



La capa de Transport

LA PILA OSI

Nivel de Aplicación
Servicios de red a aplicaciones

Nivel de Presentación
Representación de los datos

Nivel de Sesión
Comunicación entre dispositivos de la red

Nivel de Transporte
Conexión extremo-a-extremo y confiabilidad de los datos

Nivel de Red
Determinación de ruta e IP (Direccionamiento lógico)

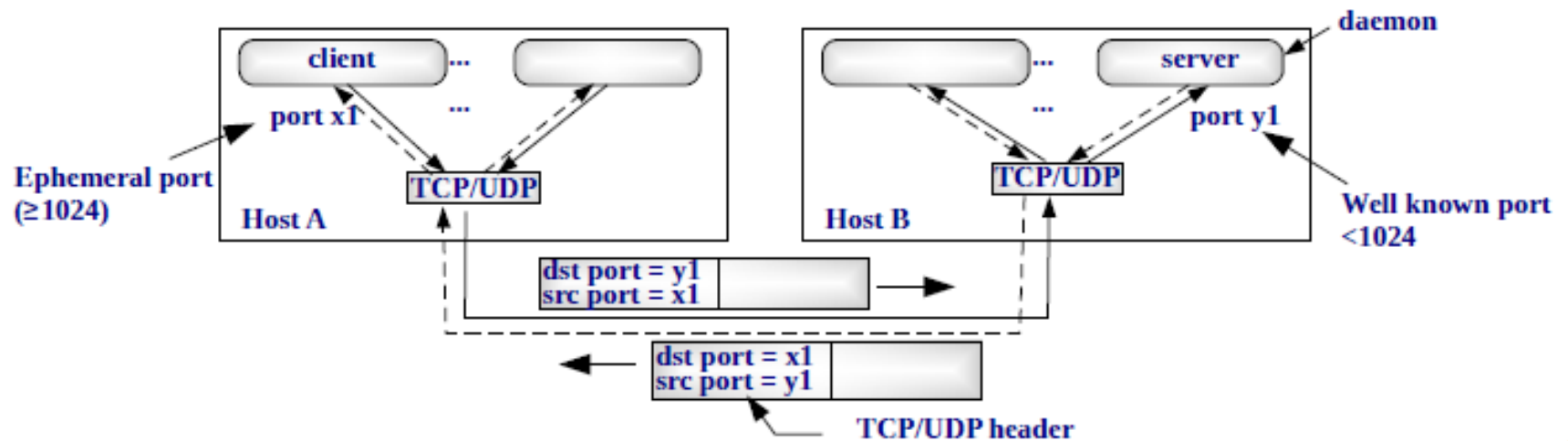
Nivel de Enlace de Datos
Direccionamiento físico (MAC y LLC)

Nivel Físico
Señal y transmisión binaria

Introducció

- Dos són els protocols que es fan servir bàsicament a la capa de transport:
 - User Datagram Protocol. Servei datagrama. No fiable
 - Transport Control Protocol. Servei fiable. QoS
- La capa de transport ofereix comunicació entre aplicacions extrem a extrem.
- Els punts d'accés de la capa de Transport estan identificats per un número de port de 16 bits.

Introducció



Stack TCP/IP

TELNET, FTP, SMTP		DNS, RIP, SNMP	
TCP		UDP	
ARP	IP		IGMP ICMP
ETHERNET	TOKEN RING	FRAME RELAY	ATM

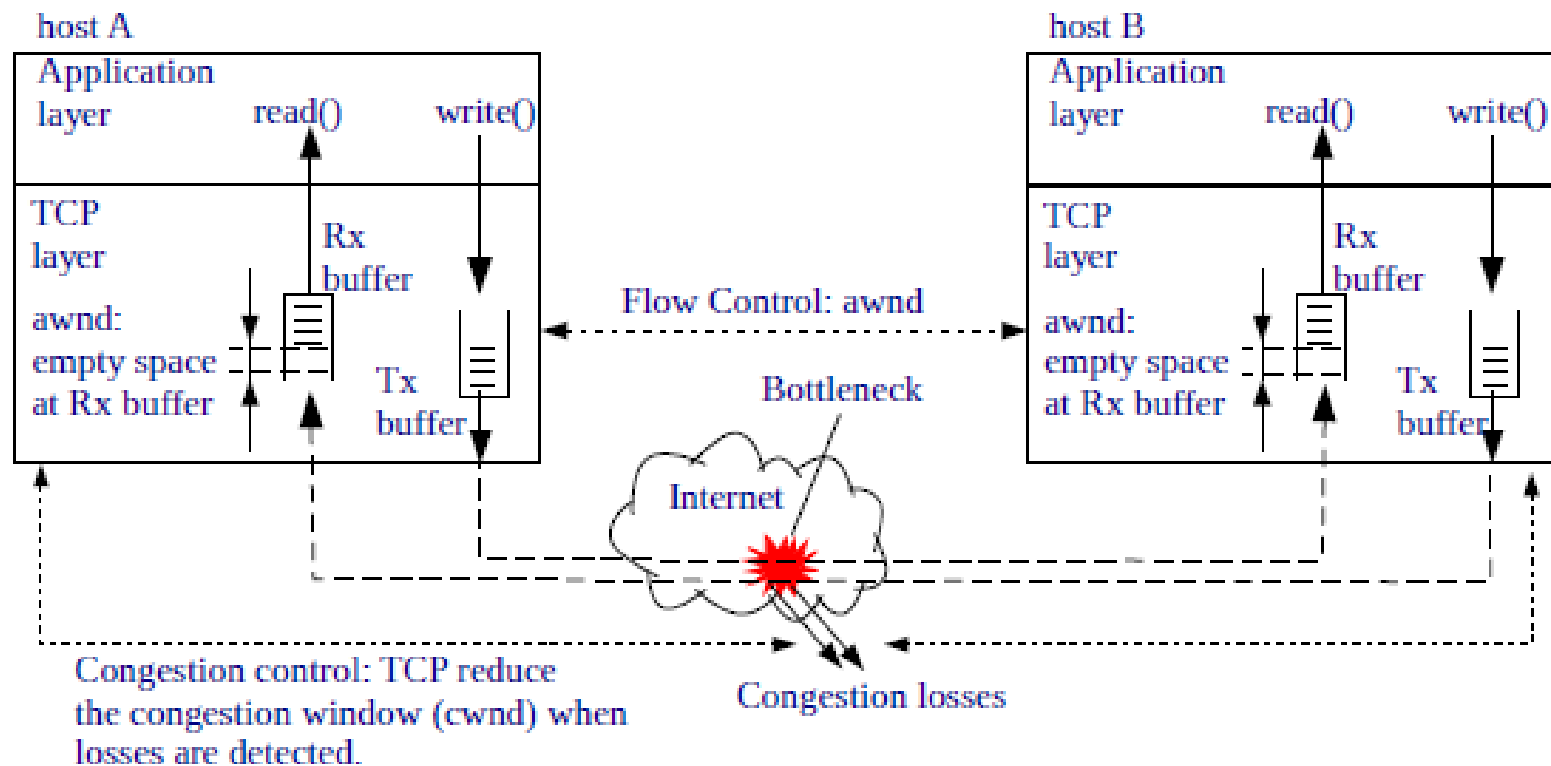
TCP: Definicions i conceptes

- El protocol de transport (TCP) es defineix en la RFC 793
- És un servei fiable d'entrega de paquets orientats a la connexió. Garanteix l'entrega de paquets IP
 - Error Recovery
 - ACKs
 - Orientat a connexió
 - Control de Flux
- TCP segmenta i torna a unir els grans blocs de dades fent servir números de seqüència. Mida màxima de segment variable (MSS)
- Opera en full dúplex

TCP – Operacions bàsiques

- ARQ implementat amb mida de finestra variable
- Cada segment de TCP rebut implica enviar un ACK sense esperar la lectura per part de la capa superior.
- Implementa dues finestres:
 - Advertised Window (awnd) utilitzada per control de flux
 - Congestion Window (cwnd) utilitzada per control de congestió
 - La mida de la finestra per ARQ es calcula fent:
$$\text{wnd} = \min(\text{awnd}, \text{cwnd})$$

TCP – Operacions bàsiques



TCP. Format i característiques

➤ Format de TCP

- Transporta i regula el flux d'informació per garantir la connectivitat d'extrem a extrem entre aplicacions de host, que s'estan comunicant en la xarxa de forma fiable, eficient i precisa.
- QoS, fiable
- Protocol orientat a la connexió
- És un protocol de flux no estructurat (bytes de control i dades)
- Protocol amb transferència de memòria intermitja
- Fa servir connexions full-dúplex

TCP. Format i característiques

Segment TCP

0										10										20										30	
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Port TCP origen															Port TCP destí																
Número de seqüència																															
Número rebut																															
HLEN			Reservat					Indicadors							Finestra																
Checksum TCP															Marcador urgent																
Opcions																									Padding						
dades																															
...																															

TCP. Format i característiques

- Port origen: 16 bits. Número del port origen
- Port destí: 16 bits. Número del port destí
- Num. Seqüència: 32 bits. El número de seqüència del primer octet de dades d'aquest segment, excepte quan l'indicador de SYN el tenim a 1. Si SYN està a 1 correspon al número de seqüència original i llavors, el primer byte de dades és ISN+1 (ISN: Initial Sequence Number)

TCP. Format i característiques

- Número rebut: 32 bits. Si el bit de control de ACK és 1, aquest camp conté el valor del següent número de seqüència que l'emissor del segment espera rebre. Un cop una connexió està establerta, aquest número s'envia sempre.
- HLEN: 4 bits. Posició de les dades. Aquest número indica on comencen les dades. La capçalera TCP sempre té un número sencer de paraules de 32 bits.
- Reservat: 6 bits. Val 0

TCP. Format i característiques

- Indicadors. 6 bits de control:
 - URG: Fa referència al camp que indica “marcador urgent”
 - ACK: Fa referència al camp : Num rebut
 - PSH: Funció de “Entregar dades immediatament” (push)
 - RST: Reset de la connexió
 - SYN: Sincronitzar els números de seqüència.
 - FIN: Darreres dades de l'emissor

TCP. Format i característiques

- Finestra: 16 bits. Nombre de dades que l'emissor del segment està disposat a acceptar.
- CRC: 16 bits. Complement a 1 de 16 bits de la suma dels complements a 1 de totes les paraules de 16 bits de la capçalera i text. L'últim byte s'omple amb 0 a la dreta per formar una paraula de 16 bits complerta i poder així fer la suma de control. En aquest càlcul el propi camp de CRC es considera omplert per 0s. Aquesta suma també inclou una pseudocapçalera de 96 bits prefixada a la capçalera TCP. Això proporciona protecció contra segments mal encaminats.

TCP. Format i característiques

- Marcador urgent: 16 bits. Consisteix en un desplaçament positiu des del número de seqüència fins a on tindrem les dades urgents. Només s'interpreta si el bit de URG està marcat a 1.
- Opcions: Variable, però sempre múltiple de 8 bits.

TCP – Flags TCP

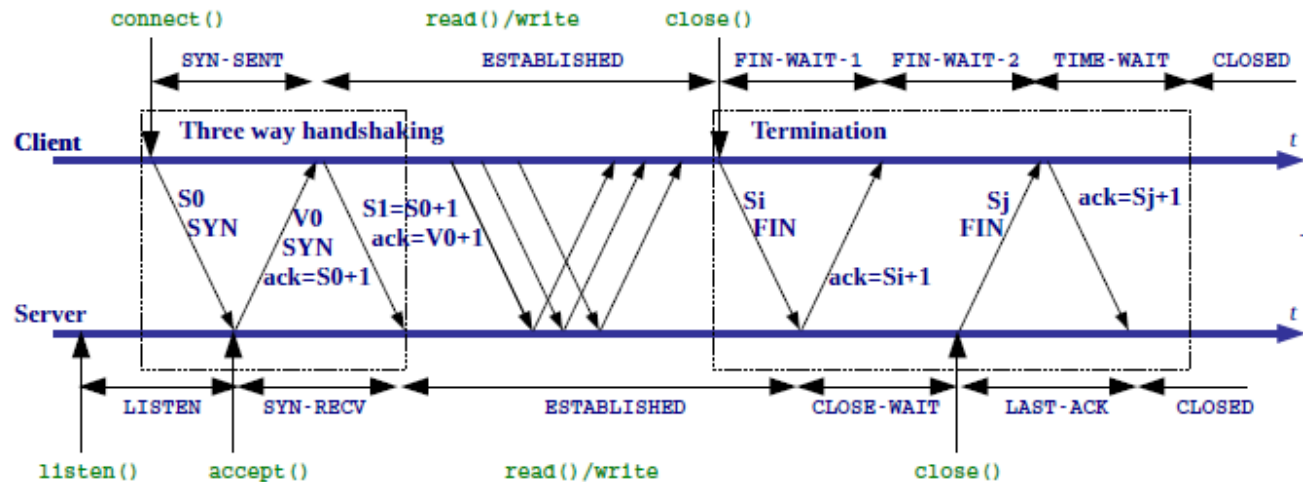
- Com hem comentat TCP implementa els següents flags:
 - URG (Urgent) -> Puntero a dades urgents. Apunta al primer (inicialment) byte urgent. No es fa servir gaire. Exemple CTRL + C en una sessió TELNET
 - ACK. Utilitzat sempre excepte en el primer segment enviat pel client.
 - PSH (Push). Qui envia indica “to push” les dades que estan al buffer a l’aplicació del receptor. El flag PSH s’activa quan el buffer està buit.
 - RST. Reset. Tallem la connexió
 - SYN (Utilitzat en el mecanisme de connexió)
 - FIN. Termina la connexió

Establiment de la connexió TCP

- Establiment de la connexió a nivell TCP
 - El receptor espera la connexió executant les primitives LISTEN i ACCEPT
 - El host que desitja iniciar la connexió executa la primitiva CONNECT, especificant IP i port on es vol connectar
 - CONNECT fa una apertura activa, enviant l'altre host un paquet que té el bit de SYN activat i on s'indica el nombre de seqüència inicial 'x' que farà servir per enviar els missatges.

➤ Establiment de connexió i terminació

- El client sempre envia el primer segment.
- El triple handshaking no té payload
- SYN i FIN tenen número de seqüència
- El nombre de seq. Inicial és aleatori.



Establiment de la connexió TCP

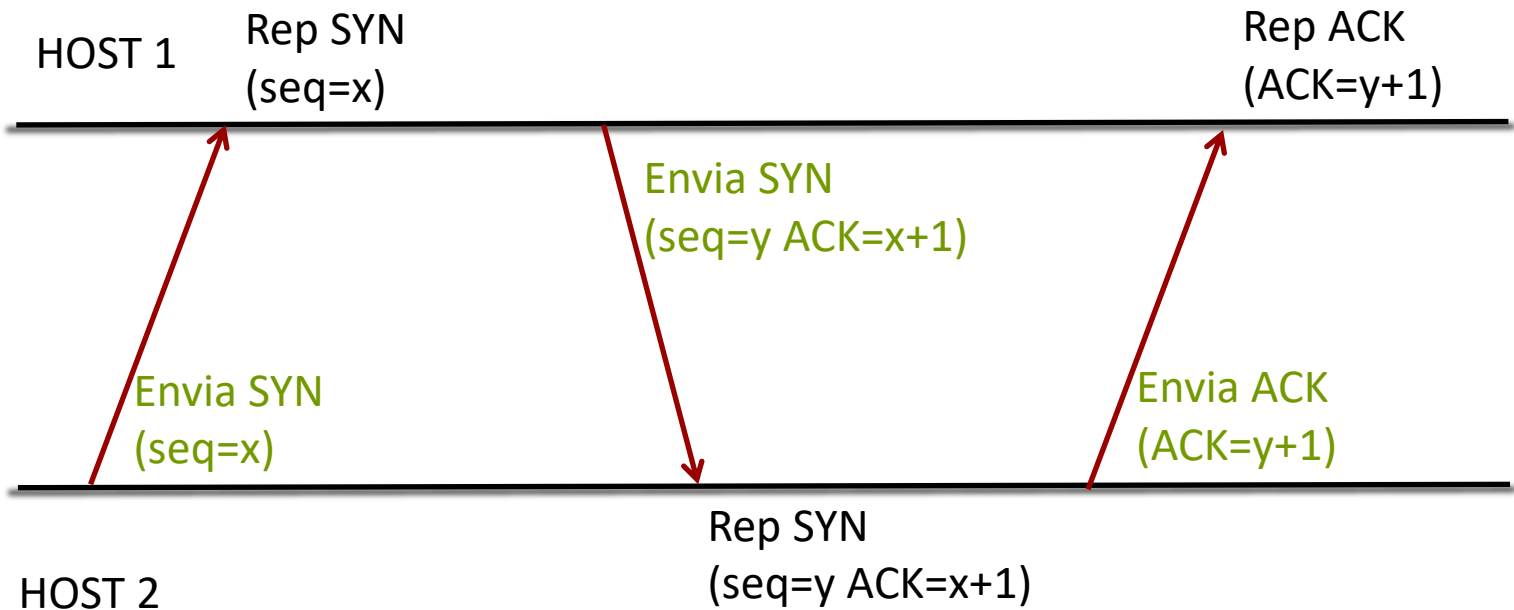
- El host receptor rep el segment i revisa si hi ha algun procés actiu que hagi executat un LISTEN en el port sol·licitat. Si existeix, el procés que rep el missatge rep el segment TCP entrant, registra el número de seqüència 'x' i, si desitja obrir la connexió, respon amb un ACK 'x+1' i el bit de SYN activat. Inclou el seu número de seqüència inicial 'y' deixant oberta la connexió al seu extrem. El número d'espera 'x+1' significa que ha rebut fins al número 'x' i espera 'x+1'. Si no desitja establir la connexió envia una resposta amb el bit RST activat.

Establiment de la connexió TCP

- El primer host rep el segment i envia la confirmació. A partir d'aquest moment pot enviar dades des del seu extrem.
- El segon host rep la confirmació i entén que l'altre host ja ha obert la seva connexió

CONNEXIÓ ESTABLERTA!!

Establiment de la connexió TCP



ESTABLIMENT CONNEXIÓ TCP
(EXTREM A EXTREM)

Tancament connexió TCP

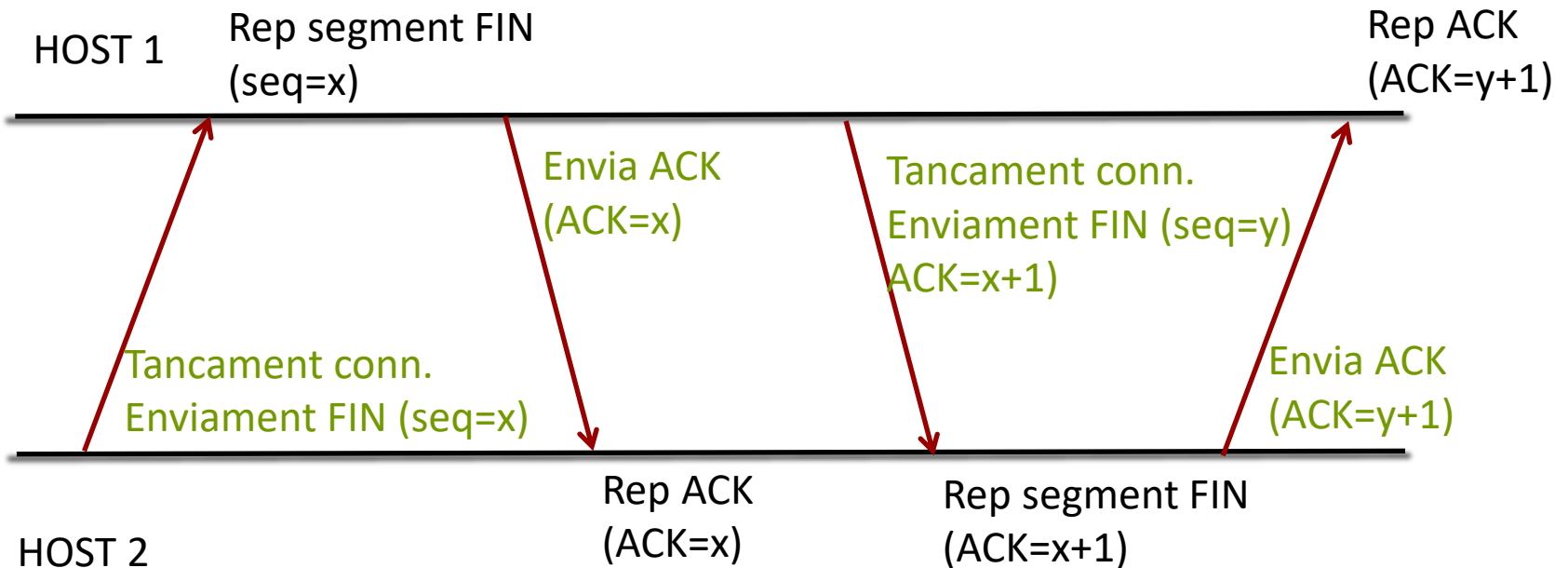
➤ Tancament de la connexió

- La màquina emissora, quan no té més dades per transmetre, envia un segment TCP amb el bit FIN activat i tanca el sentit d'enviament en la connexió, mantenint obert el sentit de recepció
- El host receptor rep el segment amb el bit de FIN activat, retorna la confirmació i informa a l'aplicació receptora de la petició de tancament. És aquesta la que decideix el tancament de l'altre sentit de la connexió. Mentre, envia un segment ACK

Tancament de la connexió TCP

- Quan la màquina receptora decideix tancar la comunicació envia un segment TCP amb el bit de FIN activat
- La primera màquina rep aquest segment i envia el corresponent ACK (tot i que ha tancat el seu sentit de connexió segueix enviant confirmacions)
- Quan la màquina receptora rep el segment ACK tanca definitivament la connexió

Tancament de la connexió TCP



ESTABLIMENT CONNEXIÓ TCP
(EXTREM A EXTREM)

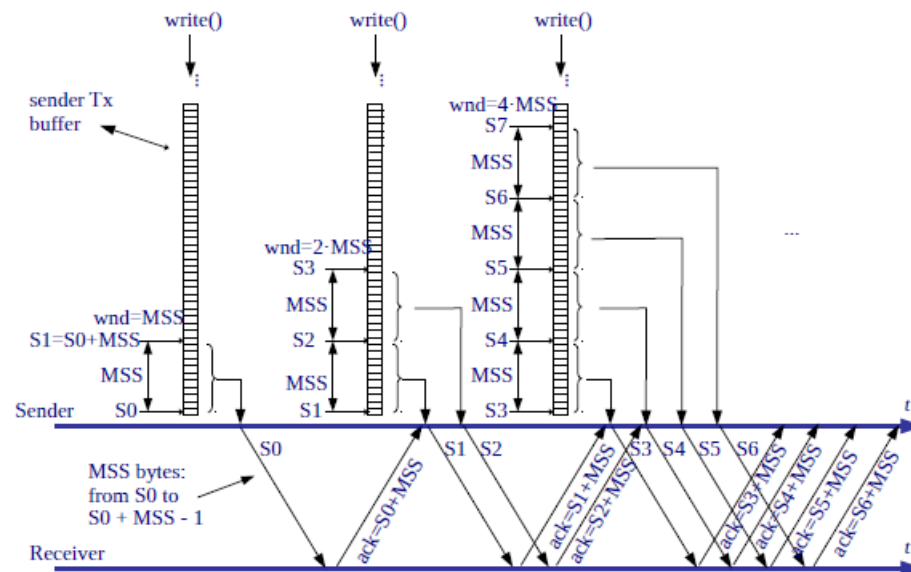
Control d'errors en TCP

- Detecció d'errors:
 - Bit de paritat: 1->num. '1s' senar, 0-> num '1s' parell
 - Checksum: Complement a 1 de la suma en mòdul $2^{16}+1$ del segment. Inclou també la capçalera IP
 - CRC: comprovació de redundància cíclica
- Confirmació positiva i expiració per interval de temps. L'emissor espera un temps i si no rep un ACK torna a enviar la trama
- Confirmació negativa i transmissió: Només per demanar reenviament

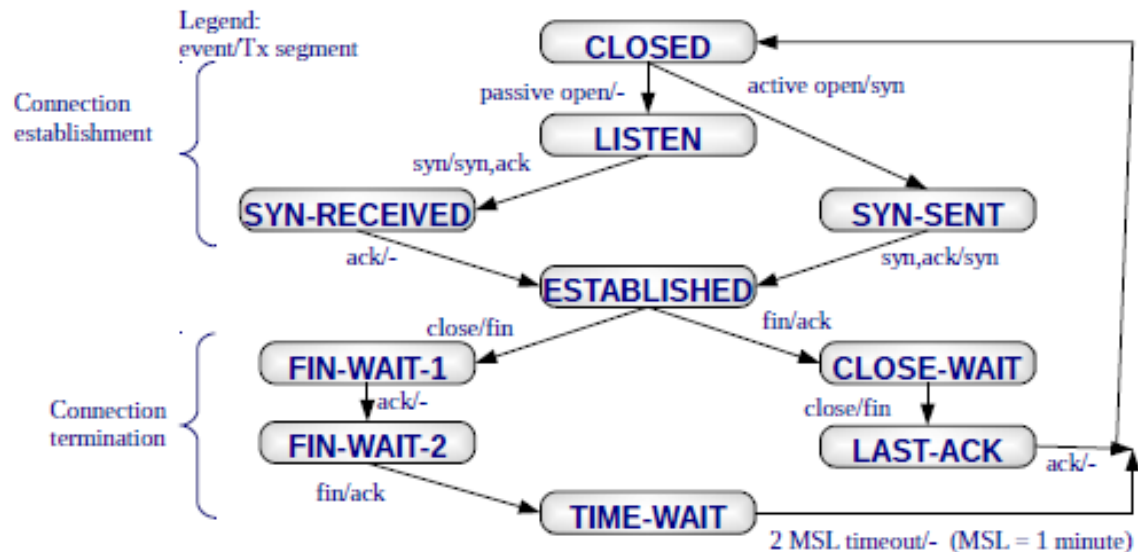
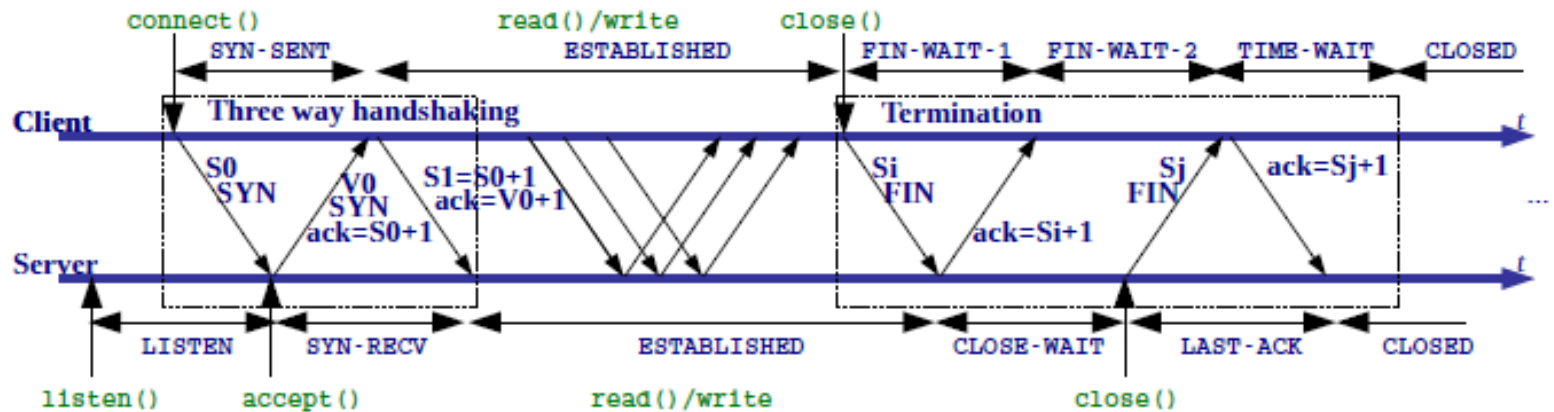
Control de Flux en TCP

- ARQ: Mètode de sol·licitud de repetició automàtica
- PAR: Recepció positiva amb retransmissió (ACK)
 - RQ-Inactiva o RQ-Continua
- Control de transmissió -> Cada datagrama es numera per al seu control
 - Datagrama _0 de 900 bytes
 - Datagrama _900 de 400 bytes
 - Datagrama _1300 de 500 bytes

- El número de seqüència identifitifa el primer byte del payload
- El número del ACK identifica el següent byte que espera el receptor.



TCP – Diagrama de estados



UDP : Definicions I conceptes

- El protocol de datagrama d'usuari (UDP) està definit en la RFC 768.
- No garanteix l'entrega de paquets IP. Són les pròpies aplicacions les que han de proporcionar mecanismes de fiabilitat
- Es fa servir per transportar dades de forma ràpida, compacta i no fiable.
- La velocitat d'entrega està per sobre de la seguretat (multimèdia en temps real)

UDP

- Protocol no fiable
- No orientat a connexió
- Premia la velocitat en l'entrega
- No existeix control de errors amb reenviament
- S'especifica amb el número 17 en el camp de capçalera IP
- Una única cua de missatges
- Important per apps que necessiten multimèdia en temps real

UDP

[illegible]

Ports

- Tant TCP com UDP fan servir ports per permetre la comunicació entre aplicacions. Aquest camp té una mida de 16 bits, per tant el rang de valors va des de 0 fins a 65535. El port 0 està reservat però es un valor permès com a port origen si el procés emissor no espera rebre missatges com a resposta.
- Els ports 1 a 1023 són els ports 'ben coneguts'. En sistemes operatius tipus Linux enllaçar amb un d'aquests ports requereix accés com a superusuari.

Ports

- Els ports 1024 a 49151 son ports registrats però no regulats
- Els ports 49.152 a 65.535 són ports efímers i es fan servir com ports temporals.

➤ Exemples de ports associats:

TCP

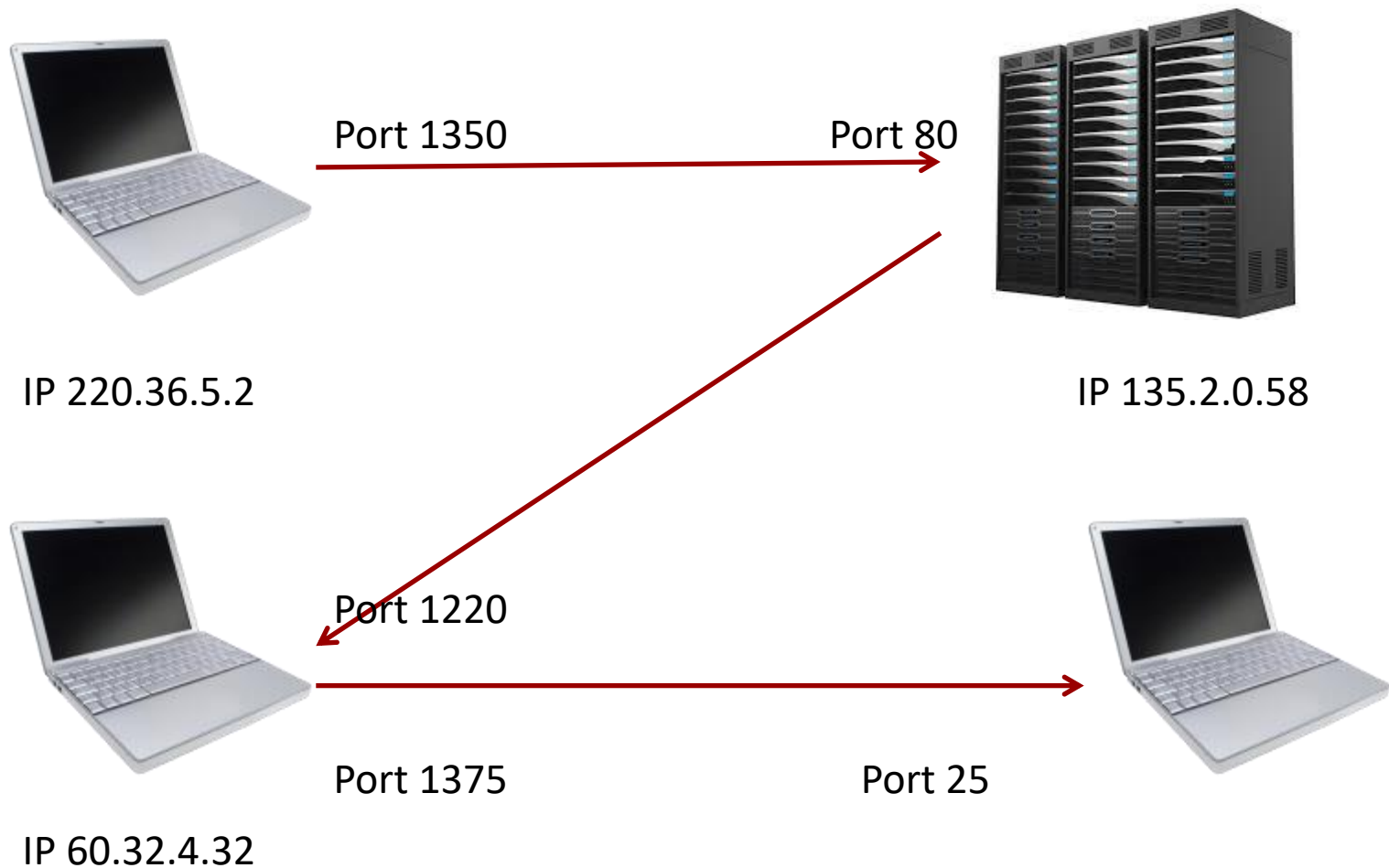
UDP

Echo	=>	7	DNS	=>	53
FTP	=>	20 (dades) 21(ccontrol)	BOOTP (server)	=>	67
SSH, scp, SFTP	=>	22	BOOTP (client)	=>	68
Telnet	=>	23	TFTP	=>	69
SMTP	=>	25	SMTP	=>	25
Time Protocol	=>	37	Time Protocol	=>	37
HTTP	=>	80	SNMP	=>	161

SOCKETS

- Sockets: Punts terminals de connexió establerts entre dues màquines
 - Socket = adreça IP + port de protocol
- Primitives associades als sockets
 - SOCKET: crea un nou punt terminal de connexió
 - BIND: connecta una adreça local a un socket
 - LISTEN: anuncia la disposició a acceptar connexions. Indica també la mida de la cua
 - ACCEPT: bloqueja al invocador fins l'arribada d'un intent de connexió
 - CONNECT: intenta establir activament la connexió
 - SEND: Envia dades a través de la connexió
 - RECEIVE: Rep dades de la connexió
 - CLOSE: Tanca la connexió

Connexions amb sockets



Sockets

- Els números inferiors a 255 es fan servir per aplicacions públiques
- Els números compresos entre 255 i 1023 s'assignen a empreses per a aplicacions comercials
- Els números superiors a 1023 no estan regulats i normalment s'utilitzen per a assignació dinàmica

Ports I sockets

- La comanda netstat proporciona connexions actives de la màquina I els sockets en us d'aquesta
- Els ports son punts d'entrada de paquets de dades a una màquina => Punts perillosos per la seva vulnerabilitat
- Només s'han de tenir oberts aquells ports que són necessaris