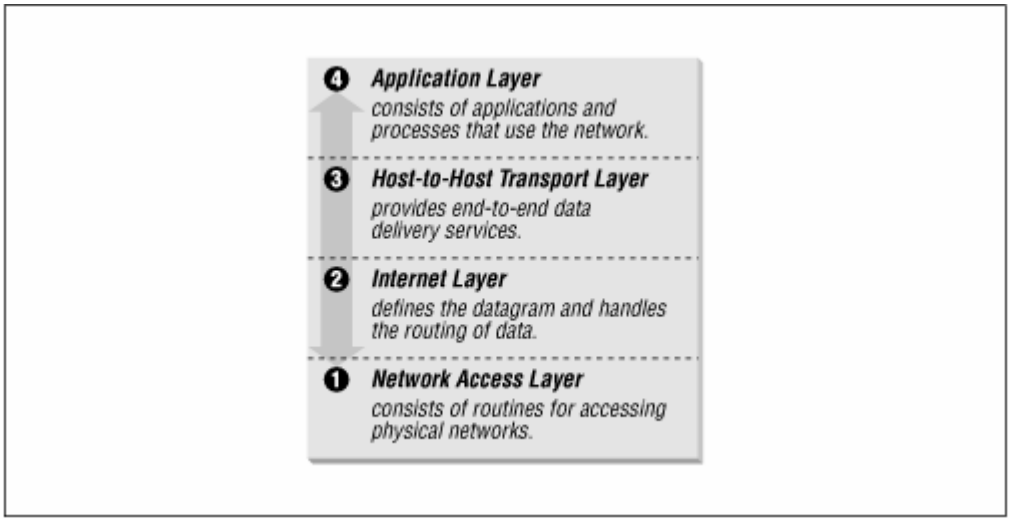
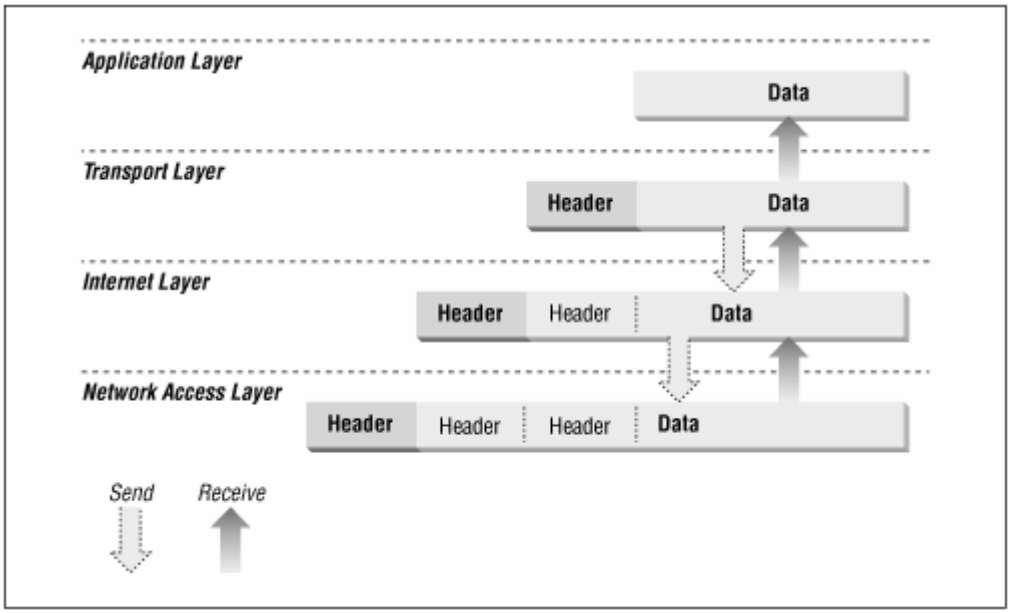
L’architettura di TCP/IP

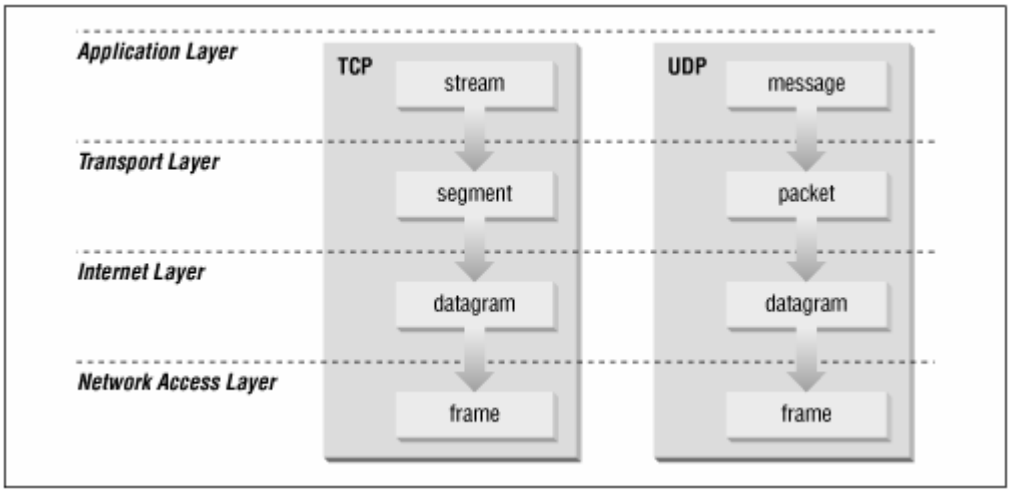
Mentre non esiste un accordo unanime su come descrivere il modello “a strati” di TCP/IP, è generalmente accettato il fatto che sia descritto da un numero di livelli inferiore ai 7 usati dal modello OSI. La maggior parte delle descrizioni del TCP/IP definiscono dai 3 ai 5 livelli funzionali. Nel nostro caso prenderemo in esame la seguente struttura:



Come nel modello OSI, in fase di invio, i dati percorrono dall’alto verso il basso la pila (stack), mentre in fase di ricezione nel senso contrario.
Ogni strato dello stack aggiunge delle informazioni di controllo per garantire la propria consegna. Questa informazione di controllo viene chiamata header (intestazione) perché è posta in testa ai dati trasmessi. Ogni strato tratta come dati tutte le informazioni ricevute dal layer soprastante e piazza il proprio header in testa a questi dati. L’aggiunta dell’header viene chiamata incapsulamento.
Quando i dati sono invece ricevuti, succede il contrario. Ogni layer elimina il proprio header prima di passare i dati al layer soprastante.



Come si può osservare dalla figura qui sotto: ogni layer ha il suo modo di chiamare il flusso di dati che lo attraversa.



Network Access Layer

Lo strato Rete è il più basso della gerarchia TCP/IP. I protocolli in questo layer forniscono le informazioni al sistema per trasmettere i dati agli altri dispositivi collegati alla rete. Esso definisce come utilizzare la rete per trasmettere un datagramma IP. A differenza degli strati superiori, il livello di Rete deve conoscere i dettagli della rete sottostante (la struttura dei suoi pacchetti, l'indirizzamento, ecc.ecc.) per formare dei dati consoni alle restrizioni della rete. I protocolli che agiscono in questo livello sono difficilmente conosciuti dagli utenti perché il design del TCP/IP nasconde il funzionamento dei livelli più bassi. Quando compare una nuova tecnologia hardware, devono essere sviluppati nuovi protocolli, in modo che le reti TCP/IP possano usare il nuovo hardware. Conseguentemente ci sono molti protocolli di accesso – uno per ogni standard di rete fisica.
Tra le funzioni di questo livello troviamo: l'incapsulamento dei datagrammi IP nei frame trasmessi dalla rete (fare riferimento alla RFC 894), e la mappatura degli indirizzi IP in indirizzi fisici usati dalla rete (MAC address) (RFC 826).

Internet Layer

Lo strato sopra quello di Rete si chiama Internet Layer. Il protocollo Internet (RFC 791) è il cuore del TCP/IP e il più importante dei protocolli che si trovano in questo strato. IP fornisce il servizio di consegna basilare sul quale le reti TCP/IP sono costruite. Tutti i protocolli che si trovano sopra e sotto l'IP, usano l'Internet Protocol per consegnare i dati: Tutti i dati TCP/IP viaggiano attraverso IP, in uscita e in entrata, a prescindere dalla loro destinazione finale.

Internet Protocol

L'Internet Protocol è il pilastro di Internet. Le sue funzioni includono:
Definire il datagramma, che è l'unita base della trasmissione in Internet

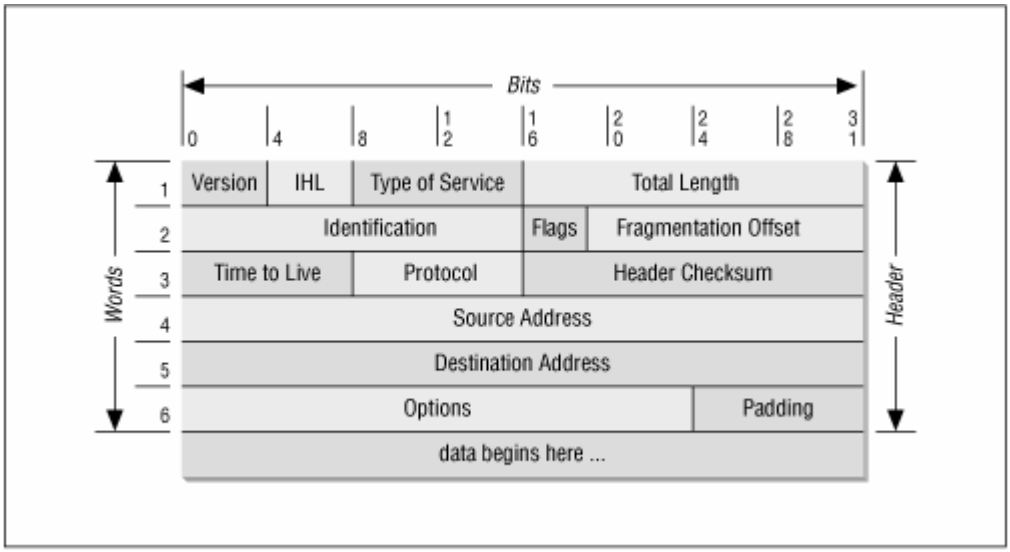
Definire lo schema di indirizzamento Internet
Trasportare i dati tra il Network layer e quello di Trasporto
Instradare (routing) i datagrammi agli host remoti
Frammentare e riassemblare i datagrammi

Alcune caratteristiche del protocollo IP:
E' senza connessione (connectionless): questo significa che IP non scambia informazioni di controllo (chiamato "handshake" (stretta di mano)) prima di stabilire una connessione punto-a-punto. L'IP si affida ad altri protocolli per stabilire la connessione (se è richiesto un servizio orientato alla connessione).
IP si affida in oltre ad altri protocolli per fornire un sistema di rilevazione degli errori e correzione degli errori.

The datagram

I protocolli TCP/IP sono stati costruiti per trasmettere dati attraverso ARPANET, che era una rete a commutazione di pacchetto (packet switching network). Un pacchetto è un blocco di dati che porta con sé l'informazione necessaria per essere spedito (proprio come una lettera postale). Una rete a commutazione di pacchetto usa le informazioni di indirizzamento contenute nei pacchetti, per commutare i pacchetti da una rete fisica all'altra, facendoli viaggiare sino alla loro destinazione finale. Ogni pacchetto attraversa la rete indipendentemente dagli altri pacchetti.
Il datagramma è il formato di pacchetto definito dall'Internet Protocol. Le prime cinque o sei word (parole) da 32-bit del datagramma sono informazioni di controllo chiamate header (intestazione). Per impostazione predefinita, l'intestazione è lunga cinque word; la sesta word è opzionale. Poiché la lunghezza dell'header è variabile, esso include un campo chiamato INTERNET HEADER LENGTH (IHL) che indica la lunghezza dell'intestazione. L'intestazione contiene tutte le informazioni necessarie per trasmettere il pacchetto.

IP datagram format

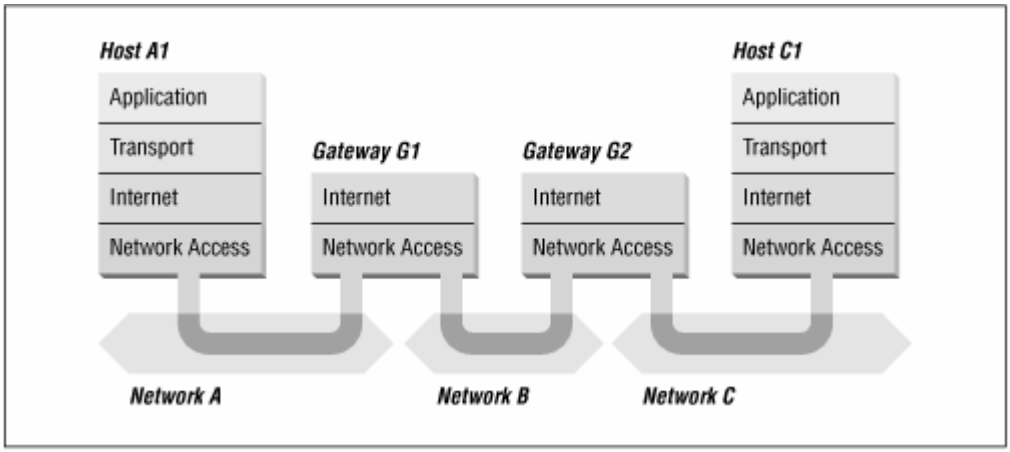


L'IP trasmette il datagramma cercando l'indirizzo di destinazione nella word 5 dell'intestazione. L'indirizzo di destinazione è un indirizzo standard a 32-bit (IP address) che identifica la rete di destinazione e l'host specifico della rete. Se la destinazione è l'indirizzo di un host che si trova nella rete locale, il pacchetto è consegnato direttamente a destinazione. Se l'indirizzo di destinazione si

trova invece in un’altra rete, il pacchetto è passato al gateway che provvederà a consegnarlo. I gateway sono dispositivi che commutano i pachetti tra reti fisiche differenti. La scelta di quale gateway utilizzare è chiamata routing. L’IP compie delle scelte di routing per ogni pacchetto.

Routing datagrams

I gateway Internet sono comunemente chiamati IP router poiché utilizzano l’Internet Protocol per instradare i pacchetti tra le reti. Nel TCP/IP JARGON tradizionale, ci sono solo due tipi di dispositivi di rete: i gateway e gli host. I gateway inoltrano i pacchetti tra le reti e gli host non lo fanno. Comunque, se un host è connesso a più di una rete (multi-homed host), può inoltrare pacchetti tra le reti. Quando un multi-homed host inoltra pacchetti, funziona esattamente come ogni altro gateway ed è dunque un gateway. In realtà ci sarebbero delle differenze tra gateway e router, ma noi useremo questi termini in maniera intercambiabile.



Fragmenting datagrams (frammentazione dei datagrammi)

Quando un datagramma è inoltrato attraverso reti differenti, potrebbe essere necessario per il modulo IP del gateway, dividere il datagramma in parti più piccole. Un datagramma ricevuto da una rete potrebbe essere troppo grande per essere trasmesso in un singolo pacchetto su una rete diversa. Questa condizione avviene solo quando un gateway interconnette reti fisiche diverse.
Ogni tipo di rete ha un’unità di trasmissione massima (MTU), che è il pacchetto più grande che può trasferire. Se il datagramma ricevuto da una rete è più grande del MTU dell’altra rete, è necessario dividere il datagramma in frammenti più piccoli per la trasmissione. Questo processo è chiamato frammentazione.

Il formato di ogni frammento è lo stesso di ogni normale datagramma. La word 2 dell’intestazione contiene informazioni che identificano ogni frammento di datagramma e fornisce informazioni su come riassemblare i frammenti nel datagramma originario. Il campo di identificazione (“Identification”) indica a quale datagramma il frammento appartiene, ed il campo “Fragmentation Offset” dice quale pezzo del datagramma è il frammento in questione. Il campo “Flags” possiede un “more fragments” bit che dice ad IP se ha riasseblato tutti i frammenti del datagramma.

Passing datagrams to the transport layer

Quando IP riceve un datagramma che è indirizzato al computer locale, deve passare la porzione di dati del datagramma al corretto protocollo di trasporto.Questa mansione viene svolta utilizzando il

“protocol number” ricavato dalla word 3 dell’intestazione del datagramma. Ogni protocollo del layer di trasporto ha un numero univoco che lo identifica.
Potete vedere da questa veloce panoramica che IP svolge molte importanti funzioni. Ad ogni modo, non aspettatevi da questa breve descrizione, di comprendere completamente i datagrammi, i gateway, il routing, gli indirizzi Ip, e tutto ciò che fa IP.

Internet Control Message Protocol

Una parte integrante di IP è l’Internet Control Message Protocol (ICMP) definito nella RFC 792. Questo protocollo è parte dello strato Internet e usa la facilità di trasporto dei datagrammi IP per spedire i suoi messaggi. ICMP spedisce messaggi che effettuano i seguenti controlli:

Controllo di flusso (flow control)
Quando i datagrammi arrivano troppo veloci per essere processati, l’host di destinazione o un gateway intermedio spedisce indietro al mittente un “ICMP source quench message”.
Questo dice di fermare temporaneamente l’invio dei datagrammi.

Identificare destinazioni irraggiungibili (Detecting unreachable destinations)
Quando una destinazione è irraggiungibile

Ridirezionamento dell’instradamento (*Redirecting routes)*
Un gateway spedisce un ICMP redirect message per dire all’host di usare un altro gateway, presumibilmente perché l’altro gateway è una scelta migliore. Questo messaggio può essere usato solo quando l’host sorgente è sulla stessa rete con entrambi i gateway.

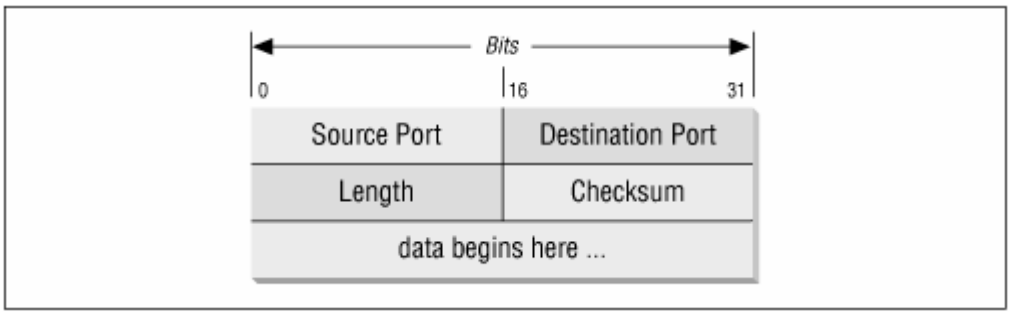
Verificare la presenza di un host remoto (Checking remote hosts)
Un host può spedire un ICMP Echo Message per vedere se l’Internet Protocol del sistema remoto è operativo e funzionante. Quando un sistema riceve un echo message, risponde e rispedisce al mittente il pacchetto di dati.

Transport Layer

Il protocollo che si trova sopra quello di trasporto si chiama *Host-to-Host Transport Layer*.
I due protocolli più importanti di questo strato sono TCP (transmission control protocol) e UDP (user datagram protocol). TCP fornisce un servizio di consegna dei dati affidabile. UDP fornisce un servizio di consegna di datagrammi senza connessione. Entrambi i protocolli trasmettono dati tra lo strato applicazione e quello Internet. I programmatori di applicazioni possono scegliere quale servizio è più appropriato per le loro applicazioni.

User Datagram Protocol

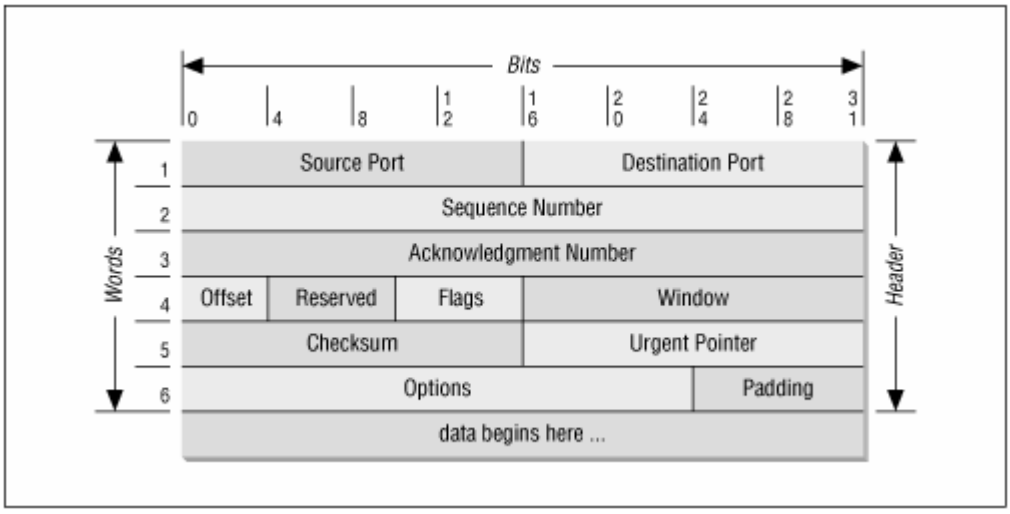
UDP fornisce ai programmi un accesso diretto al servizio di consegna dei datagrammi. Questo permette alle applicazioni di scambiare messaggi attraverso al rete che hanno un carico minore. Come già notato precedentemente, “inaffidabile” significa semplicemente che non ci sono tecniche nel protocollo per verificare che i dati abbiano raggiunto correttamente la destinazione. UDP utilizza i numeri da 16-bit: porta sorgente e porta di destinazione, nella word 1 dell’intestazione del messaggio, affinché i dati siano inviati correttamente al processo dell’applicazione.



Perché i programmatori di applicazioni dovrebbero scegliere UDP come servizio di trasporto dei dati? Ci sono un numero di buone ragioni. Se la quantità di dati trasmessi è piccola, l'overhead di creazione della connessione e la garanzia di consegna potrebbero essere più grandi del lavoro di ritrasmettere l'intero set di dati. In questo caso, UDP è la scelta più efficiente. Applicazioni che rientrano nel modello query-response (domanda-richiesta) sono altri candidati eccellenti per l'utilizzo di UDP.
La risposta può essere utilizzata come avvenuta ricezione di una query. Se una risposta non è ricevuta nell'arco di un certo periodo di tempo, l'applicazione semplicemente invia un'altra query.

Transmission Control Protocol

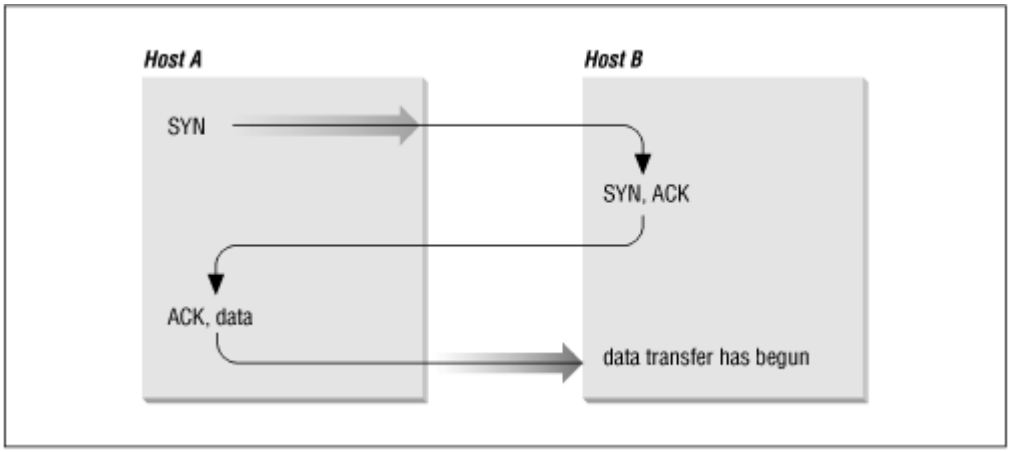
Applicazioni che richiedono una consegna affidabile dei dati usano TCP perché esso verifica accuratamente che i dati siano stati consegnati a destinazione nella giusta sequenza. TCP è un protocollo byte-stream, orientato alla connessione e affidabile.
TCP garantisce affidabilità grazie ad un meccanismo chiamato Positive Acknowledgment with Re-transmission (PAR). Un sistema che usa PAR spedisce nuovamente i dati, a meno che non sia avvisato dal sistema remoto che i dati sono arrivati correttamente. L'unità di dati scambiata tra i moduli TCP cooperanti , è chiamata segmento. Ogni segmento contiene un checksum che il destinatario utilizza per verificare che i dati non siano corrotti. Se il segmento di dati è ricevuto inalterato, il ricevente spedisce indietro al mittente un *positive acknowledgment.*
Se il segmento di dati è danneggiato, il ricevente lo scarta. Dopo un appropriato periodo di tempo, il modulo TCP di invio, ritrasmette ogni segmento per il quale non c'è stata una conferma di ricezione.



TCP è orientato alla connessione in quanto stabilisce una connessione logica punto-a-punto tra i due host comunicanti. L'informazione di controllo, chiamata handshake, è scambiata tra i due terminali

per stabilire un dialogo prima che i dati siano trasmessi. TCP indica la funzione di controllo di un segmento settando l'appropriato bit nel campo Flags nella word 4 dell'intestazione del segmento.
Il tipo di handshake usato da TCP è chiamato three-way handshake (stretta di mano a tre fasi) perché sono scambiati tre segmenti. L'host A comincia la connessione spedendo all'host B un segmento con il bit "Synchronize sequenze numbers" (SYN) impostato. Questo segmento dice all'host B che A vuole impostare una connessione, e indica a B quale numero di sequenza A utilizzerà come numero iniziale per i suoi segmenti. (I numeri di sequenza sono utilizzati per tenere i dati nella corretta sequenza). L'host B risponde ad A con un segmento cha ha il bit Acknowledgment (ACK) e il bit SYN impostati.
Il segmento di B conferma la ricevuta del segmento di A, ed informa A sul numero di sequenza con cui comincerà B. Finalmente, l'host A spedisce a B un segmento che fa intendere di aver ricevuto la ricevuta del segmento, e trasferisce per primo i dati veri e propri.

Three-way handshake

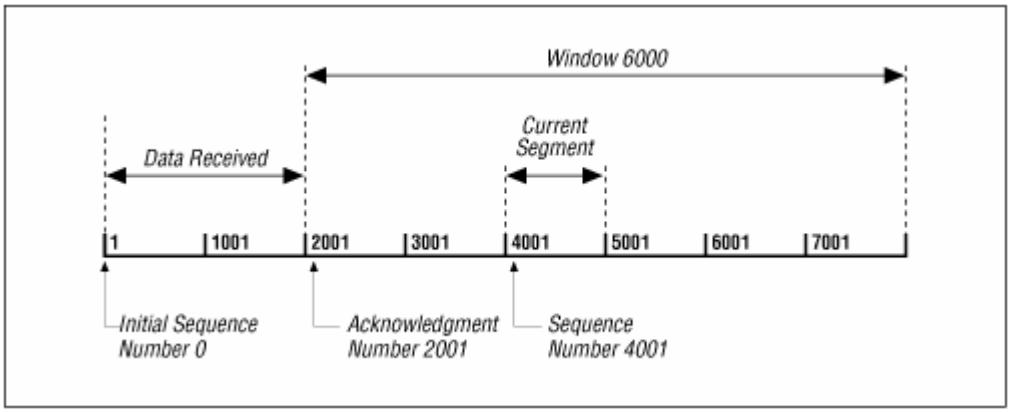


Dopo questo scambio, l'host A possiede una prova evidente che l'host B è pronto per ricevere i dati. Non appena la connessione è stata stabilita, i dati possono essere trasmessi. Quando i moduli cooperanti hanno concluso il trasferimento dei dati, scambieranno un three-way handshake con segmenti contenenti il "No more data from sender" (fine della trasmissione di dati, FIN bit) per chiudere la connessione.

TCP vede i dati che spedisce come un continuo flusso di byte, non come pacchetti indipendenti. Pertanto, TCP si prende cura di mantenere la sequenza con cui i byte sono spediti e ricevuti.
I campi sequence number e Acknowledgment Number che si trovano nell'intestazione del segmento TCP tengono traccia di questi byte.
Lo standard TCP non richiede che ogni sistema cominci a numerare i byte con un numero specifico; ogni sistema sceglie il numero che userà come punto di partenza. Per tenere traccia del flusso di dati, ogni terminale deve conoscere il numero iniziale dell'altro host.
I computer comunicanti sincronizzano i sistemi di numerazione dei byte scambiandosi segmenti SYN durante l'handshake. Il campo "Sequence Number" nel segmento SYN contiene il numero di sequenza iniziale (ISN), che è il punto di partenza per il sistema di numerazione dei byte. Per questioni di sicurezza, l'ISN dovrebbe essere un numero casuale, anche se spesso è 0.
Ogni byte di dati è numerato in sequenza a partire dall'ISN, così il primo byte di dati veri e propri, ha il numero di sequenza ISN+1. Il numero di sequenza nell'intestazione di un segmento di dati identifica la posizione sequenziale nel flusso di dati del primo byte di dati nel segmento.
Per esempio, se il primo byte nel flusso di dati avesse il sequence number 1 (ISN=0) e 4000 byte di dati fosserò già stati trasferiti, allora il primo byte di dati nel corrente segmento sarebbe il byte 4001, e il numero di sequenza sarebbe 4001.

Il segmento Acknowledgment esegue due funzioni:
positive acknowledgment e controllo di flusso.
La prima dice al mittente quanti dati sono stati ricevuti, e quanti ancora il ricevente ne può accettare.
Il numero Acknowledgment è la sequenza numerica del prossimo byte che il ricevente si aspetta di ottenere. Lo standard non richiede un acknowledgment individuale per ogni pacchetto.
L'acknowledgment number è un riconoscimento positivo di tutti i byte con quel numero. Per esempio, se il primo byte spedito fosse numerato 1 e 2000 byte fossero stati ricevuti con successo, il numero l' acknowledgment number sarebbe 2001.

Il campo Window (finestra) contiene il numero di byte che il terminale remoto è in grado di accettare. Se il ricevente è in grado di ricevere 6000 o più byte, la finestra sarà di 6000. La finestra indica al mittente che può continuare a spedire segmenti sin quando il numero totale di byte che spedisce è più piccolo della window di byte che il ricevente può accettare. Il ricevente controlla il flusso di byte provenienti dal mittente, cambiando la dimensione della finestra. Una finestra settata a 0 dice al mittente di cessare la trasmissione sino a che non riceve un valore diverso da 0.
La figura qui sotto mostra un flusso di dati TCP che comincia con un numero di sequenza iniziale di 0. Il sistema del ricevente ha ottenuto e riconosciuto 2000 byte, così il numero di riconoscimento è 2001. Anche il ricevente possiede uno spazio di buffer per altri 6000 byte, così esso annuncia una finestra di 6000. Il mittente sta spedendo un segmento di 1000 byte che cominciano con il sequence numer 4001. Il mittente non ha ricevuto nessun riconoscimento per i byte dal 2001 in su, ma continua a spedire dati sino a che riesce a rimanere dentro la finestra. Se il mittente riempie la finestra e non riceve nessun riconoscimento dei dati precedentemente spediti, esso, dopo un appropriato time-out, li spedirà ancora cominciando dal primo byte di riconoscimento.



Nella figura la ri-trasmissione comincerebbe dal byte 2001 se nessun ulteriore riconoscimento fosse ricevuto.

TCP è anche responsabile della consegna dei dati ricevuti dall'IP alla corretta applicazione.
L'applicazione a cui sono associati i dati è identificata da un numero a 16 bit chiamato numero di porta. La porta sorgente e la porta di destinazione sono contenute nella prima word dell'intestazione del segmento. Trasferire correttamente i dati dallo e allo strato applicazione è una parte essenziale del lavoro svolto dal Transport Layer.

Application Layer

Sulla cima dell'architettura del TCP/IP troviamo lo strato Applicazione (Application Layer). Questo strato include tutti i processi che usano i protocolli del Transport Layer per consegnare i dati.

Gli applications protocols più conosciuti sono:

telnet : permette il login su una macchina remota

FTP: file transfer protocol, è usato per il trasferimento di file interattivo.

SMTP: simple mail transfer protocol, che consegna la posta elettronica

HTTP: hyper transfer protocol, che trasmette pagine web attraverso la rete

Alcuni protocolli, come telnet e FTP, possono essere usati se l'utente ha qualche conoscenza delle reti. Altri invece, come OSPF, funzionano anche se l'utente non conosce la loro esistenza.