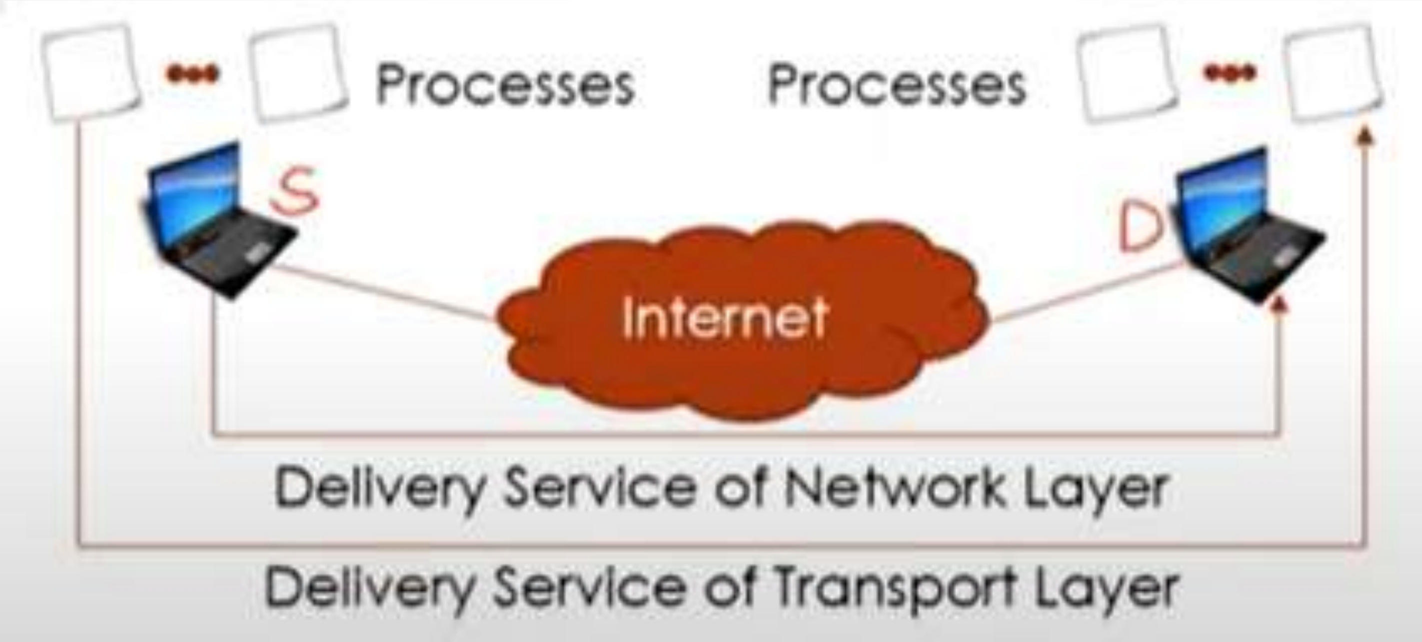
Mobile Transport layer

Syllabus

- Traditional TCP
  - Congestion Control
  - Slow start
  - Fast retransmit/Fast Recovery
- Classical TCP improvements to support Wireless and mobile environment
  - Indirect TCP
  - Snooping TCP
  - Mobile TCP
  - Selective retransmission
  - Transaction oriented TCP

Introduction

- Relation between network layer and transport layer



Overview of Transport layer

- Two protocols in transport layer
  - UDP (User Datagram protocol) : which provides an unreliable, connectionless service to the invoking application
  - TCP (Transmission control protocol): which provides a reliable, connection-oriented service to the invoking application

Overview of Transport layer

- The most fundamental responsibility of the Transport layer is to extend IP’s delivery service between two end systems to a delivery service between two processes' running on the end systems
- It also provide integrity checking by including error detection fields in their segment’s header
- It also offers additional services like reliable data transfer and congestion control

Traditional TCP

- TCP is connection oriented
- A TCP connection provides a full duplex service
- A TCP connection is also always point-to-point
- Mechanisms of traditional TCP that influence the efficiency of TCP
  - Congestion control
  - Slow start
  - Fast retransmit or fast recovery

Congestion control

- Timeouts/Packet loss typically due to (temporary) overload
- Routers discard packets when buffers are full
- TCP recognizes congestion only indirectly via missing ACKs, retransmissions unwise, since they increase congestion
- slow-start algorithm as reaction

Slow start algorithm

- sender calculates a congestion window for a receiver
- Start with a congestion window size equal to one segment (packet)
- Exponentially increase congestion window till congestion threshold, then linear increase
- Timeout/missing acknowledgement causes reduction of congestion threshold to half of the current congestion window
- congestion window starts again with one segment

Fast retransmit

- TCP sends an ACK only after receiving a packet
- If sender receives duplicate ACKs, this is due to gap in received packets at the receiver
- Receiver got all packets up to the gap and is actually receiving packets

Conclusion: packet loss not due to congestion, retransmit, continue with current congestion window (do not use slow-start)

Influences of Wireless/ mobility on TCP

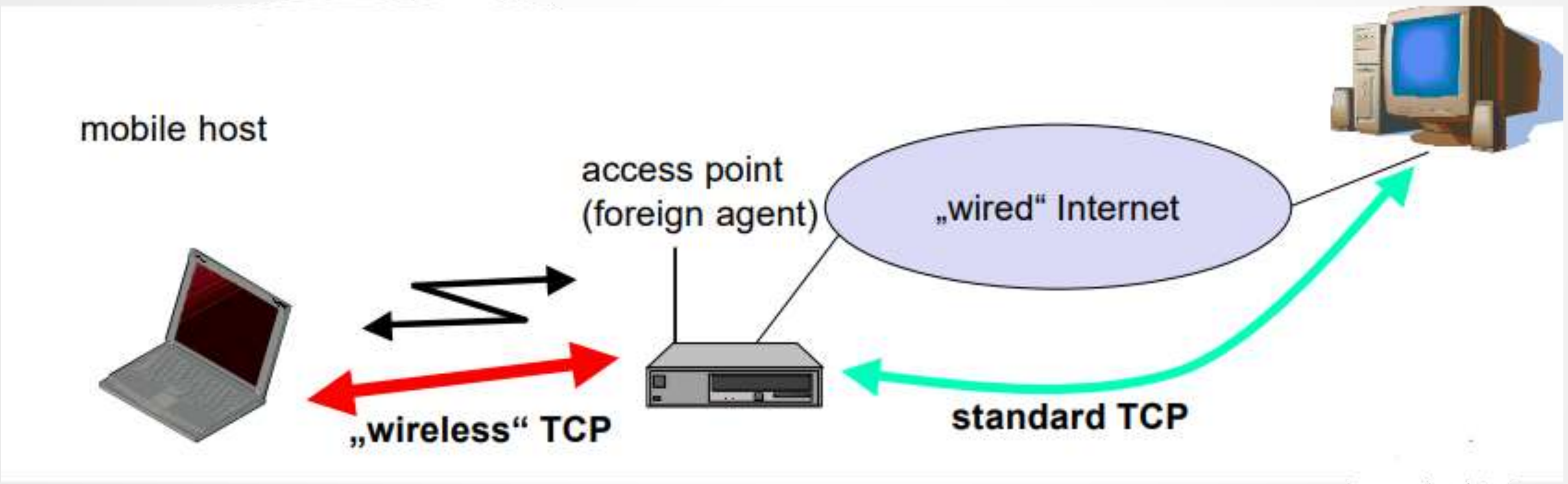
- TCP assumes congestion if packets are dropped
  - typically wrong in wireless networks, here we often have packet loss due to transmission errors
  - furthermore, mobility can cause packet loss, if e.g. a mobile node roams from one access point (e.g. foreign agent in Mobile IP) to another while packets in transit to the old access point and forwarding is not possible

  The performance of an unchanged TCP degrades severely
  - TCP cannot be changed fundamentally due to large installed base in the fixed network, TCP for mobility has to remain compatible
  - the basic TCP mechanisms keep the whole Internet together

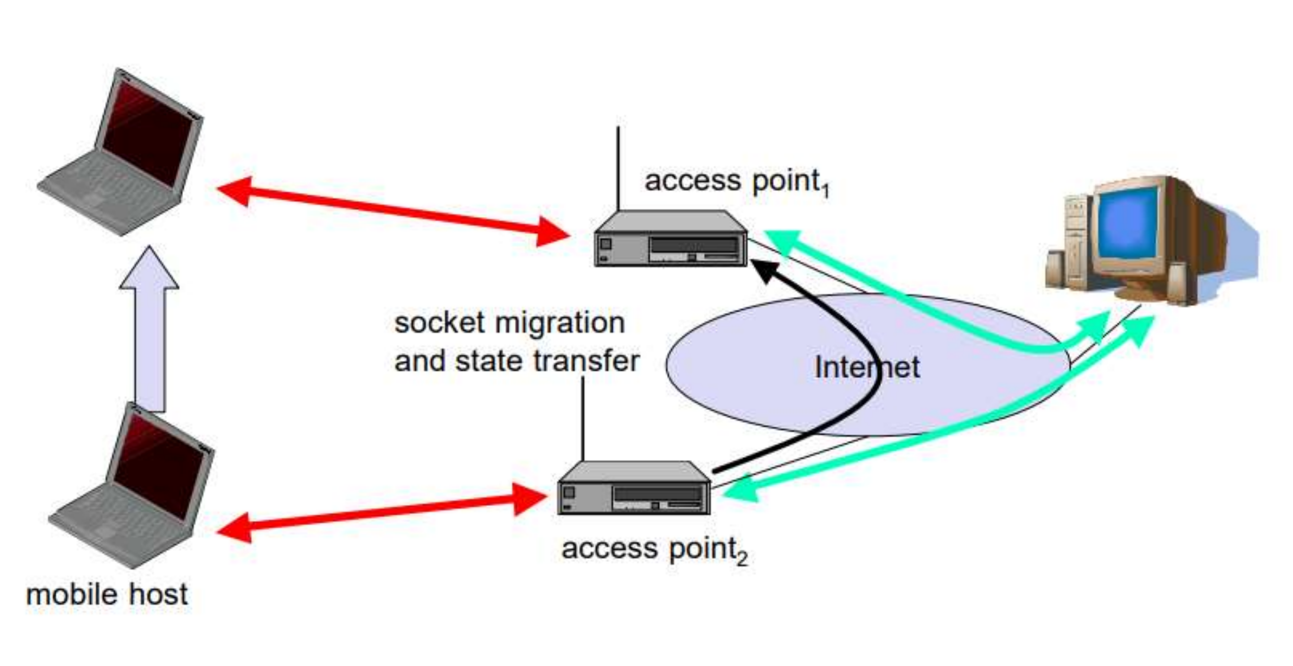
Early approach : Indirect TCP 1

- Indirect TCP or I-TCP segments the connection
  - no changes to the TCP protocol for hosts connected to the wired Internet, millions of computers use (variants of) this protocol
  - optimized TCP protocol for mobile hosts
  - splitting of the TCP connection at, e.g., the foreign agent into 2 TCP connections, no real end-to-end connection any longer
  - hosts in the fixed part of the net do not notice the characteristics of the wireless part

Early approach : Indirect TCP 1



Early approach : Indirect TCP 1



Early approach : Indirect TCP 1

Advantages

- No changes in the fixed network necessary, no changes for the hosts (TCP protocol) necessary, all current optimizations to TCP still work
- Wireless link transmission errors isolated from those in fixed network
- simple to control, mobile TCP is used only for one hop between, e.g., a foreign agent and mobile host
- therefore, a very fast retransmission of packets is possible, the short delay on the mobile hop is known

Disadvantages

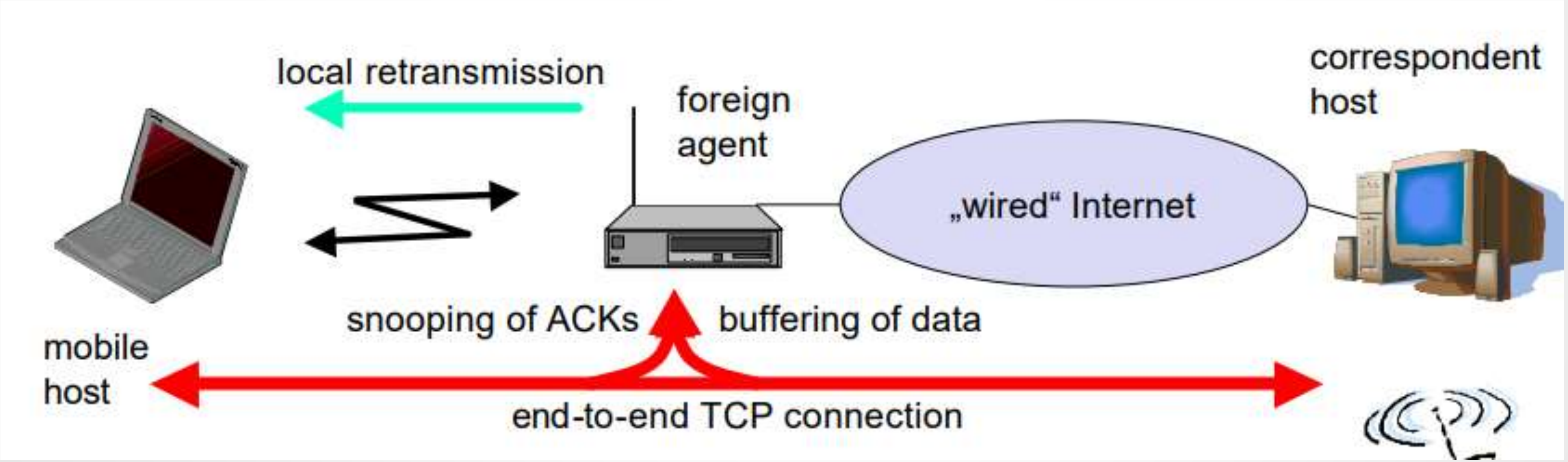
- loss of end-to-end semantics, an acknowledgement to a sender does now not any longer mean that a receiver really got a packet, foreign agents might crash
- higher latency possible due to buffering of data within the foreign agent and forwarding to a new foreign agent

Early approach : Snooping TCP 1

“Transparent ” extension of TCP within the foreign agent

- buffering of packets sent to the mobile host
- lost packets on the wireless link (both directions!) will be retransmitted immediately by the mobile host or foreign agent, respectively (so called “local” retransmission)
- the foreign agent therefore “snoops” the packet flow and recognizes acknowledgements in both directions, it also filters ACKs
- changes of TCP only within the foreign agent

Early approach : Snooping TCP 1



Early approach : Snooping TCP 1

Data transfer to the mobile host

- FA buffers data until it receives ACK of the MH, FA detects packet loss via duplicated ACKs or time-out
- fast retransmission possible, transparent for the fixed network

Data transfer from the mobile host

- FA detects packet loss on the wireless link via sequence numbers, FA answers directly with a NACK to the MH
- MH can now retransmit data with only a very short delay

Integration with MAC layer

- MAC layer often has similar mechanisms to those of TCP
- thus, the MAC layer can already detect duplicated packets due to retransmissions and discard them

Problems

- snooping TCP does not isolate the wireless link as good as I-TCP
- snooping might be tough if packets are encrypted

Early approach : Mobile TCP

Special handling of lengthy and/or frequent disconnections

M-TCP splits as I-TCP does

- unmodified TCP fixed network to supervisory host (SH)
- optimized TCP SH to MH

Supervisory host

- no caching, no retransmission
- monitors all packets, if disconnection detected
  - set sender window size to 0
  - sender automatically goes into persistent mode

Early approach : Mobile TCP

Advantages

- maintains semantics, supports disconnection, no buffer forwarding

Disadvantages

- loss on wireless link propagated into fixed network
- adapted TCP on wireless link

Selective Retransmission

TCP acknowledgements are often cumulative

- ACK n acknowledges correct and in-sequence receipt of packets up to n
- if single packets are missing quite often a whole packet sequence beginning at the gap has to be retransmitted (go-back-n), thus wasting bandwidth

Selective retransmission as one solution

- RFC2018 allows for acknowledgements of single packets, not only acknowledgements of in-sequence packet streams without gaps
- sender can now retransmit only the missing packets

Selective Retransmission

Advantage

- much higher efficiency

Disadvantage

- more complex software in a receiver, more buffer needed at the receiver

Transaction oriented TCP

TCP phases

- connection setup, data transmission, connection release
- using 3-way-handshake needs 3 packets for setup and release, respectively q
- thus, even short messages need a minimum of 7 packets!

Transaction oriented TCP

- RFC1644, T-TCP, describes a TCP version to avoid this overhead
- connection setup, data transfer and connection release can be combined
- thus, only 2 or 3 packets are needed

Transaction oriented TCP

Advantage

- efficiency

Disadvantage

- requires changed TCP
- mobility not longer transparent