4.6 Congestion Avoidance

TCP impose some methods to control congestion once it happens, instead of trying to avoid congestion.It is a prevention mechanism while congestion control is a recovery mechanism.

DECbit

DECbit means destination experiencing congestion bit. This mechanism was developed for use on the Digital Network Architecture (DNA), a connectionless network with a connection-oriented transport protocol.This mechanism could, therefore, also be applied toTCP and IP.It split the responsibility for congestion control between the routers and the end nodes.

Each router monitors the load it is experiencing and explicitly notifies the end nodes when congestion is about to occur. This notification is implemented by setting a binary congestion bit in the packets that flow through the router, hence the name DECbit.The destination host then copies this congestion bit into the ACK it sends back to the source. Finally, the source adjusts its sending rate so as to avoid congestion.

A router set this bit in a packet if its average queue length is greater than or equal to 1 at the time the packet arrives.This average queue length is measured over a time interval as , the last busy+idle cycle, plus the current busy cycle.The source records how many of its packets has set the congestion bit.If less than 50% of the packets had the bit set, then the source increases its congestion window byone packet. If 50% or more of the last window of packets had the congestion bit set, then the source decreases its congestion window to0.875 times the previous value.

Random Early Detection (RED)

RED provide congestion control at the router for TCP flows.RED was designed to work with TCP. Red notifies the sender by dropping packets. Packet dropping probability is increased as the average queue length increases. The moving average of the queue length is used to detect the long term congestion and allows short term bursts to arrive.

Properties of RED

1.RED drops packets before queue is full, in the hope of reducing the rates of some flows.

2.Drops packet for each flow roughly in proportion to its rate.

3.Red maintains average queue length.

4.Random drops desynchronize the TCP sources.

5.RED calculates the average que length using a weighted running average.

The Formula is as follows.

Average length = (1- Weight) x Average length + Weight x Sample length

Sample length is the queue length each time a packet arrives. The weight parameter is between 0 and 1.

RED has two queue length thresholds that trigger certain activity: MinThreshold and MaxThreshold.

When a packet arrives at the gateway, RED compares the current AvgLen with these two thresholds,according to the following rules:

```
 ifAvgLen≤ MinThreshold
Then queue the packet
ifMinThreshold<AvgLen<MaxThreshold
calculate probability P
drop the arriving packet with probability P
ifMaxThreshold≤AvgLength
!drop the arriving packet
```

If the average queue length is smaller than the lower threshold, no action is taken, and if the average queue length is larger than the upper threshold, then the packet is always dropped. If the average queue length is between the two thresholds, then the newly arriving packet is dropped with some probability P.