UIC UNIVERSITY OF ILLINOIS AT CHICAGO Department of Computer Science
COLLEGE OF ENGINEERING

CS450 – Introduction to Networking
Lecture 14 – TCP

Phu Phung

Feb 13, 2015

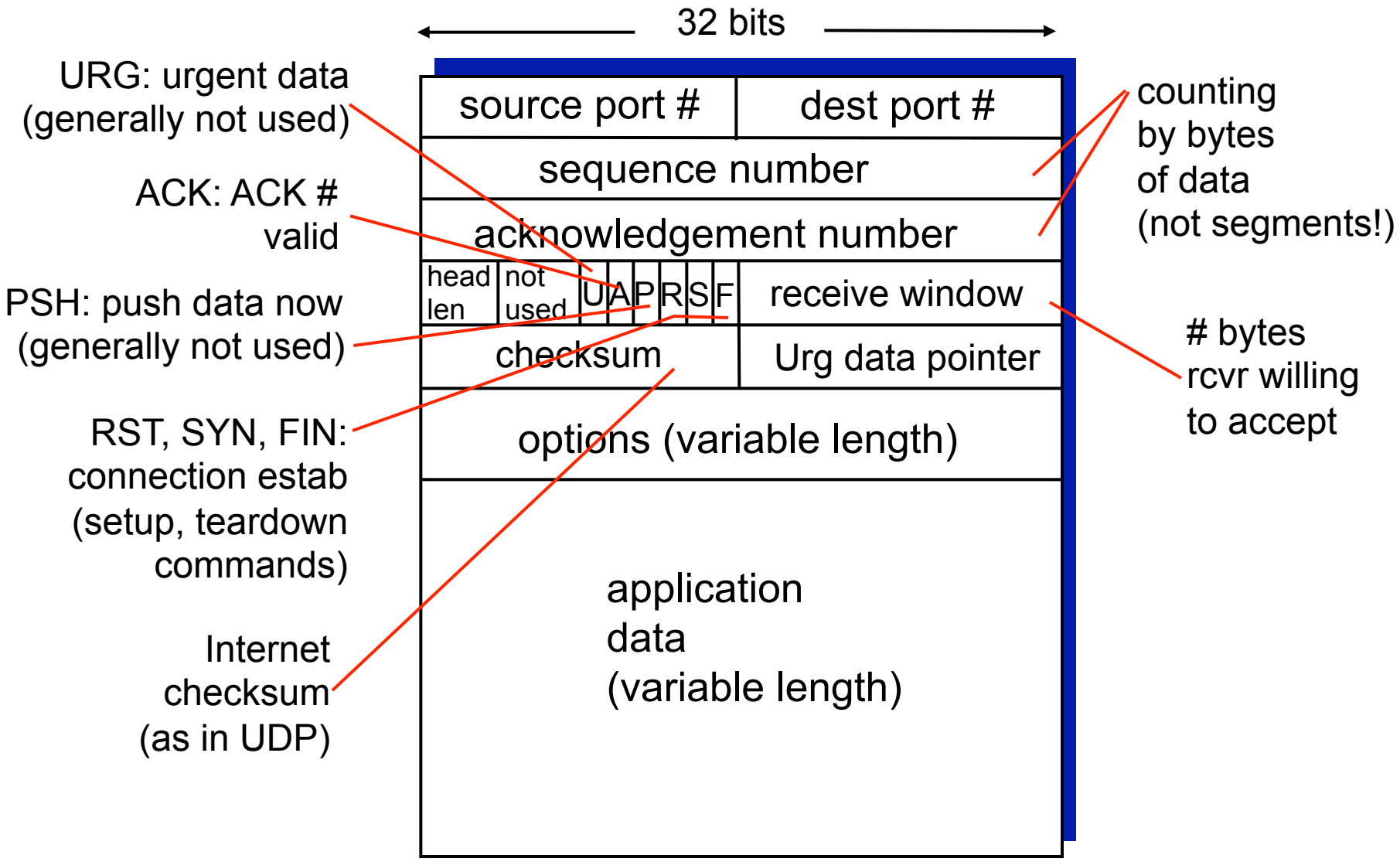
Next lecture (Feb 16)

- Assignment 3 (No iclicker questions)
  - Wireshark links
- Guest lecture on Monday Feb 23rd
  - DNS Security
- Midterm exam in class
  - 1 PM Friday, March 6th

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

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

TCP segment structure



TCP seq. numbers, ACKs

sequence numbers:

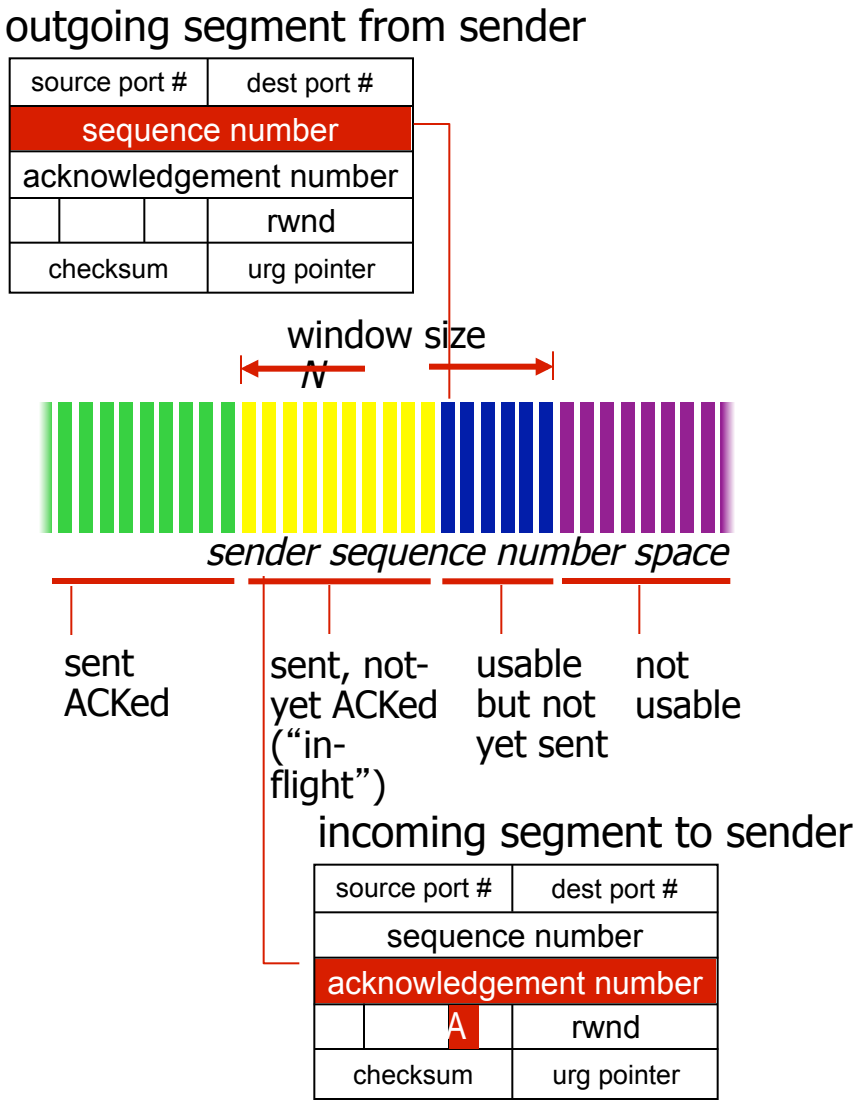
- byte stream "number" of first byte in segment's data

acknowledgements:

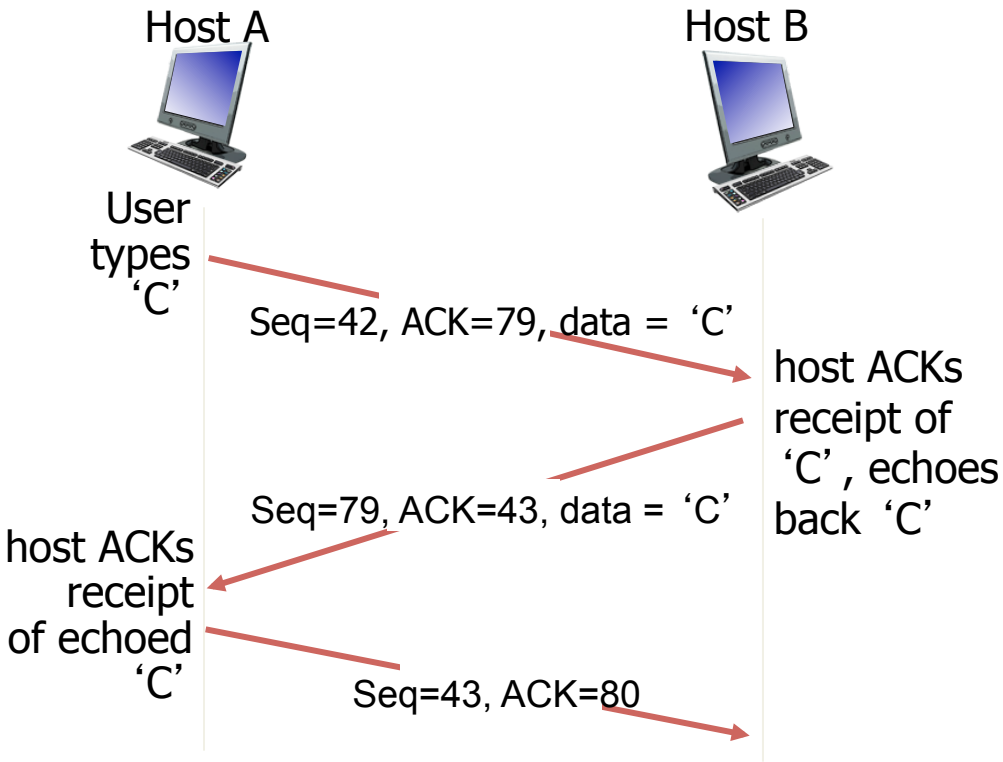
- seq # of next byte expected from other side
- cumulative ACK

Q: how receiver handles out-of-order segments

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



TCP seq. numbers, ACKs



simple telnet scenario

Sender sends seq#=101, with data length = 20 bytes, and receives ack = 91. What is true?

A. The receiver has successfully received up to seq# 90

B. The packet with seq#=101 was lost

C. The packet with seq#=91 was lost

D. A and B

E. A and C

TCP round trip time, timeout

Q: how to set TCP timeout value?

- longer than RTT
  - but RTT varies
- *too short:* premature timeout, unnecessary retransmissions
- *too long:* slow reaction to segment loss

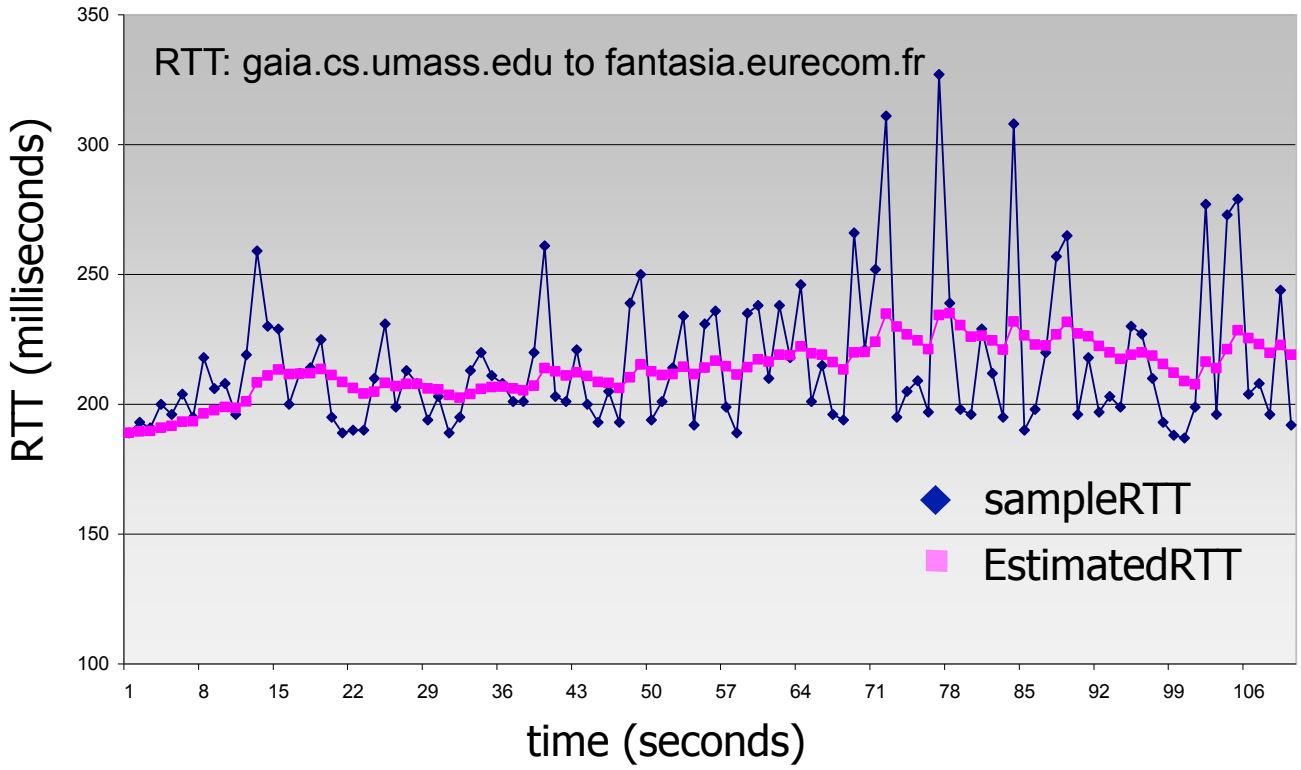
Q: how to estimate RTT?

- **`SampleRTT`**: measured time from segment transmission until ACK receipt
  - ignore retransmissions
- **`SampleRTT`** will vary, want estimated RTT "smoother"
  - average several *recent* measurements, not just current **`SampleRTT`**

TCP round trip time, timeout

EstimatedRTT = (1- α)*EstimatedRTT + α*SampleRTT

- exponential weighted moving average
- influence of past sample decreases exponentially fast
- typical value: α = 0.125



TCP round trip time, timeout

- timeout interval: **`EstimatedRTT`** plus "safety margin"
  - large variation in **`EstimatedRTT`** -> larger safety margin
- estimate SampleRTT deviation from EstimatedRTT:

```
DevRTT = (1-β)*DevRTT +
                β*|SampleRTT-EstimatedRTT|

(typically, β = 0.25)
```

```
TimeoutInterval = EstimatedRTT + 4*DevRTT
```

estimated RTT

"safety margin"

What is DevRTT?

A. Relative difference between SampleRTT and EstimatedRTT

B. Is used as a parameter for safe margin of timeout interval

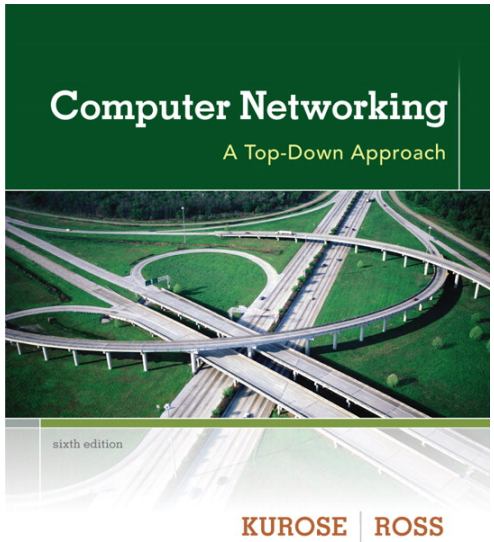
C. Is used to re-calculate SampleRTT

D. A and B

E. A and C

Next lecture

- TCP
  - Readings 3.5
- Guest lecture on Monday Feb 23rd
  - DNS Security
- Midterm exam in class
  - In class: 1 PM Friday, March 6th



Copy right notice:
These slides are adapted from J.F Kurose and K.W. Ross's ones

Computer Networking: A Top Down Approach

6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012