

## – 1 – Introduzione alle reti di calcolatori.

### – 1.1 – Reti e funzionamento.

Internet può essere descritto come una rete di calcolatori interconnessi e composta da:

- Host, sistemi terminali, ovvero dispositivi della rete;
- Communication Link, ovvero una rete di collegamento (costituita da fili di rame, fibre ottiche ecc) dotata di velocità di trasmissione e lunghezza di banda;
- Commutatori di pacchetti, ovvero ulteriori dispositivi che inoltrano i pacchetti;
- ISP, Internet Service Provider, attraverso i quali si accede ad Internet;
- Protocolli di comunicazione, che controllano invio e ricezione dei messaggi.

#### – 1.1.1 – Accesso alle informazioni.

Esistono diversi modelli con funzioni diverse per lo scambio di informazioni.

##### – 1.1.1.1 – Modello Client-Server.

Il modello Client-Server, che è alla base delle applicazioni network, è utilizzato per l'accesso alle risorse dell'Internet. Vi sono:

- Client: colui che richiede le informazioni o qualsiasi servizio di rete;
- Server: colui che fornisce le informazioni o qualsiasi servizio di rete.

In esso sono coinvolti almeno due processi:

- il processo client che manda un messaggio, attraverso la rete, al processo server e resta in attesa di una risposta;
- il processo server che, una volta ricevuta una richiesta, esegue il lavoro/recupera dati e restituisce la risposta al processo client.

##### – 1.1.1.2 – Peer to Peer.

Nel modello P2P ogni elaboratore può assumere il ruolo sia di client che di server. Data la multifunzionalità degli elaboratori, essi hanno un proprio database delle informazioni quindi non sono necessari server dedicati. Utile al file sharing o telefonia internet.

##### – 1.1.1.3 – Comunicazioni da persona a persona.

Grazie al programma talk di UNIX si sono potute sviluppare applicazioni per la comunicazione tra persone come: instant message, email, wiki, ecc.

**– 1.1.2 – Tipologie delle reti.**

Le reti si possono distinguere per l'accesso a dei tipi di dati. Vi sono:

- Rete di accesso a banda larga e mobile;
- Reti di distribuzioni per contenuti o data center: esse muovono grandi quantità di dati tra i server tramite banda trasversale ed inoltre sfruttano le CDN(Content Delivery Network) per connettere l'utente richiedente con il server più vicino;
- Reti di transito per l'accesso ai data center, può capitare che il servizio richiesto non risieda nella propria rete quindi esso deve attraversare internet dal data center alla rete di accesso per poi finire nel dispositivo richiedente;
- Reti aziendali, che permette il file sharing tra dispositivi della rete.

NOTA: i contenuti sono replicati in un singolo CDN e per decidere quale copia servire si tiene conto della distanza tra client e replica, carico su ciascun server della CDN e traffico e congestione della rete.

**– 1.1.3 – Tecnologia di rete.**

Come già accennato, le reti possono differire per ampiezza del raggio d'azione. Abbiamo:

**– 1.1.3.1 – PAN.**

Le reti PAN connettono i dispositivi entro un raggio molto piccolo. Un esempio sono le reti che consentono la connessione di periferiche al computer. Solitamente viene adottato il paradigma master-slave, dove il computer funge da master e comunica con le periferiche con informazioni come indirizzi da usare, frequenza ecc.

**– 1.1.3.2 – LAN.**

le reti LAN sono reti private in grado di operare all'interno o nelle vicinanze di un edificio. Consentono connettere tra loro gli elaboratori tramite due tipi di connessione:

- LAN wireless: ogni PC è dotato di ricevitore radio e antenna e si connette agli altri tramite Access Point, router o direttamente tramite configurazione mesh (reti simili a quelle wi fi di casa con punti di accesso simili);
- LAN cablata: usato diverse tecnologie di trasmissione. Sono estremamente veloce con una latenza molto bassa e quasi assenza di errori di trasmissione.

**– 1.1.3.3 – MAN.**

Le MAN sono reti che ricoprono un'intera città. In esse i segnali vengono inseriti nella stazione di testa centralizzata per la successiva distribuzione.

**– 1.1.3.4 – WAN.**

Le WAN ricoprono un'area geografica molto estesa, spesso una nazione o un continente. La rete è costituita da:

- Host e PC delle varie sedi;
- Subnet formata da linee di trasmissione e elementi di commutazione, generalmente router e/o switch. Quest'ultimi, all'arrivo dei dati, devono scegliere la linea di uscita tramite forwarding e inoltrare i dati su di essa.

La maggior parte delle volte, le WAN sono formate da reti connesse da coppie di router. Per la comunicazione tra due si applicano degli algoritmi di routing per trovare i percorsi e scegliere quello migliore.

**– 1.1.3.5 – Internetwork.**

Date le differenze hardware/software tra le reti, si deve trovare un adattamento per poter effettuare la loro connessione, questo è detto gateway. Quindi possiamo definire una internetwork come una rete di reti connesse tra di loro distinte e indipendenti.

**– 1.1.4 – Architettura di Internet.**

Per collegarsi ad Internet, bisogna effettuare una connessione ad un ISP che fornisce diversi tipi di accesso (TV, fibre ottiche, DSL). Una volta connessi all'ISP i nostri dati attraversano diversi ISP, tramite store & forward, fino a quello del ricevente.

Gli ISP sono dotati di un POP, Point of Presence, che permette appunto la connessione ad essi. Tale connessione può avvenire con:

- Peering Point: i router degli ISP sono connessi a coppie, utile per ISP molto estese;
- NAP: Neutral Access Point: i router hanno possono avere più di una connessione.

Gli ISP adottano una gerarchia a 3 livelli:

- ISP Tier 1: essi formano la dorsale di Internet, quindi tutti devono connettersi a loro per raggiungere il resto dell'Internet, non pagano quindi il transito;
- ISP Tier 2: detti ISP regionali data la loro copertura, si connettono a Tier 1, pagando il transito, e altri Tier 2;
- ISP Tier 3: detti anche di accesso o last hop network. Sono i primi ISP a cui si effettua normalmente l'accesso.

**– 1.2 – Modelli protocollari.**

Un protocollo definisce il formato e l'ordine dei messaggi scambiati tra più entità in comunicazione, così come le azioni in fase di trasmissione e/o ricezione.

**– 1.2.1 – Progettazione.**

Nella progettazione di un protocollo si devono garantire affidabilità e evolvibilità.

**– 1.2.1.1 – Affidabilità.**

L'affidabilità di un protocollo è definita da:

- Capacità di ripristino dagli errori o guasti tramite error detection e error correction;
- Assegnazione delle risorse in maniera dinamica in base alle necessità tramite statistical multiplexing;
- Controllo del flusso affinché non vi sia un'inondazione di informazioni;
- Congestione, dovuta al traffico eccessivo di dati che la rete non riesce a gestire, non garantendo la consegna dei pacchetti.

**– 1.2.1.2 – Evolvibilità.**

L'evolvibilità consiste in:

- Implementazioni sia incrementali che scalari tramite protocol layering;
- identificazione di chi trasmette e riceve un messaggio. Questa procedura è detta Addressing nei livelli bassi e Naming nei livelli alti.

**– 1.2.2 – Stratificazione.**

Una volta definito il sistema, si ricorre ad un modello di servizio per definire le operazioni possibili. Si individuano dei livelli, ognuno dei quali implementa uno specifico servizio tramite operazioni che utilizzano servizi dello stesso livello o di un livello inferiore. Ciò rende possibile identificare e relazionare elementi del sistema in modo da introdurre un modello di riferimento. Questa procedura modulare facilita: progettazione, gestione e aggiornamento del sistema, implementazione di ulteriori servizi in maniera trasparente a quelli già presenti.

**– 1.2.3 – Architettura di rete.**

La maggior parte delle reti sono organizzate a livelli, ognuno dei quali offre servizi diversi ai livelli superiori. Ovviamente i dati non sono trasferiti da un livello di una macchina allo stesso di un'altra in maniera diretta bensì passano attraverso i livelli sottostanti fino al primo livello, il supporto fisico, che mette in atto la vera comunicazione e trasferimento. Tra ciascun livello vi è un'interfaccia che definisce le operazioni e servizi offerti dal livello inferiore a quello superiore.

---

**Nota:** l'insieme dei protocolli usati da un sistema è detto protocol stack, che insieme ai livelli definisce la network architecture.

#### – 1.2.3.1 – Impacchettamento dati.

Come già detto i dati per essere trasmessi devono attraversare i livelli sottostanti. Durante questi passaggi vengono aggiunte delle informazioni. Tra esse vi è l'header, aggiunto durante il primo attraversamento, che contiene un identificativo e informazioni per la consegna.

Possono essere imposti limiti dai protocolli come per la dimensione che porta a dover suddividere il messaggio in pacchetti dotati di header e trailer. Una volta arrivati al destinatario il messaggio risale i vari livelli rimuovendo le informazioni aggiuntive e consegnando il messaggio.

#### – 1.2.4 – Connessione e affidabilità.

Ogni livello offre a quello superiore due tipi di servizi e vari livelli di affidabilità. I servizi di connessione vengono distinti tra orientati alla connessione e senza connessione.

- Orientato alla connessione: viene stabilita prima una connessione, durante la quale le due parti concordano i parametri da usare (dimensione, qualità ecc), e una volta usata la si rilascia. I pacchetti arrivano a destinazione nell'ordine in cui sono stati trasmessi;
- Senza connessione: anche detto datagram, ogni pacchetto ha l'indirizzo di destinazione ed è instradato in maniera indipendente dagli altri pacchetti. Nel caso in cui il messaggio deve passare per altri nodi prima di arrivare a destinazione, ogni nodo applicherà un approccio store & forward.

Per quanto riguarda l'affidabilità di un servizio si intende la capacità di non perdere dati e consegnarli nell'ordine giusto. Generalmente quindi si opta per un servizio orientato alla connessione con conferme di ricezione, che però possono appesantire il sistema e causare ritardi. Alcune tecniche sono:

- message sequence: vengono rispettati i confini dei messaggi;
- byte stream: la connessione è un flusso di byte senza alcuna divisione.

#### – 1.2.5 – Primitive di servizio.

Un servizio è formalmente descritto da un insieme di primitive, operazioni, che permettono di accedere al servizio. Questi set di primitive differiscono in base al tipo di servizio, con o senza connessione. Tra quelle conosciute vi sono:

- LISTEN: attesa bloccante di una connessione in arrivo, indica che un server è pronto ad accettare una connessione;
- CONNECT: stabilisce una connessione con un peer in attesa, va specificato a chi connettersi, il client manda un pacchetto e resta sospeso finché non riceve risposta;

- **ACCEPT**: accetta una richiesta di connessione da un peer, restituendo una risposta al client sbloccandolo;
- **RECEIVE**: attesa bloccante per un messaggio in arrivo, blocca il server in attesa di richieste;
- **SEND**: manda un messaggio dal peer al server e viceversa, sblocca il server per eseguire la richiesta o sblocca il client per esaminare la risposta;
- **DISCONNECT**: termine della connessione, è una chiamata bloccante che sospende il client e spedisce un pacchetto al server per rilasciare la connessione, il server eseguirà una **DISCONNECT** per confermare l'operazione.

**– 1.2.6 – Protocolli non proprietari.**

L'impossibilità a comunicare tra reti diverse fece sviluppare due pile protocollari per creare uno standard di comunicazione. Nacquero il TCP/IP e ISO/OSI.

**– 1.2.6.1 – ISO/OSI.**

L'ISO/OSI è una pila protocollare strutturata in 7 livelli, dall'alto 3 orientati al contenuto dell'informazione e 4 orientati alla rete. Si ha:

- Livello applicazione: si occupa della gestione della comunicazione delle applicazioni;
- Livello presentazione: si occupa della gestione della sintassi e codifica dei dati anche con codifiche diverse;
- Livello Sessione: si occupa di gestire le sessioni di utenti su macchine diverse, ovvero:
  - organizza il dialogo tra due terminali;
  - effettua il controllo del dialogo, uni o bidirezionale;
  - gestisce la sincronizzazione.
- Livello trasporto: si occupa di:
  - trasferimento dati da sorgente a destinatario;
  - gestione connessioni multiple;
  - gestione canale punto-punto o senza garanzia dei dati;
  - invio del messaggio a più destinazioni;
  - frammentazione flusso in frame;
  - correzione errori e prevenzione congestione.
- Livello rete: si occupa di:
  - instradare pacchetti in maniera dinamica o statica;
  - congestione e qualità servizio.
- Livello collegamento: datalink, si occupa di creare dei frame per la trasmissione tra elementi della rete locale e controllo degli errori e del flusso;
- Livello fisico: interfacce meccaniche ed elettriche per la trasmissione dei bit tramite canale di comunicazione.

Quindi questo modello è utilizzabile per la progettazione di un applicativo di rete.

**– 1.2.6.2 – TCP/IP.**

Lo standard TCP/IP è diviso in 5 livelli:

- Applicazione: per i servizi;
- Trasporto: per il trasferimento dati host-host e può essere TCP, connection oriented, o UDP, connectionless;
- Rete o Internet: per l'instradamento dei datagram e protocolli di routing;
- Data Link o Collegamento: per il trasferimento dei frame lungo i mezzi di comunicazione;
- Fisico: trasferimento dei bit per mezzi fisici.

NOTA: I livelli di presentazione e sessione del modello OSI vengono compensati dal sistema operativo del modello TCP.

**– 1.2.6.3 – Comunicazione.**

- Comunicazione fisica: i dati seguendo un percorso fisico scendono la pila protocollo e dal mezzo fisico vengono instradati al destinatario e risalgono fino al rispettivo livello di partenza;
- Comunicazione logica: ogni livello è distribuito in entità, ognuna delle quali implementano delle funzionalità, che scambiano messaggi solo con i loro pari.

**– 1.2.6.4 – Incapsulamento.**

L'incapsulamento è un processo attuato nel trasferimento dei dati. Dato il sistema di riferimento, ogni livello prende i dati dal livello superiore, aggiunge un header per creare una nuova unità dati (PDU, Protocol Data Unit) e la passa al livello sottostante.

**– 1.2.6.5 – Dati.**

I dati assumono nomi diversi in base al livello:

- Applicazione: messaggio o stream;
- Trasporto: segmenti o pacchetti;
- Rete: datagram;
- Collegamento: frame o trame.