1. **Il candidato descriva le procedure di Fast Retransmit e Fast Recovery del TCP. Il candidato mostri anche un esempio applicativo delle due procedure.**

La procedura di Fast Retransmit migliora le prestazioni se si perde un singolo segmento. Il ricevitore che riceve un segmento fuori sequenza invia un ACK per ogni sequenza ricevuta finché non riceve il segmento mancante. Quando la sorgente riceve un ACK può significare o che il segmento è in ritardo o che si è perso. Dunque, per essere sicuri che l’ACK si sia perso, il segmento viene ritrasmesso solo una volta che la sorgente riceve 3 ACK duplicati per lo stesso segmento e un ACK normale anche se il timeout non è scaduto.

La procedura di Fast Recovery è associata al Fast Retransmit per evitare la congestione, essa salta la fase di slow start.

Si pone sstresh=cwdn/2 (sstresh è il slow start threshold); Viene ritrasmesso il segmento perduto;

1. **Il candidato mostri le maggiori differenze dei protocolli ARQ conosciuti. In particolare, si dovrà evidenziare il comportamento dei protocolli in caso di perdita. Il candidato può utilizzare esempi pratici al fine di evidenziare i meccanismi di funzionamento.**

Quando una stazione riceve il pacchetto inviatole da un’altra stazione, per prima cosa deve controllare se il pacchetto presenta errori, a questo scopo può far uso di protocolli detti ARQ (automatic repeat request).

I protocolli ARQ da noi conosciuti sono : Stop and wait, Continuous RQ.

* STOP AND WAIT: la stazione sorgente una volta trasmesso il generico pacchetto, deve necessariamente aspettare il pacchetto di riscontro prima di poter riprendere la trasmissione, se il riscontro ottenuto è un NAK (Negative ACK) oppure il tempo massimo (time out) scade, il pacchetto deve essere ritrasmesso fin quando non si riceverà un riscontro positivo (ACK).
* CONTINUOUS RQ : (operano su pacchetti ordinati numericamente) La stazione sorgente può emettere un certo numero di pacchetti senza dover attendere conferma della ricezione di essi, ma inviato il numero massimo di pacchetti (detto finestra) dovrà fermarsi e attendere il riscontro del primo di tali pacchetti prima di poter ritrasmettere.

Se il riscontro ottenuto è negativo (o il Tout è scaduto) ci sono due modi di operare:

* GO-BACK-N : la sorgente ritrasmette tutti i pacchetti a partire da quello che la destinazione ha segnalato come errore
* SELECTIVE REPEAT: la sorgente ritrasmette solo il pacchetto contrassegnato come errato

1. **Il candidato illustri il comportamento del protocollo TCP in caso di rilevazione di congestione. Si descriva in dettaglio come il TCP adatta la finestra di congestione ai diversi casi che si possono verificare**

Se l’errore è causato dalla perdita di molteplici pacchetti allora si procede con una fase detta di CA. A seguito della rilevazione della perdita che avviene dopo lo scadere del RTO, la finestra CW viene riportata ad 1 e si impone una nuova soglia SSTH=CW/2, per cui inizia una nuova fase di SS fino alla soglia e dopo inizia una fase di CA in cui la finestra CW viene linearmente incrementata di 1 fino all’invio di tutti i dati o al raggiungimento della soglia iniziale.

Se l’errore è causato invece dalla perdita di un solo pacchetto si può ricorrere a tecniche migliori dette:

* FAST RETRASMIT: consiste nel non attendere un tempo RTO bensì inviare l’ack relativo al pacchetto perso, per ogni pacchetto successivo e quando vengono ricevuti 3 ack duplicati (oltre il primo) il pacchetto si dichiara perso e inizia la fase SS + CA.
* FAST RECOVERY: consiste nel saltare la fase di SS e impostare la finestra CW da subito pari alla soglia SSTH così che inizi subito una fase CA fino al raggiungimento della soglia precedente.

Le 2 tecniche ottimale sono usate solitamente insieme.

1. **Il candidato descriva nel dettaglio la problematica della silly window.**

La problematica nota come “Sindrome della Silly Window” è stata riscontrata nelle prime implementazioni del TCP. Il problema si riscontra quando l’applicazione sorgente passa i dati al suo TCP a grossi blocchi mentre l’applicazione ricevente assorbe i byte lentamente, ad esempio 1 bit alla volta. Il buffer ricevente inizialmente è pieno, il mittente ne è a conoscenza, dunque il ricevente legge il primo byte ed invia una notifica alla sorgente che può nuovamente inviare un bit , la sorgente invia 1 byte ed il tutto si ripete. Naturalmente inviare segmenti da 1 byte porta ad uno spreco di risorse a causa di un elevato overhead.

SOLUZIONI

**-**Il lato ricevitore non deve inviare ‘Windows Update’ per piccole variazioni della finestra di ricezione.

**-**Il sender d’altro canto deve evitare di emettere segmenti troppo piccoli

1. **Il candidato illustri in dettaglio le maggiori differenze tra protocollo TCP e UDP**

* **UDP**

Protocollo di trasporto più semplice. Aggiunge 2 funzionalità a IP: indirizzamento delle applicazioni, blando controllo d’errore sull’header (CheckSum). Fornisce un servizio di tipo datagram, dunque esso è:

* senza connessione (nessun controllo sui pacchetti fuori sequenza);
* non affidabile (pacchetti persi in quanto non si ha nessuna garanzia di consegna)
* senza controllo di flusso (nessun controllo sulla saturazione del reicever)
* nessun meccanismo di recupero d’errore.
* **TCP**
* con connessione, fornisce un servizio affidabile “end-to-end”.
* Protocollo orientato al byte e non al messaggio.

Effettua funzioni di:

* indirizzamento di uno specifico utente all’interno di un host
* trasferimento di un flusso informativo continuo e bidirezionale, ma non strutturato, di dati tra host remoti
* gestione delle connessioni
* controllo di flusso
* controllo di congestione
* riordinamento delle unità informative

1. **Il candidato descriva la struttura di una trama del protocollo HDLC, le tipologie di trama, le loro funzioni evidenziando le maggiori differenze**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Flag  1byte | Address  N byte | Control  1 o 2 byte | Info  #byte >=0 | Frame check sequence  2 byte | Flag   1. byte |

* **Flag:** indica l’inizio e la fine del frame
* **Address:** L’indirizzo contenuto può essere quello della stazione destinataria o sorgente. Nelle modalità

sbilanciate (NRM,ARM) è sempre quello della stazione secondaria, nelle modalità bilanciate (ABM) è quello della stazione detinataria

* **Information**: Contiene l’informazione d’utente, è presente solo nelle trame di tipo I ed U
* **Frame Check Sequence**: Contiene il codice rilevatore d’errore usato per riconoscere le trame errate
* **Control:** Il campo distingue i tipi di trama e contiene informazioni di controllo. I primi bit distinguono il tipo che può essere:
* Information(I): trame numerate per la trasmissione di informazioni d’utente
* Supervisory (S): trame numerate per il controllo dell’invio dell’informazione e per inviare riscontri. In base ai bit nel campo type (2 bit) si ha:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 0 | Type | P/F | RN |

* **RR (Receiver Ready)** [00]: la trama S viene usata come ack e contiene la prossima trama attesa.
  + **RNR (Receiver Not Ready)** [10]: blocca l’invio delle prossime trame e riscontra le trame fino alla precedente.
  + **REJ (Reject)** [01]: richiede la trasmissione delle prossime trame e riscontra le trame precedenti.
  + **SREJ (Selective Reject)** [11]: richiede la ritrasmissione della trama corrente.
* Unnumbered(U): sono trame non numerate usate per l’invio di informazioni di controllo

1. **Il candidato descriva in maniera dettagliata le differenze tra i protocolli link state e distance vector.**

**DISTANCE VECTOR:**

Si basa sull’algoritmo di Bellman-Ford, è più semplice, impegna meno risorse sul router ma è meno efficiente e adatto a reti piccole. Ogni router mantiene una struttura dati detta Distance Vector, ad ogni modifica invia le informazioni di routing ai router adiacenti. L’informazione trasmessa è una stima del costo del cammino fra 2 nodi, ed essa è trasmessa periodicamente.

**LINK STATE:**

Si assume che ogni router dispone della mappa completa della rete su cui calcolare gli instradamenti ottimali utilizzando l’algoritmo di Dijkstra. La topologia di rete è costruita tramite invio di LSP, inizialmente tramite protocolli di greeting per la conoscenza degli adiacenti. Ogni router ha un LSP database tramite la quale ogni router calcola la propria tabella di routing. L’informazione trasmessa è il valore esatto del costo dei link verso le reti adiacenti ed è trasmessa ad ogni cambiamento topologico.

1. **Descrivere il protocollo IPv4 evidenziandone le funzionalità associate alle informazioni di controllo trasportate**

Il protocollo IPv4 definisce lo schema di indirizzamento dei dati attraverso l'internet, l’instradamento dei dati, le modalità di frammentazione dei dati, quindi definisce il formato dell’unità. Fornisce un servizio inaffidabile e senza connessione.

È “senza connessione” perché ogni unità informativa può seguire una strada diversa per arrivare a destinazione.

È un protocollo “inaffidabile” poiché non garantisce la consegna delle unità informative. Queste infatti possono andare perse. La garanzia di qualità è un compito che spetta ai livelli superiori.

Le unità-dati dello strato IP sono dette datagrammi e sono composti da un campo informativo che contiene i

dati di utente e da un’intestazione (header). L’header contiene diversi tipi di informazione come: **TTL** (time to live) che indica il n° max di hop di un pacchetto, **padding** (riempitivo), **flags** (bit di controllo), **versione, source, protocol, identification, options.**

1. **Il candidato descriva come il protocollo TCP effettua il controllo di flusso, di errore e di congestione**

A seconda della tecnica implementa e dal tipo di congestione rilevata, è possibile adottare metodi diversi come il FAST RETRASMIT ad esempio. In generale, la congestione si rileva quando non si ha il riscontro su un pacchetto allo scadere del tempo limite RTO. In quel caso inizia una fase di SS + CA di recupero. Durante la fase di SS, il numero di pacchetti inviati cresce esponenzialmente e ciò permette di raggiungere più velocemente la soglia SSTH, dopo la quale inizia la fase di CA in cui la finestra di CW invece incrementa linearmente fino al raggiungimento eventualmente della fase di regime.

1. **Il candidato descriva il modello di riferimento ISO/OSI indicando i livelli protocollari conosciuti e le rispettive funzionalità associate ai livelli**

Riguarda la connessione di sistemi “aperti” verso la comunicazione con altri. Se più aziende utilizzano questa architettura l’ISO/OSI favorisce l’interoperabilità tra i sistemi.

Ha 7 livelli protocollari. Ovvero 7 strati diversi.

Ogni livello riceve dati dal livello superiore che aggiunge header per creare l’unità dati (incapsulamento).

Livelli protocollari:

1. **FISICO**: fornisce i mezzi fisici per attivare e disattivare le connessioni fisiche. Le unità dati sono bit o simboli.
2. **COLLEGAMENTO (DATALINK)**: Si occupa di rilevazione e recupero di errori tramite invio di Ack e del controllo del flusso. Si suddivide in:

* MAC: che si occupa della condividisione in modo ottimale delle risorse (evitare collisioni).
* LLC: che si occupa di framing, controllo errori, controllo flusso

1. **RETE**: Funzioni fondamentali sono l’instradamento, indirizzamento e inoltro. Quindi è responsabile della costruzione delle tabelle di routing. Controlla la congestione, schedula e controlla le connessioni di rete
2. **TRASPORTO**: Si occupa di garantire affidabilità tra sorgente e destinazione, effettua la frammentazione dei dati in “segmenti”. Si occupa di stabilire una connessione e del controllo della congestione (cioè evitare che troppi pacchetti dati arrivino allo stesso router contemporaneamente).
3. **SESSIONE**: è il responsabile dell’organizzazione del dialogo fra due programmi applicativi di sistemi diversi. Le sue funzioni sono: gestione del dialogo e sincronizzazione tra eventi e maschera le interruzioni del servizio trasporto.
4. **PRESENTAZIONE**: Risolve i problemi di compatibilità per quanto riguarda la rappresentazione dei dati da trasferire. Può fornire servizi di cifratura delle informazioni
5. **APPLICAZIONE**: Fornisce ai processi applicativi i mezzi per accedere all’ambiente OSI. I protocolli delle applicazioni tipiche di questo livello realizzano operazioni come, ad esempio, trasferimento file o posta elettronica.
6. **Indicare per linee generali cosa cambia tra protocolli di accesso al mezzo centralizzati e protocolli di accesso al mezzo distribuiti. Indicare vantaggi e svantaggi delle due tipologie di MAC.**

**Nell’accesso centralizzato** una delle stazioni (MASTER) provvede ad abilitare le stazioni secondarie ad emettere

a seguito di una fase di POLLING: ciascuna stazione aspetta il proprio turno per trasmettere, se quando arriva non hanno nulla da trasmettere il turno passa alla stazione successiva.

Vantaggi: consente coordinamento tra le stazioni

Svantaggi: bassa efficienza nel caso di molte stazioni, il messaggio di polling viene inviato ad un gran numero di stazioni che non ne hanno bisogno sprecando la risorsa trasmissiva, dunque efficienza bassa e tempi di attesa elevati

**Accesso distribuito**: le N stazioni sono disposte secondo un ordine logico ad anello (ring), non necessariamente coincidente con quello fisico, ed ognuna ha un preciso indirizzo. Ogni stazione conosce l’indirizzo della successiva e della precedente, secondo l’ordine logico. Una stazione può accedere al canale solo quando riceve un particolare pacchetto detto token dalla stazione che la precede; se ha un pacchetto da trasmettere, vi appenderà l’indirizzo della stazione destinataria e lo immetterà sul canale comune. La stazione che riconosce il proprio indirizzo catturerà il pacchetto. Quando una stazione termina i dati da trasmettere o dopo un certo tempo massimo, invia il token alla stazione successiva e così via.