eecs489 COMPUTER NETWORKS

Lecture 28:
Lab6 and PA3

Sequence Number

Sequence number is per byte, not per packet

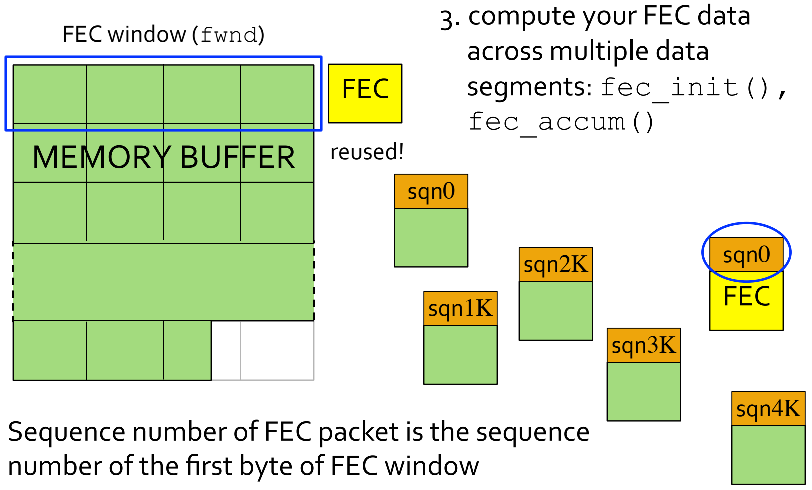
The sequence number attached to a packet is the sequence number of its first byte

The sequence number of a byte is its byte offset from the start of image buffer

This enables out-of-order data to be placed in its right position in the image buffer

Lab6: FEC Sender

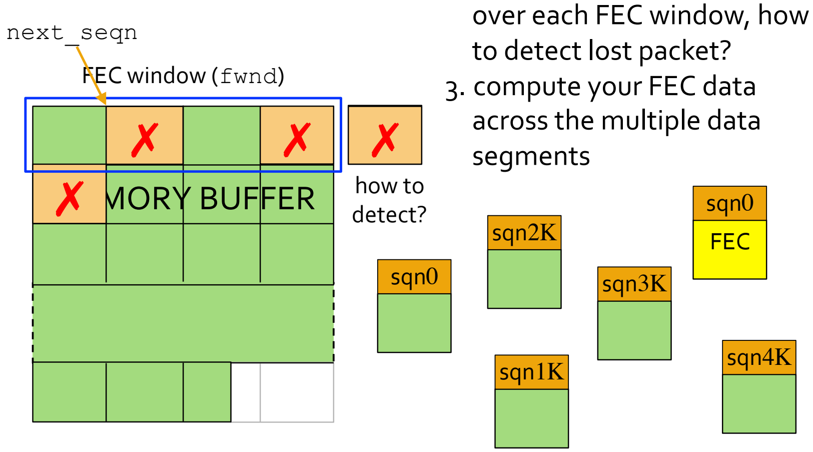
1. maintain your FEC window,
2. keep track of your progress over each FEC window, and
3. compute your FEC data across multiple data segments: `fec_init()`, `fec_accum()`



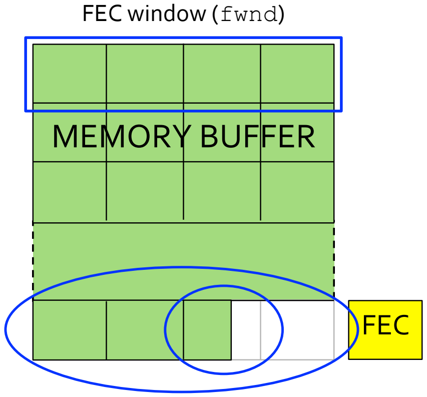
Sequence number of FEC packet is the sequence number of the first byte of FEC window

Lab6: FEC Receiver

1. maintain your FEC window over image buffer
2. keep track of your progress over each FEC window, how to detect lost packet?
3. compute your FEC data across the multiple data segments



Lab6: Last FEC



The last FEC window may be smaller than the `fwnd` specified by the client

- last FEC packet may be computed over a smaller window

The last data packet may be smaller than the `mss` specified by the client

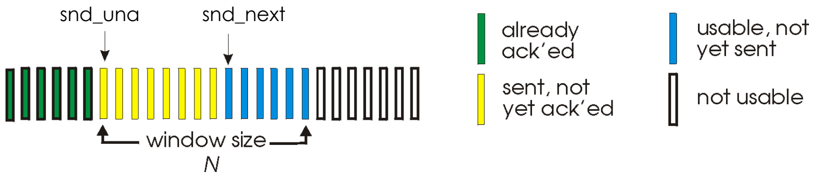
- last FEC packet computed as if last segment padded with 0

In all cases, FEC packet is of fixed size

Go-Back-N [PA3]

Sender:

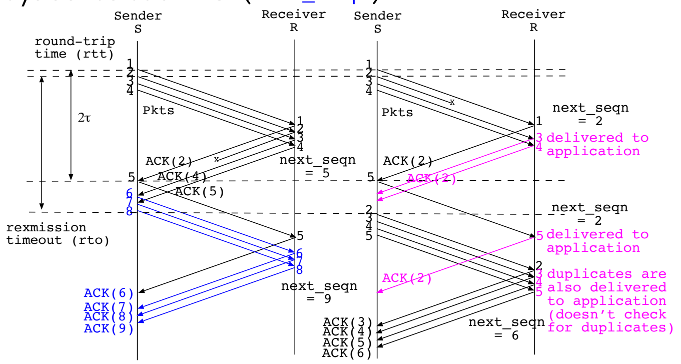
- ACK(n) is cumulative: ACKs all bytes up to n-1, not including n
- maintains only one timer, for snd_una
- timeout(snd_una): retransmits snd_una and all higher seq#s in window
  - resets snd_next to snd_una
  - resets timer for snd_una



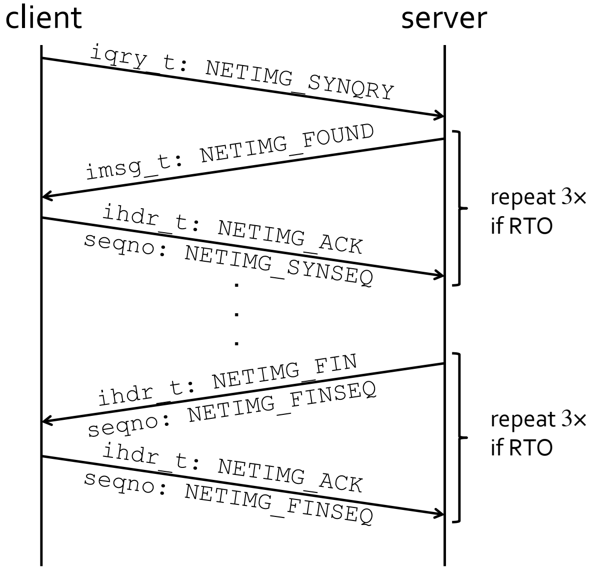
Go-Back-N [PA3]

Receiver:

- only needs to remember next expected seq# (next_seqn)
- cumulative ACKs acknowledge all bytes received in-order, up to (next_seqn – 1)
- ACKs and delivers out-of-order packets to application
- always sends back ACK(next_seqn)



Connection Setup and Teardown



PA3: Task 2 Go-Back-N

Session handshaking

- `imgdb::sendpkt()` must wait for ACK, with `NETIMG_MAXTRIES` retransmissions
- must wait for FIN, else server won't be able to handle next client

Server side Go-Back-N

- keep track of "usable" window (add member variables)
- can send, including retransmit, only when usable window > 1 mss
- greedy grab of ACKs: non-blocking with `ioctl()`
- assume fixed RTO (may need to be > 20 secs if on ADSL)

Client side Go-Back-N

- prepare and send ACK packets, including for FIN packet

PA3 support code covers only this task; build off Lab5

PA3: Go-Back-N with FEC

If a single segment is lost in an FEC window, recreate packet and cumulative ACK the whole window

If multiple segments are lost, enter Go-Back-N

FEC improve performance on network with low loss rate

PA3: Task 4 GBN with FEC

Server side

- upon RTO, enter GBN and reset FEC window

Client side

- if FEC patched, cumulative ACK last byte of FEC window
- Go-Back-N mode tracks inferred sender's Go-Back-N
  - already in-flight packets mess up packet count
  - enter GBN modeif lost more than 1 packets within an FEC window
  - or when FEC packet lost and an FEC window worth of data, or more, has been received since the last lost packet
  - "deactivate" FEC until next_seqn is received
- exit Go-Back-N mode
  - when retransmitted next_seqn is received
- detect last FEC window, adjust variables