Capa de TRANSPORTE

Ing. José Martín Calixto Cely
Original: Galo Valencia P.

Capa de Transporte

- *La Capa 1 crea y transporta las corrientes de bits;*
- *La Capa 2 encapsula los paquetes de datos en tramas, y posibilita así su entrega en las LAN;*
- *La Capa 3 empaqueta los datos de las capas superiores en paquetes y permite el enrutamiento y entrega en las WAN.*

Pero se ha tomado medidas para garantizar que los datos viajen de manera confiable de extremo a extremo.

La Capa 4 ejecuta múltiples funciones para brindar esta "calidad de servicio."

Sus funciones principales son transportar y regular el flujo de información desde el origen hasta el destino de manera confiable y precisa.

Capa de Transporte: Propósito

- Principales funciones de la capa de Transporte:
  - **Control de de flujo: extremo a extremo**
  - **Multiplexaje: datos de distintas aplicaciones transmitidos de manera simultánea mediante un medio físico único**
  - **Administración de circuitos virtuales**
  - **Verificación y recuperación de errores.**

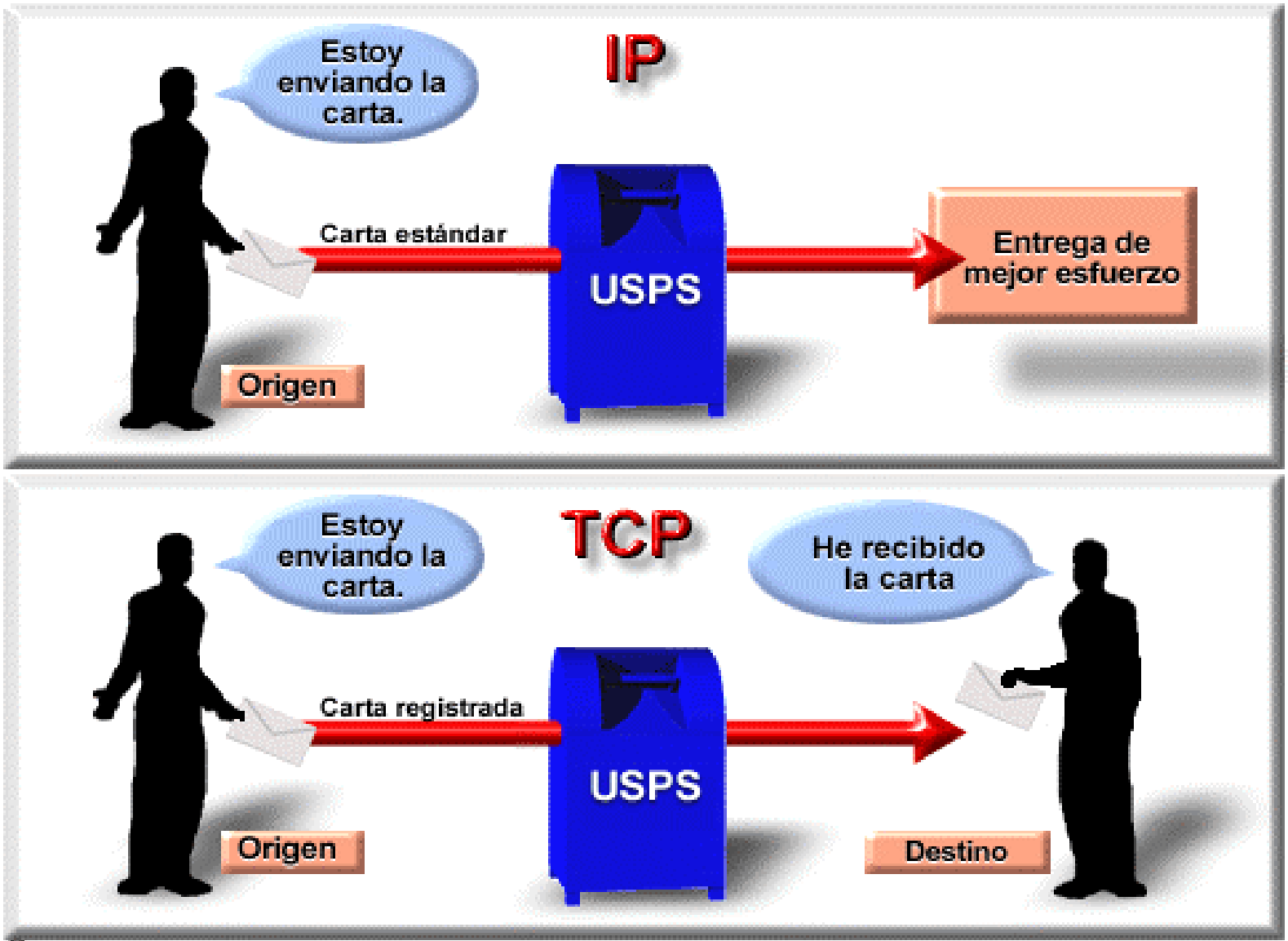
Protocolos de la Capa de Transporte

- El protocolo TCP/IP de la Capa 4 del modelo OSI consta de dos protocolos: **TCP y UDP.**
- **TCP ofrece un circuito virtual entre aplicaciones de usuario final**
- Características:
  - **Orientado a conexión**
  - **Confiable**
  - **Divide los mensajes salientes en segmentos**
  - **Reensambla los mensajes en la estación destino**
  - **Vuelve a enviar lo que no se ha recibido**
  - **Reensambla los mensajes a partir de segmentos entrantes.**

Protocolos de la Capa de Transporte

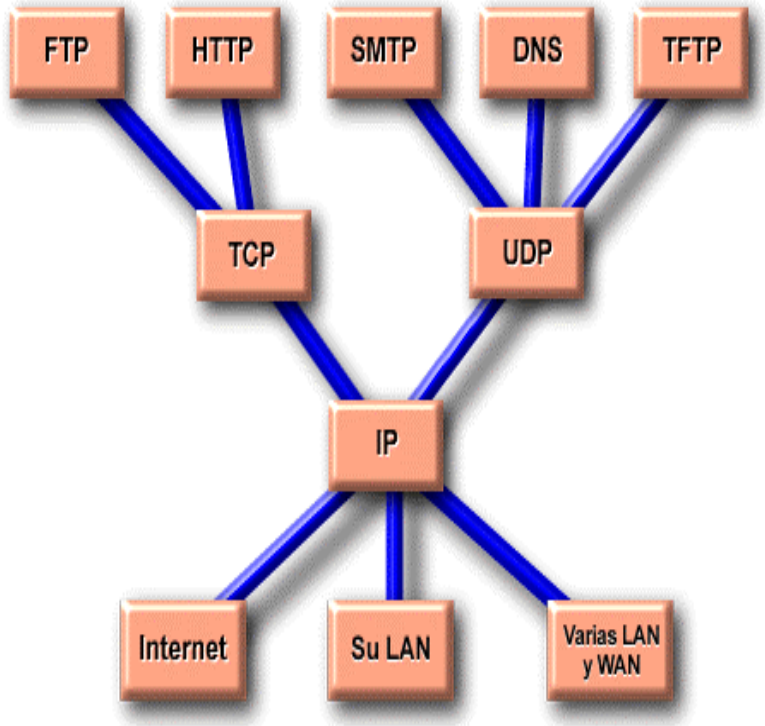
- **UDP transporta datos de manera no confiable entre hosts.**
- Características:
  - **No orientado a la conexión**
  - **Poco confiable**
  - **Transmite mensajes (llamados datagramas del usuario)**
  - **No ofrece verificación de software para la entrega de segmentos (poco confiable)**
  - **No reensambla los mensajes entrantes**
  - **No usa acuses de recibo**
  - **No proporciona control de flujo**

Comparación entre TCP e IP



TCP

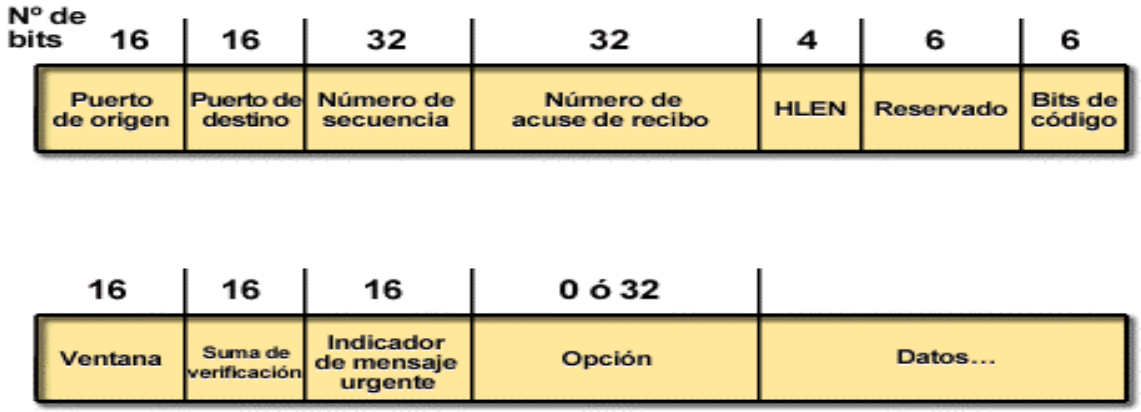
- *El Protocolo de control de transmisión (TCP)* es un protocolo de la Capa 4 (la capa de transporte).



- Es orientado a conexión
  - Datos en ráfagas
  - Confiabiliadad
  - Control de Flujo eficiente
  - Operación Full Duplex
  - Multiplexaje

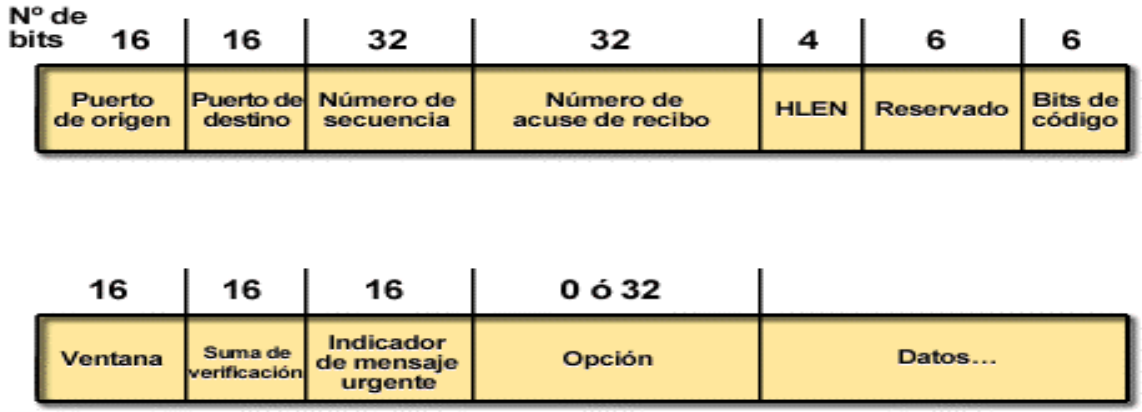
- TCP forma parte de la pila de protocolos TCP/IP

Segmento TCP



- ***Puerto origen***: **Número del puerto que realiza la llamada**
- ***Puerto destino*** : **Número del puerto que recibe la llamada**
- ***Número de secuencia***: **Número que se usa para garantizar el secuenciamiento correcto de los datos entrantes**
- ***Número de acuse de recibo***: **Próximo octeto TCP esperado**
- ***HLEN***: **Cantidad de palabras de 32 bits del encabezado**
- ***Reservado***: **Se establece en cero**
- ***Bits de código:*** **Funciones de control (Ejemplo: configuración y terminación de una sesión)**

Segmento TCP

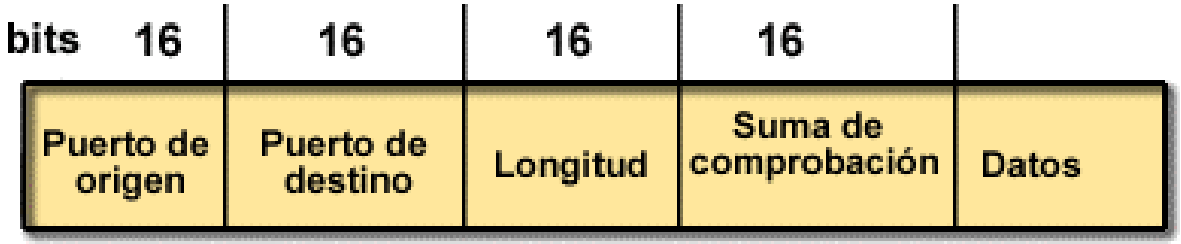


- ***Ventana***: **Cantidad de octetos que el emisor desea aceptar**
- ***Suma de comprobación***: **Suma de comprobación calculada del encabezado y de los campos de datos**
- ***Marcador urgente***: **Indica el final de los datos urgentes**
- ***Opción***: **Tamaño máximo de segmento TCP**
- ***Datos***: **Datos de protocolo de capa superior**

UDP Protocolo de Datagrama de usuario

- Es el protocolo de transporte no orientado a conexión de la pila de protocolo TCP/IP.
- UDP es un protocolo simple que intercambia datagramas, sin acuse de recibo ni entrega garantizada.
- El procesamiento de errores y retransmisión deben ser manejados por otros protocolos de capa superior.
- UDP está diseñado para las aplicaciones que no necesitan agrupar secuencias de segmentos.
- Entre los protocolos que usan UDP se incluyen: TFTP, SNMP, DHCP, DNS
- UDP, por lo tanto es una interfase entre IP y los procesos de capas superiores.

Segmento UDP



- ***Puerto origen***: **Número del puerto que realiza la llamada**
- ***Puerto destino***: **Número del puerto que recibe la llamada**
- ***Longitud***: **Longitud del segmento en bytes**
- ***Suma de comprobación***: **Suma de comprobación calculada del encabezado y de los campos de datos**
- ***Datos***: **Datos de protocolo de capa superior**

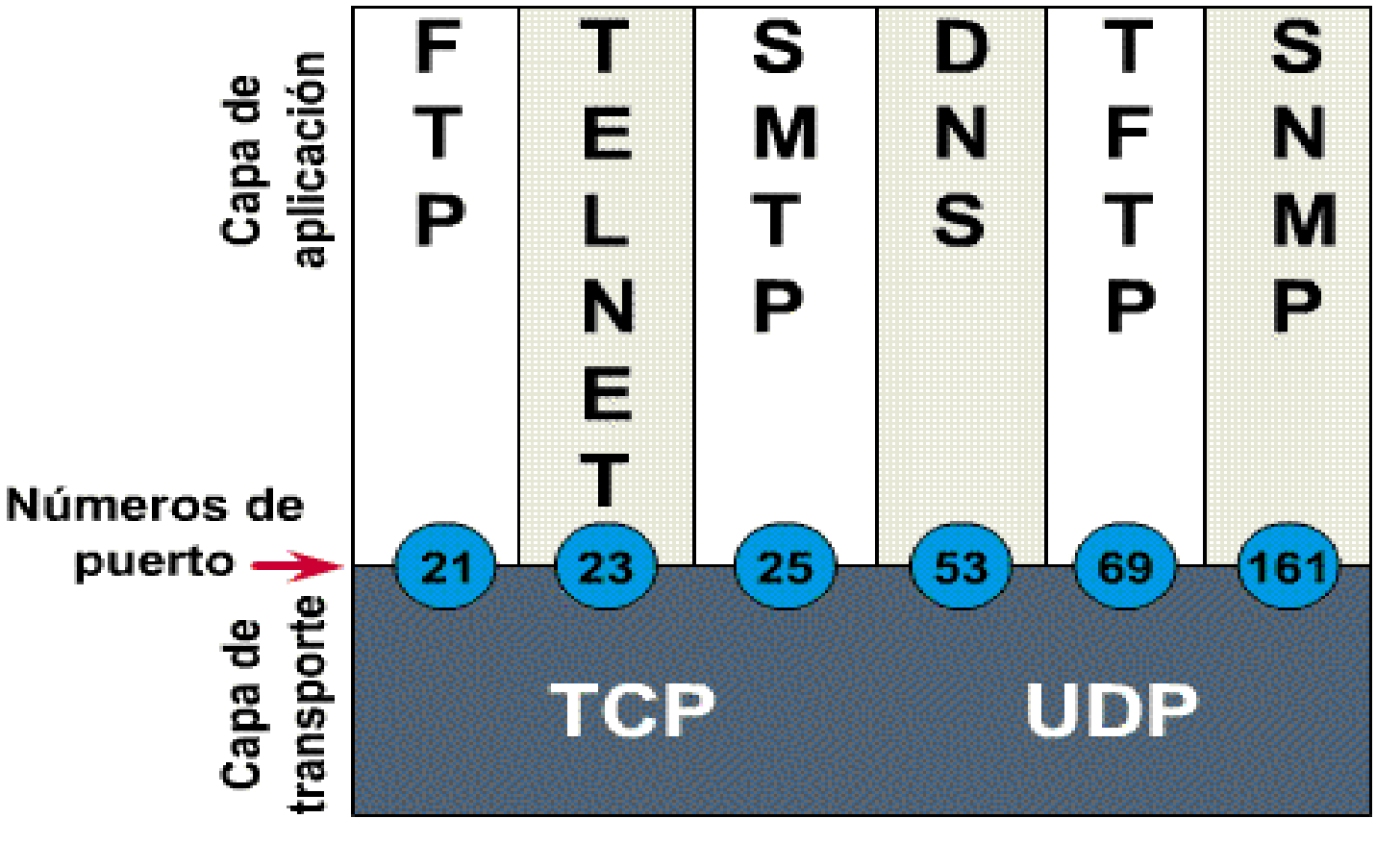
Números de Puerto

- Tanto TCP como UDP usan *números de puerto* para enviar información a las capas superiores.
- Permiten mantener un seguimiento de las distintas conversaciones que atraviesan la red al mismo tiempo.
  - Los desarrolladores de software han acordado utilizar los # de puerto conocidos definidos en RFC1700.
    - FTP utiliza el número de puerto estándar 21.
- Las aplicaciones que no tienen un # de puerto predefinido se les asignan un # de puerto un intervalo específico. Estos números de puerto se usan como direcciones origen y destino en el segmento TCP.

Números de Puerto

- Algunos puertos se reservan tanto en TCP como en UDP
- Los números de puerto tienen los siguientes intervalos asignados:
  - **Los números inferiores a 255 se usan para aplicaciones públicas.**
  - **Los números del 255 al 1023 son asignados a empresas para aplicaciones comercializables**
  - **Los números superiores a 1023 no están regulados.**
- Los sistemas finales usan números de puerto para seleccionar las aplicaciones adecuadas.
- Los números de puerto origen son asignados dinámicamente por el host origen; normalmente es un número mayor que 1023.

Números de puerto



3-Way Handshake

- Los servicios orientados a conexión se dividen en 3 fases, se les conoce como *saludo en tres direcciones* o *intercambio de señales de tres vias*.
  - ***Fase de establecimiento de la conexión***: Determina una ruta única entre origen y destino. Los recursos quedan reservados para garantizar un servicio constante.
  - ***Fase de transferencia de datos***: Los datos se transmiten secuencialmente siguiendo la ruta establecida, llegando a su destino en el orden en que se enviaron.
  - ***Fase de terminación de la conexión***: Termina la conexión entre el origen y el destino cuando ya no se necesita.

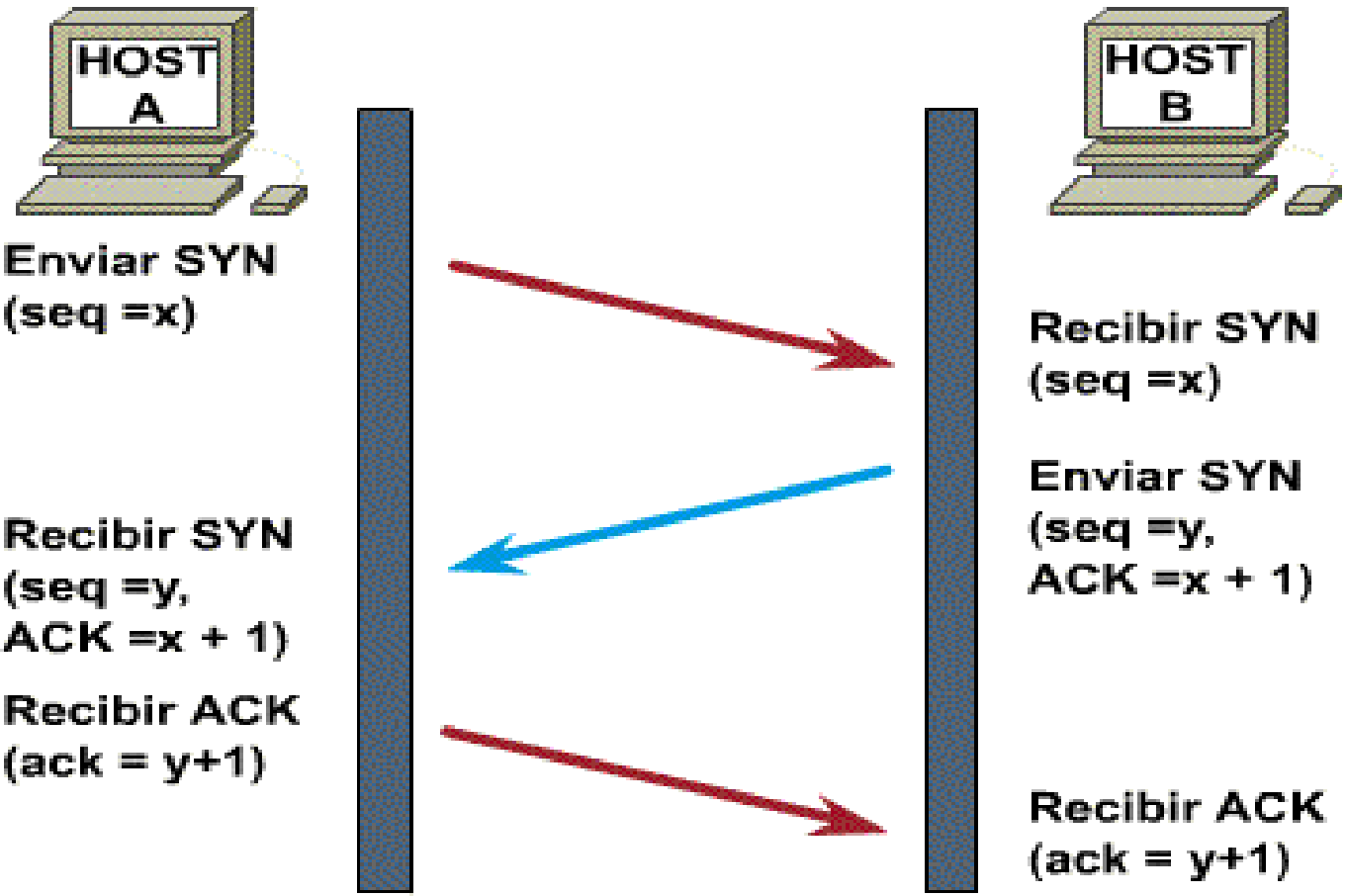
Saludo de 3 Vías TCP

- Los hosts TCP establecen una sesión orientada a conexión entre sí mediante un **3-Way Handshake**, lo cual sincroniza una conexión en ambos extremos antes de transferir datos.
- Garantiza que, si se pierden datos debido a problemas de transmisión, se puedan recuperar.
  - El host A inicia una conexión enviando un paquete con el # de secuencia inicial: X y el bit SYN activado en el encabezado para indicar una solicitud de conexión.
  - El host B recibe el SYN, graba el número de secuencia X, responde confirmando el SYN (con un ACK=X+1) e incluye su propio número de secuencia inicial Y.

Saludo de 3 Vías TCP

- El número de confirmación X + 1 significa que el host ha recibido todos los octetos enviados incluyendo el X, y espera X + 1 a continuación.
- El acuse de recibo y retransmisión positivos, o ***PAR*** es una técnica utilizada por muchos protocolos para proporcionar confiabilidad.
- Se inicializa un temporizador y espera confirmación antes de enviar un nuevo paquete, si no se recibe confirmación, el origen retransmite el paquete.
- La utilización de PAR implica un uso ***deficiente*** del ancho de banda, ya que se debe esperar la confirmación antes de enviar un nuevo paquete, y sólo se puede enviar un paquete a la vez.

Saludo de tres vías TCP/ Conexión abierta



Ventana deslizante

- **Permite un uso más eficiente del ancho de banda.**
- **El tamaño de ventana determina la cantidad de datos que se pueden transmitir antes de recibir una confirmación desde el destino.**
- **Cuanto mayor sea el tamaño de ventana (bytes), mayor será la cantidad de datos que el host puede transmitir.**
- **TCP usa *acuses de recibo de expectativa*, lo que significa que el número de confirmación se refiere al siguiente octeto esperado.**
- **La parte "deslizante" de la *ventana deslizante*, se refiere al hecho de que el tamaño de la ventana se negocia de forma dinámica durante la sesión TCP.**

Ventana deslizante

- El uso de ventanas es un mecanismo de control de flujo que requiere que el origen reciba una confirmación desde el destino después de transmitir una cantidad determinada de datos.
- Si el origen no recibe un acuse de recibo, sabe que los octetos se deben retransmitir, y que la velocidad de transmisión (tamaño de ventana) debe reducirse.
- TCP proporciona secuenciamiento de segmentos, cada datagrama se numera antes de la transmisión.
- En el host de destino el TCP reensambla los segmentos hasta formar un mensaje completo.
- Si falta algún número de secuencia en la serie, ese segmento se vuelve a transmitir.

Ventana deslizante TCP