

# Nuove Versioni Protocolli Di Trasporto

- **TCP Cubic:**

Il *TCP CUBIC* è una variante del protocollo TCP progettata per migliorare le prestazioni del controllo della congestione, in particolare nelle reti ad alta larghezza di banda e latenza elevata. Ecco i punti chiave che descrivono il funzionamento e i vantaggi del TCP CUBIC:

## Funzionamento del TCP CUBIC

### 1. Probing per la Banda Disponibile:

- Dopo aver rilevato una perdita di pacchetti, TCP CUBIC riduce la velocità di invio (finestra di congestione) a metà del valore massimo raggiunto prima della perdita ( $W_{max}$ ).
- Inizialmente, TCP CUBIC aumenta la velocità di invio rapidamente fino a raggiungere di nuovo  $W_{max}$ .
- Man mano che si avvicina a  $W_{max}$ , l'aumento della velocità di invio diventa più lento per evitare di sovraccaricare la rete.

### 2. Utilizzo della Funzione Cubica:

- La finestra di congestione ( $W$ ) aumenta in base al tempo trascorso dalla rilevazione dell'ultimo evento di congestione, seguendo una curva cubica.
- Punto temporale in cui la finestra di congestione raggiungerà di nuovo  $W_{max}$ . Questo punto è configurabile e aiuta a modulare la crescita della finestra di congestione.

### 3. Adattamento alla Congestione:

- TCP CUBIC è progettato per mantenere una crescita della finestra di congestione che sia aggressiva quanto basta per sfruttare la capacità disponibile, ma abbastanza prudente da evitare un sovraccarico della rete.
- Quando la finestra di congestione è lontana da  $W_{max}$ , l'aumento è più grande; quando è vicina, l'aumento è più piccolo e cauto.

- **BBR:**

BBR (Bottleneck Bandwidth and Round-trip propagation time) è un algoritmo di controllo della congestione per il protocollo TCP, sviluppato da Google. Esso è progettato per migliorare le prestazioni della rete misurando e utilizzando in modo più accurato la capacità effettiva della rete, rispetto agli approcci tradizionali basati su perdita di pacchetti o latenze crescenti. Ecco in cosa consiste:

### Principi Fondamentali di BBR

1. Stima della Capacità di Banda:

- BBR misura la larghezza di banda disponibile (bottleneck bandwidth) utilizzando il rate di invio e ricezione dei pacchetti.
- Calcola la massima velocità a cui i pacchetti possono essere inviati attraverso il collegamento senza causare congestione.

2. Tempo di Propagazione di Andata e Ritorno:

- Misura il round-trip time (RTT), che è il tempo impiegato da un pacchetto per viaggiare dal mittente al destinatario e tornare.
- Utilizza il RTT per determinare la latenza base del percorso di rete, che rappresenta la minima latenza osservata.

3. Ciclo di Probing:

- BBR esegue cicli regolari di probing per rilevare cambiamenti nella capacità di banda e nella latenza di rete.
- Durante il ciclo, BBR varia la velocità di invio dei pacchetti per trovare un equilibrio ottimale tra utilizzo della larghezza di banda e minimizzazione della latenza.

- **Indirect TCP (I-TCP):**

L'*indirect TCP* (I-TCP) è una variante del protocollo TCP progettata per migliorare la performance delle comunicazioni in reti mobili, in particolare quando il dispositivo mobile si sposta tra diverse celle o reti. Il concetto chiave di I-TCP è la divisione della connessione TCP in due segmenti distinti: uno fisso e uno mobile. Ecco in cosa consiste e come funziona:

### Funzionamento dell'Indirect TCP

1. Segmentazione della Connessione:

- La connessione TCP originale viene divisa in due segmenti separati.
  - Segmento fisso: tra il server applicativo e un agente di supporto mobile (MSA) fisso.
  - Segmento mobile: tra l'MSA e il dispositivo mobile (MD).
- Questa divisione permette di gestire separatamente i problemi di mobilità e quelli di trasporto, migliorando le prestazioni complessive.

2. Agente di Supporto Mobile (MSA):

- Un punto di accesso fisso (tipicamente situato in una stazione base o in un gateway) che funge da intermediario tra il server e il dispositivo mobile.
- Gestisce il segmento fisso della connessione TCP con il server e il segmento mobile con il dispositivo mobile.

3. Indipendenza dei Segmenti:

- Ogni segmento TCP può essere ottimizzato separatamente. Ad esempio, il segmento mobile può essere configurato per tollerare ritardi più elevati e perdite di pacchetti, tipici delle reti wireless.
- Il segmento fisso mantiene la connessione stabile con il server, anche se il dispositivo mobile si sposta o cambia rete.

- **QUIC:**

QUIC (Quick UDP Internet Connections) è un protocollo di trasporto sviluppato da Google, progettato per migliorare le prestazioni delle applicazioni web. Combina i vantaggi di TCP, TLS e HTTP/2, offrendo una soluzione più efficiente per il trasferimento dei dati su Internet. Ecco una panoramica delle sue caratteristiche principali e del suo funzionamento:

### Caratteristiche Principali di QUIC

1. Basato su UDP:

- Utilizza UDP come trasporto sottostante, consentendo una gestione più flessibile dei flussi di dati rispetto a TCP.
- Evita i problemi di ritardo associati all'handshake di connessione di TCP.

2. Connessioni Veloci:

- QUIC riduce significativamente la latenza iniziale grazie a un handshake crittografico che richiede solo un round-trip.
- Supporta la ripresa delle connessioni interrotte senza la necessità di un nuovo handshake completo, riducendo ulteriormente i tempi di ritardo.

3. Sicurezza Integrata:

- Integra TLS 1.3, garantendo che tutte le connessioni siano crittografate end-to-end.
- Fornisce autenticazione e crittografia dei dati per migliorare la sicurezza.

4. Multiplexing di Flussi:

- Permette a più flussi di dati di viaggiare in parallelo all'interno della stessa connessione QUIC, migliorando l'efficienza.
- I flussi di dati sono indipendenti, quindi la perdita di pacchetti in un flusso non influisce sugli altri flussi.

5. Gestione della Congestione e del Controllo del Flusso:

- Utilizza algoritmi di controllo della congestione avanzati per adattarsi meglio alle condizioni di rete variabili.
- Implementa meccanismi per il controllo del flusso che permettono una gestione più efficace delle risorse di rete.

**ADOPERA MESSAGGI FEC INVECE DI REINVIARE I  
PACCHETTI**