

[[archivos/ejercicios/Enunciados/CheckSumUDP.pdf|CheckSumUDP]] ![[archivos/ejercicios/Solucion...]]

[[archivos/ejercicios/Enunciados/UtilizacionARQ.pdf|UtilizacionARQ]] ![[archivos/ejercicios/Solucion...]]

[[archivos/ejercicios/Enunciados/ProtoTipoTCP.pdf|ProtoTipoTCP]]

![[archivos/ejercicios/SolucionesPropias/QuirogaLinaresAdrian_ProtoTipoTCP.pdf|QuirogaLinaresAdri...]]

[[archivos/ejercicios/Enunciados/CongestionTCP.pdf|CongestionTCP]] ### 1. Mecanismos de Control de Congestión en TCP

El protocolo TCP utiliza varios mecanismos para manejar la congestión en redes, permitiendo que el emisor adapte su tasa de transmisión según las condiciones de la red. A continuación, se explican los mecanismos principales:

a. Inicio Lento (Slow Start) El objetivo del **Inicio Lento** es evitar una sobrecarga repentina de la red. TCP comienza con una **ventana de congestión (cwnd)** pequeña, generalmente establecida en un valor mínimo (por ejemplo, 1 o 2 segmentos). Esta ventana determina el número máximo de bytes en tránsito permitidos en la red sin recibir un acuse de recibo (ACK) del receptor.

1. La **ventana de congestión** se incrementa exponencialmente: con cada ACK recibido, TCP incrementa **cwnd** en el mismo tamaño de segmento transmitido.
2. Esto duplica **cwnd** en cada ronda, hasta alcanzar el umbral de **ssthresh** (slow start threshold).
3. Cuando **cwnd** alcanza **ssthresh**, el protocolo cambia al siguiente mecanismo, **Incremento Aditivo/Decremento Multiplicativo**.

b. Incremento Aditivo/Decremento Multiplicativo (Congestion Avoidance)

Este mecanismo se encarga de **evitar la congestión** de manera gradual cuando la red se encuentra estable.

1. Cuando **cwnd** ha alcanzado o superado el valor de **ssthresh**, TCP incrementa la ventana de congestión de forma aditiva, en lugar de exponencial.
2. Este incremento es lineal, aumentando **cwnd** en una unidad (el tamaño de un segmento) por cada RTT.
3. Si ocurre una pérdida de paquetes, TCP asume que hay congestión en la red y reduce **cwnd** de forma drástica (por ejemplo, estableciendo **cwnd = cwnd / 2**) y ajusta **ssthresh** a este nuevo valor de **cwnd** reducido.

Este enfoque, conocido como **Incremento Aditivo/Decremento Multiplicativo (AIMD)**, permite a TCP adaptarse a las condiciones de la red de forma gradual.

c. Recuperación Rápida (Fast Recovery) El mecanismo de **Recuperación Rápida** ayuda a mejorar el rendimiento de TCP cuando se detecta una pérdida de paquetes sin que haya una interrupción total de la transmisión.

1. TCP interpreta tres **ACK duplicados** consecutivos como una pérdida de paquete.
2. En lugar de volver al Inicio Lento, TCP reduce **cwnd** a la mitad (multiplicación por 0.5) y ajusta **ssthresh** a este valor reducido.

3. Durante la **Recuperación Rápida**, TCP utiliza la ventana de congestión actualizada y continúa enviando nuevos paquetes sin esperar el fin del proceso de Inicio Lento.
4. Cuando recibe el siguiente ACK no duplicado (confirmando la llegada de un nuevo paquete), TCP sale de la Recuperación Rápida y reinicia **cwnd** a **ssthresh**.

Este mecanismo permite que TCP recupere la transmisión rápidamente sin un reinicio completo de **cwnd**, lo cual reduce las caídas drásticas en el rendimiento de la transmisión.

2. Notificación Explícita de Congestión (ECN)

La **Notificación Explícita de Congestión (ECN)** es una técnica para informar de la congestión sin que haya una pérdida de paquetes, permitiendo que el emisor ajuste la tasa de transmisión de manera preventiva.

1. **Función de ECN en la cabecera IP y TCP:** Los routers que detectan congestión marcan los paquetes con bits especiales en la cabecera IP (bits ECN-Capable y Congestion Experienced).
2. **Manejo de la congestión:** El receptor, al identificar estos bits, notifica al emisor que ha recibido un paquete con marca de congestión.
3. **Ajuste de la ventana de congestión:** El emisor, al recibir esta notificación, reduce su ventana de congestión de manera similar a cuando detecta pérdida de paquetes, disminuyendo la tasa de transmisión antes de que se produzca un desbordamiento de la red.

ECN mejora la eficiencia al reducir la pérdida de paquetes y la latencia de la red, lo cual beneficia especialmente a redes donde los paquetes se acumulan antes de ser descartados.