



## Capítulo 9: Capa de transporte



## Introducción a Networks v6.0

Cisco | Networking Academy®  
Mind Wide Open™



# Capítulo 9: Secciones y objetivos

## 9.0 Introducción

### 9.1 División de una red IPv4 en subredes

- Describir el propósito de la capa de transporte en la administración del transporte de datos en la comunicación de extremo a extremo.
- Describir las características de los protocolos TCP y UDP, incluidos los números de puerto y sus usos.

### 9.2 Esquemas de direccionamiento

- Explicar la forma en que los procesos de establecimiento y finalización de sesión TCP promueven una comunicación confiable.
- Explicar la forma en que se transmiten y se reconocen las unidades de datos del protocolo TCP para garantizar la entrega.
- Describir los procesos de cliente UDP para establecer la comunicación con un servidor.
- Comparar UDP y TCP.

### 9.3 Resumen



## 9.1 Protocolos de capa de transporte



Cisco | Networking Academy®  
Mind Wide Open™



## Protocolos de la capa de transporte

# Transporte de datos

### ■ Función de la capa de transporte

- Es responsable de establecer una sesión de comunicación temporaria entre dos aplicaciones y de transmitir datos entre ellas.
- Proporciona compatibilidad con el flujo de datos orientado a la conexión, confiabilidad, control de flujo y multiplexión.

### ■ Tareas de la capa de transporte

- Realizar un seguimiento de conversaciones individuales.
- Segmentar datos y volver a armar segmentos.
- Identificar las aplicaciones.

### ■ Multiplexión de conversaciones

- Segmenta datos en fragmentos pequeños.
- Etiqueta fragmentos de datos según la conversación.

### ■ Confiabilidad de la capa de transporte

- Dos protocolos provistos: TCP y UDP.
- TCP aporta confiabilidad, UDP no.



## Protocolos de la capa de transporte

# Transporte de datos (continuación)

### ■ TCP

- Admite la confirmación de entrega de paquetes.
- Existen tres operaciones básicas que habilitan la confiabilidad con TCP:
  - La numeración y el seguimiento de los segmentos de datos transmitidos hacia un host específico desde una aplicación determinada
  - El acuse de recibo de datos
  - La retransmisión de cualquier dato sin acuse de recibo después de un período determinado

### ■ UDP

- UDP proporciona las funciones básicas para distribuir segmentos de datos entre las aplicaciones adecuadas, con muy poca sobrecarga y comprobación de datos.
- Es ideal para aplicaciones que no requieren confiabilidad.

### ■ Protocolo de la capa de transporte correcto para la aplicación adecuada

- TCP es mejor para bases de datos, navegadores web, clientes de correo electrónico, etc.
- UDP es mejor para transmisiones en vivo de audio o video, VoIP, etc.



## Protocolos de la capa de transporte

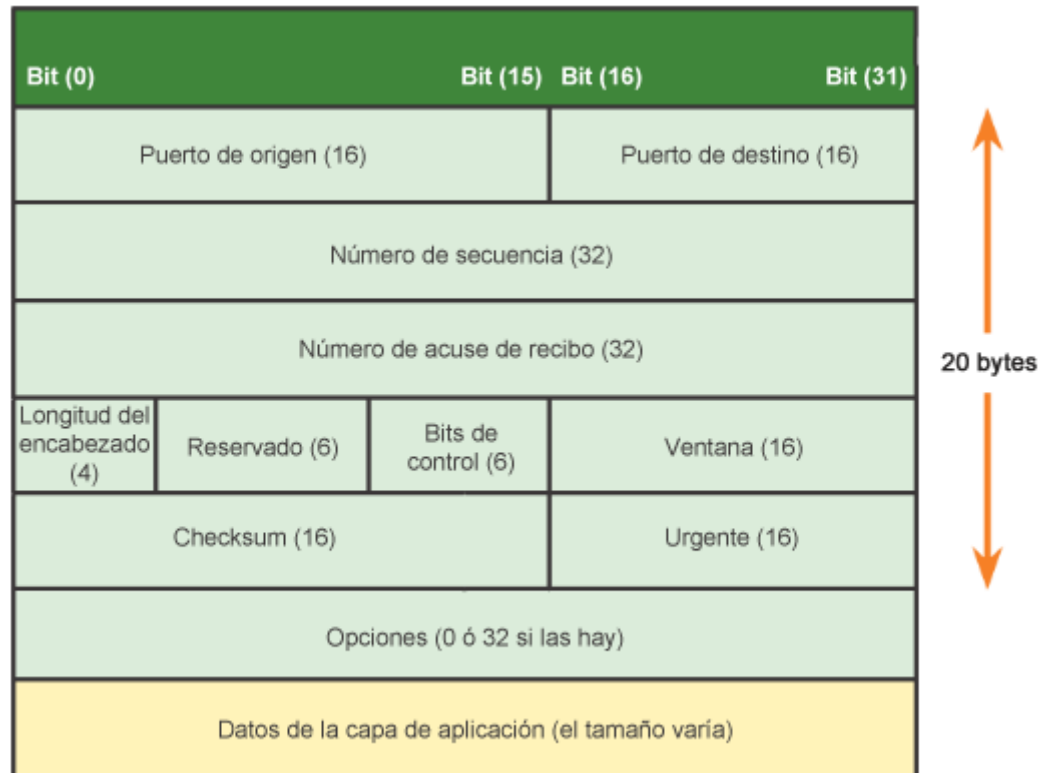
# Descripción general de TCP y UDP

### ■ Características de TCP

- Establecimiento de una sesión
- Entrega confiable
- Entrega en el mismo orden
- Control de flujo

### ■ Encabezado TCP

- TCP es un protocolo con información de estado.
- TCP agrega 20 bytes de sobrecarga en el encabezado del segmento.





## Protocolos de la capa de transporte

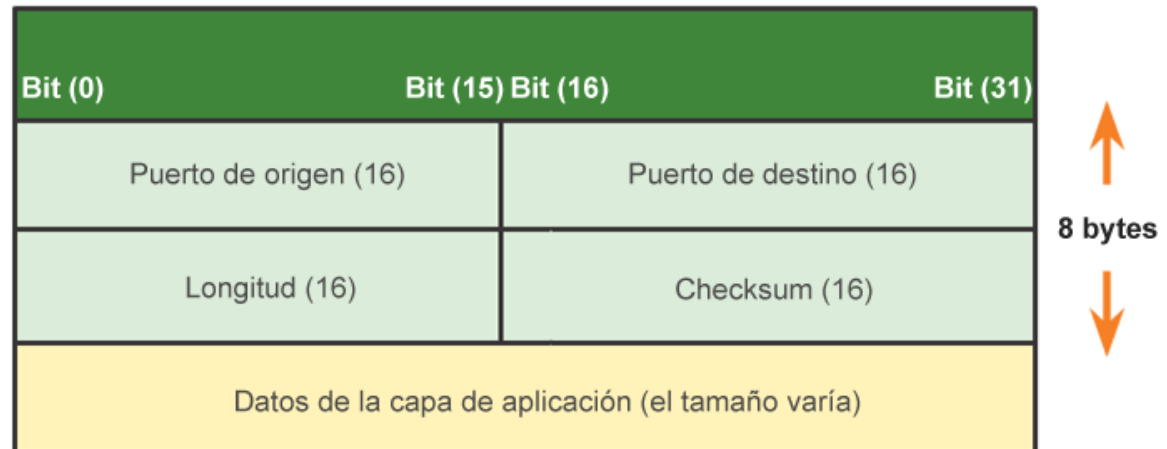
# Descripción general de TCP y UDP (continuación)

### ■ Características de UDP

- Simple y rápido.

### ■ Encabezado UDP

- UDP es un protocolo sin información de estado.
- La aplicación debe manejar la confiabilidad.
- Las porciones de comunicación en UDP se denominan datagramas.
- UDP agrega solo 8 bytes de sobrecarga.

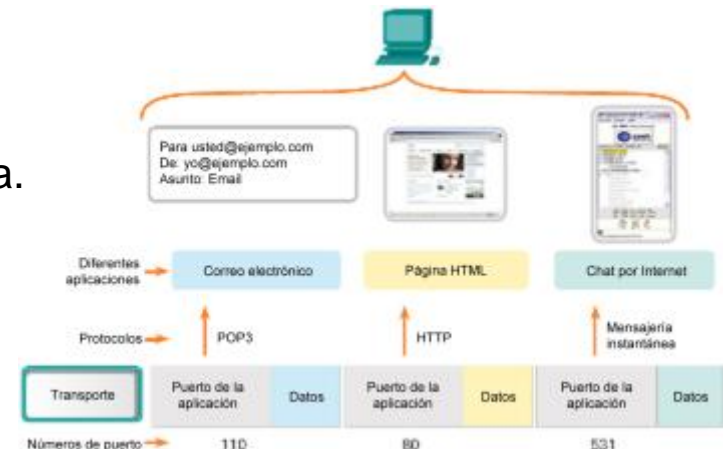




## Protocolos de la capa de transporte

# Descripción general de TCP y UDP (continuación)

- Varias conversaciones por separado
  - La capa de transporte separa y administra varias comunicaciones con diferentes requisitos de transporte.
  - Diferentes aplicaciones envían y reciben datos en la red de manera simultánea.
  - Los valores únicos de encabezado permiten que TCP y UDP administren estas distintas conversaciones simultáneas por medio de la identificación de estas aplicaciones.
  - Estos identificadores únicos son números de puertos.
- Números de puerto
  - Suelen verse en pares: puerto de origen y puerto de destino.
  - El emisor elige el puerto de origen en forma dinámica.
  - El puerto de destino se utiliza para identificar una aplicación en el servidor (destino).







## Protocolos de la capa de transporte

# Descripción general de TCP y UDP (continuación)

### ■ Pares de sockets

- La combinación de la dirección IP de origen y el número de puerto de origen, o la dirección IP de destino y el número de puerto de destino, se conoce como “socket”.
- El socket se utiliza para identificar el servidor y el servicio que solicita el cliente.
- Se combinan dos sockets para formar un par de sockets: (192.168.1.5:1099, 192.168.1.7:80).
- Los sockets permiten que varios procesos que se ejecutan en un cliente y que varias conexiones a un proceso de servidor se distingan entre sí.

### ■ Grupos de números de puerto

- La IANA ha creado tres grupos de números de puerto:
- Puertos conocidos (0 a 1023)
- Puertos registrados (1024 a 49151)
- Puertos privados y/o dinámicos (49152 a 65535)

```
C:\> netstat

Active Connections

Proto  Local Address           Foreign Address         State
TCP    kenpc:3126             192.168.0.2:netbios-ssn ESTABLISHED
TCP    kenpc:3158             207.138.126.152:http   ESTABLISHED
TCP    kenpc:3159             207.138.126.169:http   ESTABLISHED
TCP    kenpc:3160             207.138.126.169:http   ESTABLISHED
TCP    kenpc:3161             sc.msn.com:http        ESTABLISHED
TCP    kenpc:3166             www.cisco.com:http     ESTABLISHED

C:\>
```

### ■ El comando netstat

- Netstat permite que un usuario vea las conexiones activas en un host.
- Netstat también muestra el proceso que está utilizando la conexión.



## 9.2 TCP y UDP



Cisco | Networking Academy®  
Mind Wide Open™



## Protocolos de la capa de transporte

# Proceso de comunicación en TCP

### ■ Procesos del servidor TCP

- Cada proceso de aplicación que se ejecuta en el servidor utiliza un número de puerto.
- Un servidor individual no puede tener dos servicios asignados al mismo número de puerto dentro del mismo servicio de la capa de transporte.
- Una aplicación de servidor activa asignada a un puerto específico se considera abierta.
- Toda solicitud entrante de un cliente dirigida a un puerto abierto se acepta y se procesa en la aplicación de servidor conectada a dicho puerto.
- Pueden existir muchos puertos abiertos simultáneamente en un servidor, uno para cada aplicación de servidor activa.

### ■ Establecimiento de conexiones TCP

- Una conexión TCP se establece en tres pasos:
  - El cliente de origen solicita una sesión de comunicación de cliente a servidor con el servidor.
  - El servidor acusa recibo de la sesión de comunicación de cliente a servidor y solicita una sesión de comunicación de servidor a cliente.
  - El cliente de origen acusa recibo de la sesión de comunicación de servidor a cliente.



## Protocolos de la capa de transporte

# Proceso de comunicación en TCP (continuación)

### ■ Terminación de una sesión TCP

- El indicador TCP FIN se utiliza para terminar una conexión TCP.
  - Cuando el cliente no tiene más datos para enviar en la transmisión, envía un segmento con el indicador FIN establecido.
  - El servidor envía un ACK para acusar recibo del FIN para terminar la sesión de cliente a servidor.
  - El servidor envía un FIN al cliente para terminar la sesión de servidor a cliente.
  - El cliente responde con un ACK para dar acuse de recibo del FIN desde el servidor.
  - Una vez reconocidos todos los segmentos, la sesión se cierra.

### ■ Análisis del enlace de tres vías de TCP

- El enlace de tres vías:
  - Establece que el dispositivo de destino está presente en la red.
  - 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.
  - Informa al dispositivo de destino que el cliente de origen intenta establecer una sesión de comunicación en dicho número de puerto

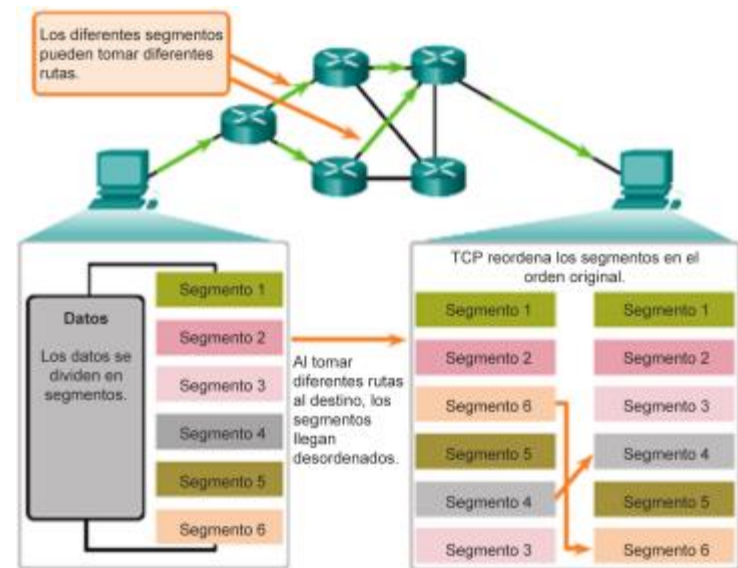


## Protocolos de la capa de transporte

# Confiabilidad y control de flujo

### ■ Confiabilidad de TCP: entrega ordenada

- Los segmentos TCP utilizan números de secuencia para identificar exclusivamente a cada segmento y dar acuse de recibo de ellos, hacer un seguimiento del orden de segmentos e indicar la forma en que se vuelven a armar y a reordenar los segmentos recibidos.
- Durante la configuración de la sesión TCP, se elige un número de secuencia inicial (ISN) al azar. Después, el ISN se incrementa con el número de bytes transmitidos.
- El proceso de TCP receptor reúne los datos de los segmentos en el búfer hasta que se reciban y se vuelvan a armar todos los datos.
- Los segmentos que no se reciben en el orden correcto se conservan para su posterior procesamiento.
- Los datos se entregan a la capa de aplicación solo cuando se hayan recibido y vuelto a armar por completo.





## Protocolos de la capa de transporte

# Confiabilidad y control de flujo (continuación)

- Control del flujo en TCP: Tamaño de la ventana y acuses de recibo
  - TCP también proporciona mecanismos para el control del flujo.
  - El control del flujo asegura que las terminales TCP puedan recibir y procesar datos de manera confiable.
  - TCP administra el control del flujo mediante el ajuste de la velocidad del flujo de datos entre el origen y el destino en una sesión determinada.
  - La función de control del flujo en TCP depende de un campo del encabezado TCP de 16 bits denominado Tamaño de ventana. El tamaño de ventana es la cantidad de bytes que el dispositivo de destino de una sesión TCP puede aceptar y procesar al mismo tiempo.
  - El origen y el destino TCP acuerdan el tamaño de ventana inicial cuando se establece la sesión TCP.
  - De ser necesario, los terminales TCP pueden ajustar el tamaño de ventana durante una sesión.



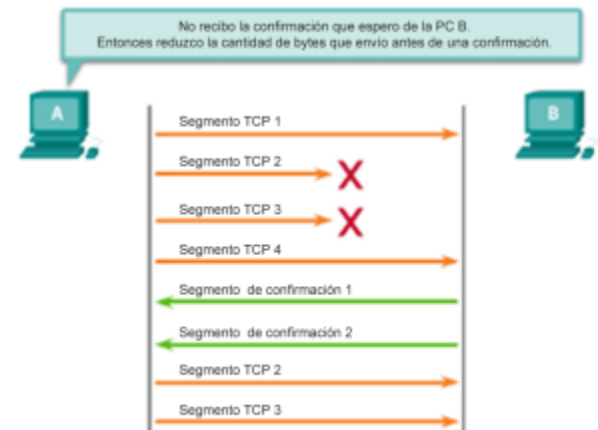
El tamaño de la ventana determina la cantidad de bytes que se pueden enviar para recibir un reconocimiento.  
El número de reconocimiento es el número del siguiente byte esperado.



## Protocolos de la capa de transporte

# Confiabilidad y control de flujo (continuación)

- Control del flujo de TCP: prevención de congestiones
  - La congestión de red suele dar como resultado el descarte de paquetes.
  - Los segmentos TCP que no se entregan activan la retransmisión. La retransmisión de segmentos TCP puede empeorar la congestión.
  - El origen puede estimar un nivel determinado de congestión de red al observar la velocidad a la que se envían los segmentos TCP sin acusar recibo.
  - El origen puede reducir la cantidad de bytes que envía antes de recibir un acuse de recibo cuando se detecta la congestión.
  - El origen reduce la cantidad de bytes que envía sin acuse de recibo y no el tamaño de ventana, el cual lo determina el destino.
  - El destino suele desconocer la congestión de red y considera que no es necesario sugerir un nuevo tamaño de ventana.



Los números de confirmación son para el próximo byte esperado y no para un segmento. El número de segmento solo se utiliza aquí con fines de simplificación.





## Protocolos de la capa de transporte

# Comunicación en UDP

- UDP: Baja sobrecarga vs. confiabilidad
  - UDP tiene mucha menos sobrecarga que TCP.
  - UDP no está orientado a la conexión y no proporciona los mecanismos sofisticados de retransmisión, secuenciación y control del flujo.
  - Las aplicaciones que ejecutan UDP aún pueden utilizar la confiabilidad, pero se debe implementar en la capa de aplicación.
  - Sin embargo, UDP no es inferior.
- Rearmado de datagramas UDP
  - UDP simplemente vuelve a armar los datos en el orden en el que se recibieron.
  - Si es necesario, la aplicación debe identificar la secuencia correcta.
- Procesos y solicitudes del servidor UDP
  - A las aplicaciones de servidor basadas en UDP se les asignan números de puerto conocidos o registrados.
  - Las solicitudes que se reciben en un puerto específico se reenvían a la aplicación adecuada según los números de puerto.





## Protocolos de la capa de transporte

# Comunicación en UDP (continuación)

### ■ Procesos de cliente UDP

- La comunicación entre cliente y servidor UDP también se inicia con una aplicación cliente.
- El proceso de cliente UDP selecciona de manera dinámica un número de puerto y lo utiliza como puerto de origen.
- Por lo general, el puerto de destino es el número de puerto conocido o registrado que se asigna al proceso de servidor.
- Se utiliza el mismo par de puertos de origen o destino en el encabezado de todos los datagramas usados en la transacción.
- En la devolución de datos del servidor al cliente, se invierten los números de puerto de origen y de destino en el encabezado del datagrama.



# Protocolos de la capa de transporte

## TCP o UDP

### ■ Aplicaciones que utilizan TCP

- TCP maneja todas las tareas relacionadas con la capa de transporte.
- Esto hace que la aplicación no tenga que administrar ninguna de dichas tareas.
- Las aplicaciones simplemente pueden enviar el flujo de datos a la capa de transporte y utilizar los servicios de TCP.

### ■ Aplicaciones que utilizan UDP

- Aplicaciones multimedia y de video en vivo: pueden tolerar cierta pérdida de datos, pero requieren demoras breves o que no haya demoras. Los ejemplos incluyen VoIP y la transmisión de vídeo en vivo.
- Aplicaciones con solicitudes y respuestas simples: aplicaciones con transacciones simples en las que un host envía una solicitud y existe la posibilidad de que reciba una respuesta o no. Los ejemplos incluyen DNS y DHCP.
- Aplicaciones que manejan la confiabilidad por sí mismas: comunicaciones unidireccionales en las que no se requiere control de flujo, detección de errores, acuses de recibo ni recuperación de errores, o en las que la aplicación pueda ocuparse de estas tareas.  
Los ejemplos incluyen SNMP y TFTP.



## 9.3 Resumen



Cisco | Networking Academy®  
Mind Wide Open™



## Resumen del capítulo

# Resumen

- Implementar un esquema de direccionamiento IPv4 para permitir una conectividad completa en una red de pequeña o mediana empresa.
- Dado un conjunto de requisitos, implementar un esquema de direccionamiento VLSM para proporcionar conectividad a usuarios finales en una red pequeña o mediana.
- Explicar las consideraciones de diseño para implementar IPv6 en una red comercial.

