

Teleinformática

Universidad de Mendoza

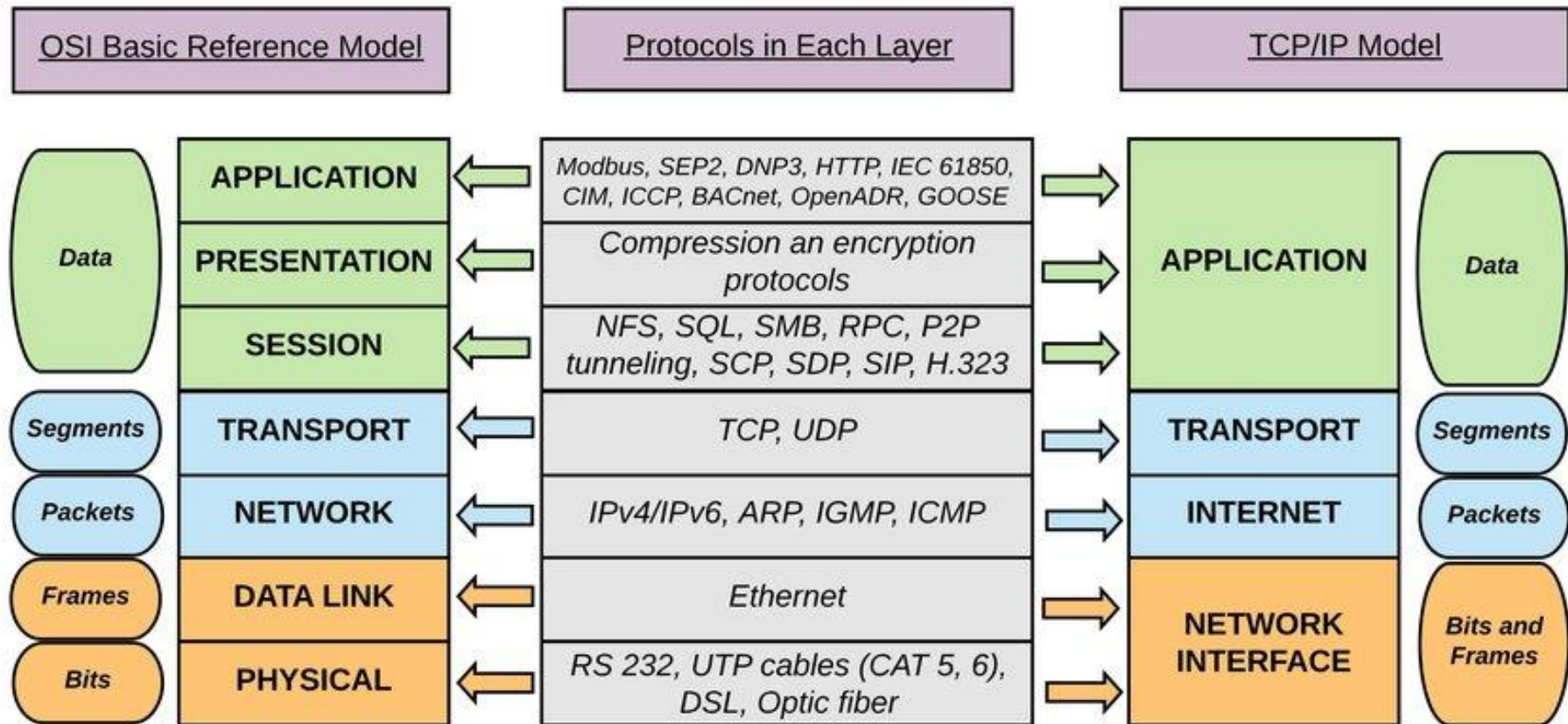
Protocolos TCP/IP

Ms. Ing. Diego Navarro

Ms. Ing JuanJo Ciarlante

Stack TCP/IP

Comparación TCP/IP - OSI



5-upla identifica “conexión/asociación”: < IPorig, PORTorig, PROTO, IPdest, PORTdest >

Proto. de transporte TCP/IP

Caract.

	TCP	SCTP	UDP
Orient. conexión	Sí	Sí	No
Unidad transp.	bytes	msgs	msgs<65k
Corr.err, no dupl.	Sí	Sí	-
Entrega ordenada	Sí	Opt -	-
Control de flujo	Sí	Sí	-
Control congest.	Sí	Sí	-
Multi-stream	-	Sí	-
Multihome	-	Sí	-

TCP: Transmission Control Protocol: Características

- TCP: RFCs: 792, 1122 y 2581
- capa de transporte
- orientado a conexión
- orientado a stream **de bytes:**
 - espacio de secuencia
- confiable + entrega ordenada
 - ACK “positivo”
 - retransmisión si ACKto (timeout)

TCP: Características (cont.)

- **circuito virtual:**
 - **establecimiento, uso, fin: handshake, ACKs**
- **gestión I/O hacia aplicación (interfaz tpte-app)**
 - ports: identifican procesos/servicios
 - stream: flujo continuo
 - apertura activa ó pasiva
- **gestión I/O hacia la red (interfaz tpte-red)**
 - stream continuo <-> segmentos (paquetes)
 - confiabilidad: retransmisión de segmentos hasta ACK
 - MTU, PMTU
- **control de flujo en transmisión: ventanas**
 - ventana de recepción (lado RX)
 - ventana de congestión (no determinística)

TCP: manejo de I/O y de congestión (1/2)

- **SWS: Silly Window Syndrome:** lado RX lee de a pequeños bloques
 - Solución: publico **W=0** hasta que **W ≥ PMTU** ó **W > RCV.WND/2**
- **Nagle:** lado TX escribe de a pequeños bloques
 - Solución: demorar el envío de segmentos hasta que
Seg.Size ≥ PMTU ó todos los seg. anteriores **ACK**-eados
- **Ventana de congestión:** cuando la **red** es el cuello de botella
 - **cwnd = cwnd/2** si ACK duplicado
 - **cwnd = PMTU** si ACK timeout (retrans.)
- **Slow start:** para evitar el “burst” de arranque
 - Solución: arranco **cwnd = PMTU** (1 segmento),
aumento exponencial haciendo **cwnd++** por cada ACKeo
- **Congestion avoidance:**
 - aumento lineal cuando **cwnd** llega a la mitad del valor de la ventana al momento de congestión

TCP: manejo de I/O y de congestión (2/2)

- **Delayed ACKs**

- tratar de no enviar ACK inmediatamente para:
 - permitir *piggybacking*
 - disminuir tráfico mediante la acumulación de espacio a ACK-ear
- no más de 500 mS.
- no más de 2 segmentos full size (PMTU)

- **RTT: Promedio dinámico:** $RTT = \alpha RTT + (1 - \alpha) RTT$; $\alpha \approx 0.8$

- $ACKto = (1...2) * RTT$

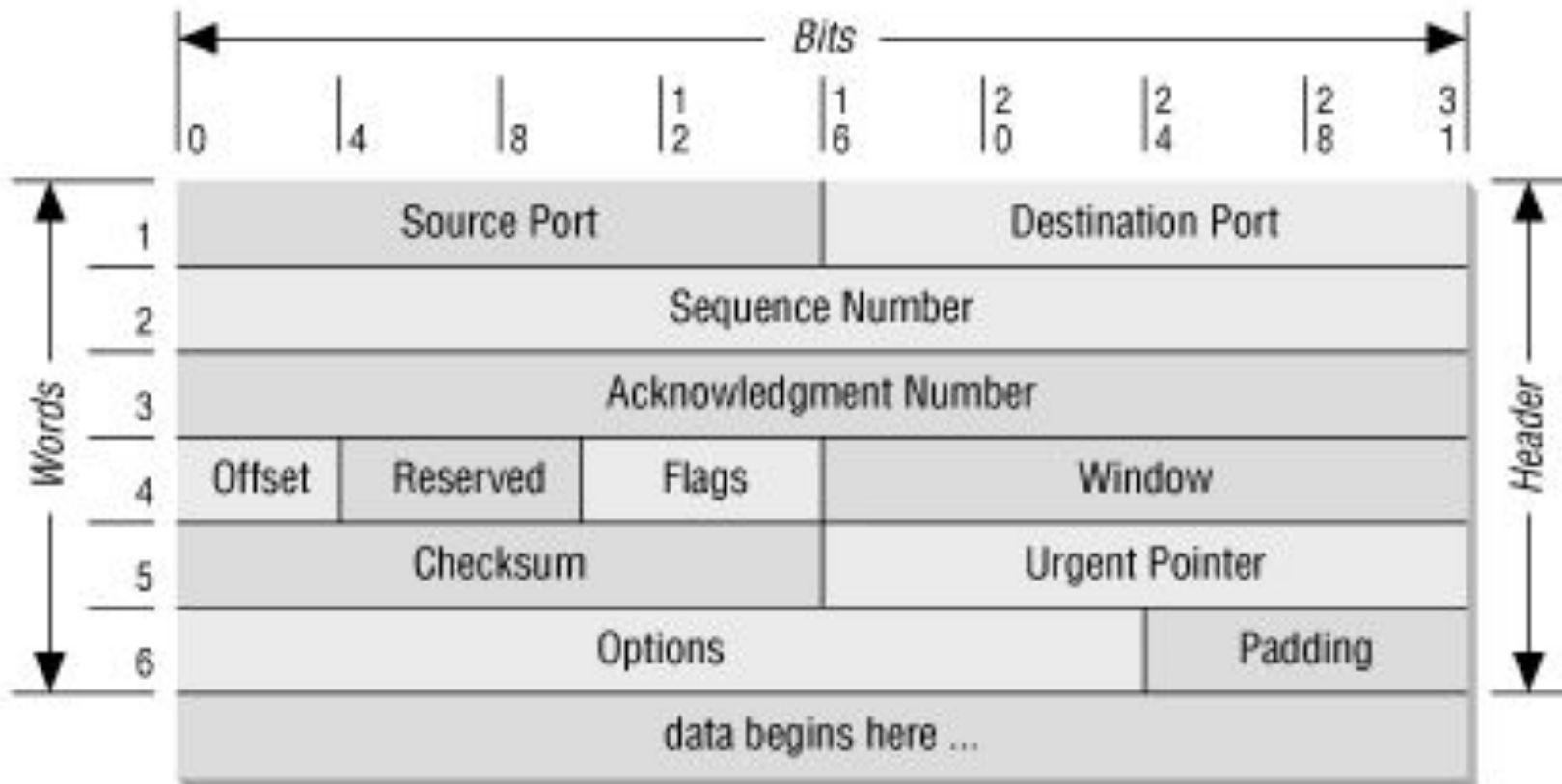
- **Fast retransmit (RFC 2581)**

- no esperar $ACKto$ si hay ≥ 3 dups

- **Fast recovery (RFC 2581)**

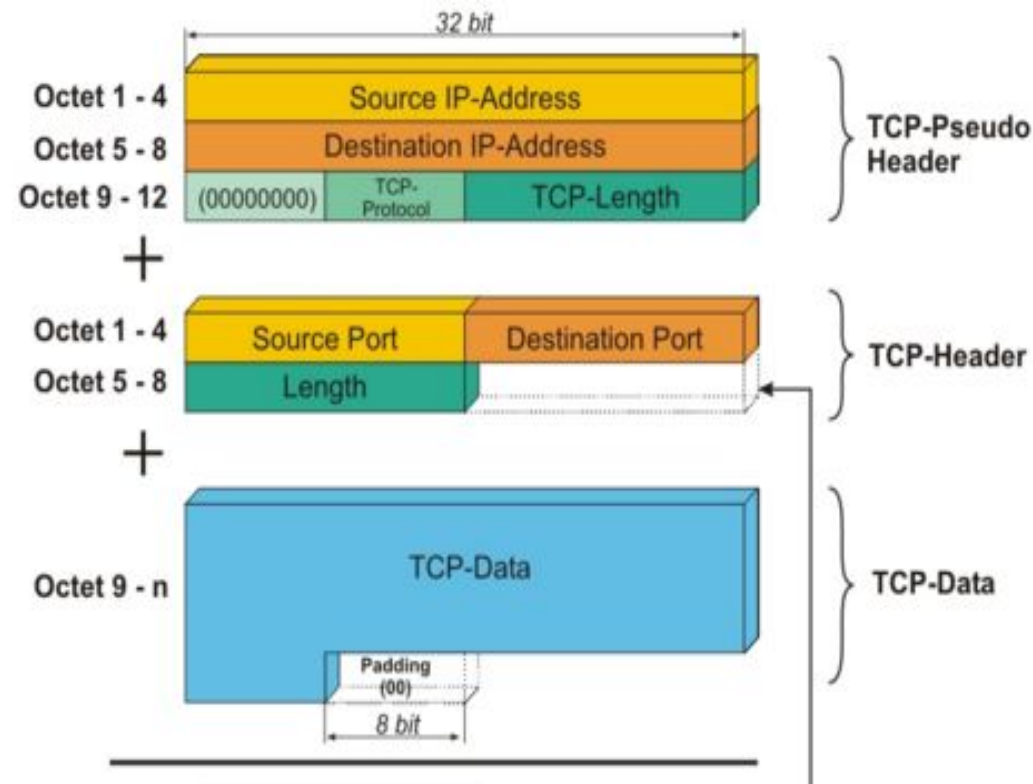
- reenvío solamente el segmento “consecutivo” (optimista resp. del resto)
- “infla” la **cwnd** momentáneamente

TCP: Cabecera



TCP: Cacerera: campos

- **Source, Destination Port** [16]: Puertos origen, destino
- **Sequence Number** [32]: Número de seq. del 1er byte del este segmento (excepto SYN,FIN)
- **Ack. Number** [32]: Próximo número de seq. que se espera recibir (válido si el bit ACK está seteado)
- **Data Offset** [4]: Tamaño del header tcp; en palabras de 32bits
- **Reserved** [6]: --
- **Control Bits** [6]:
 - URG: Urgent Pointer field significant
 - ACK: Acknowledgment field significant
 - PSH: Push Function
 - RST: Reset the connection
 - SYN: Synchronize sequence numbers
 - FIN: No more data from sender
- **Window** [16]: Cant. de bytes (desde el ACK que se pueden recibir).
- **Checksum** [16]: Checksum de PSEUDO-HEADER
- **Urgent Pointer** [16 bits]: Offset del byte siguiente al urgent data



TCP: Cabecera:

opciones más usadas

- **EOL:** End of Option List: 1 byte

- Solamente usada si el fin de las opciones no coincide con el fin del header

- **NOP:** No-Operation: 1 byte Para alineado (relleno)

- **MSS:** Maximum Segment Option : 2+2 bytes (RFC 793)

- Indica el tamaño máximo de segmento que es capaz de recibir (quien envía este segmento), sólo presente en SYN.

- **TCP Window Scale Option:** 2+1 bytes (RFC 1072)

*"efficient operation over a path with a high bandwidth*delay product..."*

- ambos deben negociar "en SYN" para poder usarlo cualquiera.
- Se calcula $\text{window} = \text{window} \ll (\text{shift.cnt})$
- permite ventanas de hasta 2^{30} (1Gbyte).

- **TCP Selective ACK options** (RFC 1072):

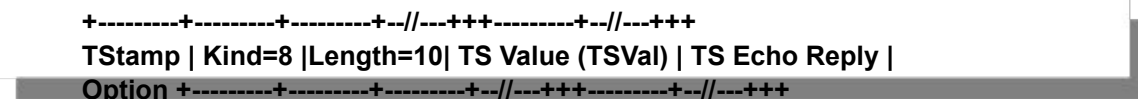
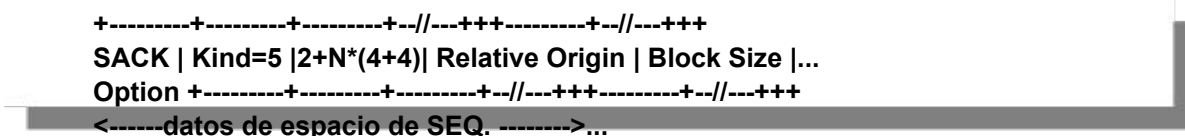
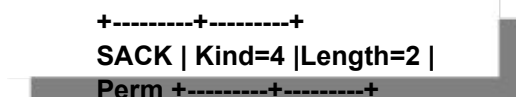
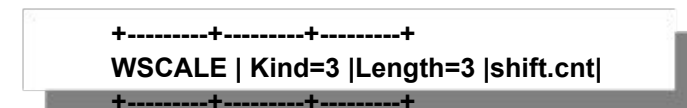
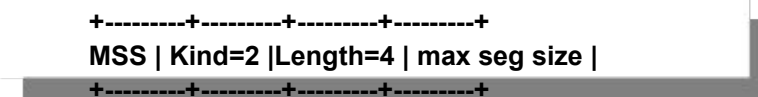
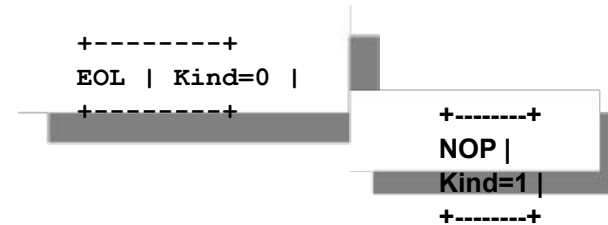
- **SACK-Permitted:** 2 bytes: Negociado "en SYN".

- **SACK Option:** 2+N*(4+4) bytes

- Usado durante la conexión; contiene una lista de bloques de espacios de seq. contiguos recibidos y encolados.
- Relative Origin es relativo al ACK del header tcp
- Block size: medido en bytes.

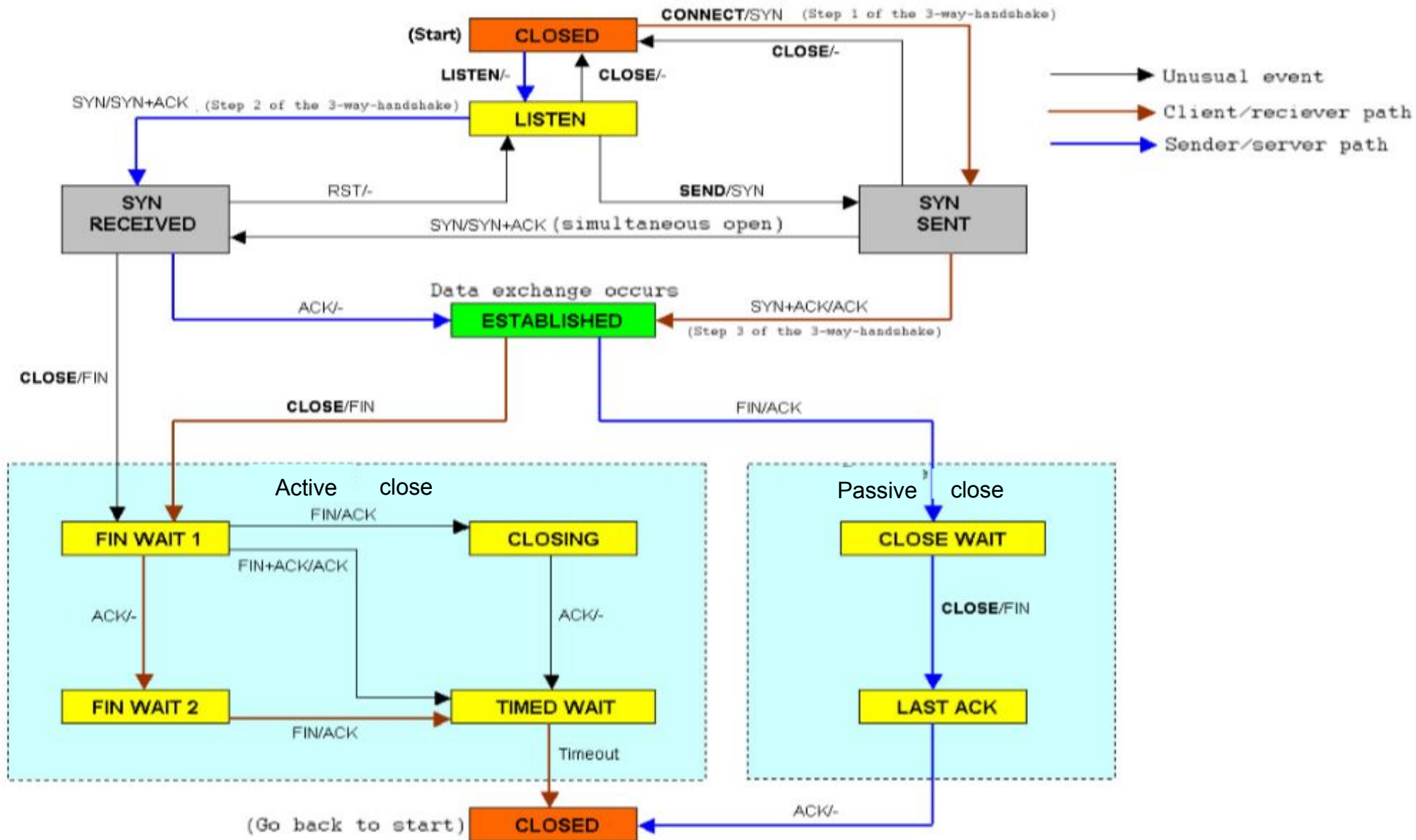
- **TCP Timestamps Option:** 2+4+4 bytes

- El TCP peer "copia" en TSecr el TSVal enviado



TCP: diag. de estados

http://en.wikipedia.org/wiki/Transmission_Control_Protocol

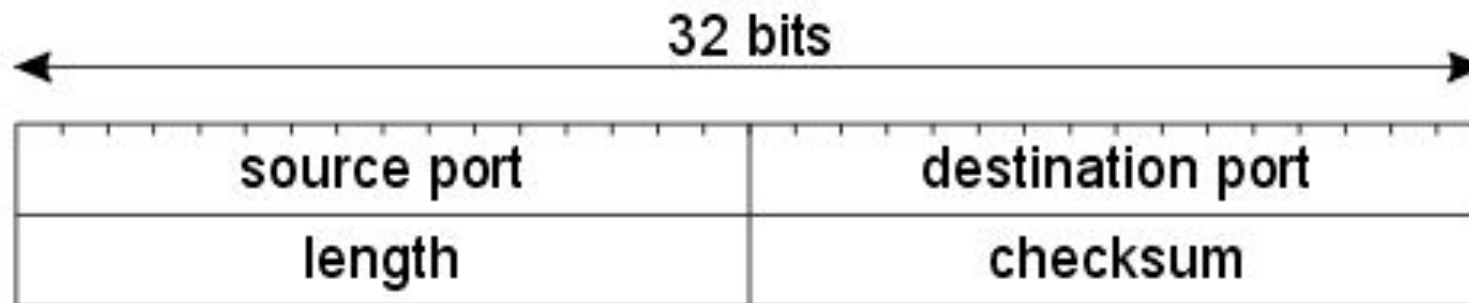


UDP: User datagram protocol

- RFC 768, 1122
- Características:
 - no orientado a conexión: *datagramas*
 - stateless
 - no confiable
 - I/O de aplicación: *mensajes*
- Multiplexación via puertos
- Permite **multicast (y broadcast)**
- Permite DHCP,BOOTP (datagramas con IPsrc=0.0.0.0)

UDP: Cabecera

UDP header format



- Largo datagrama:
 - 16bits
 - aplicación debería ajustar a PMTU
- Checksum opcional
 - si falla se descarta sin aviso

DNS: Domain Name System

- RFC 1035, 2535 (DNSSEC)
- Protocolo client-server
 - servers: primario , secundario (autoridad de zona)
 - “caché”: server intermediario (local)
- espacio de nombres jerárquico con delegación de autoridad
- mapea “strings” a valor
- tipos de R.R.:
 - SOA, NS
 - A, PTR, AAAA, A6
 - CNAME
 - MX
 - HINFO, TXT
 - KEY

DNS: zona de ejemplo

thematrix.org

```
; Archivo de zona "thematrix.org"
$TTL 86400 ; 1 dia
@ IN SOA neo root (
2006031001 ; serial
2h ; refresh
300 ; retry
30d ; expire
60 ; mínimo TTL
)
IN NS neo
IN NS morpheus
IN MX 0 trinity
IN MX 10 agent.smith.com
neo IN A 1.2.3.4
morpheus IN A 1.2.3.5
trinity IN A 1.2.3.10
www IN CNAME trinity
```

TCP: Protocolos de aplicación

(algunos)

- HTTP: port 80
 - Bajo overhead, baja latencia
 - “muchas” conexiones efímeras
- SSH: port 22
 - terminal remota segura
 - sesiones de “larga” duración
- SMTP: port 25
 - transporte de mail “inter”-servidores
- POP3: port 110
 - acceso a mailbox: copia server->cliente + borrado en server
- IMAP: port 143
 - acceso a mailbox: “control remoto”, mails **quedan** en server
- P2P: port <de todo>

UDP: Protocolos de aplicación

(algunos)

- DHCP: ports 67,68
 - UDP permite
 - **IPsrc**=0.0.0.0 (DHCPDISCOVER)
 - broadcast (DHCPDISCOVER, DHCPREQUEST)
- DNS: port 53
 - muy liviano
 - 2 paquetes: pregunta, respuesta
- NFS: Network File System
 - montaje de recursos remotos
 - UDP evita “taconamiento de cabeza” de TCP
- RTP, RCTP:
 - streaming de medios: audio, video
 - multicast Ok