



Gestione delle Reti di Telecomunicazioni

Modello TCP/IP

Ing. Tommaso Pecorella

Ing. Giada Mennuti

{pecos,giada}@lenst.det.unifi.it

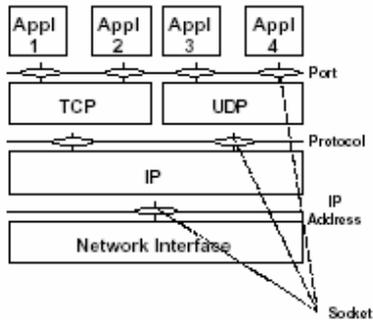
Protocolli di trasporto



Lo strato di trasporto fornisce un servizio di trasferimento allo strato applicativo conforme ai requisiti di qualità richiesti dall'applicazione

- User Datagram Protocol (UDP)
 - è utilizzato quando l'applicazione non richiede funzioni di controllo di flusso e controllo d'errore
- Transport Control Protocol (TCP)
 - è utilizzato per applicazioni che generano flussi informativi di una certa complessità che richiedono funzioni di controllo d'errore e di flusso

Mux / DeMux



Distingue tra i programmi applicativi (processi) che sono utenti dello stesso servizio di trasporto

Port

- identifica un utente (applicazione) dello strato di trasporto
- è rappresentato da un intero (16 bit)

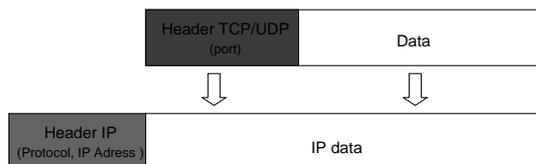
Socket

- identifica l'interfaccia tra l'applicazione ed i protocolli di comunicazione
- è rappresentata dalla tripletta (port; protocol; IP_Address)

Mux / DeMux



- La componente "Port" è contenuta nell'intestazione dell'unità dati di TCP/UDP
- Le componenti "Protocol" e "IP_Address" sono contenute nell'intestazione dell'unità dati di IP
- Lo strato IP esegue la moltiplicazione di una molteplicità di flussi di trasporto



Numeri di porta



Il numero di porta può essere

- statico
 - sono identificativi staticamente associati ad applicazioni largamente utilizzate
 - sono utilizzati identificativi inferiori a 256

Numero	Applicazione
21	FTP (File Transfer Protocol)
23	Telnet
25	SMTP (Simple Mail Transport Protocol)

- dinamico
 - sono identificativi assegnati direttamente dal sistema operativo al momento dell'apertura della connessione
 - si utilizzano valori maggiori di 1023

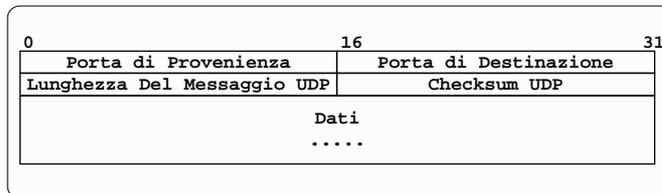
UDP



User Datagram Protocol (UDP)

- E' un protocollo senza connessione
- Non supporta meccanismi di riscontro e di controllo d'errore
- E' utilizzato per il supporto di transazioni semplici tra applicativi
 - interrogazioni di database
 - risoluzione di indirizzi
 - messaggi di management

UDP



- **Source Port (16 bit) e Destination Port (16 bit)**
 - identificano i processi sorgente e destinazione dei dati
- **Datagram Length (16 bit)**
 - è la lunghezza totale (espressa in byte) del datagramma, compreso l'header UDP
- **Checksum (16 bit)**
 - protegge il datagramma UDP e i campi indirizzo, protocol e datagram length dell'header IP

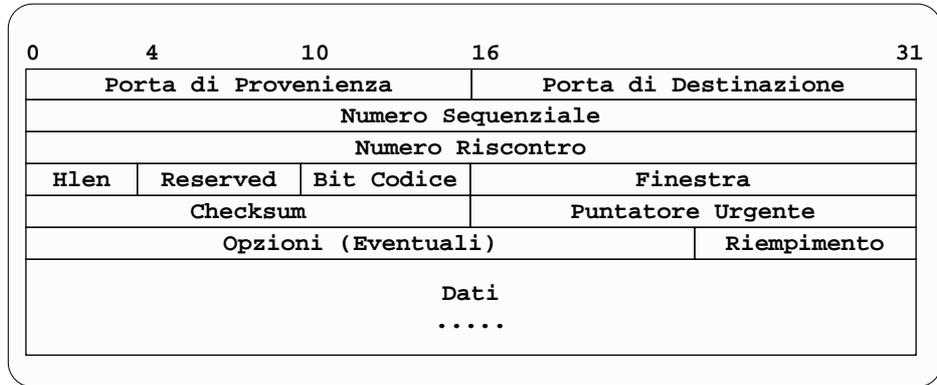
TCP



Transmission Control Protocol (TCP)

- Gestisce un flusso informativo bi-direzionale tra due host
- E' un protocollo con connessione
- Funzioni eseguite
 - indirizzamento di una specifica applicazione all'interno di un host
 - ri-ordinamento delle unità informative
 - controllo e recupero di errore
 - controllo di flusso
 - controllo di congestione

TCP



TCP



Source Port (16 bit) e Destination Port (16 bit)

- identificano i processi sorgente e destinazione

Sequence Number (32 bit)

- numero di sequenza in trasmissione
- contiene il numero di sequenza del primo byte di dati contenuti nel segmento a partire dall'inizio della sessione TCP

Acknowledgement Number (32 bit)

- numero di sequenza in ricezione
- contiene il numero di sequenza del prossimo byte che il trasmettitore del segmento si aspetta di ricevere
- è possibile la modalità piggybacking di riscontro (ovvero il riscontro attraverso un segmento dati nell'altro verso di trasmissione)

TCP



- HLEN (4 bit)
 - contiene il numero di parole di 32 bit contenute nell'intestazione TCP
 - l'intestazione TCP non supera i 60 byte ed è sempre un multiplo di 32
- Reserved (6 bit)
 - riservato per usi futuri, per ora contiene degli zeri
- Window (16 bit)
 - larghezza della finestra in byte (controllo di flusso è orientato al byte)
 - è il numero di byte che, ad iniziare dal valore del campo Ack Number, l'host che trasmette il segmento è in grado di ricevere
- Checksum (16 bit)
 - protegge l'intero segmento più alcuni campi dell'header IP (es. indirizzi)

TCP

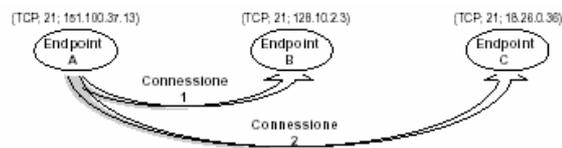


- Urgent Pointer (16 bit)
 - contiene il numero di sequenza dell'ultimo byte dei dati che devono essere consegnati urgentemente al processo ricevente
 - tipicamente sono messaggi di controllo (out-of-band traffic)
- Options (di lunghezza variabile)
 - sono presenti solo raramente
 - Esempio: Maximum Segment Size (MSS)
- Padding (di lunghezza variabile)
 - impone che l'intestazione abbia una lunghezza multipla di 32 bit

Connessione TCP



- Il protocollo TCP è un protocollo orientato alla connessione
- Nella fase di instaurazione della connessione, le due entità TCP remote si sincronizzano scambiandosi i propri numeri di sequenza iniziale
- Il numero di sequenza iniziale rappresenta il numero a partire dal quale tutti i byte trasmessi saranno numerati in sequenza
- Una connessione TCP è identificata dalla coppia di socket associati agli endpoint tra cui vengono scambiate informazioni
- Un endpoint può essere impegnato allo stesso tempo in più connessioni TCP



Controllo d'errore in TCP



- Il TCP prevede esclusivamente riscontri positivi (ACK)
- La ritrasmissione dei segmenti è innescata dalla mancata ricezione degli ACK entro un fissato tempo limite (Retransmission Timeout - RTO)
- Il dimensionamento dell'RTO è critico
 - se il suo valore è troppo piccolo, alcuni segmenti in ritardo a causa di congestione, potrebbero essere considerati persi e quindi ritrasmessi inutilmente
 - se il suo valore è troppo grande, la risposta ad un evento di perdita sarebbe troppo lenta

Controllo d'errore in TCP



- Il Retransmission TimeOut (RTO) è determinato con uno schema adattativo
- Il TCP misura dinamicamente il Round Trip Time (RTT)
 - RTT = ritardo tra l'invio di un segmento e la ricezione del relativo ACK
- Il valore di RTO è scelto maggiore del valore medio di RTT
- La misura del RTT è affetta da errori
 - l'emissione degli ACK del ricevente può essere non immediata
 - in caso di ritrasmissioni è impossibile distinguere se l'ACK si riferisce al segmento iniziale o al segmento ritrasmesso
 - lo stato di congestione della rete può cambiare rapidamente

Controllo di flusso in TCP



- Il controllo di flusso ha lo scopo di limitare il tasso di generazione dei dati da parte di un host
- Il controllo di flusso è indispensabile in Internet dove sono presenti host di potenzialità molto diverse
- TCP utilizza un controllo di flusso a finestra basato su
 - una finestra scorrevole di ampiezza variabile
 - uno schema di allocazione di crediti

Controllo di congestione in TCP



- Ha lo scopo di recuperare situazioni di sovraccarico nella rete limitando il traffico offerto alla rete
- Difficoltà:
 - il protocollo IP (protocollo di rete) non possiede alcun meccanismo per rivelare e controllare la congestione
 - il TCP è un protocollo end-to-end e può rivelare e controllare la congestione solo in modo indiretto
 - la rete non coopera con gli host per il controllo della congestione
 - la conoscenza dello stato della rete da parte delle entità TCP è imperfetta a causa dei ritardi di rete
 - le entità TCP che usano la rete non cooperano tra loro, anzi competono per l'uso delle risorse distribuite

Controllo di Congestione



- In caso di congestione, il controllo di flusso a finestra protegge implicitamente, oltre al destinatario, anche la rete
 - se la rete è congestionata arriveranno meno riscontri e quindi saranno emessi un numero minore di segmenti
 - il meccanismo adattativo di timeout evita ritrasmissioni che porterebbero ad un aumento della congestione invece che ad una sua diminuzione
- Il controllo di flusso end-to-end riesce ad adattare il rate di emissione della sorgente a quello corrispondente al bottleneck della rete (proprietà di self-clocking)

QOS in internet



- La Internet attuale è di tipo "Best Effort"
- Non esistono modi per garantire la Qualità del Servizio ricevuto su Internet
- Tutti i flussi sono trattati allo stesso modo
- La pila protocollare TCP/IP è stata sviluppata con l'idea di far accedere tutti allo stesso modo e con le stesse priorità
- La maggior parte dei nodi di rete (Router) ha delle code interne di tipo FIFO; se arrivano più pacchetti di quelli gestibili, vengono scartati

Instradamento



- **All'accensione una macchina sa:**
 - il suo indirizzo MAC
 - il suo indirizzo IP (e la rete locale a cui appartiene)
 - il suo indirizzo alfanumerico
- **NON sa:**
 - chi ha attorno (macchine "visibili" direttamente)
 - qual'è il modo per raggiungere l'esterno (indirizzo dei gateways)

Instradamento



- Un cammino attraversato da un *datagramma* IP è composto da sotto-reti interconnesse da router
- Un datagramma è interpretato da una sotto-rete come un'unità di dati di servizio (SDU)
- Una sotto-rete consegna la SDU al router successivo o alla destinazione (se la destinazione è all'interno della sottorete) utilizzando i propri meccanismi protocollari
- Due tipi di instradamento
 - diretto
 - indiretto

Instradamento diretto



si applica quando il pacchetto deve essere rilanciato nella sotto-rete di destinazione

- l'host di destinazione è connesso alla stessa sotto-rete dell'host sorgente o del router che emette il *datagramma*
- Il trasferimento dei *datagrammi* IP non coinvolge router intermedi
- E' necessaria la traduzione dell'indirizzo IP dell'host di destinazione nel suo indirizzo fisico (es. indirizzo MAC)
- Il *datagramma* IP viene incapsulato nell'unità dati della sotto-rete che viene inviata direttamente all'host di destinazione
- L'instradamento all'interno della sotto-rete utilizza i meccanismi specifici della sotto-rete

Instradamento indiretto



si applica quando il pacchetto deve essere instradato in sotto-reti diverse da quella di destinazione

- quando l'host di destinazione è connesso ad una sotto-rete diversa da quella dell'host sorgente o del router che emette il datagramma
- L'host mittente identifica il router a cui inviare il datagramma IP ed individua il suo indirizzo fisico
- Il router esamina il datagramma IP ricevuto e decide il router successivo verso cui instradarlo
 - l'instradamento attraverso la sotto-rete che connette i due router avviene secondo i meccanismi della sottorete
- Il processo si ripete di router in router sino alla sotto-rete di destinazione
 - nella sotto-rete di destinazione è utilizzato l'instradamento diretto