

Fundamentos de Networking - Parte III

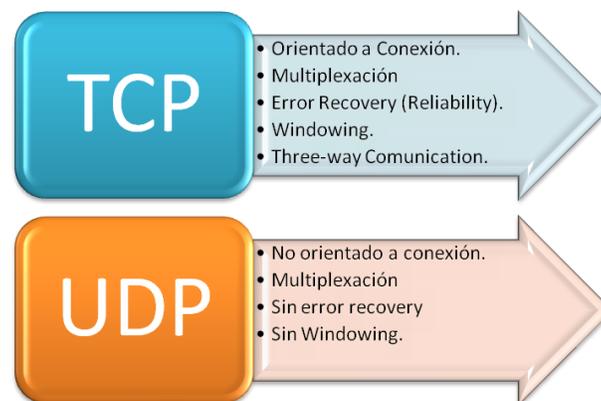
TCP vs UDP

Encapsulamiento de la Data

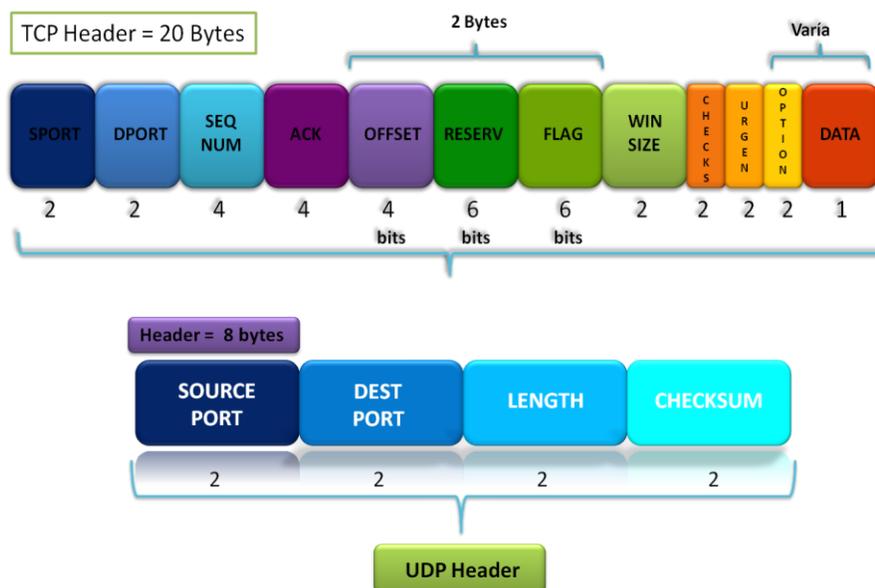
Modelos de Resolución de Problemas de Conexiones.

TCP vs UDP:

Todas las comunicaciones de las redes y de internet se establecen gracias a **Protocolos de Comunicación**. En el siguiente gráfico y describimos las principales características de **TCP** y **UDP**:



A continuación mostramos la cabecera de cada **protocolo de comunicación**, siendo **TCP** la **cabecera** de mayor tamaño con **20 bytes** en comparación con la cabecera **UDP** que tiene un tamaño de **8 bytes**.



Por ser **TCP** un **Protocolo Orientado a la Conexión** requiere de más campos donde los equipos puedan insertar data para efectos de sincronismo y mantenimiento de la conexión a diferencia de **UDP** que no requiere de sincronismo porque es un protocolo que deja la administración de la conexión y la corrección de errores como responsabilidad de otras capas.

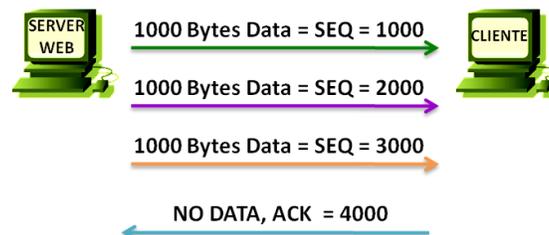
TCP ofrece una serie de funcionalidades entre las cuales tenemos:

MULTIPLEXACION: Esta es la única función compartida por **UDP** y **TCP**. Permite a los hosts receptores escoger la aplicación correcta para la cual la data ha sido enviada basado en el **número de puerto destino**.

N°. PUERTO	PROTOCOLO	APLICACION
20	TCP	FTP Data
21	TCP	FTP Control
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	TCP/UDP	DNS
67/68	UDP	DHCP
69	UDP	TFTP
80	TCP	HTTP(W) ³
110	TCP	POP3
161	UDP	SNMP
443	TCP	SSL
16.384-30.767	UDP	RIP VoIP and Video

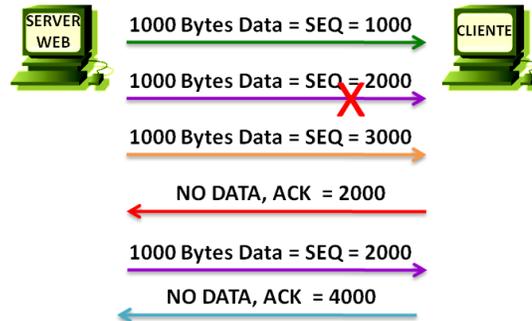
CORRECCION DE ERRORES: Es conocido como un mecanismo de **Confabilidad**. **TCP** provee **recuperación de errores** durante la sesión de transmisión de datos entre dos dispositivos finales que han establecido una conexión.

Para este caso los campos de **SEQUENCE** y **ACKNOWLEDGMENT (ACK)** de la cabecera **TCP** son usados para llevar un registro de cada byte transferido para asegura que los bytes perdidos sean **retransmitidos**.

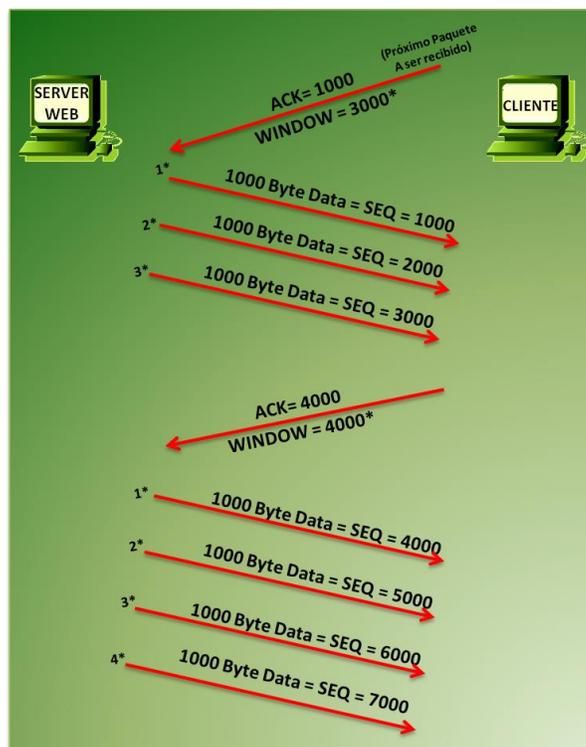


El escenario anterior es un escenario donde no existe perdida de paquetes y es llamado **FORWARD ACKNOWLEDGMENT**.

En el siguiente gráfico veremos la **pérdida de un paquete** y la solicitud del mismo, donde se puede observar la **recuperación de la data** realizada por **TCP** y el mecanismo de recuperación de errores vía **ACK**.

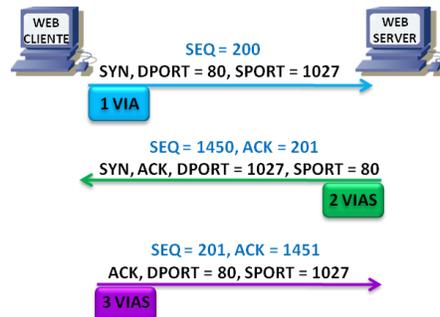


CONTROL DE FLUJO: El control de flujo en **TCP** se logra mediante el mecanismo **WINDOWING**, el cual, es un **proceso entre dos dispositivos finales que negocian dinámicamente el tamaño de la ventana de transferencia de Datos al momento de establecer la conexión**. El tamaño de la ventana se especifica en el campo **WINDOW** de la cabecera **TCP**. Los hosts negocian dinámicamente el tamaño de la ventana durante la vida de la conexión y la ventana pueda incrementar su tamaño hasta el valor máximo en bytes que es de **65.535** o hasta que ocurra un error.



ESTABLECIMIENTO Y TERMINACION DE LA CONEXION: Se basa en el uso del **Puerto Destino** y el **Puerto Origen** de las aplicaciones, así como en el uso de **Sincronismo**, como lo son los números de **Secuencia** y los **Mensajes ACK**.

Para establecer una conexión en **TCP** se realiza una “**Conexión de Tres Vías**”.



Para **finalizar** una conexión **TCP** se utiliza una **Secuencia de 4 vías** y un **FLAG** adicional llamado **FIN bit**.

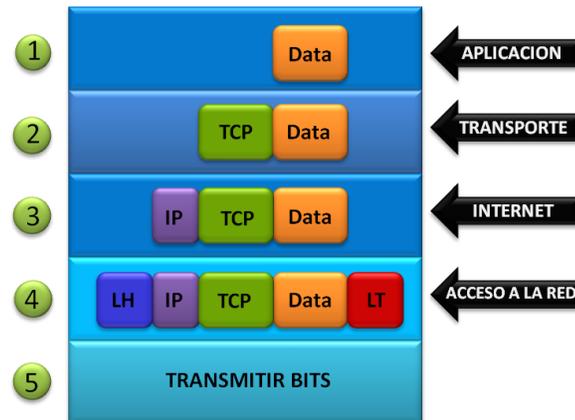


Finalmente **TCP** ofrece soporte a la **Segmentación y Ordenamiento de la Data** durante la transferencia, permitiendo que la data sea organizada en el mismo orden de byte en el host que recibe la información desde el origen.

[Volver al inicio...](#)

Encapsulamiento de la Data

Cada capa del **Modelo TCP/IP** agrega su propia información de Cabecera en cuanto la data viaja hacia abajo a través de las capas. Este es un procedimiento de **Cinco pasos** el cual se describe en la imagen siguiente:



“Solo la **Capa de Enlace de Datos** agrega **dos campos**, una **Cabecera** y un **Trailer**”.



[Volver al inicio...](#)

Modelos de Resolución de Problemas de Conexiones

Existen **3 métodos principales** para la **Resolución de Problemas en una red**, utilizando como referencia el [Modelo OSI](#). Estos métodos, los describimos a continuación:

De Abajo hacia Arriba: Este modelo es utilizado para **resolver problemas de internetwork**. Consiste en **ir desde la capa física hacia la capa de aplicaciones**.

De Arriba hacia Abajo: Este modelo es utilizado para **resolver problemas cuando se considera y cree que el problema es derivado por una aplicación**. Consiste en **ir desde la capa de aplicación hasta la capa física**.

Divide y Vencerás: Basado en la **experiencia del usuario**. Consiste en **documentar los síntomas y utilizar esta información para suponer en cual capa OSI comenzar la resolución del problema**. **No tiene una capa de inicio** definida y se puede **ir de arriba hacia abajo en las capas y viceversa, en cualquier momento**.

[Volver al inicio...](#)

The House of Routing

<https://cursos.thehouseofrouting.com>