

Reti di elaboratori

Leonardo Ganzaroli

Indice

Introduzione	1
1 Reti ed Internet	3
1.1 Internet	3
1.2 Comunicazione	5
1.3 Protocolli	6
2 Livelli	7
2.1 Collegamento	7
2.1.1 Protocolli	8
2.2 Rete	9
2.2.1 IP	9
2.2.2 Router	10
2.2.3 Algoritmi di instradamento	11
2.2.4 Protocolli	12
2.3 Trasporto	12
2.3.1 UDP	13
2.3.2 TCP	14
2.4 Applicazione	15
2.4.1 HTTP	16
2.4.2 Posta elettronica	18
2.4.3 DNS	19
2.4.4 FTP	20
2.4.5 BitTorrent	20

Introduzione

Questi appunti sono derivanti principalmente dalle slide del corso di *Reti di elaboratori* che ho svolto durante la laurea Triennale di informatica all'università "La Sapienza".

1 Reti ed Internet

1.1 Internet

Definizione Una rete è un'infrastruttura composta da *nodi* collegati tramite *link*.

I nodi si possono dividere in 2 categorie:

1. Terminali
 - **Server**
 - **Host**
2. Dispositivi di interconnessione
 - **Router**
Connette più reti.
 - **Switch**
Connette più terminali in una rete.
 - **Modem**

I link possono essere diversi:

- Cavo coassiale
- Fibra ottica
- Doppino intrecciato
- Onde elettromagnetiche

Le reti stesse si possono classificare in base alla loro dimensione:

- **PAN** (Personal Area Network)
- **LAN** (Local ...)
- **MAN** (Metropolitan ...)
- **WAN** (Wide ...)
- **Internet**

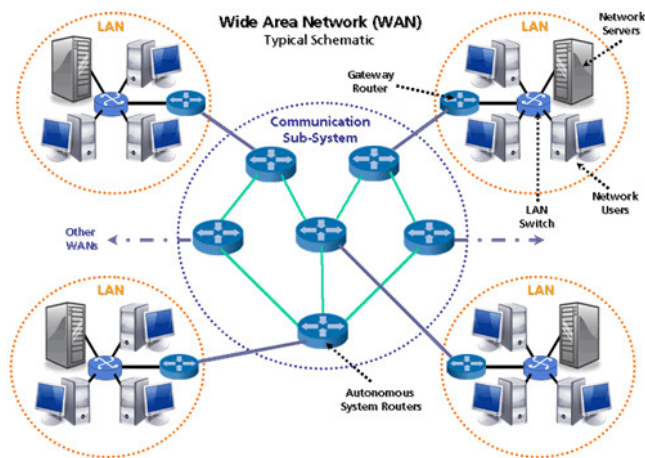


Figura 1: Esempio di rete

Definizione Internet è la rete di tutte le reti.

Si può dividere in 3 elementi:

1. **Network Edge**

L'insieme di tutti i nodi terminali connessi.

2. **Access Network**

I collegamenti tra i terminali e il primo Router.

3. **Nucleo**

La connessione tra le sottoreti.

Il nucleo stesso presenta una divisione a livelli, la connessione tra questi avviene tramite gli *Internet Exchange Point*.

L'accesso ad Internet può avvenire in diversi modi:

- Via cavo
- Via DSL
- Via rete cellulare
- Via WLAN

1.2 Comunicazione

Definizione Dividendo un messaggio da inviare in blocchi si ottengono i pacchetti.

Definizione La commutazione è l'azione che sposta l'informazione in input ad un nodo all'output giusto, avviene in 2 modi:

1. Pacchetto

I singoli pacchetti arrivano al destinatario "rimbalzando" tra i nodi, ognuno potrebbe seguire un percorso diverso.

2. Circuito

Per far comunicare due terminali viene instaurato un percorso fisico che non può essere usato da nessun'altro fino alla sua "liberazione", si può effettuare una divisione in tempo/frequenza per permettere a più circuiti di usare un link contemporaneamente.

Definizione L'instradamento è la determinazione del percorso che un pacchetto deve seguire.

Definizione La larghezza di banda è:

1. La larghezza dell'intervallo di frequenze del mezzo trasmissivo
2. Il numero di *bit/s* che il mezzo può trasmettere

Definizione Il throughput è la quantità di *bit/s* passanti per un nodo.

Definizione Il tempo necessario per inviare un pacchetto tra 2 nodi è detto latenza, è composto da:

- Lat. di trasmissione
Tempo necessario per immetterlo sul link.
- Lat. di propagazione
Tempo necessario per arrivare alla destinazione.
- Lat. di queueing
Eventuale tempo passato in attesa nella coda di trasmissione.
- Lat. di elaborazione
Tempo dovuto ad operazioni del nodo.

1.3 Protocolli

Definizione Un protocollo è un insieme di regole.

Per permettere la comunicazione vengono divisi i vari compiti in livelli indipendenti, ognuno con il suo protocollo. Così facendo si possono far comunicare sistemi diversi instaurando dei collegamenti logici tra i vari livelli.

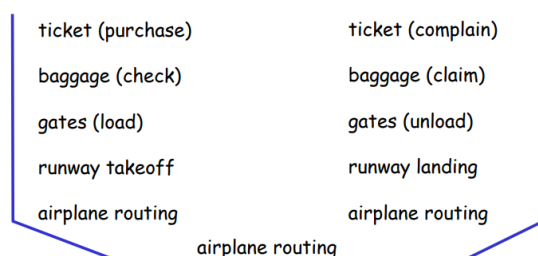


Figura 2: Esempio divisione a livelli

I 2 modelli principali nell'ambito delle reti sono:

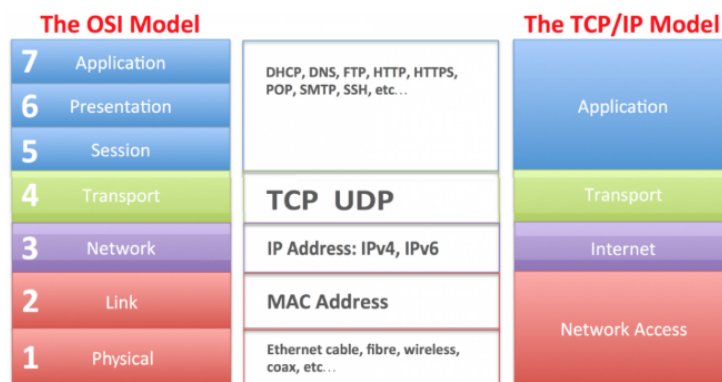


Figura 3: ISO/OSI e TCP/IP

Quando un messaggio deve essere inviato attraversa prima tutti i livelli partendo dall'alto, quando arriva al destinatario avviene il processo inverso.

Ogni livello aggiunge un header con delle informazioni al messaggio prima di passarlo al livello inferiore (incapsulamento).

2 Livelli

2.1 Collegamento

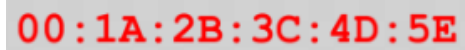
Fornisce i seguenti servizi:

- Framing
- Consegna tra nodi adiacenti
- Rilevamento e correzione errori
- Controllo del flusso
- Accesso al link

Viene diviso in 2 sottolivelli:

1. **LLC**
2. **MAC**

Definizione L'indirizzo MAC è un indirizzo identificativo usato localmente, esso viene assegnato ad ogni scheda di rete ed è usato per inviare frame ad un'altra scheda fisicamente connessa.



00:1A:2B:3C:4D:5E

Figura 4: Indirizzo MAC

Per trovare ed eventualmente risolvere errori ci sono diversi modi:

- **Parity Checking**
- **Parity Checking multidimensionale**
- **CRC**

Si individuano 2 tipi di collegamenti:

1. **Punto-punto**
2. **Broadcast**

Essendo diversi presentano un funzionamento che richiede protocolli specifici per effettuare correttamente trasmissioni di dati.

2.1.1 Protocolli

- **PPP**

Usato per comunicazioni dirette tra due nodi.

- **ALOHA**

- **Slotted**

Assumendo che i frame abbiano dimensioni uguali:

- * Il tempo si divide in intervalli di lunghezza pari al tempo necessario per trasmettere un frame
- * I nodi sono sincronizzati
- * Quando arriva un frame aspetta il prossimo slot temporale per inviarlo
- * In caso di collisioni riprova la trasmissione nello slot successivo con una certa probabilità

- **Pure**

Le uniche differenze con il precedente sono:

- * Il tempo non viene diviso
- * Quando un frame arriva viene subito inviato

- **CDMA**

- **TDMA**

- **FDMA**

- **CSMA**

Protocollo ad accesso casuale, a grandi linee:

- Prima di trasmettere ascolta il canale
- Se inattivo trasmetti, altrimenti aspetta

Ce ne sono di diversi tipi, 2 sono:

- **CD**

In caso di collisione il nodo trasmittente interrompe la trasmissione ed invia un apposito segnale per indicarla, poi torna in attesa.

- **CA**

- * Non c'è rilevamento delle collisioni
- * Si usa un ACK per capire se la trasmissione è riuscita
- * C'è un ascolto anche per il messaggio ACK
- * La collisione può avvenire anche per gli ACK

- **ARP**

Permette di associare un indirizzo MAC ad un indirizzo IP, ogni nodo possiede una tabella che contiene le associazioni.

2.2 Rete

Lo scopo di questo livello è di effettuare l'inoltro e l'instradamento dei pacchetti, garantisce quindi una comunicazione logica tra sorgente e destinatario.

Un fatto fondamentale è che ci si basa su un modello *best effort*, ossia la consegna dei pacchetti, il tempo necessario, etc... non è garantita.

2.2.1 IP

Il protocollo usato a questo livello è l'IP, esso è formato da:

- **Interfaccia**

Collegamento alla rete fisica.

- **Sottorete**

L'insieme di interfacce direttamente raggiungibili.

- **Indirizzo IP**

Un identificatore associato ad un'interfaccia.

Version	Length (IHL)	Type of Service (TOS)	Total Length			
Identification			Flag	Fragment Offset		
Time To Live (TTL)		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
Options						

Figura 5: Header IP

Ci sono 2 versioni dell'indirizzo IP:

1. **IPv4**

32 bit divisi in quattro gruppi.

2. **IPv6**

128 bit divisi in 8 gruppi di 2 byte.

Dato che il passaggio a IPv6 ancora non è completo si usa il *NAT* per risparmiare indirizzi, ossia il Router associa un unico indirizzo IP esterno a tutti i dispositivi della rete interna.

2.2.2 Router

I Router sono i dispositivi principali nel caso la comunicazione non sia interna alla sottorete, mantengono delle tabelle contenenti le varie destinazioni di una certa rete ed eventualmente la sua "distanza".

La struttura generale è la seguente:

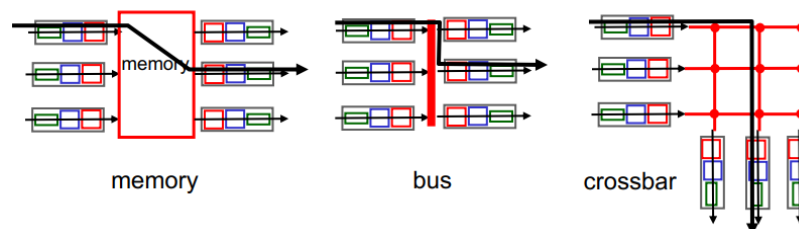
- Porte Input/Output connesse ai link
 - Composte da:
 - Memoria interna
 - Terminazione di linea
 - Interfaccia con livello inferiore
- Struttura di commutazione tra le porte
- Processore di instradamento

In alcuni casi si potrebbe verificare un accodamento nelle porte che può portare ad una perdita di pacchetti, per minimizzare il problema si usano degli algoritmi di scheduling:

- **FCFS**
- **Con priorità**
- **Round Robin**
- **WFQ**

Alcuni collegamenti possono avere una dimensione di trasferimento massima sul link, in questo caso avviene la frammentazione del pacchetto con successiva ricostruzione da parte del destinatario.

Per effettuare lo spostamento verso l'output ci sono 3 modi:



2.2.3 Algoritmi di instradamento

Gli algoritmi di routing vengono classificati in:

- Statici/Dinamici
- Globali/Decentralizzati

Notazione:

- $c_{x,y}$ è il costo del link tra i nodi x e y
- $D(v)$ è la stima del costo dalla sorgente a v
- $p(v)$ è il nodo predecessore

Considerando la sua struttura è possibile rappresentare una rete come un grafo, di conseguenza si può usare una variante dell'algoritmo di Dijkstra:

```
Link-State(G,x):
R = {x}
for v ∈ V(G) do
    if ∃(u,v) ∈ E(G) then
        D(v) = cu,v
    else
        D(v) = ∞
    end if
end for
while R ≠ V(G) do
    w = minw ∈ V(G) \ R(D(w))
    R.add(w)
    for x ∈ V(G) \ R do
        if ∃(w,x) ∈ E(G) then
            D(x) = min(D(x), D(w) + cw,x)
            p(x) = w
        end if
    end for
end while
```

Un'altra possibilità è l'algoritmo Distance-vector, è basato sull'equazione di Bellman-Ford $D_x(y) = \min_{v \in V(G)}(c_{x,v} + D_v(y))$. Si può riassumere in:

1. Inizializza la tabella delle distanze
2. Attendi un cambio di costo
3. Ricalcola
4. Invia le eventuali modifiche ai vicini

2.2.4 Protocolli

- **DHCP**
Permette di assegnare dinamicamente indirizzi IP.
- **ICMP**
Usato per scambiare messaggi a questo livello.
- **RIP**
Instradamento tramite Distance-vector.
- **OSPF**
Instradamento tramite Link-State.
- **BGP**
Instradamento tramite Path-vector.
- **IGMP**
Usato per richiedere l'adesione ad un gruppo multicast.

2.3 Trasporto

Lo scopo di questo livello è di instaurare una connessione logica tra i processi applicativi in esecuzione sui dispositivi.

Per identificare i processi viene fatto uso di un numero di porta ad essi associato, nell'Header inserito qui + Header IP saranno generalmente presenti:

- IP + porta origine
- IP + porta destinatario

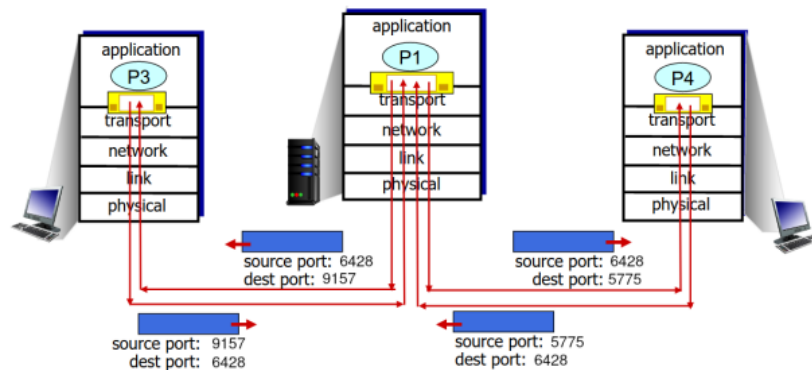


Figura 6: Esempio di comunicazione

2.3.1 UDP

Il protocollo UDP è senza connessione e *barebone*, quindi le informazioni inviate potrebbero arrivare in qualsiasi ordine o non arrivare proprio.

Le sue caratteristiche lo rendono utile per i seguenti motivi:

- Comunicazione veloce
- Dimensione minima dell'Header
- "Ignora" la congestione
- Presenta comunque un minimo controllo degli errori tramite Checksum

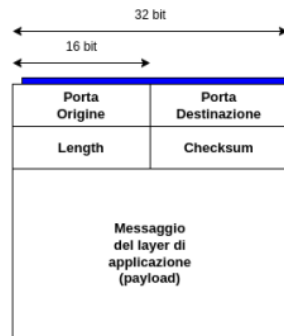


Figura 7: Header UDP

Il checksum è essenzialmente la somma in complemento ad uno (con wrap-around) degli altri campi.

Per i motivi visti UDP viene usato dalle applicazioni che tollerano perdita di dati ma sono dipendenti dal tempo, per esempio il traffico audio/video (anche in tempo reale) è un candidato perfetto.

2.3.2 TCP

Questo protocollo invece è orientato alla connessione e la inizializza tramite *handshaking* (a 3 vie), offre inoltre uno stream di byte affidabile ed ordinato.

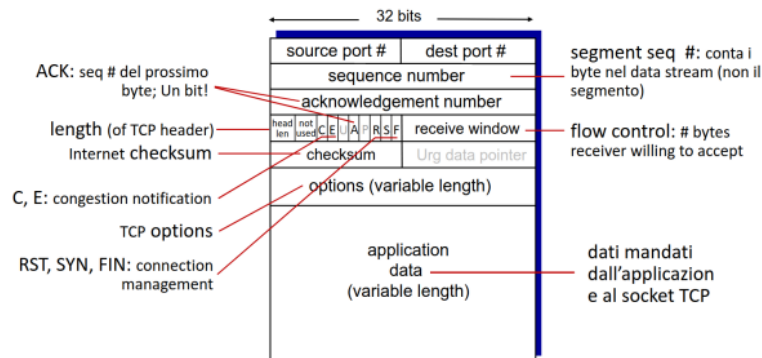


Figura 8: Header TCP

Per garantire il trasporto usa:

- **ACK cumulativi**
- **Numero di sequenza**

Corrisponde al numero di sequenza del primo byte del settore data, invece per gli ACK è il byte successivo aspettato.

- **Timer**

Fa riferimento al pacchetto più vecchio non confermato, se scade viene rinviato e si resetta.

Definizione L'RTT è il tempo impiegato da un piccolo pacchetto per effettuare il percorso client→server→client.

Questo protocollo gestisce inoltre sia il flusso (tramite la receive window) che la congestione con diversi metodi:

- **Fast recovery**

Si dimezza il rate di invio in caso di perdite rilevate.

- **Slow start**

Partendo da un valore prefissato all'inizio della connessione, ogni RTT si raddoppia il numero di byte inviati fino alla prima perdita.

2.4.1 HTTP

- Porta 80
- TCP
- Stateless
- Client-Server

Le connessioni di questo tipo originariamente erano non persistenti, dalla versione 1.1 si possono creare anche connessioni persistenti.

I messaggi vengono formattati in Ascii, la struttura di un messaggio è:

- **Richiesta**
 - Riga di richiesta
 - Header
 - Body
- **Risposta**
 - Riga di stato
 - Header
 - Body

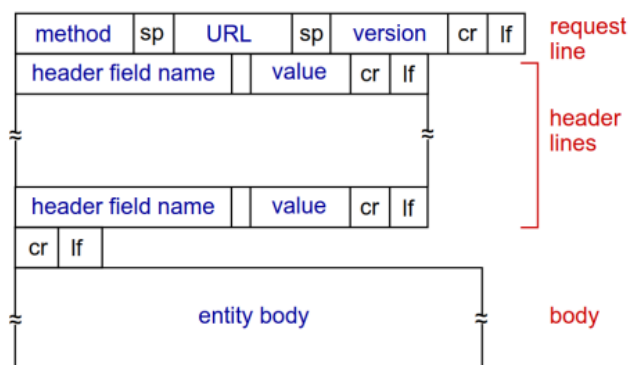


Figura 10: Struttura richiesta

I possibili metodi usati nella riga di richiesta sono:

- **GET**
Invio dati al server, presenti nell'URL.
- **POST**
Invio dati al server, presenti nel Body.
- **HEAD**
Richiesta esclusiva dell'Header che si riceverebbe usando GET.
- **PUT**
Caricamento/Sostituzione file, non più usato.

I possibili codici presenti nella riga di stato sono:

- **1xx**
Risposta contiene solo informazioni.
- **2xx**
Richiesta andata a buon fine.
- **3xx**
Avvenuto un reindirizzamento.
- **4xx**
Errore nella richiesta.
- **5xx**
Errore del server.

Definizione Un cookie è un file di testo contenente informazioni, viene salvato nel client da parte del server.

Definizione Un server proxy è un intermediario tra client e server.

2.4.2 Posta elettronica

Definizione Uno User Agent (UA) è un processo attivo sul client che informa l'utente di nuove e-mail ricevute, permette anche la lettura/scrittura di mail che vengono inviate poi all'MTA.

Definizione Un Mail Transfer Agent (MTA) è un processo attivo sul server che invia e-mail ad un altro MTA.

Definizione Un Mail Access Agent (MAA) è un processo attivo sul server che permette di leggere la posta in arrivo.

SMTP

- Porta 25
- Ascii
- Interazione comando-risposta
- Invio diretto tra MTA

MIME

Un protocollo avanzato che permette di usare messaggi multimediali.

POP3

- Porta 110
- Stateless
- Scarica le mail dal server (vengono cancellate sul server)

IMAP

- Porta 143
- Stateful
- Come POP3 ma le mail non vengono cancellate

2.4.3 DNS

Definizione Il Domain Name System viene usato per mappare i nodi della rete ad un nome identificativo, alcune delle sue funzioni sono:

- Traduzione nome in IP
- Distribuzione del carico
- Host Aliasing

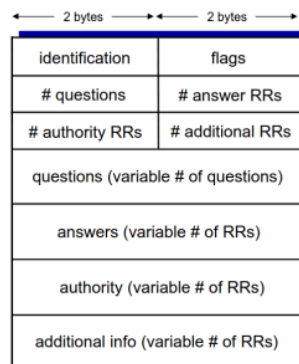
Essendo un sistema decentralizzato presenta una gerarchia a livelli:

1. **Root**
2. **TLD**
3. **Server autoritativi**
4. **Server locali**

Il protocollo omonimo:

- Porta 53
- UDP
- Stateless

Richieste e risposte hanno la stessa struttura:



2.4.4 FTP

- Porta 20 (dati)
- Porta 21 (controllo)
- 2 connessioni TCP
- Stateful

Permette di scambiare file tra client e server.

2.4.5 BitTorrent

- Solitamente porte [6881-6889]
- TCP
- P2P

Definizione Un torrent è un insieme di peer che si scambiano pezzi di file.

Definizione Un tracker è un dispositivo che tiene traccia dei peer in un torrent.

Durante il download un peer svolge anche la funzione di uploader.