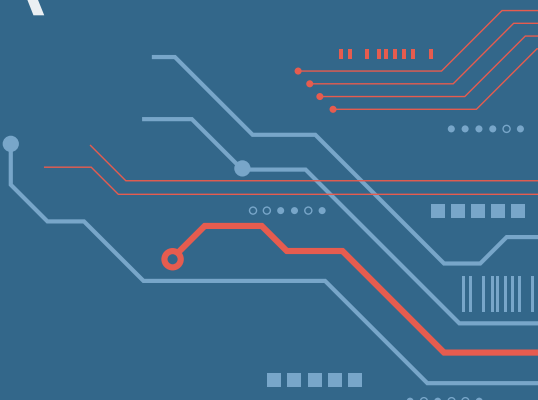


REDES DE DATOS TUIA | FCEIA UNR

Docentes | 1C 2023

Juan Pablo Michelino
Emiliano Pavicich
Andrea León Cavallo
Iván Pellejero
Esteban Toribio

jpmich@fceia.unr.edu.ar
pavicich@fceia.unr.edu.ar
aleoncavallo@gmail.com
ivan.pellejero97@gmail.com
toribio@fceia.unr.edu.ar



03



CAPA DE TRANSPORTE

3.1. Funciones de la capa de transporte

3.2. Puertos y sockets.

3.3. Protocolo UDP.

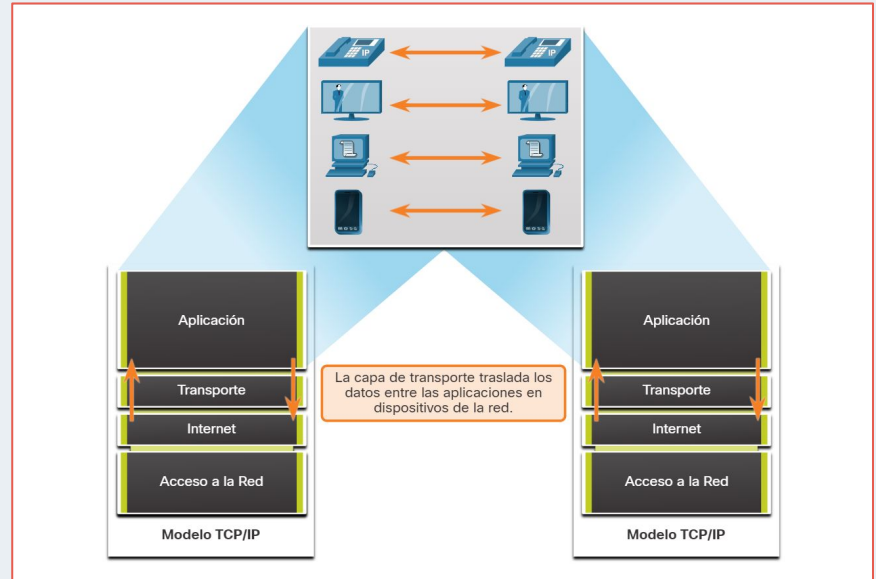
3.4. Protocolo TCP. Intercambio de tres vías. Ventana deslizante

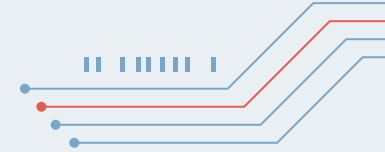
3.5. Aplicaciones y puertos.



Funciones

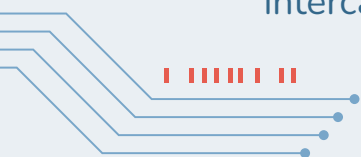
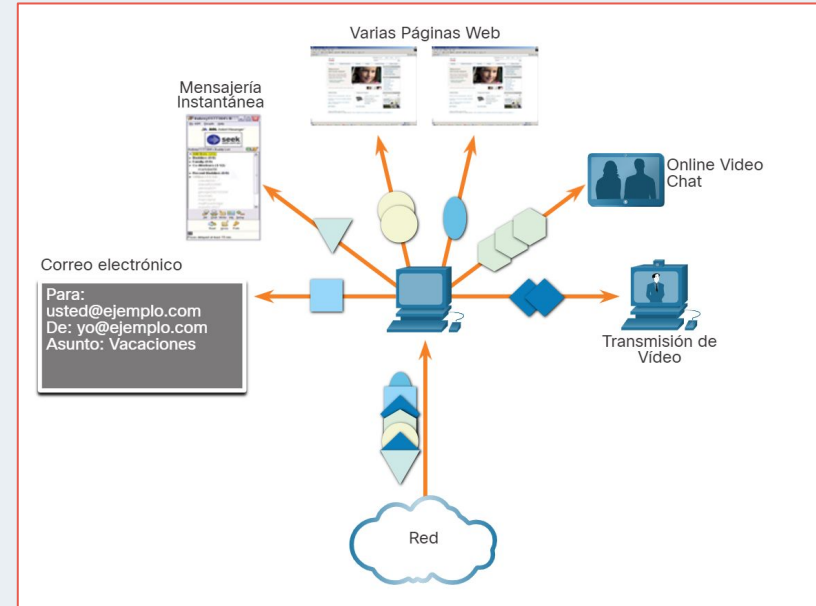
- Responsable de las **comunicaciones lógicas entre aplicaciones** que se ejecutan en diferentes hosts.
- Enlace entre la capas de **aplicación y las capas inferiores** que se encargan de la transmisión a través de la red.





Responsabilidades

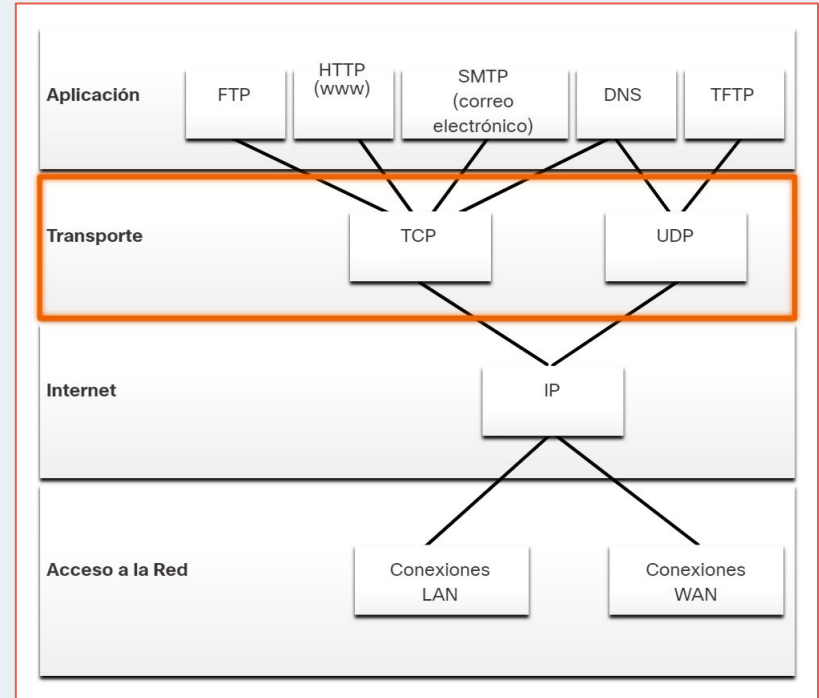
- Seguimiento de conversaciones individuales
- Segmentación de datos y rearmado de segmentos
- Agregar información de encabezado
- Identificar, separar y administrar múltiples conversaciones
- Utiliza segmentación y multiplexación para permitir que diferentes conversaciones de comunicación se intercalen en la misma red



Protocolos

Las diferentes aplicaciones tienen diferentes requisitos de confiabilidad de transporte. Por lo tanto, TCP/IP proporciona dos protocolos de capa de transporte: TCP y UDP.

TCP tiene que establecer una conexión entre el emisor y el receptor para transmitir datos, mientras que **UDP** transmite paquetes a otro dispositivo sin tener que establecer un canal ni preguntar si el receptor está listo. UDP es más rápido que TCP porque no ordena ni comprueba los errores de los paquetes de datos.



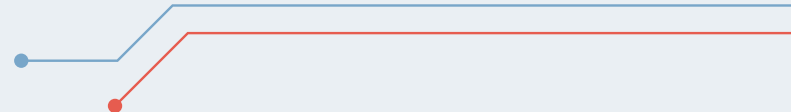
Puertos

Los protocolos de capa de transporte TCP y UDP utilizan **números de puerto** para administrar **múltiples conversaciones simultáneas**. Los campos de encabezado TCP y UDP identifican un número de puerto de aplicación de origen y destino.

Un puerto es un número de 16 bits utilizado por el protocolo para identificar a qué protocolo de nivel superior (aplicación o proceso) debe entregar el mensaje entrante.

Puerto de origen (16)

Puerto de destino (16)





Grupos de números de puerto

La Autoridad de Números
Asignados de Internet (IANA)
ha dividido el rango de
números en los siguientes tres
grupos de puertos:

Puertos bien conocidos:
[0 - 1023]

Puertos registrados:
[1024 - 49151]

Puertos privados y/o dinámicos:
[49152 - 65535]



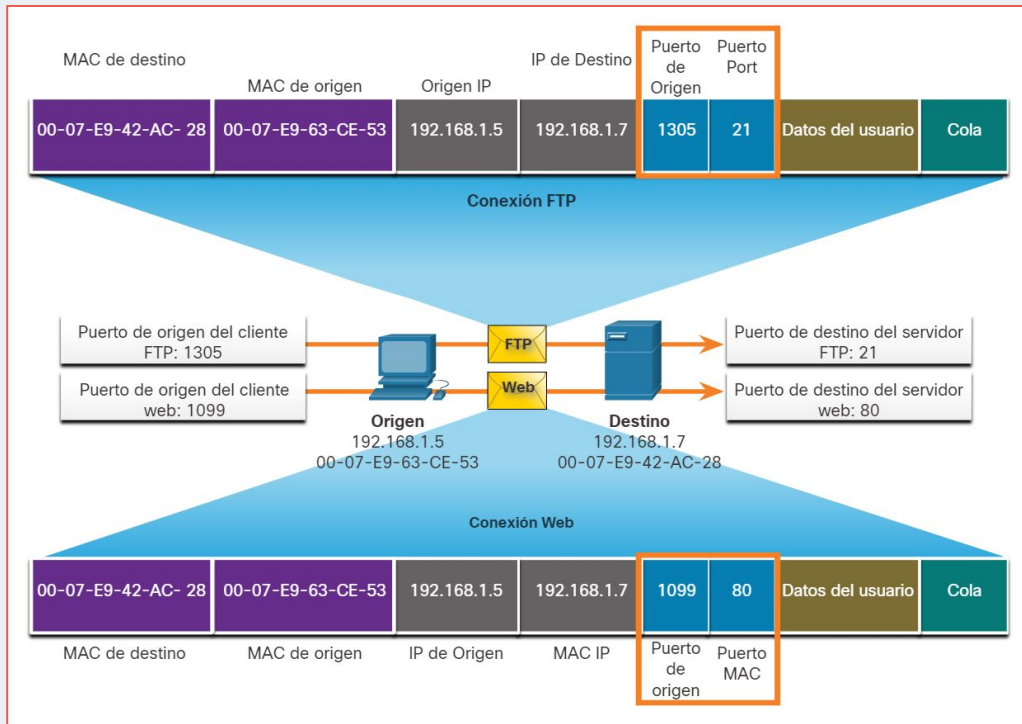
Puerto	Protocolo	Aplicación
20	TCP	Protocolo de transferencia de archivos (FTP) - Datos
21	TCP	Protocolo de transferencia de archivos (FTP) - Control
22	TCP	Secure Shell (SSH)
23	TCP	Telnet
25	TCP	Protocolo simple de transferencia de correo (SMTP)
53	UDP, TCP	Servicio de nombres de dominio (DNS, Domain Name Service)
67	UDP	Protocolo de configuración dinámica de host (DHCP): servidor
68	UDP	Protocolo de configuración dinámica de host: cliente
69	UDP	Protocolo trivial de transferencia de archivos (TFTP)
80	TCP	Protocolo de transferencia de hipertexto (HTTP)
110	TCP	Protocolo de oficina de correos, versión 3 (POP3)
143	TCP	Protocolo de acceso a mensajes de Internet (IMAP)
161	UDP	Protocolo simple de administración de redes (SNMP)
443	TCP	Protocolo seguro de transferencia de hipertexto (HTTPS)

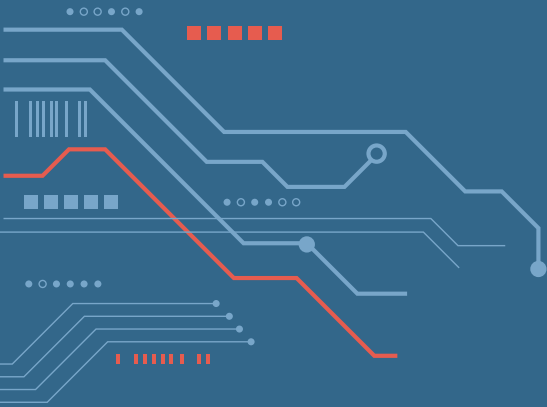
Sockets

Se conoce como **socket** a la combinación de la dirección IP de origen y el número de puerto de origen, o de la dirección IP de destino y el número de puerto de destino.

Por ejemplo: 192.168.1.5:1305

Permiten que diversos procesos que se ejecutan en un cliente se distingan entre sí y la diferenciación de conexiones a un proceso de servidor.





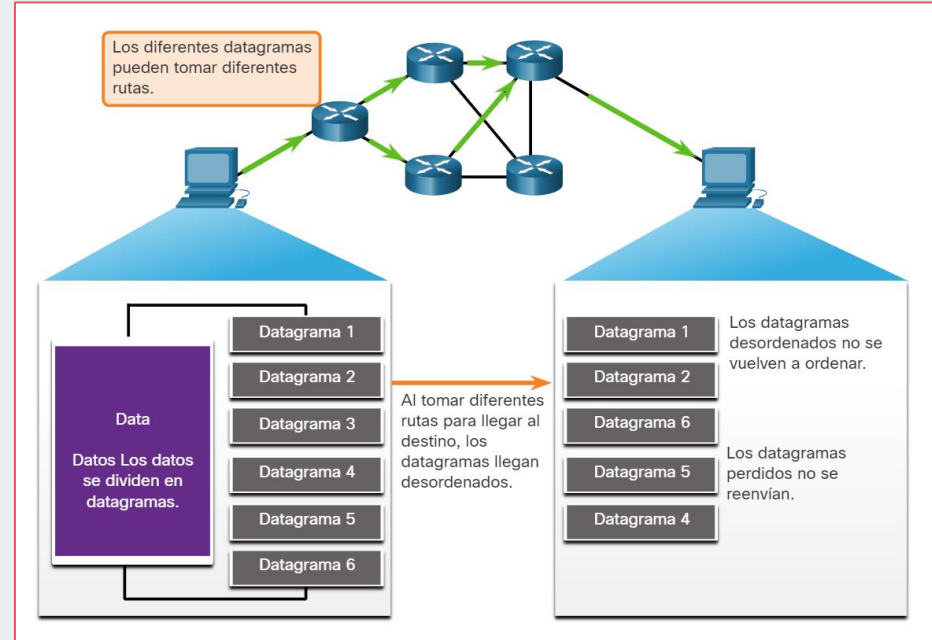
| | | | | | | |

UDP

User Protocol Datagram (UDP)

Permite el envío de datagramas de forma rápida en redes IP sin establecer previamente una conexión, dado que el propio datagrama incorpora suficiente información sobre el destinatario en su cabecera.

- Los datos se reconstruyen en el orden en que se recibieron.
- Los segmentos perdidos no se vuelven a enviar.
- No hay establecimiento de sesión.
- El envío no está informado sobre la disponibilidad de recursos.



Encabezado

Puerto de origen [16 bits]: identificar la aplicación de origen por número de puerto.

Puerto de destino [16 bits]: identificar la aplicación de destino por número de puerto.

Longitud [16 bits]: indica la longitud del encabezado del datagrama UDP.

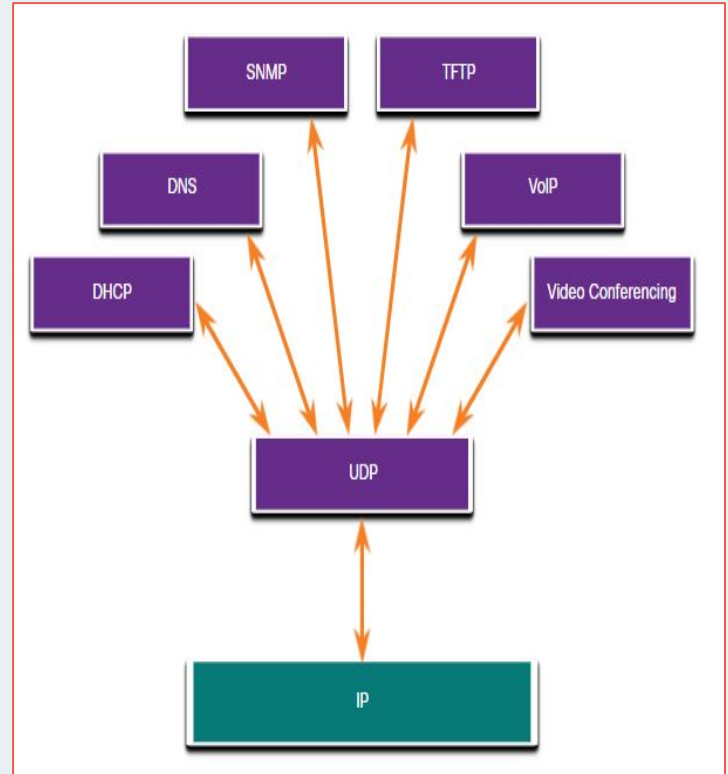
Suma de comprobación [16 bits]: utilizado para la comprobación de errores del encabezado y los datos del datagrama.

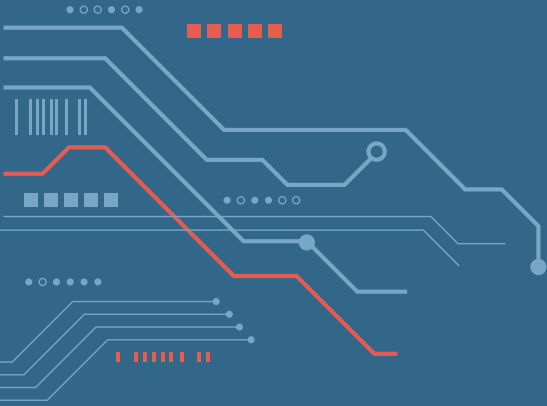
Ref.: IEEE [RTF768](#) (1980)



Aplicaciones

- Video y multimedia en vivo: estas aplicaciones pueden tolerar cierta pérdida de datos, pero requieren poco o ningún retraso. Ej.: VoIP y la transmisión de video en vivo.
- Solicitudes y respuestas simples: transacciones simples en las que un host envía una solicitud y existe la posibilidad de que reciba una respuesta o no. Ej.: DNS y DHCP.
- Aplicaciones que manejan la confiabilidad por sí mismas. Ej.: SNMP y TFTP





| | | | | | | |

TCP

Transmission Control Protocol (TCP)



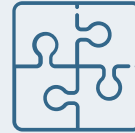
Establece una sesión

Negocia y establece una conexión permanente entre los dispositivos de origen y destino antes de reenviar cualquier tráfico.



Entrega confiable

TCP asegura que cada segmento que envía la fuente llega al destino.



Entrega en el mismo orden

Al numerar y secuenciar los segmentos, TCP garantiza que los segmentos se vuelvan a ensamblar en el orden correcto.



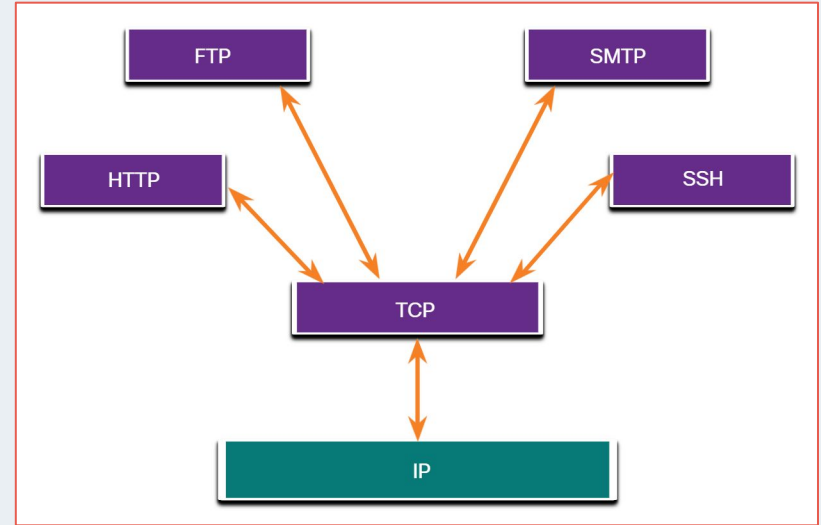
Control de flujo

Puede regular la cantidad de datos que transmite en función de la disponibilidad de recursos (memoria y potencia de procesamiento).



Aplicaciones

TCP es un buen ejemplo de cómo las diferentes capas del conjunto de protocolos TCP / IP tienen roles específicos. TCP maneja todas las tareas asociadas con la división del flujo de datos en segmentos, proporcionando confiabilidad, controlando el flujo de datos y reordenando segmentos. TCP libera a la capa de aplicación de tener que administrar estas tareas y simplemente pueden enviar el flujo de datos a la capa de transporte y utilizar los servicios de TCP.



Encabezado

TCP es un protocolo con estado: para hacer un seguimiento del estado de una sesión, TCP registra qué información se envió y qué información se reconoció. La sesión con estado comienza con el establecimiento de la sesión y termina con la finalización de la sesión.



Un segmento TCP agrega 20 bytes (es decir, 160 bits) de sobrecarga al encapsular los datos de la capa de aplicación.

Ref.: [RFC 793](#)

Encabezado

Campo de Encabezado TCP	Descripción
Puerto de Origen [16 bits]	Utilizado para identificar la aplicación de origen por número de puerto.
Puerto de Destino [16 bits]	Utilizado para identificar la aplicación de destino por puerto número.
Secuencia de Números [16 bits]	Utilizado para reensamblar datos.
Número de Acuse de Recibo [32 bits]	Utilizado para indicar que se han recibido datos y el siguiente byte esperado de la fuente.
Longitud del Encabezado [4 bits]	Indica la longitud del encabezado del segmento TCP.
Reservado [6 bits]	Un campo que está reservado para uso futuro.
Bits de Control [6 bits]	Códigos de bits, o indicadores, que indican el propósito y función del segmento TCP.
Tamaño de la ventana [16 bits]	Indica el número de bytes que se pueden aceptar a la vez.
Suma de Comprobación [16 bits]	Chequeo de error del encabezado y los datos.
Urgente [16 bits]	Campo de 16 bits utilizado para indicar si los datos contenidos son urgentes.

U	A	P	R	S	F
R	C	S	S	V	I
G	K	H	T	N	

Encabezado

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N

URG: Indica que el campo urgente tiene sentido. (Urgent)

ACK: Indica que el campo número de recibo tiene sentido.
(Acknowledge)

PSH: El receptor debe pasar los datos a la aplicación tan pronto como sea posible. (Push)

RST: Indica que se debe reiniciar la conexión. (Reset)

SYN: Sincronizar la secuencia de números. (Sync.)

FIN: No hay más datos para transmitir.(Finish)



Comparación encabezados UDP / TCP



UDP

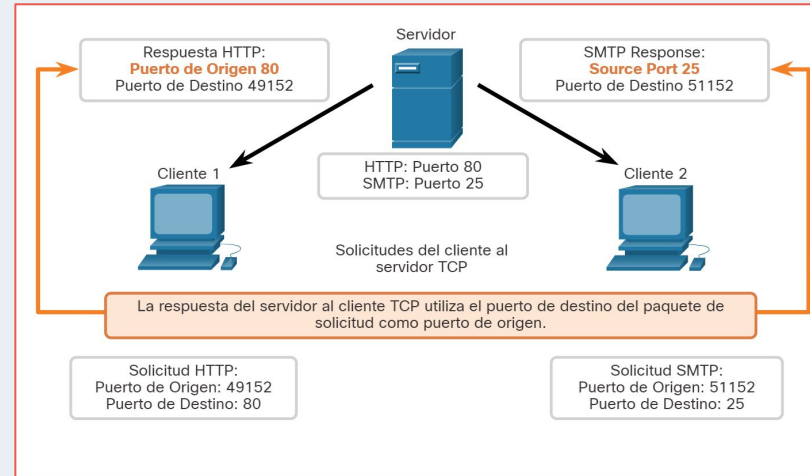


TCP

Servidor TCP

Cada proceso de aplicación que se configura en el servidor para utilizar un número de puerto.

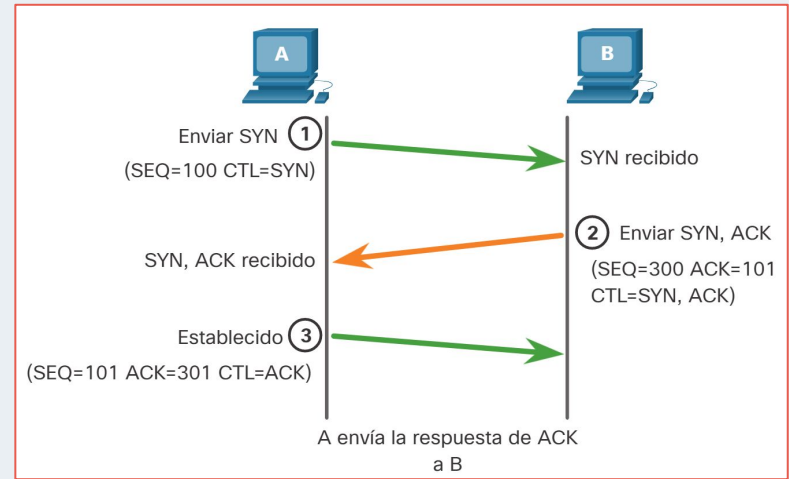
- Un servidor individual no puede tener dos servicios asignados al mismo número de puerto dentro de los mismos servicios de la capa de transporte.
- Una aplicación de servidor activa asignada a un puerto específico se considera abierta, lo que significa que la capa de transporte acepta y procesa los segmentos dirigidos a ese puerto.
- Toda solicitud entrante de un cliente direccionada al socket correcto es aceptada y los datos se envían a la aplicación del servidor.



Establecimiento de conexiones TCP

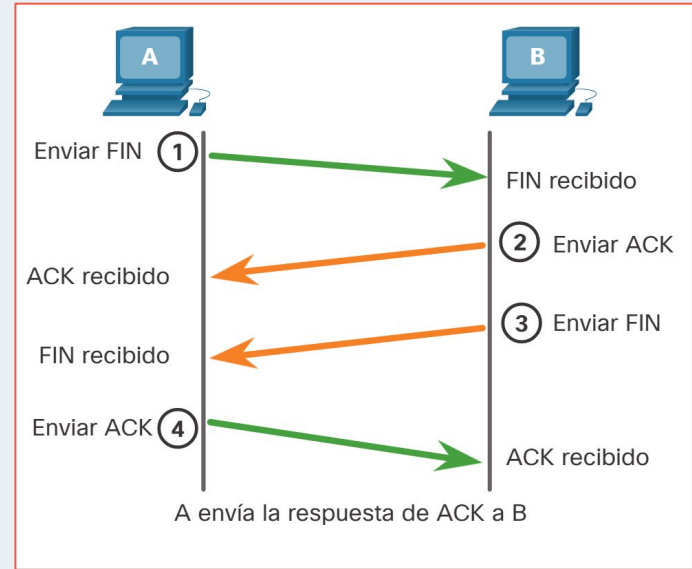
En las conexiones TCP, el cliente host establece la conexión con el servidor mediante el proceso de **enlace de tres vías** (“three way handshake”):

1. El cliente de origen solicita una sesión de comunicación con el servidor.
2. El servidor reconoce la sesión de comunicación del cliente y también solicita una sesión de comunicación.
3. El cliente de origen reconoce la sesión de comunicación del servidor.



Finalización de la sesión TCP

1. Cuando el cliente no tiene más datos para enviar en la transmisión, envía un segmento con el indicador FIN establecido.
2. El servidor envía un ACK para confirmar el indicador FIN y finalizar la sesión de cliente a servidor.
3. El servidor envía un FIN al cliente para finalizar la sesión de servidor a cliente.
4. Paso 4: El cliente responde con un ACK para confirmar el FIN desde el servidor.



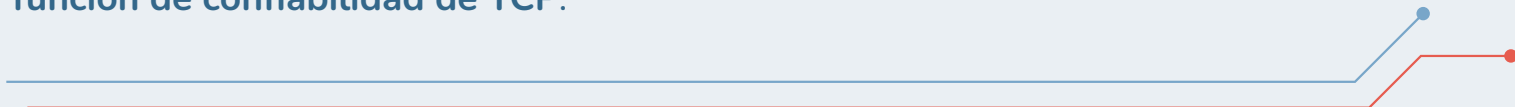


Comunicación TCP

Funciones del enlace de tres vías:

1. Establece que el dispositivo de destino está presente en la red.
2. Verifica que el dispositivo de destino tenga un servicio activo y acepte solicitudes en el número de puerto de destino que el cliente de origen desea utilizar.
3. Informa al dispositivo de destino que el cliente de origen intenta establecer una sesión de comunicación en dicho número de puerto

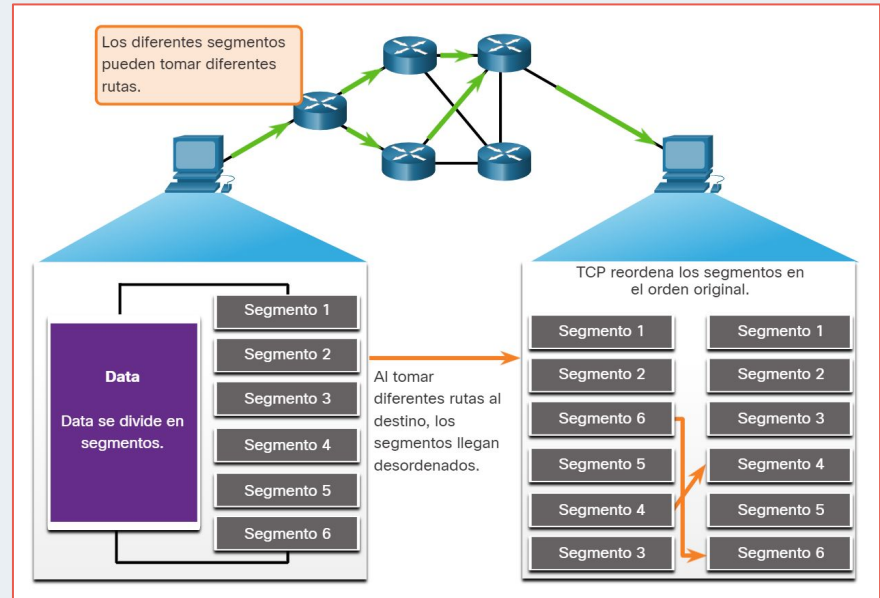
Una vez que se completa la comunicación, se cierran las sesiones y se finaliza la conexión. **Los mecanismos de conexión y sesión habilitan la función de confiabilidad de TCP.**



Confiabilidad TCP: Entrega garantizada y ordenada

En TCP **todos los datos deben ser recibidos** y los datos de estos segmentos **deben ser reensamblados en el orden original**.

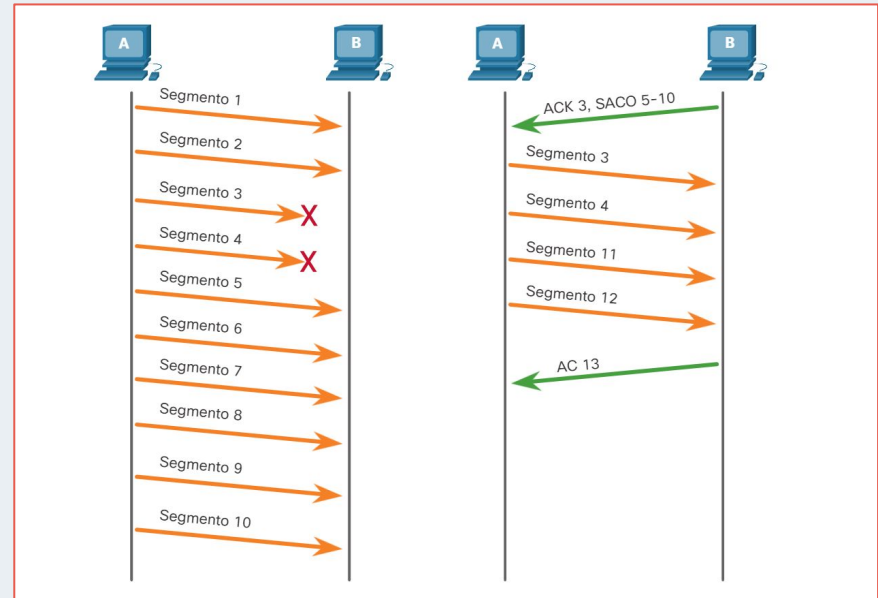
Para lograr esto, se asignan **números de secuencia** en el encabezado de cada paquete. A medida que se transmiten los datos durante la sesión, el número de secuencia se incrementa según el número de bytes que se han transmitido. Este seguimiento de bytes de datos permite **identificar y reconocer cada segmento de manera exclusiva**.



Confiabilidad TCP: Pérdida y retransmisión de datos

TCP proporciona **métodos para administrar la pérdida de segmentos**. Entre estos está un mecanismo para retransmitir segmentos para los datos sin reconocimiento.

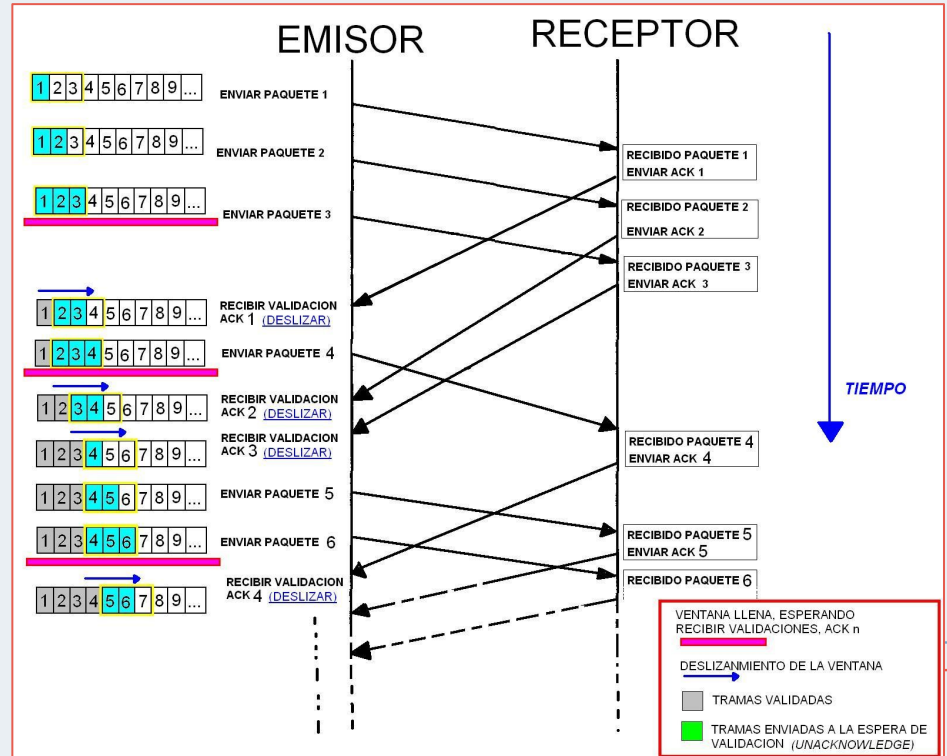
El número SEQ identifica el primer byte de datos en el segmento que se transmite. TCP utiliza el número de ACK re-enviado al origen para indicar el próximo byte que el receptor espera recibir. Esto se llama acuse de recibo de expectativa.



Ventana deslizante

El protocolo de **ventana deslizante** de TCP le permite al emisor transmitir múltiples segmentos de información antes de comenzar la espera para que el receptor le confirme la recepción de los segmentos.

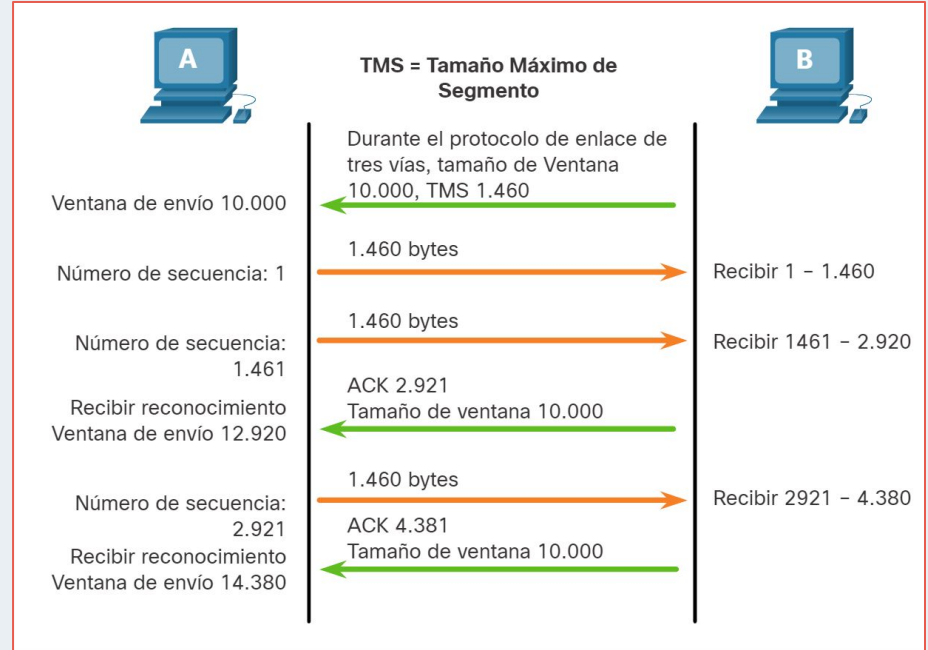
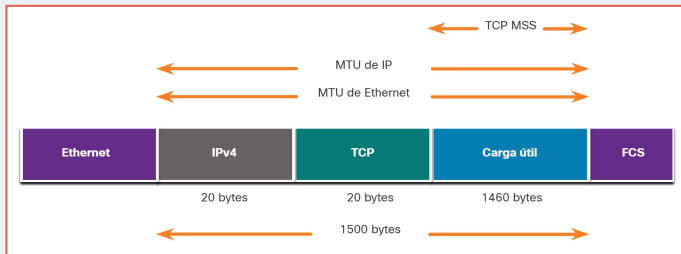
Esto hace que exista una continua transmisión de información, mejorando el desempeño de la red.



Control de flujo de TCP

El control de flujo permite mantener la confiabilidad de la transmisión de TCP mediante el **ajuste de la velocidad del flujo de datos** entre el origen y el destino para una sesión dada.

Tamaño máximo de segmento (TMS) es la cantidad máxima de datos que puede recibir el dispositivo de destino y es informado por el mismo.

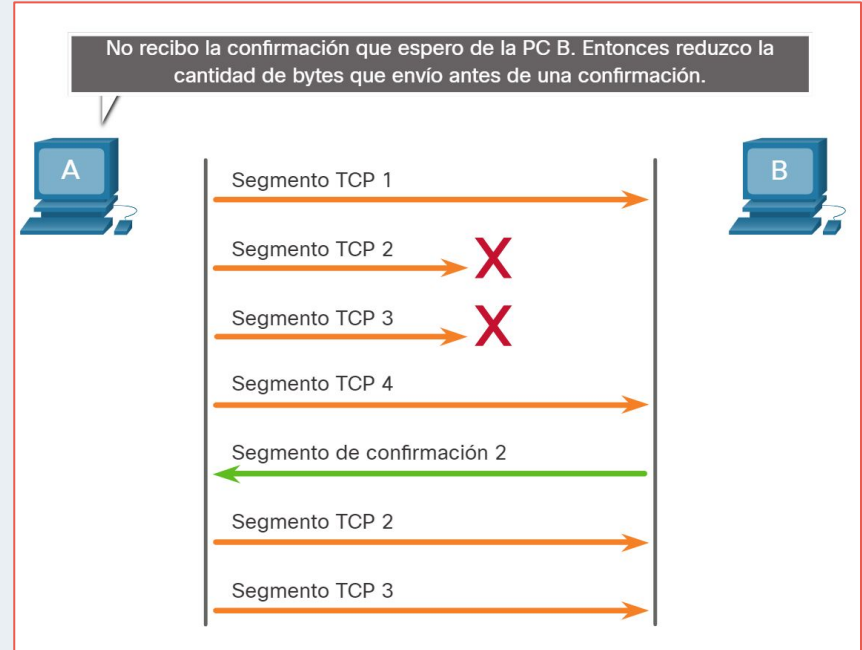


Control de flujo de TCP

Siempre que haya congestión, se producirá la retransmisión de los segmentos TCP perdidos del origen.

Si el origen determina que los segmentos TCP no están siendo reconocidos o que sí son reconocidos pero no de una manera oportuna, entonces **puede reducir el número de bytes que envía antes de recibir un reconocimiento**.

Para evitar y controlar la congestión, TCP emplea varios mecanismos, temporizadores y algoritmos de manejo de la congestión.



RECURSOS BIBLIOGRÁFICOS

- Cisco NetAcad Introduction to Networks:
 - Módulo 14: “Capa de transporte”
- Redes de Computadoras | Tannenbaum - Wetherall (2012) | 5ta Edición WordPress:
 - Capítulo 6: “La capa de transporte”
- Comunicaciones y Redes de Computadores | Stallings (2018) | 7ma Edición Pearson:
 - Capítulo 17: “Protocolos de transporte”