19.16-1 (2019-01-17) x86_64
Build Date	Mar 7 2019 19:43:34
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.3/apache2
Loaded Configuration File	/etc/php/7.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.3/apache2/conf.d
Additional .ini files parsed	/etc/php/7.3/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.3/apache2/conf.d/10-opcache.ini, /etc/php/7.3/apache2/conf.d/10-pdo.ini, /etc/php/7.3/apache2/conf.d/15-xml.ini, /etc/php/7.3/apache2/conf.d/20-bcmath.ini, /etc/php/7.3/apache2/conf.d/20-calendar.ini, /etc/php/7.3/apache2/conf.d/20-ctype.ini, /etc/php/7.3/apache2/conf.d/20-curl.ini, /etc/php/7.3/apache2/conf.d/20-dom.ini, /etc/php/7.3/apache2/conf.d/20-exif.ini, /etc/php/7.3/apache2/conf.d/20-fileinfo.ini, /etc/php/7.3/apache2/conf.d/20-ftp.ini, /etc/php/7.3/apache2/conf.d/20-gettext.ini, /etc/php/7.3/apache2/conf.d/20-iconv.ini, /etc/php/7.3/apache2/conf.d/20-json.ini, /etc/php/7.3/apache2/conf.d/20-mbstring.ini, /etc/php/7.3/apache2/conf.d/20-mysqli.ini, /etc/php/7.3/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.3/apache2/conf.d/20-phar.ini, /etc/php/7.3/apache2/conf.d/20-posix.ini, /etc/php/7.3/apache2/conf.d/20-readline.ini, /etc/php/7.3/apache2/conf.d/20-shmop.ini, /etc/php/7.3/apache2/conf.d/20-simplexml.ini, /etc/php/7.3/apache2/conf.d/20-sockets.ini, /etc/php/7.3/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.3/apache2/conf.d/20-sysvsem.ini, /etc/php/7.3/apache2/conf.d/20-sysvshm.ini, /etc/php/7.3/apache2/conf.d/20-tokenizer.ini, /etc/php/7.3/apache2/conf.d/20-wddx.ini, /etc/php/7.3/apache2/conf.d/20-xmlreader.ini, /etc/php/7.3/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.3/apache2/conf.d/20-xsl.ini, /etc/php/7.3/apache2/conf.d/20-zip.ini
PHP API	20180731
PHP Extension	20180731
Zend Extension	320180731
Zend Extension Build	API320180731,NTS

FIGURA 3 Pagina di avvio di PHP 7.3

2.3 Configurazione PHP

Occorre fornire a PHP i permessi per scrivere nella cartella del web server allo user www-data. Assegnare allo user e al gruppo www-data la proprietà della cartella /var/www/html con permesso di scrittura:

```
sudo chown -R www-data:www-data /var/www/html
sudo chmod 775 /var/www/html
```

Aggiungere l'utente pi al gruppo www-data:

```
sudo usermod -a -G www-data pi
```

Accedere al web server

Raggiungere il web server da Internet.

Se si desidera raggiungere il web server da rete internet, configurare sul proprio router l'inoltro della porta 80, protocollo TCP, verso l'indirizzo IP locale del server Raspberry Pi.

- Individuare il proprio indirizzo IP pubblico dal router stesso o tramite un servizio web.
- Digitare sul browser di un dispositivo connesso in Internet l'indirizzo appena trovato.
- Verificare che si apra la pagina di prova vista in precedenza.

FISSA LE CONOSCENZE

- Perché la scheda Raspberry Pi consente di realizzare un server con migliori prestazioni?
- Cos'è NGINX?

COMPETENZE IN GIOCO

Competenze disciplinari

- Conoscere i parametri di una scheda di rete per Arduino.
- Configurare la scheda di rete in base alla rete a cui è collegata.

Competenze chiave di cittadinanza

- Interpretare il testo.
- Risolvere problemi.
- Comunicare.
- Competenza digitale.

obiettivi formativi

- Consultare fonti Internet.
- Esporre i risultati della ricerca alla classe.

tempi

- Ricerca di informazioni in rete sul tema proposto: 1 ora.
- Personale risoluzione del tema proposto: 1 ora.
- Preparazione di una presentazione con la propria soluzione: 1 ora.
- Illustrazione dei risultati e dibattito in classe: 1 ora.
- Autovalutazione: 10 minuti.

strumenti

- Libro di testo.
- Dispositivo connesso a Internet.
- Carta e penna.
- Software per le presentazioni.
- Proiettore o LIM in classe o in laboratorio.

TEMA PROPOSTO

File sorgenti
Scarica il file

Occorre monitorare temperatura, umidità e luminosità ambientale di un laboratorio in cui si studia la crescita di alcune piante e trasmettere le informazioni su una pagina web di monitoraggio tramite connessione Internet via cavo.

– Scegliere i sensori adeguati alle esigenze (intervallo di valori da rilevare, precisione).
– Configurare opportunamente la scheda di rete.
– Determinare la struttura della pagina web.
– Determinare la modalità di memorizzazione delle informazioni (tempistiche, memorie di massa da utilizzare, tipologia di rappresentazione grafica).

SVOLGIMENTO

– Individuare su siti Internet i sensori più adatti e i loro costi.
– Ricercare esempi disponibili in rete per l'acquisizione, la visualizzazione e la memorizzazione delle singole grandezze.
– Disegnare uno schema dei vari blocchi utilizzati.
– Disegnare lo schema dei collegamenti elettrici.
– Verificare il funzionamento del software con i singoli sensori tramite simulatore online.
– Costruire una pagina HTML di prova e determinare i comandi necessari per la visualizzazione dei parametri richiesti.
– Integrare i vari listati in modo da gestire contemporaneamente le varie grandezze.
– Simulazione complessiva del programma e gestione della pagina web.
– Ricerca di possibili ottimizzazioni.
– Stesura della relazione.

A CASA

- Predisposizione del progetto in modo completo, inserendo la valutazione dei costi dei vari dispositivi utilizzati.
- Valutazione delle possibili migliorie al sistema.

IN CLASSE

- Condividi la presentazione dei tuoi risultati con i compagni.
- Confrontate e discutete insieme i casi che sono stati presentati.
- Stabilite quale caso rappresenta l'esempio migliore sia dal punto di vista tecnico, sia dal punto di vista economico per completezza e che meglio si adatta alla soluzione del tema proposto.
- Procedi con l'autovalutazione.

AUTOVALUTAZIONE

ATTIVITÀ	LIVELLO			
	INIZIALE	BASE	INTERMEDIO	AVANZATO
Ho compreso senza difficoltà le richieste dell'attività proposta?	Ho compreso solo alcune delle richieste aiutato dal docente. ☐	Con la guida del docente e la collaborazione dei compagni ho compreso quasi tutte le richieste. ☐	Ho compreso le richieste e in parte le ho svolte autonomamente. ☐	Ho identificato le richieste e le ho svolte senza difficoltà. ☐
Ho reperito in rete le informazioni su come configurare la scheda di rete e ottimizzare la visualizzazione dei dati acquisiti?	Ho reperito solo alcune delle informazioni utili. ☐	Con la guida del docente e la collaborazione dei compagni ho reperito quasi tutte le informazioni. ☐	Ho reperito le informazioni utili autonomamente e le ho elaborate. ☐	Ho identificato le informazioni utili e le ho elaborate senza difficoltà. ☐
Sono riuscito a realizzare una presentazione convincente?	Ho preparato una presentazione di 3 slide con poche informazioni. ☐	Ho preparato una presentazione con molte informazioni un po' confuse. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni abbastanza strutturate. Non sono riuscito a spiegarmi bene. ☐	Ho preparato una presentazione con molte informazioni ben strutturate. Sono riuscito a far capire a tutti i concetti che volevo esprimere. ☐

RISPOSTE AI QUESITI DI FINE UNITÀ

UNITÀ 1 LE ARCHITETTURE DI RETE

Vero (V) o falso (F) pag. 54

1. V **2.** F **3.** V **4.** F **5.** F **6.** V **7.** V **8.** V **9.** F **10.** V
11. F **12.** F **13.** F **14.** F **15.** F **16.** F **17.** V **18.** F
19. V **20.** F **21.** F **22.** V **23.** V **24.** F

Domande a scelta multipla pag. 54

1. C **2.** B **3.** B **4.** A **5.** D **6.** C **7.** A **8.** C **9.** D
10. B **11.** C **12.** A **13.** D **14.** C **15.** C **16.** D **17.** C
18. B

In English, please pag. 56

5 4 6 2 7 1 3

UNITÀ 2 IL PHYSICAL LAYER DEL TCP/IP

Vero (V) o falso (F) pag. 96

1. F **2.** V **3.** F **4.** F **5.** V **6.** V **7.** V **8.** F **9.** F **10.** F
11. V **12.** V **13.** V **14.** V **15.** V **16.** F **17.** V **18.** V
19. V **20.** V **21.** F **22.** V **23.** V **24.** V **25.** V **26.** F
27. V **28.** F **29.** F

Domande a scelta multipla pag. 97

1. C **2.** A **3.** C **4.** D **5.** D **6.** B

In English, please pag. 98

5 1 6 3 8 4 2 7
1. B **2.** C

UNITÀ 3 IL NETWORK LAYER DEL TCP/IP

Vero (V) o falso (F) pag. 152

1. F **2.** V **3.** F **4.** V **5.** V **6.** F **7.** V **8.** F **9.** V **10.** F
11. F **12.** F **13.** V **14.** F

Domande a scelta multipla pag. 152

1. B **2.** D **3.** A **4.** C **5.** B **6.** D **7.** C **8.** D **9.** A
10. D **11.** B **12.** A **13.** A **14.** B **15.** A

In English, please pag. 154

5 7 6 1 8 2 3 4

UNITÀ 4 L'EVOLUZIONE DI IP E IL MONITORING DELLA RETE

Vero (V) o falso (F) pag. 190

1. F **2.** F **3.** V **4.** V **5.** F **6.** V **7.** V **8.** F **9.** V **10.** V
11. V **12.** F

Domande a scelta multipla pag. 190

1. B **2.** B **3.** D **4.** D **5.** B **6.** B **7.** D **8.** A **9.** A **10.** B
11. D **12.** A **13.** B **14.** D **15.** A **16.** D **17.** C **18.** C

In English, please pag. 192

6 4 7 2 1 8 5 3

UNITÀ 5 INSTRADAMENTO E INTERCONNESSIONE DI RETI GEOGRAFICHE

Vero (V) o falso (F) pag. 252

1. V **2.** V **3.** F **4.** F **5.** F **6.** V **7.** V **8.** V **9.** V **10.** F
11. V **12.** F **13.** V **14.** F **15.** V **16.** V **17.** F **18.** F

Domande a scelta multipla pag. 252

1. D **2.** B **3.** B **4.** C **5.** B **6.** A **7.** C **8.** D **9.** C **10.** C
11. A **12.** D **13.** C

In English, please pag. 254

8 7 1 6 3 2 5 4

UNITÀ 6 IL TRANSPORT LAYER DEL TCP/IP

Vero (V) o falso (F) pag. 304

1. V **2.** F **3.** V **4.** V **5.** V **6.** F **7.** F **8.** F **9.** F **10.** V
11. V **12.** V **13.** F **14.** F **15.** V **16.** F **17.** F **18.** V

Domande a scelta multipla pag. 304

1. C **2.** B-C **3.** B **4.** B **5.** A **6.** B **7.** B **8.** A **9.** D
10. A

In English, please pag. 306

4 6 2 8 1 7 3 5
1. D **2.** C

UNITÀ 7 LA CONFIGURAZIONE DEL DHCP E DEL DNS

Vero (V) o falso (F) pag. 352

1. V **2.** V **3.** F **4.** F **5.** F **6.** V **7.** V **8.** F **9.** V **10.** V
11. V **12.** F **13.** V **14.** V **15.** F **16.** F **17.** V **18.** V
19. F **20.** V **21.** F **22.** F **23.** F **24.** F **25.** V **26.** V
27. F

Domande a scelta multipla pag. 352

1. D **2.** A **3.** C **4.** A **5.** A **6.** B **7.** C **8.** C

In English, please pag. 354

2 6 7 1 4 8 3 9 5

UNITÀ 8 L'APPLICATION LAYER DEL TCP/IP

Vero (V) o falso (F) pag. 408

1. V **2.** F **3.** V **4.** V **5.** V **6.** F **7.** F **8.** V **9.** V **10.** F
11. V **12.** V **13.** V **14.** V **15.** F **16.** F **17.** V

Domande a scelta multipla pag. 408

1. D **2.** C **3.** A **4.** B **5.** B **6.** A-B **7.** D **8.** C **9.** B
10. D **11.** D **12.** A **13.** C **14.** A **15.** B

In English, please pag. 410

4 3 1 6 2 8 5 7

LEGENDA IMMAGINI

APPARATI DI RETE

Router

Switch

PoE Power Injector

Hub

Antenna mobile

Modem-Router-Switch-
Access point Rete domestica

Access point aziendale

END SYSTEM

Telefono cordless

Telefono VoIP

Termostato wi-fi

Computer desktop

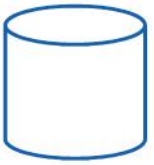
Computer laptop

Tablet

Smartphone

Server

Server rack



Disco del computer
Database

Database

REFERENZE ICONOGRAFICHE

Archivio GettyImages

3DSculptor
4FR
AtnoYdur
chebyraha
Chunumunu
cosmin4000
denisk0
Floortje
Franck-Boston
gilaxia
JackF
Jasmina007
kai zhang
leminuit
Mitja Mladkovic
neopicture
Nikuwka
pingingz
Rawpixel Ltd
sqback
Thomas_EyeDesign
uschools

SOFTWARE CITATI NEL TESTO

Software utilizzati

Cisco Packet Tracer 7.3.1
Wireshark 3.x
Network Mapper (Nmap)
Processing
NGINX
PHP

Sistemi Operativi affrontati

Windows (Windows 10)
GNU/Linux (UBUNTU 20.04 LTS desktop)
Raspberry Pi4 (Raspberry Pi OS)
Cisco IOS (Internetwork Operating System)

Software/marchi citati

Cisco
Microsoft
Active Directory
Internet Information Services (IIS)
Apache
Outlook
Thunderbird
Gmail
Hotmail
Yahoo!Mail.
Filezilla
Firefox
Edge
Opera
Chrome
Android
Unix BSD (Berkeley Software Distribution)
VirtualBox
VMWare
Boson NetSim Network Simulator
GNS3 (Graphical Network Simulator-3)
RADb by Merit Network, Inc.
Google Maps
YouTube
Twitter
Amazon Prime Video
Netflix
Hulu
Asterisk
WhatsApp
Facebook Live
Periscope
Skype
Akamai