



TCP/IP in pillole – Parte III

DARIO DE JUDICIBUS

Internet collega reti diverse tramite gateway e protocolli comuni, creando una rete virtuale basata su pacchetti. Ogni pacchetto contiene un indirizzo IP, che identifica rete e host di destinazione, mentre il DNS associa agli indirizzi nomi più semplici. Poiché hardware e sistemi operativi rappresentano i dati in modi diversi, TCP/IP adotta uno standard comune: gli interi sono trasmessi in formato Big Endian. Per collegare un indirizzo IP all'indirizzo fisico della rete interviene ARP, che usa richieste in broadcast e tabelle locali aggiornate dinamicamente; RARP svolge il compito inverso per macchine senza disco. L'architettura TCP/IP è organizzata in tre livelli: applicazioni, trasporto affidabile e consegna dei pacchetti. Alla base c'è IP, un servizio connectionless, non garantito e best-effort.

Terza parte del nostro breve corso sul TCP/IP. Nella prima parte abbiamo spiegato cos'è un sistema di reti interconnesse, od *internet*, abbiamo introdotto i concetti di protocollo ed interfaccia, abbiamo spiegato come una internet si basa su una serie di regole che formano appunto il TCP/IP e su alcune macchine dedicate come i *gateway* ed i *name server*. Nella seconda parte abbiamo introdotto il concetto di *host*, di indirizzo IP (*IP address*) e spiegato quali sono le convenzioni utilizzate per identificare univocamente ogni connessione ad una internet. In questa puntata cominceremo a parlare in dettaglio dei vari protocolli TCP/IP, ed in particolare dell'IP e del TCP.

Dato che quanto abbiamo descritto nelle prime due puntate è importante per capire quanto ci accingiamo a spiegare, facciamo qui un breve riassunto dei concetti base per coloro che non hanno avuto la possibilità di leggere i primi due articoli.

Innanzitutto una internet è un sistema di interconnessione fra reti differenti che utilizza sia dei sistemi dedicati per la connessione, detti gateway, sia uno strato (*layer*) di protocolli che mostrano alle applicazioni una visione omogenea di una rete virtuale e che sono basati sulla trasmissione di piccoli pacchetti di dati. Ogni pacchetto porta con sé l'indirizzo del destinatario il quale identifica univocamente sia la rete di destinazione che la connessione alla quale deve essere recapitato il pacchetto. Un sistema connesso a più reti della stessa internet avrà quindi più indirizzi IP. Un opportuno software, spesso installato su macchine dedicate, permette di associare ad ogni indirizzo un nome di più facile utilizzo da parte degli utenti del sistema. Il formato di questo nome si basa su un insieme di regole dette

DNS. Quella che è universalmente conosciuta come **Internet** è di fatto la principale rete interconnessa esistente, che si estende praticamente su tutto il pianeta.

Data questa premessa, vediamo di approfondire il discorso dei protocolli TCP/IP. Innanzi tutto qualunque trasferimento di dati implica la trasmissione di bit da un sistema ad un altro. Tali dati devono essere correttamente interpretati dai vari sistemi connessi alla rete. Data l'enorme varietà di hardware e di sistemi operativi questo è tutt'altro che banale. Nei protocolli di trasmissione i bit vengono convenzionalmente raggruppati per multipli di otto, detti *ottetti*. Una volta questo corrispondeva al bus da 8 bit, cioè un byte, tipico dei computer. Oggi la maggior parte dei computer usa *parole* di almeno 32 bit. Tuttavia non tutte le macchine memorizzano tali parole nello stesso modo. Esistono vari modi per memorizzare un intero rappresentato da 32 bit. In quello detto *Little Endian*, la posizione più bassa in memoria contiene il byte di ordine più basso dell'intero. Nei sistemi *Big Endian* avviene esattamente il contrario, cioè la posizione più bassa in memoria contiene il byte di ordine più elevato. In altri sistemi ancora il raggruppamento viene fatto con parole da 16 bit, in cui la parola meno significativa viene appunto prima. Il risultato è lo stesso del Little Endian ma con i byte invertiti all'interno di ogni singola parola (*Figura 1*). É evidente che non è pensabile che sia la rete a gestire tutti questi modi diversi di interpretare i dati, anche perché di solito i protocolli di trasmissione non entrano nel merito di come ragionano i singoli sistemi, ma si occupano solamente di trasferire in modo più o meno affidabile i dati a loro affidati. Ne consegue la necessità di definire un formato standard valido per tutti i dati che corrono lungo i collegamenti, lasciando ad i vari sistemi il compito di effettuare le opportune conversioni locali. Lo standard internet prevede che gli interi vengano trasmessi a partire dal byte più significativo, secondo lo stile del Big Endian. Così in un pacchetto, un intero ha il byte più significativo verso la testa del pacchetto e quello meno significativo verso la coda dello stesso.

Formati di memorizzazione dei dati

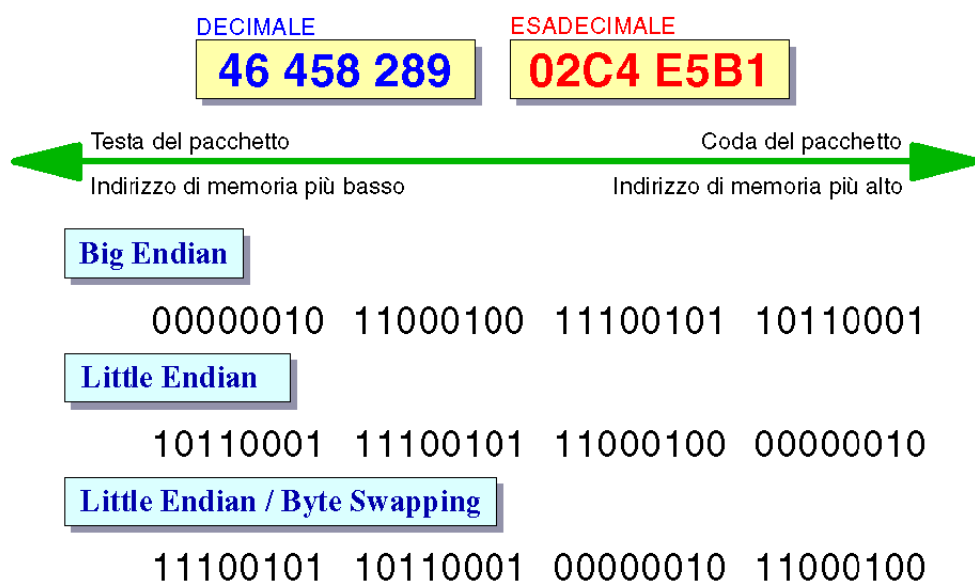


Figura 1 - Formati di memorizzazione dei dati

A questo punto i sistemi sono in grado di scambiarsi i dati in modo non equivoco. Ma come fa a sapere la rete internet che un sistema è collegato, e soprattutto, come avviene l'associazione tra l'IP address e l'indirizzo fisico di rete? Ogni rete fisica infatti ha un suo formato per gli indirizzi fisici assegnati alle connessioni di rete. In generale esistono due modi di assegnare indirizzi fisici alle macchine connesse in rete. In una rete piccola, come può essere una *Token Ring*, cioè un anello di un paio di centinaia di macchine al massimo, ad ogni connessione può essere assegnato un intero basso, ad esempio compreso tra **1** e **254**. Questo sistema ha il vantaggio di associare l'indirizzo fisico alla connessione piuttosto che alla scheda che permette la stessa. Per cui, se la scheda si rompe, l'utente può cambiarla senza dover tuttavia modificare l'indirizzo fisico di rete, purché imposti sulla nuova scheda lo stesso indirizzo di quella vecchia. Lo svantaggio è che non esiste alcun controllo che impedisca a due utenti sulla stessa rete di impostare lo stesso indirizzo fisico, creando così una *collisione*. In altri tipi di reti, come ad esempio *Ethernet*, ogni scheda ha già preimpostato da parte del costruttore un indirizzo fisico fisso, per cui non c'è alcun rischio di collisione, ma cambiare la scheda vuol dire dover necessariamente cambiare indirizzo fisico. Inoltre, dato che questo indirizzo è unico non solo fra le schede installate su una certa rete, ma in assoluto fra tutte le schede costruite, esso è generalmente molto lungo. Nel caso di Ethernet è di ben 48 bit.

Associare un IP address ad un sistema con indirizzi formati da piccoli numeri e per giunta tali che a parità di connessione l'indirizzo non cambia mai, come nel caso di una rete proNET-10, è molto semplice. Ad esempio, per un IP address di classe C, si può usare l'indirizzo fisico come *host identifier*. Così, se la rete ha IP address del tipo **10.214.32.x**, l'host con indirizzo fisico **16** avrà IP address **10.214.32.16**. Un altro paio di maniche è gestire indirizzi molto più lunghi dei 32 bit utilizzati per gli indirizzi internet, e per giunta che possono cambiare nel tempo a parità di connessione. Ovviamente si potrebbe tenere da qualche parte una tabella per gli accoppiamenti, e di fatto si fa così, ma non è certo molto pratico pensare che qualcuno la tenga aggiornata a mano. Il problema è stato risolto efficacemente utilizzando un meccanismo di risoluzione dinamica implementato dal protocollo **ARP**, o **Address Resolution Protocol**.

ARP funziona più o meno così. Quando un host deve spedire un pacchetto ad un certo destinatario, spedisce a tutti gli host nella stessa rete fisica un messaggio in cui chiede chi è l'host con quel ben preciso IP address. Nello stesso messaggio mette anche i propri indirizzi, sia quello fisico che quello IP. Si adopera cioè una tecnica di *broadcasting*. L'host il cui IP è quello cercato, rimanda indietro al richiedente il proprio indirizzo fisico, permettendo così l'associazione tra i due. Questo in quanto esso ha comunque ricevuto anche l'indirizzo fisico del mittente. Ma allora per ogni pacchetto che va spedito ad un certo IP address è necessario prima mandare un pacchetto a tutti gli host nella rete? E perché allora non mandare direttamente il pacchetto da trasmettere a tutti, invece di chiedere prima chi è che ha un certo indirizzo IP? Ovviamente la cosa non funziona così, anche perché si rischierebbe di appesantire inutilmente la rete con pacchetti che vengono

recapitati ai sistemi sbagliati. Quello che si fa è di mantenere presso ogni host una tabellina con tutti gli accoppiamenti già trovati, e di aggiornarla periodicamente per evitare che diventi obsoleta. A questo punto i meccanismi di broadcasting servono ad aggiornare tali tabelle (*Figura 2*). Ad esempio, se un host deve spedire un pacchetto ad un certo indirizzo IP, prima controlla nella sua tabella se non ha già l'indirizzo fisico del destinatario. Solo nel caso l'informazione manchi, l'host spedisce a tutti gli altri host il messaggio di richiesta. Quando questo arriva ad un qualunque host, sia esso il vero destinatario o no, ogni host aggiorna la sua tabella con l'indirizzo fisico e quello IP del mittente, tanto per guadagnare tempo. Il destinatario, in più, spedisce indietro anche il suo indirizzo fisico al mittente, così da potergli permettere di aggiornare la sua tabella di indirizzi. Un'ulteriore tecnica che si usa per assicurarsi che tali tabelle siano sempre aggiornate, è quella di far distribuire la propria coppia di indirizzi, fisico ed IP, ogni qual volta un sistema si connette alla rete, ad esempio al *reboot*.

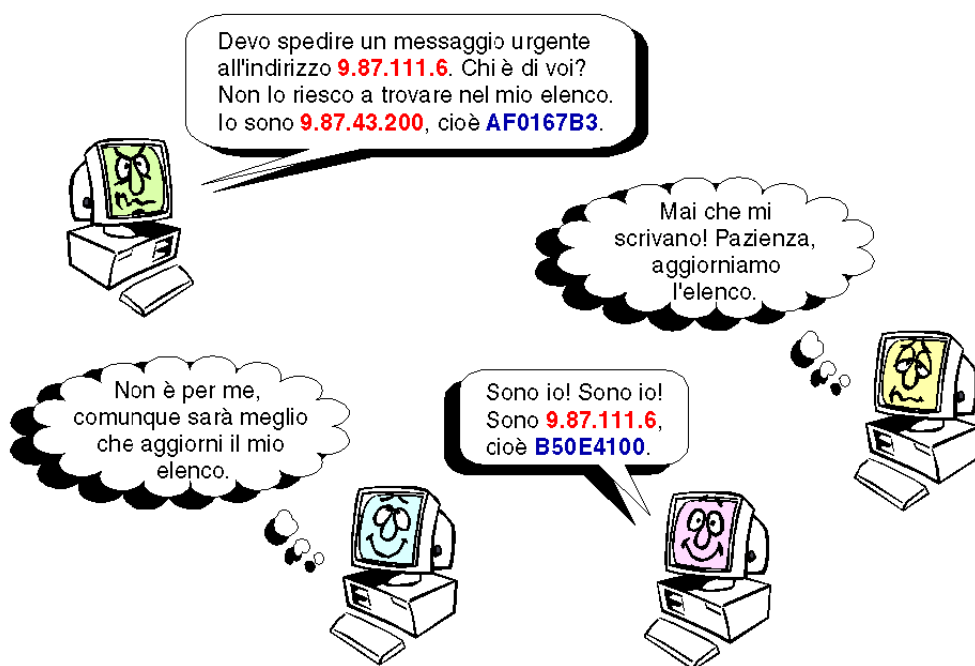


Figura 2 – Meccanismi di broadcasting

ARP non viene considerato propriamente un protocollo internet, quanto un meccanismo della rete fisica. É comunque un protocollo su cui si basa il protocollo IP per far comunicare fra loro le varie macchine quando non è possibile risolvere in altro modo gli indirizzi IP in indirizzi fisici. Un protocollo analogo è il **RARP**, o **Reverse Address Resolution Protocol**, con il quale una macchina *diskless* è in grado di conoscere il proprio indirizzo IP a partire da quello fisico. Per far ciò la rete deve avere uno o più *RARP server*, i quali contengono una tabella di associazione fra gli indirizzi IP e quelli fisici di tutte le macchine *diskless*. Anche questo protocollo si basa su un messaggio mandato in broadcasting. L'esistenza di questo protocollo è legata al fatto che una macchina *diskless* non può memorizzare il proprio indirizzo IP in alcun posto, non avendo memoria secondaria.

E veniamo ora al TCP/IP vero e proprio. Come detto nelle scorse puntate l'architettura internet è basata su tre livelli (*Figura 3*). L'**Application Services** è il livello più alto, cioè quello delle applicazioni. I programmi che utilizzate quando usate internet ricadono in questo livello. Il **Reliable Stream Transport Service** è il livello intermedio. Esso si occupa dell'affidabilità della comunicazione, gestendo gli errori di trasmissione e la perdita di eventuali dati. Esso inoltre fornisce una visione della comunicazione ad alto livello, in cui esiste una connessione tra i due host che si trasmettono grandi volumi di dati. Il livello più basso, chiamato **Connectionless Packet Delivery Service** è quello che effettua la spedizione vera e propria dei singoli pacchetti, senza garantire l'affidabilità sulla singola trasmissione, nella modalità detta *connectionless*.



Figura 3 – I livelli dell'architettura di internet

Il protocollo su cui si basa il livello più basso della torre internet è appunto l'**Internet Protocol**, od **IP**. Tale protocollo si basa su alcuni concetti fondamentali. Innanzi tutto il servizio che fornisce è detto *unreliable*, cioè inaffidabile, in quanto non dà alcun garanzia che il singolo pacchetto arrivi effettivamente a destinazione. In secondo luogo è detto *connectionless*, cioè senza connessione diretta, in quanto la trasmissione non avviene direttamente verso il destinatario, ma il messaggio è lanciato nella rete lasciando poi a questa il compito di portarlo a destinazione utilizzando l'indirizzo IP dell'host destinatario. Infine si parla di *best-effort delivery*, cioè spedizione al meglio delle possibilità, in quanto la rete fa tutto il possibile per portare comunque a destinazione il pacchetto. In pratica l'IP si comporta come un naufrago su un'isola deserta che lancia nella corrente un messaggio in una bottiglia per un tizio che si trova su di un'altra isola dello stesso arcipelago, contando sul fatto che se la bottiglia arriva sull'isola sbagliata qualcuno ributterà a mare il messaggio fintanto che non arriverà a destinazione. Detta così c'è quasi da stupirsi che internet funzioni così bene. Anzi, che funzioni del tutto! In realtà non dimentichiamoci che sopra al livello più basso ce n'è un altro che garantisce appunto l'affidabilità della comunicazione.

Torniamo comunque all'IP. Esso è formato da tre regole base: come è fatto il pacchetto da trasmettere, detto *IP datagram*, come avviene la scelta del cammino che il pacchetto segue per raggiungere il destinatario, come gli host ed i gateway devono trattare i pacchetti ed in particolare le modalità per l'emissione dei messaggi di errore e quelle per la soppressione dei pacchetti.

Nella prossima puntata prenderemo spunto dal formato del datagram per studiare in dettaglio il protocollo IP. Vedremo cioè come funziona realmente internet al livello più basso. Arrivederci alla prossima puntata.