CS 204 : COMPUTER NETWORKS

Assignment – 02

Roll no : 180010011

Congestion Avoidance and Control

1. The author says that flow of packets in a TCP connection need to follow the "conservation principle". Explain, what is a conservation principle and how can we ensure that for a given TCP connection, conservation principle is followed.

Ans) Conservation Principle: Conservation principle in packets means a new packet isn't put back into the network until old packet leaves.

There are only three ways for packet conservation to fail.They are:

a) The connection doesn't get equilibrium.
b) Sender injects new packet before old packet exits.
c) Equilibrium cant be reached because of resource limits along the path.

If we ensure none of the above three doesn't happen then we can ensure Packet Conservation.

2. According to the author, why it is not a good idea to estimate Time Out (TO) as twice of estimated Round Trip Time (RTT)(i.e TO = 2*RTT) under normal load scenarios. Instead, What have the authors proposed to do to compute the TO.

Ans)TO = K * RTT , but k should not be exactly 2 because all the RTTS will not be same.So, the authors proposed to take variance of RTT'd into cosideration to calculate TIMEOUT.

3. In the case of Retransmissions, how should the TO be calculated? Explain your answer.

Ans) To retransmit lost segment, TCP uses retransmission timeout(TO). When TCP sends a segment, the timer starts and stops when the acknowledgment is received. If the timer expires timeout occurs and the segment is retransmitted. TO is for 1RTT. To calculate TO we need to calculate RTT.

RTT are of three types:

a)Measured RTT(RTTm)
b)Smoothed RTT(RTTs)

After the first measurement: RTTs = RTTm
After each measurement: RTTs = (1-t) * RTTs + t * RTTm

c)Deviated RTT(RTTd)

After the first measurement: RTTd = (RTTm) / 2
After each measurement: RTTd = (1-k) * RTTd + k * (RTTm – RTTs)
Retransmitted timeout (TO) = RTTs + 4 * RTTd.

4. Figures 3 and 4 depict the behaviour of TCP without and with implementing Slow Start respectively. Study both the graphs and explain in your own words, why do you think that TCP slow start phase is necessary for implementing congestion control algorithms.

Ans) In Fig.3 Trace data of the start of a TCP conservation between two Sun 3/50s running Sun OS 3.5. The two Suns were on differnet Ethernets connected by IP gateways driving a 230.4 kbps point-to-point link. The window size for the connection was 16KB and there were 30 packets of buffer avaliable at the bottleneck gateway.The actual path contains six store and forward hops so the pipe plus gateway queue has enough capacity for a full window but the gateway queue alone does not.

The dashed line show the 20kbps bandwidth available for this sonnection. Only 35% of this bandwidth was used, the rest was wasted on retransmitts. Almost everything is retransmitted at least once and data from 54 to 58 KB is sent five times.
In Fig.5 no bandwidth is wasted on retransmitts but two seconds is spent on the slow-start sp the effective bandwidth of this part of the trace is 16kbps- two times bigger than fig.3.

5. Figures 8 and 9 depict the behaviour of multiple TCP connections when implementing congestion avoidance algorithm.

Ans) Figure 8 gives the data where there is no congestion avoidance for multiple and simultaneously TCP connections. Data from those 4 simultaneous TCP's says 4,000 out of 11,000 packets sent were retransmitted, it means more 50% of data was retransmitted. Since the data link bandwidth is 25kbps each TCP should receive 6kbps, instead one got 8kbps, two got 5kbps and other got 0.5kbps.

Figure 9 gives the data where there is congestion avoidance for multiple and simultaneous TCP connections. Data from those 4 simultaneous TCP's says 89 out of 8281 packets sent were retransmitted, it means only 1% of data was retransmitted. Two fot 8kbps, teoo got 4.5kbps. The 4.5kbps senders were talking to 4.3BSD receivers whcih would delay an ack until 35% of the window was filled ot 200ms had passed(i.e an ack was delayed for 5-7 packets on the average.) an ack for at most one packet.

So if we use congestion avoidance algorithm there will negligible retransmission of data.

6. Explain the combined slow start and congestion avoidance algorithm.

Ans)Slow Start Algorithm: TCP slow start algorithm balances the speed of a network connection. Slow start gradually increases the amount of data transmitted until it finds the network's maximum carrying capacity.
Congestion avoidance algorithm: This mechanism is used in TCP implementations. Once the threshold value is hit, TCP slows the rate of increased window size (slows down from exponential growth in the slow start state to linear growth in the songestion avoidance state). After a period of time, TCP transmission rate exceeds the network link capacity and thus including packet loss. TCP will immediately detect the packet loss and reduces the congestion window size to almost half of its actual value.

Slow start algorithm increases the amount of data transmitted exponentially till it reaches threshold value. When threshold value is reached Congestion avoidance algorithm slows the rate increase in window size, it slows down from exponential growth to linear growth. After sometime TCP transmission rate exceeds network link capacity Tcp will detect packet loss and reduces congestion window sixe to half of its value.

This is combined operation of Slow Start Algorithm and Congestion Avoidance Algorithm.

A Protocol for Packet Network Intercommunication

1. What is framing and why it is needed?

Ans) Framing is a point to point connection between two computers or devices consists of a wire in which data is transmitted as stream of bits. However, These bits must be framed into discernible blocks of information. Framing is a function of the data link layer. It provides a way for a sender to transmit a set of bits that are meaningful to the receiver. Ethernet, token ring, frame relay and other data link layer technologies have their own frame structures.

Frames have headers that contain information such as error checking codes. A frame works to help identify data packets used in networking and telecommunications structures. Frames also help to determine how data receivers interpret a stream of data from a source.

2. Why do we need TCP addressing when transmitting segments using TCP from one host to another.

Ans)TCP addressing is intimately bound up in routing issues, since a host must choose a suitable destination host for an outing internetwork packet. The choice for network identification (8 bits) allows up to 256 distinct networks. Similarly, the TCP identifier field permits up to 65,536 distinct TCP's to be addressed, which seems more than sufficient for any given network.

As each packet passes through a gateway, the gateway observes the destination ID to determine how to route the packet. If the destination network is connected to the gateway, the lower 16 bits of the TCP address are used to produce a local TCP address in the destination network. If the destination network is not connected to the gateway, the upper 8 bits are used to select a subsequent gateway. We make no effort to specifyhow each individual network shall associate the internetwork TCP identifier with its local address. We also do not rule out the possibility that the local network understands the internetwork addressing scheme and thus alleviates the gateway of the routing.

3. What is 'ES' and 'EM' flag in the TCP segment? How does it help in reassembly and sequencing?

Ans) The splliting of messages into segments by the **TCP** and the potential splitting of segments into smaller pieces by gateways creates the necessity for indicating to the destination TCP when the end of a segment (**ES**) has arrived and when the end of message (**EM**) has arrived. The flag field of the internetwork header is used for this purpose.

The **ES** flag is set by the source **TCP** each time it prepares a segment for transmission. If it should happen that the message is completely contained int he segment, then the **EM** flag would also be set. The **EM** flag is also set on the last segment of a message, if the message could not be contained in one segment. These two flags are used by the destination **TCP,** respectively to discover the presence of a check sum for a given segment and to discover that a complete message has arived.

The **ES** and **EM** flags in the internetwork header are known to the gateway and are of special importance when packets must be split apart for propagation through the next local network.

4. What are Transmit and Receive Control Block (TCP and RCB), explain its purpose. What is typical format of a TCB.

Ans) In order to send a message, a process sets up its next in a buffer region in its own address space, inserts the requisite control information in a transmit control block (TCB) and passes control to the TCP. The exact form of a TCB is not specified, but it might take the form of a passed pointer, a pseudointerrupt or various other forms.To receive a message in its address space, a process sets up a receive buffer, inserts the requisite control information in a receive control block (RCB) and again passes control to the TCP.

The TCB contains information necessary to allow the TCP to extract and sesnd the process data.Some of the information might be implicity known, but we are not concerned with that level of detail. The various fields in the TCB are described as follows:

1)Source Address: This is the full net/host/TCP/port address of the transmitter.

2)Destination Address: This is the full net/host/TCP/port of the receiver.

3)Next Packet Sequence Number: This is the sequence number to be used for the next packet the TCP will transmit from this port.
4)Current Buffer Size: This is present size of the process transmit buffer.
5)Next Write Position: This is the address of next position in the buffer at which the process can place new data for transmission.
6)Next Read Position: This is the address at which the TCP should begin reading to build the next segment for output.
7)End Read Position: This is the addresss at which the TCP should halttransmission. Initially bound the message which the process which the process wishes to transmit.
8)Number of Retransmissions/Maximum Retransmission: These fields enable the TCP to keep track of the number of times it has transmitted the data nad could be ommited if the TCP is not to give up.
9)Timeout/Flags: The timeout field specifies the delay after which unacknowledge data should be transmitted. The flag field is used for semaphores and other TCP/process synchronization status reporting etc..
10)Current Acknowledgment/Window: The current acknowledgment field identifies the first byte of data still unacknowledgment by the destination TCP.

5. What is the difference between connection and association ? Explain, what do you mean by connection free protocol with assocation ?

Ans) The term connection has a wide variety of meanings. It can prefer to a physical or logical path between two identities, it can refer to the flow over the path, it can inferentially refer to am action associated with the setting up of a path, or it can refer to an association between two or more entities, with or without regard to any path between them.

The relationship between two or more ports that are in communication or are prepared to communicate to be an association. Ports that are associated with each other are called associates.

The interface message processors do not have to open and close connections from source to destination. The reason for this is that connections are in effect always open since the address of every source and destination is never reassigned. When the name and the place are static and unchanging it is only necessary to label a packet with source and destination to transmit it through the network.

However, we find that port address are continually being used and reused. Some ever present process could be assigned fixed address whcih do not change.If we supposed, that every TCP had an infinite supply of port address so that no old address would ever be reused, then anuy dynamically created port would be assigned the next unused address.