CN-Basic
L26

TCP Overview

Dr. Ram P Rustagi
rprustagi@ksit.edu.in
http://www.rprustagi.com
https://www.youtube.com/rprustagi

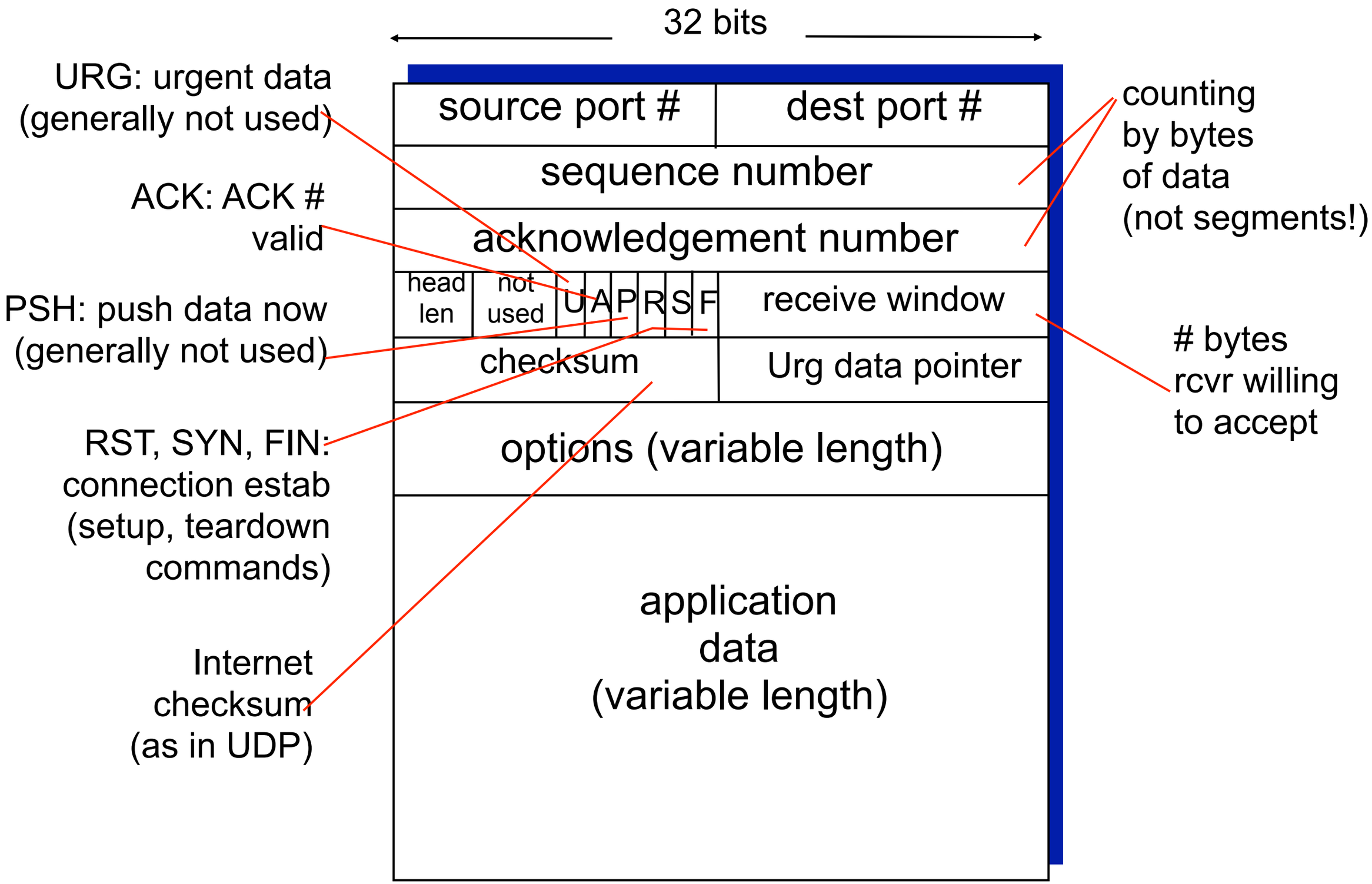
TCP: Overview RFCs: 793,1122,1323, 2018, 2581

- point-to-point:
  - one sender, one receiver
- reliable, in-order *byte steam:*
  - no "message boundaries"
- pipelined:
  - Sliding window size of TCP congestion and flow control
- full duplex data:
  - bi-directional data flow in same connection
  - MSS: maximum segment size, determined from link/frame size
- connection-oriented:
  - handshaking (exchange of control msgs) inits sender, receiver state before data exchange
- flow controlled:
  - sender will not overwhelm receiver

TCP Headers

- What should it contain?
  - Source and destination ports
  - Sequence numbers and acknowledgements
    - Data exchange in both directions
    - (implicit) Indication of stream oriented
  - Info about managing sliding window
  - Info about handling out or order delivery by internet
  - Checksum for errors
  - Identification for setup, data transfer and teardown
  - Window size (buffer size for what it can receive)
  - Length of data
  - Options?

TCP segment structure



TCP seq. numbers, ACKs

sequence numbers:

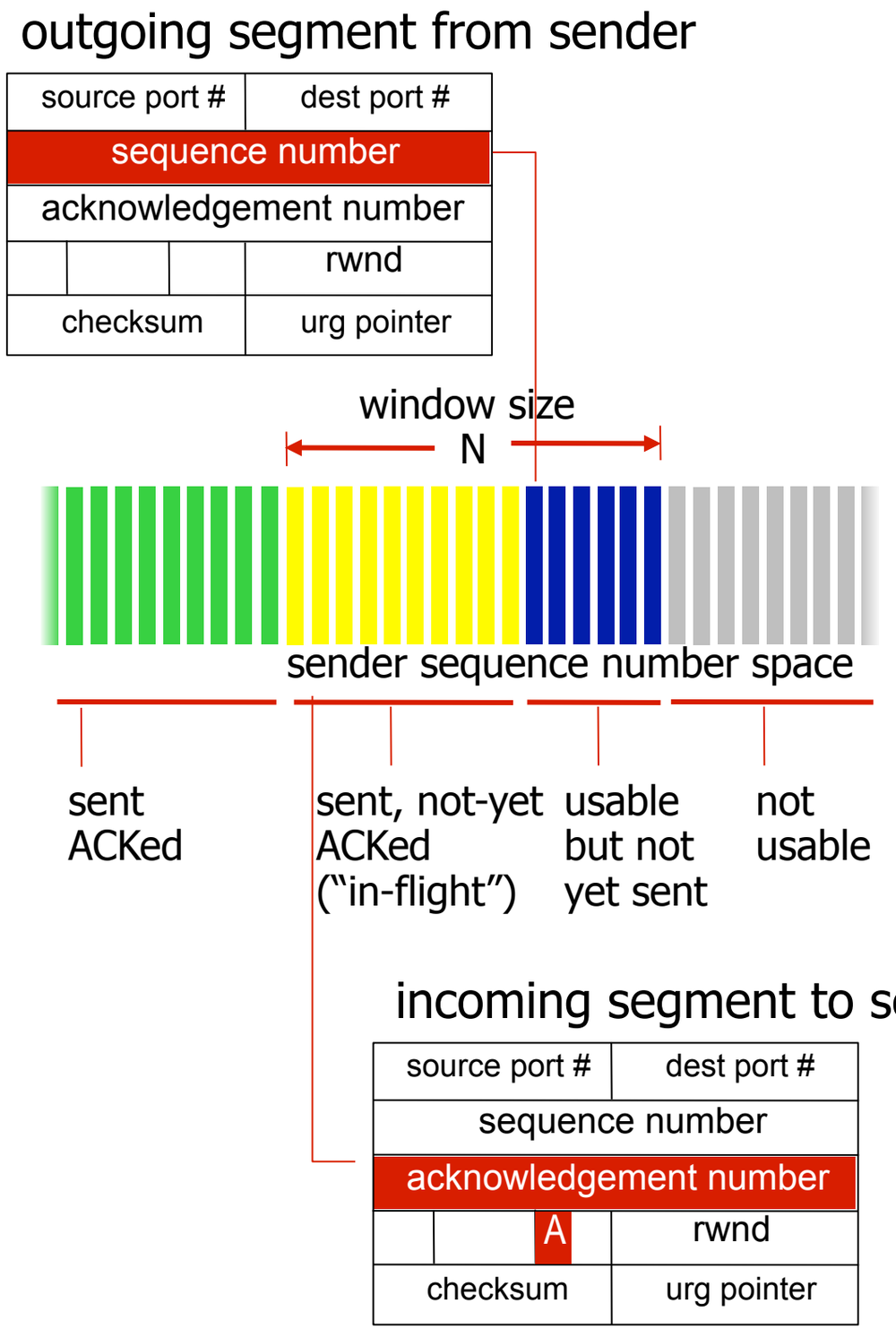
- byte stream "number" of first byte in segment's data
- does not reflect segment num

acknowledgements:

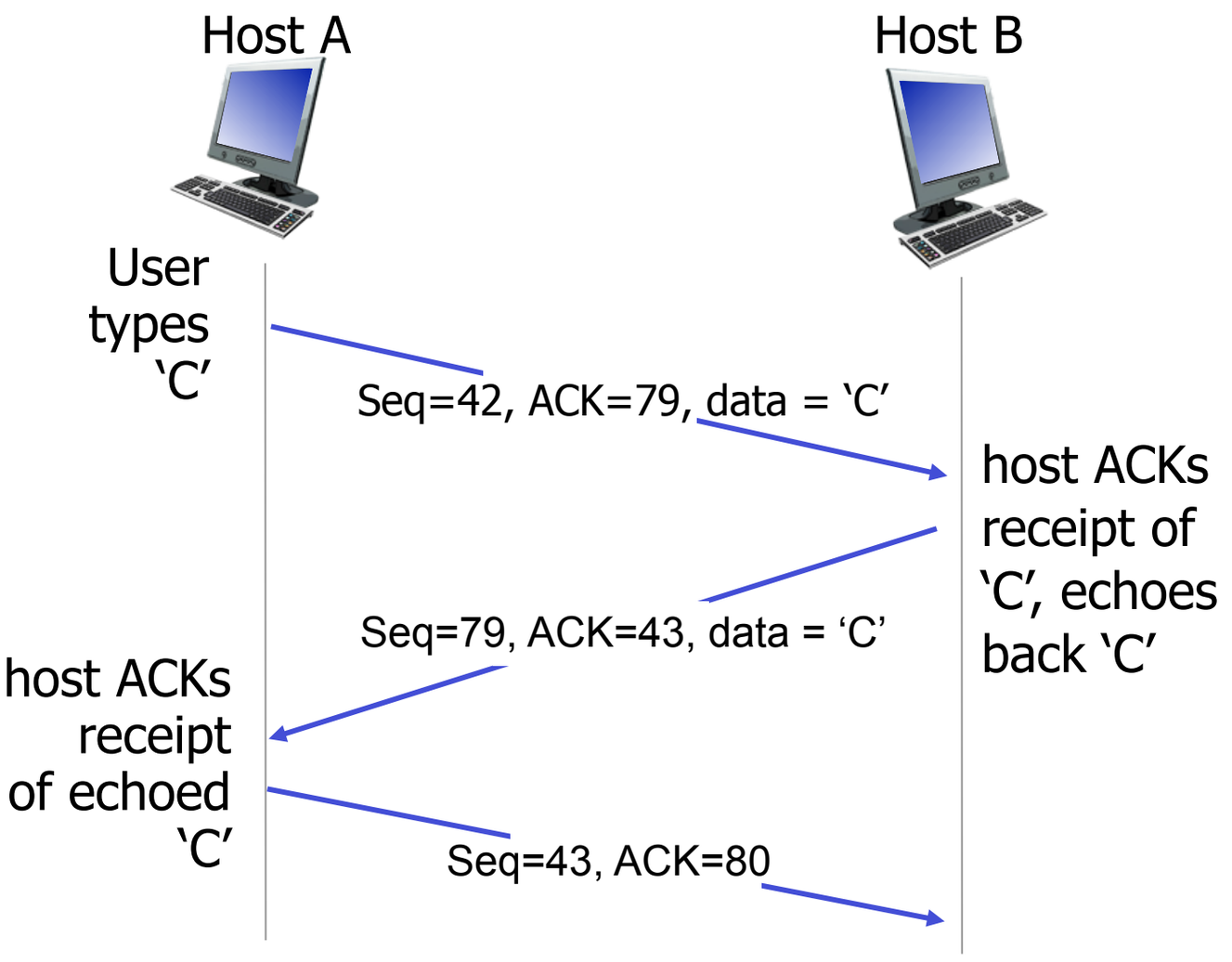
- seq # of next byte expected from other side
- cumulative ACK

Q: how receiver handles out-of-order segments

- A: TCP spec doesn't say, - up to implementor



TCP seq. numbers, ACKs



simple telnet scenario

Do an HTTP capture with Google or your website
Study the flow of sequence numbers.

TCP seq. numbers, ACKs

- TCP Bi-Directional Data Xfer
- Server program:

```
(i=6;while [ $i -gt 0 ];  do sleep 1; echo
"From Server, the value is $i"; i=$(($i - 1));
done) | nc -l 8080
```

- Client programs:

```
(i=0;while  [ $i -le 5 ];  do echo "Request
$i"; sleep 1; i=$(($i + 1)); done) | nc
localhost 8080
```

- File Capture:
  - **TCP-Bidirectional-Data-Xfer.pcap**

Summary

- TCP Header format
- TCP Communication
  - Sequence number and ack