

Livello di collegamento e Reti Locali

Introduzione

Collegamento: canale di comunicazione che connette nodi adiacenti sul percorso tra sorgente e destinazione. Possono essere di diversi tipi:

- Wired
- Wireless
- LANs

Una comunicazione tra nodi senza router è un **collegamento**.

Il pacchetto viene chiamato **frame** al livello 2.

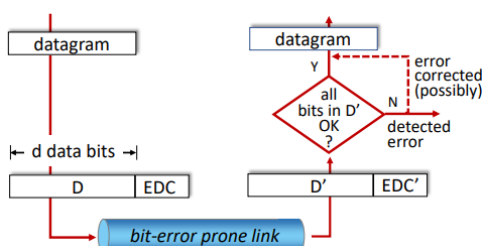
Diverse tipologie di protocolli di collegamento possono essere utilizzate sui vari link durante una comunicazione. La tipologia di protocollo utilizzato **dipende dal mezzo fisico adottato**. Si ha una grande quantità di protocolli e possono offrire diversi servizi tra di loro (affidabilità etc.).

Servizi

Ogni protocollo può offrire o meno un sottoinsieme dei servizi elencati.

- **Framing**: incapsulamento di un datagramma in una trama e aggiunta del header e trailer (coda aggiunta in fondo al datagramma);
- **Accesso al collegamento**: il collegamento può essere condiviso tra tanti;
- **Consegna affidabile tra nodi adiacenti**: simile alla consegna affidabile a livello di Trasporto. Perché ha senso? Si evita di sovraccaricare dei collegamenti per ritrasmissioni che non sono necessari.
- **Controllo di flusso**;
- **Rilevazione di errore**;
- **Correzione dell'errore** (codici che oltre a capire l'errore e sono in grado di correggere l'errore senza ritrasmettere);
- **Half-duplex e full duplex**.

Rilevazione e correzione dell'errore



ECD: Error Detection and Correction bits (bit aggiuntivi);

D: data protected by error checking (e.g. header fields).

È possibile aggiungere dei bit ridondanti di rilevazione e correzione di errore. Quando prendo il datagramma, aggiungo questi bit fino alla destinazione finale. Non sono al 100% affidabili, ma più aggiungo bit, più migliora la correzione dell'errore. Oltre ad una certa soglia, non posso correggere.

Più aggiungo i bit, lo svantaggio è che sto impiegando parte della trasmissione per trasmettere gli errori che possono rallentare la velocità di trasmissione.

Tipologie di collegamenti

1. **Collegamenti punto-punto:** collegamenti fisici tra nodi (host collegati punto-punto con switch).
2. **Broadcast:** se uno trasmette, tutti possono ricevere. Ovviamente i dati vengono letti e decifrati solo dalle persone interessate, ma tutti possono sapere che c'è trasmissione (bus, collegamento fisico ma effettua broadcast, Wi-Fi).

Problema: se è di tipo condiviso, può verificarsi trasmissioni simultanee e generazione di interferenza che prende nome di collisione. Questo accade quando i nodi non sono più in grado di riconoscere quali messaggi dovrebbero ricevere e quali no e la comunicazione non va a buon fine.

Esistono dei protocolli utilizzati per determinare quando i nodi possono trasmettere: **multiple access protocols**.

Multiple access protocols

Unico canale di trasmissione.

Interferenza: se ci sono due o più trasmissioni simultanee dai nodi;

Collisione: se un nodo riceve due o più segnali allo stesso tempo.

- **Channel partitioning**

Suddividiamo il canale in sotto canali, suddividendo la risorsa per la comunicazione tra ognuno dei vari nodi. Quando il nodo può utilizzare in canale, trasmette alla velocità massima (*Time slots*). Un'alternativa può essere la frequenza.

- **Random access**

Il canale non è diviso e ogni nodo o utente può trasmettere quando vuole, seguendo determinate regole. Esistono collisioni, ma ci sono metodi per recuperare dalle collisioni.

- **Taking turns**

I nodi hanno i turni, ma certi nodi che hanno più cose da trasmettere hanno turni più lunghi.

Random access protocols

Non si ha una coordinazione a priori. Se un nodo deve trasmettere, lo fa alla velocità massima consentita dal canale. Se due nodi decidono di trasmettere nello stesso momento, si ha una collisione.

I protocolli specificano:

- Come riconoscere le collisioni
- Come recuperare dalle collisioni

Principali: *ALOHA*, *slotted ALOHA*, *CSMA*, *CSMA/CD*, *CSMA/CA* (*non vediamo l'ultimo*).

Slotted ALOHA

- Trame tutte della **stessa dimensione**
- **Tempo suddiviso in slot**: Ogni singolo slot di tempo può essere acceduto da un qualsiasi nodo che vuole trasmettere; il tempo viene "slottato" in base al tempo per trasmettere un frame.
- Possono trasmettere all'inizio di ogni slot temporale;
- I nodi sono **sincronizzati**;
- Se due nodi trasmettono nello stesso time slot, **collisione**;

Come funziona?

1. Se si ha una trama da trasmettere, si cerca il primo slot di tempo disponibile e trasmettere;
2. Se non si verifica una trasmissione e ho ancora da trasmettere, il nodo trasmette nello slot di tempo immediatamente successivo;
3. Se si verifica una collisione, il nodo ritrasmette in frame successivi a quello dove si verifica una collisione con una probabilità p fino a quando non ha successo. Perché randomizzato (probabilità p)? Se io ritrasmettessi con una probabilità 100% per due nodi che sono sincronizzati, continuerò ad avere sempre una collisione. Con questa randomicità, si è sicuri che almeno uno dei due ritrasmette.

Pro:

- Se c'è un solo nodo che trasmette, velocità massima;
- Decentralizzato, ognuno fa come vuole;
- Semplice.

Cons:

- Quando ho delle collisioni, spreco degli slot;
- Quando non ho nessuno che trasmette, spreco degli slot;
- Devo avere un clock di sincronizzazione.

Perché slotto il tempo?

Efficienza: frazione di slot in cui è avvenuta una trasmissione di successo tra tutti gli slot tenendo conto di tutti gli slot.

ALOHA puro

Uguale ma senza slot.

Problema: posso aumentare il tempo in cui si verifica una collisione. Invece quando è slottato, si verifica quando due nodi sono sincronizzati con il clock e trasmettono nello stesso istante.

CSMA (Carrier Sense Multiple Access)

Va ad ascoltare il canale prima di andare a trasmettere, in modo tale da evitare le collisioni:

- Se il canale è idle: trasmetto tutto il frame;
- Se il canale è occupato: aspetto e trasmetto dopo.

CSMA/CD (CSMA con Collision Detection)

Se io mentre sto trasmettendo mi accorgo che c'è una collisione, la comunicazione viene abortita immediatamente così non spreco tempo.

Ma perché possono verificarsi comunque le collisioni? Il tempo di propagazione.

Se uno inizia a trasmettere ma il segnale dell'inizio della trasmissione non è ancora arrivato per il tempo di propagazione, si verifica una collisione. A causa della collisione, il tempo di trasmissione di un pacchetto è sprecato.

Con CSMA/CD posso rendermi conto che si verifica una collisione e mette in atto una procedura che fa abortire la sua comunicazione e spreco meno il canale. Invio di segnale di jam, segnale di disturbo per indicare che si è verificato una collisione. Riduco il tempo.

Nel caso di Ethernet:

1. Ricevo il datagramma
2. Si ascolta il canale:
 - a. Se libero: inizio la trasmissione
 - b. Se occupato: aspetto fino a quando non è libero e poi trasmetto;
3. Se trasmetto senza collisione, top.
4. Se un altro trasmette mentre trasmetto, abort e mando segnale di jam.
5. Nel momento in cui si verifica una collisione, i due nodi entrano nella fase di **binary exponential backoff**. Aspetto tanto più tempo per ritrasmettere quante le collisioni avvenute in passato (16 bit inizialmente). Dopo un tot di collisioni, si sceglie randomicamente un numero K e si moltiplica per 512 bit. Più collisioni, più la fase è lunga. Poi torno al passaggio 2.

$$Efficienza = \frac{1}{1 + 5 \frac{t(prop)}{t(trans)}}$$

- T_{prop} = massimo ritardo di propagazione tra due nodi;
- T_{trans} = tempo per trasmettere il frame massimo.

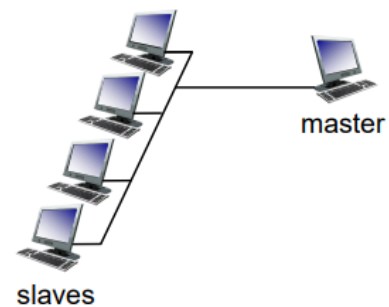
Migliore performance di ALOHA: semplice, economico e decentralizzato.

“Taking turns” access protocols

Protocolli ad accesso a rotazione. Evita che esistano delle collisioni andando a coordinare la comunicazione.

Polling

Usato dal Bluetooth, trasmettere i dispositivi collegati a turno. Nei protocolli di tipo polling abbiamo una gerarchia: **master** (smartphone) e **slaves** (cuffiette). Il master tramite un messaggio di polling interroga a turno i slaves e se uno ha bisogno di trasmettere dati, li trasmette e poi chiede al prossimo. È di tipo round robin e poi ricomincia.

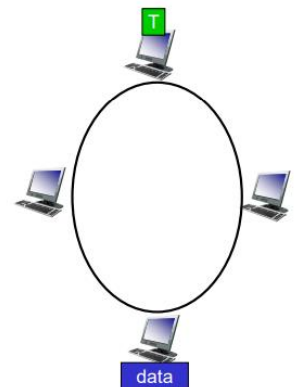


Svantaggi:

- **Overhead** dovuto al messaggio di polling;
- **Latenza** (polling agli altri nodi);
- **Single point of failure** (se master cade, tutto il sistema non funziona più).

Token passing

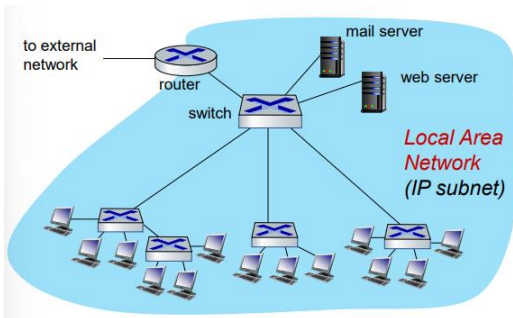
Messaggio di controllo chiamato token e tutti i nodi sono allo stesso livello della gerarchia e la comunicazione è ad anello (senso anti o orario, solitamente antiorario). Un nodo se ha il token può trasmettere, una volta finito inoltra il token a quello vicino (antiorario). Non ci sono collisioni.



Svantaggi:

- **Overhead** dal messaggio di token
- **Latenza** (giro di tutti i nodi)
- **Single point of failure** (se il token sperimenta corruzione, errore, si rompe tutto. Si può rigenerare il token, ma serve tempo).

Local Area Network



Le reti locali possono essere implementate da canali condivisi. Oggi ci sono gli switch che effettuano la commutazione tra nodi e i router ci permette il contatto esterno. Possono essere complicate.

All'interno di una rete locale i dispositivi parlano a livello 2. Utilizzano degli indirizzi a livello 2 chiamato indirizzo MAC, utilizzato per inviare le trame da un'interfaccia ad un'altra interfaccia della sottorete.

Indirizzo MAC

Indirizzo MAC	Indirizzo IP
Funzione: utilizzato localmente per spostare una trama da un'interfaccia ad un'altra nella stessa sottorete (forwarding livello 2).	Funzione: utilizzato per forwarding al livello 3 (livello di Rete). È l'indirizzo per un'interfaccia.
Lunghezza: 48-bit è immutabile per interfaccia (amministrato da IEEE).	Lunghezza: 32 bit. Non è immutabile, può essere assegnato in base a dove ti colleghi (esempio Bicocca).
Esempio: 1A-2F-BB-76-09-AD	Esempio: 128.119.40.136

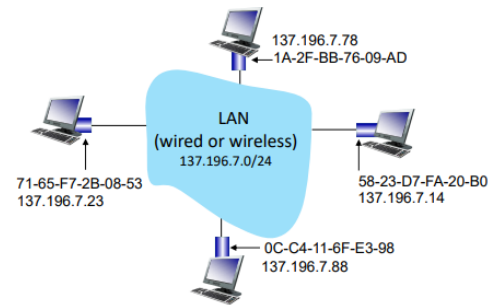
Base esadecimale. A differenza dell'indirizzo IP, il MAC è immutabile per interfaccia. Sono fisicamente collegati all'interfaccia. Il livello IP è più corto, utilizzato per inoltrare per livello 3 e tipicamente cambia perché dipende dalla sottorete alla quale siamo connessi. Tutte le interfacce della rete locali che parlano al livello 3, hanno sia IP che MAC. Gli switch non hanno associati IP e MAC perché non serve.

Gli indirizzi MAC vengono allocati dalla IEEE e si occupano di assegnare alle manifatture che devono comprare un pool di MAC.

Mi sposto da una rete all'altra, IP cambia ma MAC rimane.

In un'interfaccia LAN:

- Indirizzo univoco MAC da 48-bit;
- Indirizzo localmente univoco IP da 32-bit (indirizzo IP privato e pubblico);



ARP: Address Resolution Protocol

Come posso determinare un indirizzo MAC di un'interfaccia sapendo il suo indirizzo IP?

Tabella ARP: mantiene l'associazione indirizzo IP e MAC. Mantenuta da tutti gli host della rete locale e dal router che si connette alla rete locale. Abbiamo anche un TTL per una entry, dopo un tot la entry si elimina (circa 20 minuti).

Guardare le slide per capire il funzionamento di ARP.

Voglio inviare un datagramma ad un host che appartiene ad un'altra sottorete: tra A e B c'è un router di bordo.

- Tra le due reti locali c'è un router che permette la comunicazione;
- Voglio mandare un datagramma da A a B;
- Tutte le tabelle ARP sono vuote;
- A conosce l'indirizzo IP di B e l'indirizzo IP dell'interfaccia R che mi permette di uscire verso altre reti locali.

Ethernet

Semplice, prima tecnologia LAN. Velocità tra 10 Mbps – 400 Gbps.

Fino a metà degli anni 90 veniva implementata secondo una topologia fisica a bus. Si possono avere delle collisioni.

Oggi prevalgono le reti locali commutate in cui sono presenti switch che è il centro dei collegamenti. I nodi non possono avere delle collisioni tra di loro perché devono comunicare mediante lo switch.



Struttura della trama Ethernet

A livello 2 si possono avere anche delle code in fondo al payload.

HEADER:

1. **Preamble:** 8 byte di info che hanno sempre un pattern ben definito. Serviva per il bus: ascoltando il canale potevo capire se fosse una trama o un rumore. Serve anche per garantire il clock rate. Vedendo la velocità di arrivo di questi byte, il ricevitore può capire a che velocità stanno arrivando i pacchetti e sincronizzare la propria velocità su quello del trasmettitore.
2. **Indirizzi:** MAC del sorgente e destinatario. Manteniamo solo le trame per noi stessi oppure broadcast. Le altre vengono scartate.
3. **Type:** serve per multiplazione e demultiplazione. In passato venivano usati per altri protocolli (AppleTalk).

PAYLOAD:

Da 46 a 1500 byte. Ethernet ha pure una dimensione minima. Se i dati che devo trasmettere sono inferiori a 46 byte, vengono aggiunti dei bit di stuffing per arrivare fino a 46 byte che è la dimensione minima della mia trama.

TRAILER:

CRC: 4 byte. Simile ai checksum. Rilevazione di errore, in caso di errore viene scartato il frame.

Pro:

- Connectionless: non c'è bisogno di fare handshaking tra interfaccia mittente e destinatario.
- Inaffidabile (niente ACK NACK etc.)

CSMA/CD con Binary Backoff viene utilizzato da Ethernet per quando la struttura era ancora a bus.

Ethernet switch

Switch è un dispositivo che mi consente di inoltrare trame in modo corretto tra host in comunicazione. Dispositivo di livello 2. (Router fino al livello 3). Ha un ruolo attivo.

- **Store & forward** (prima di inoltrare una trama devo averla ricevuta completamente).
- Esamina l'intestazione di livello 2, per recuperare MAC dest ed effettuare l'inoltro a uno o più destinatari.
- L'esistenza degli switch è trasparente agli host e router. Gli host si connettono allo switch ma l'host non ha visibilità sul funzionamento dello switch.

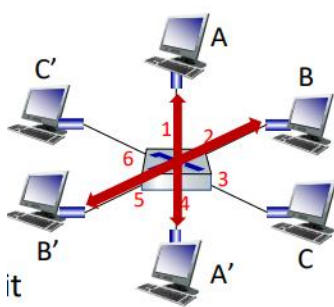
- Gli switch non hanno associati indirizzi MAC perché non gli serve.

Gli switch sono **plug and play**, senza effettuare una configurazione funziona. Questo è possibile per mezzo di un meccanismo self-learning. Uno switch ethernet è diverso da l'ethernet hub/repeater (dispositivo di livello 1 che ha più interfacce al quale possono connettersi gli host che prende il segnale ricevuto sull'interfaccia e amplificarlo e inoltrarlo sulle interfacce di uscita. Se più host sono collegati a questo hub, sarà il loro dominio di collisione).

Switch

Comunicazione full-duplex. Governa la comunicazione fra gli host e permette di avere più comunicazioni senza avere collisioni. Ovviamente non può permettere una comunicazione contemporanea allo stesso destinatario. Per questo avviene un accodamento dei pacchetti e sono quindi necessari dei buffer.

Attenzione allo switching



A-to-A' e B-to-B' possono trasmettere simultaneamente senza collisioni.

Una collisione avviene quando due nodi mandano simultaneamente allo stesso nodo.

Quindi se è attiva la comunicazione A-to-A', non può essere attivo C-to-A' simultaneamente.

Lo switch deve sapere come sono collegati i nodi per fare in modo che ci sia l'inoltro corretto. Come funziona?

Abbiamo una tabella (**switching-table**, come tabella di inoltro). Effettua inoltro su interfaccia corretta. Le entry sono: indirizzo MAC dest e interfaccia da raggiungere per quell'indirizzo. Non si hanno algoritmi di instradamento ma self-learning. Lo switch apprende autonomamente quali host possono raggiungere chi.

1. Self-learning

Lo switch impara quali host possono essere raggiunti tramite quali interfacce, riempiendo la switching-table.

Manda un pacchetto ad un'interfaccia, trova informazioni, inserisce nella tabella.

2. Frame forwarding

Quando lo switch riceve una trama, memorizza il link di invio e l'indirizzo MAC di sorgente del mittente.

Cerca di capire dove mandare la trama, quindi effettua una query alla switch-table tenendo conto della MAC destination.

Se esiste una entry per la destinazione:

1. Se il link di destinazione è uguale a quello del mittente, scarta;
2. Se trovo una entry e non siamo nel primo caso, manda a quello corretto;
3. Altrimenti flood: inoltra a tutte le interfacce tranne a quello dalla quale arriva la trama.

Switch interconnessi

Se io voglio inviare ad un host e la sorgente e dest sono collegati da due switch interconnessi tra di loro, si effettua il self-learning che funziona come nel caso dello switch singolo.

SWITCH	ROUTER
Livello 2 – Collegamento Commutatori Store & forward Self-learning per popolare switching-table	Livello 3 – Rete Commutatori Store & forward Algoritmi di instradamento.