Unit 6

Mobile Transport Layer

TCP over Wireless Networks – Indirect TCP – Snooping TCP – Mobile TCP – Fast Retransmit / Fast Recovery Transmission/Timeout Freezing-Selective Retransmission – Transaction Oriented TCP, TCP over 2.5 / 3G wireless Networks

TCP over Wireless Networks:

1. **Congestion control:**
   - A transport layer protocol such as TCP has been designed for fixed networks with fixed end-systems.
   - Data transmission takes place using network adapters, fibre optics, copper wires, special hardware for routers etc.
   - This hardware typically works without introducing transmission errors.
   - If the software is mature enough, it will not drop packets or flip bits, so if a packet on its way from a sender to a receiver is lost in a fixed network, it is not because of hardware or software errors.
   - The probable reason for a packet loss in a fixed network is a temporary overload some point in the transmission path, i.e., a state of congestion at a node.
2. **Slow start:**
   - TCPs reaction to a missing acknowledgement is quite drastic, but it is necessary to get rid of congestion quickly.
   - The behaviour TCP shows after the detection of congestion is called slow start.
   - The sender always calculates a congestion window for a receiver.
   - The start size of the congestion window is one segment (TCP packet).
   - The sender sends one packet and waits for acknowledgement. If this acknowledgement arrives, the sender increases the congestion window by one, now sending two packets (congestion window = 2).
   - After arrival of the two corresponding acknowledgements, the sender again adds 2 to the congestion window, one for each of the acknowledgements. Now the congestion window equals 4.
   - This scheme doubles the congestion window every time the acknowledgements come back, which takes one round trip time (RTT).

- This is called the exponential growth of the congestion window in the slow start mechanism.
- It is too dangerous to double the congestion window each time because the steps might become too large.
- The exponential growth stops at the congestion threshold.

3. **Fast retransmit/fast recovery:**
   - Two things lead to a reduction of the congestion threshold.
   - One is a sender receiving continuous acknowledgements for the same packet.
   - This informs the sender of two things.
   - One is that the receiver got all packets up to the acknowledged packet in sequence.
   - In TCP, a receiver sends acknowledgements only if it receives any packets from the sender.
   - Receiving acknowledgements from a receiver also shows that the receiver continuously receives something from the sender.
   - The gap in the packet stream is not due to severe congestion, but a simple packet loss due to a transmission error.
   - The sender can now retransmit the missing packet(s) before the timer expires.
   - This behavior is called fast retransmit.
4. **Implications on mobility:**
   - Mobility itself can cause packet loss.
   - There are many situations where a soft handover from one access point to another is not possible for a mobile end system.
   - For example, when using mobile IP, there could still be some packets in transit to the old foreign agent while the mobile node moves to the new foreign agent.
   - The old foreign agent may not be able to forward those packets to the new foreign agent or even buffer the packets if disconnection of the mobile node takes too long.
   - This packet loss has nothing to do with wireless access but is caused by the problems of rerouting traffic.

1. ***Indirect TCP:***

- Two competing insights led to the development of indirect TCP (I-TCP).
- One is that TCP performs poorly together with wireless links; the other is that TCP within the fixed network cannot be changed.
- I-TCP segments a TCP connection into a fixed part and a wireless part.

- Figure 9.1 shows an example with a mobile host connected via a wireless link and an access point to the wired internet where the correspondent host resides.
- The correspondent node could also use wireless access.
- The following would then also be applied to the access link of the correspondent host.
- Standard TCP is used between the fixed computer and the access point.
- No computer in the internet recognizes any changes to TCP.
- Instead of the mobile host, the access point now terminates the standard TCP connection, acting as a proxy.
- This means that the access point is now seen as the mobile host for the fixed host and as the fixed host for the mobile host.
- Between the access point and the mobile host, a special TCP, adapted to wireless links, is used.
- However, changing TCP for the wireless link is not a requirement
- A good place for segmenting the connection between mobile host and correspondent host is at the foreign agent of mobile IP.

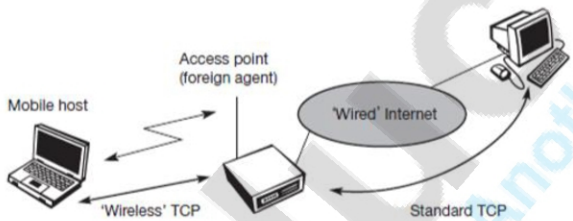
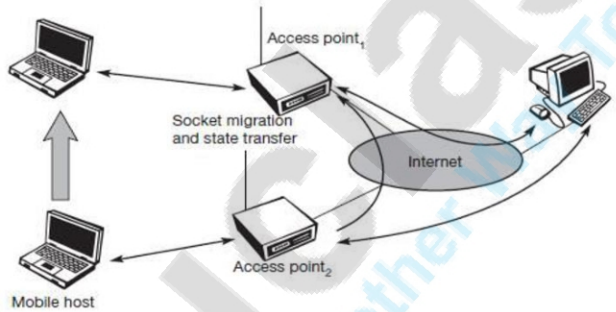


Figure 9.1
Indirect TCP segments a TCP connection into two parts

- The foreign agent controls the mobility of the mobile host anyway and can also hand over the connection to the next foreign agent when the mobile host moves on.
- However, one can also imagine separating the TCP connections at a special server, e.g., at the entry point to a mobile phone network.
- The correspondent host in the fixed network does not notice the wireless link or the segmentation of the connection.
- The foreign agent acts as a proxy and relays all data in both directions.
- If the correspondent host sends a packet, the foreign agent acknowledges this packet and tries to forward the packet to the mobile host.
- If the mobile host receives the packet, it acknowledges the packet.
- However, this acknowledgement is only used by the foreign agent.
- If a packet is lost on the wireless link due to a transmission error, the correspondent host would not notice this.

- In this case, the foreign agent tries to retransmit this packet locally to maintain reliable data transport.
- Similarly, if the mobile host sends a packet, the foreign agent acknowledges this packet and tries to forward it to the correspondent host.
- If the packet is lost on the wireless link, the mobile hosts notice this much faster due to the lower round trip time and can directly retransmit the packet.
- Packet loss in the wired network is now handled by the foreign agent.
- I-TCP requires several actions as soon as a handover takes place.
- As Figure 9.2 demonstrates, not only the packets have to be redirected using, e.g., mobile IP.

Figure 9.2 Socket and state migration after handover of a mobile host



There are several advantages with I-TCP:

- I-TCP does not require any changes in the TCP protocol as used by the hosts in the fixed network or other hosts in a wireless network that do not use this optimization.
- Due to the strict partitioning into two connections, transmission errors on the wireless link, i.e., lost packets, cannot propagate into the fixed network.
- It is always dangerous to introduce new mechanisms into a huge network such as the internet without knowing exactly how they will behave. But with I-TCP only between the mobile host and the foreign agent.
- The authors assume that the short delay between the mobile host and foreign agent could be determined and was independent of other traffic streams. An optimized TCP could use precise time-outs to guarantee retransmission as fast as possible.
- Partitioning into two connections also allows the use of a different transport layer protocol between the foreign agent and the mobile host or the use of compressed headers etc. The foreign agent can now act as a gateway to translate between the different protocols.

But the idea of segmentation in I-TCP also comes with some disadvantages:

- The loss of the end-to-end semantics of TCP might cause problems if the foreign agent partitioning the TCP connection crashes. If a sender receives an acknowledgement, it assumes that the receiver got the packet. Receiving an acknowledgement now only means (for the mobile host and a correspondent host that the foreign agent received the packet. The correspondent node does not know anything about the partitioning, so a crashing access node may also crash applications running on the correspondent node assuming reliable end-to end delivery.
- The foreign agent must be a trusted entity because the TCP connections end at this point. If users apply end-to end encryption, e.g., according to RFC 2401 (Kent, 1998a), the foreign agent has to be integrated into all security mechanisms.

2. Snooping TCP:

- In this approach, the foreign agent buffers all packets with destination mobile host and additionally snoops the packet flow in both directions to recognize acknowledgements.
- The reason for buffering packets toward the mobile node is to enable the foreign agent to perform a local retransmission in case of packet loss on the wireless link.
- The foreign agent buffers every packet until it receives an acknowledgement from the mobile host.
- If the foreign agent does not receive an acknowledgement from the mobile host within a certain amount of time, either the packet or the acknowledgement has been lost.
- Alternatively, the foreign agent could receive a duplicate ACK which also shows the loss of a packet.
- Now the foreign agent retransmits the packet directly from the buffer, performing a much faster retransmission compared to the correspondent host.
- The time out for acknowledgement scan be much shorter, because it reflects only the delay of one hop plus processing time.

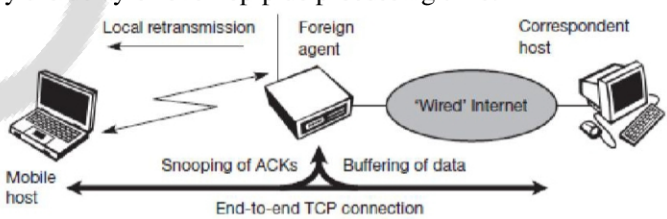


Figure 9.3 Snooping TCP as a transparent TCP extension

- To remain transparent, the foreign agent must not acknowledge data to the correspondent host.
- This would make the correspondent host believe that the mobile host had received the data and would violate the end-to-end semantic in case of a foreign agent failure.
- However, the foreign agent can filter the duplicate acknowledgements to avoid unnecessary retransmissions of data from the correspondent host.
- If the foreign agent now crashes, the time-out of the correspondent host still works and triggers a retransmission.
- The foreign agent may discard duplicates of packets already retransmitted locally and acknowledged by the mobile host. This avoids unnecessary traffic on the wireless link.
- Data transfer from the mobile host with destination correspondent host works as follows.
- The foreign agent snoops into the packet stream to detect gaps in the sequence numbers of TCP.
- As soon as the foreign agent detects a missing packet, it returns a negative acknowledgement (NACK) to the mobile host.
- The mobile host can now retransmit the missing packet immediately.
- Reordering of packets is done automatically at the correspondent host by TCP.

Extending the functions of a foreign agent with a snooping TCP has several advantages:

- The end-to-end TCP semantic is preserved. No matter at what time the foreign agent crashes, neither the correspondent host nor the mobile host have an inconsistent view of the TCP connection as is possible with ITCP.
- It does not need a handover of state as soon as the mobile host moves to another foreign agent. Assume there might still be data in the buffer not transferred to the next foreign agent.
- All that happens is a time-out at the correspondent host and retransmission of the packets, possibly already to the new care-of address.
- It does not matter if the next foreign agent uses the enhancement or not. If not, the approach automatically falls back to the standard solution.

However, the simplicity of the scheme also results in some disadvantages:

- Snooping TCP does not isolate the behavior of the wireless link as well as I-TCP. The problems on the wireless link are now also visible for the correspondent host and not fully isolated. It is problematic that the

wireless link exhibits very high delays compared to the wired link due to error correction on layer 2.

- Using negative acknowledgements between the foreign agent and the mobile host assumes additional mechanisms on the mobile host. This approach is no longer transparent for arbitrary mobile hosts.
- All efforts for snooping and buffering data may be useless if certain encryption schemes are applied end-to-end between the correspondent host and mobile host.
- Retransmitting data from the foreign agent may not work because many security schemes prevent replay attacks – retransmitting data from the foreign agent may be misinterpreted as replay.

3. **Mobile TCP:**

- The M-TCP (mobile TCP)1 approach has the same goals as I-TCP and snooping TCP: to prevent the sender window from shrinking if bit errors or disconnection but not congestion cause current problems.
- M-TCP wants to improve overall throughput, to lower the delay, to maintain end-to-end semantics of TCP, and to provide a more efficient handover.
- Additionally, M-TCP is especially adapted to the problems arising from lengthy or frequent disconnections.
- M-TCP splits the TCP connection into two parts as I-TCP does.
- An unmodified TCP is used on the standard host-supervisory host (SH) connection, while an optimized TCP is used on the SH-MH connection.
- The supervisory host is responsible for exchanging data between both parts similar to the proxy in ITCP(see Figure 9.1).

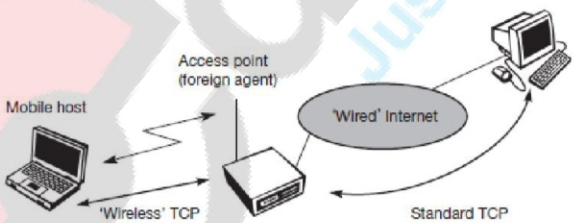


Figure 9.1
Indirect TCP segments a TCP connection into two parts

- The M-TCP approach assumes a relatively low bit error rate on the wireless link.
- Therefore, it does not perform caching/retransmission of data via the SH.
- If a packet is lost on the wireless link, it has to be retransmitted by the original sender.
- This maintains the TCP end-to-end semantics.

- The SH monitors all packets sent to the MH and ACKs returned from the MH.
- If the SH does not receive an ACK for some time, it assumes that the MH is disconnected.
- It then chokes the sender by setting the senders window size to.
- Setting the window size to 0 forces the sender to go into persistent mode, i.e., the state of the sender will not change no matter how long the receiver is disconnected.
- This means that the sender will not try to retransmit data.
- As soon as the SH (either the old SH or a new SH) detects connectivity again, it reopens the window of the sender to the old value.
- The sender can continue sending at full speed. This mechanism does not require changes to the senders TCP.
- The wireless side uses an adapted TCP that can recover from packet loss much faster.
- This modified TCP does not use slow start, thus, M-TCP needs a bandwidth manager to implement fair sharing over the wireless link.

The advantages of M-TCP are the following:

- It maintains the TCP end-to-end semantics. The SH does not send any ACK itself but forwards the ACKs from the MH.
- If the MH is disconnected, it avoids useless retransmissions, slow starts or breaking connections by simply shrinking the senders window to. Since it does not buffer data in the S(as)-TCP does, it is not necessary to forward buffers to a new SH.
- Lost packets will be automatically retransmitted to the new SH.

The lack of buffers and changing TCP on the wireless part also has some disadvantages:

- As the SH does not act as proxy as in I-TCP, packet loss on the wireless link due to bit errors is propagated to the sender. M-TCP assumes low bit error rates, which is not always a valid assumption.
- A modified TCP on the wireless link not only requires modifications to the MH protocol software but also new network elements like the bandwidth manager.

4. Fast Retransmit / Fast Recovery:

- The idea presented by Caceres is to artificially force the fast retransmit behaviour on the mobile host and correspondent host side.
- As soon as the mobile host registers at a new foreign agent using mobile IP, it starts sending duplicated acknowledgements to correspondent hosts.
- The proposal is to send three duplicates.
- This forces the corresponding host to go into fast retransmit mode and not to start slow start, i.e., the correspondent host continues to send with the same rate it did before the mobile host moved to another foreign agent.
- As the mobile host may also go into slow start after moving to a new foreign agent, this approach additionally puts the mobile host into fast retransmit.
- The mobile host retransmits all unacknowledged packets using the current congestion window size without going into slow start.
- The advantage of this approach is: o Its simplicity.
    - Only minor changes in the mobile hosts software already result in a performance increase.
    - No foreign agent or correspondent host has to be changed.
- The main disadvantage of this scheme is:
    - The insufficient isolation of packet losses.
    - Forcing fast retransmission increases the efficiency, but retransmitted packets still have to cross the whole network between correspondent host and mobile host.
    - If the handover from one foreign agent to another takes a longer time, the correspondent host will have already started retransmission.
- The approach focuses on loss due to handover: packet loss due to problems on the wireless link is not considered.
- This approach requires more cooperation between the mobile IP and TCP layer making it harder to change one without influencing the other.

5. Transmission/Timeout Freezing:

- While the approaches presented so far can handle short interruptions of the connection, either due to handover or transmission errors on the wireless link, some were designed for longer interruptions of transmission.
- Examples are the use of mobile hosts in a car driving into a tunnel, which loses its connection to, e.g., a satellite, or a user moving into a

cell with no capacity left over. In this case, the mobile phone system will interrupt the connection.

- Quite often, the MAC layer has already noticed connection problems, before the connection is actually interrupted from a TCP point of view.
- Additionally, the MAC layer knows the real reason for the interruption and does not assume congestion, as TCP would.
- The MAC layer can inform the TCP layer of an upcoming loss of connection or that the current interruption is not caused by congestion.
- TCP can now stop sending and freezes the current state of its congestion window and further timers.
- If the MAC layer notices the upcoming interruption early enough, both the mobile and correspondent host can be informed.
- With a fast interruption of the wireless link, additional mechanisms in the access point are needed to inform the correspondent host of the reason for interruption.
- Otherwise, the correspondent host goes into slow start assuming congestion and finally breaks the connection.
- The advantage of this approach is:
    - That it offers a way to resume TCP connection seven after longer interruptions of the connection.
    - It is independent of any other TCP mechanism, such as acknowledgements or sequence numbers, so it can be used together with encrypted data.
- However, this scheme has some severe disadvantages.
    - Not only does the software on the mobile host have to be changed, to be more effective the correspondent host cannot remain unchanged.
    - All mechanisms rely on the capability of the MAC layer to detect future interruptions.
    - Freezing the state of TCP does not help in case of some encryption schemes that use time-dependent random numbers.
    - These schemes need resynchronization after interruption.

6. Selective Retransmission:

- A very useful extension of TCP is the use of selective retransmission.
- TCP acknowledgements are cumulative, i.e., they acknowledge in-order receipt of packets up to a certain packet.
- If a single packet is lost, the sender has to retransmit everything starting from the lost packet (go-back-n retransmission).
- This obviously wastes bandwidth, not just in the case of a mobile network, but for any network.

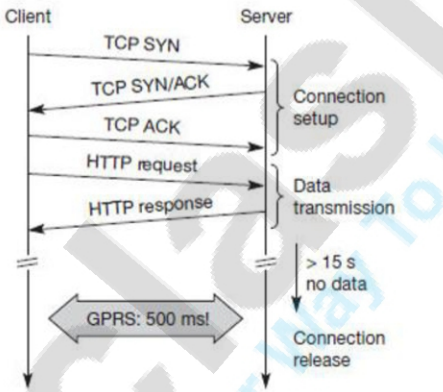
- TCP can indirectly request a selective retransmission of packets.
- The receiver can acknowledge single packets, not only trains of in-sequence packets.
- The sender can now determine precisely which packet is needed and can retransmit it.
- The advantage of this approach is obvious:
  - a sender retransmits only the lost packets.
  - This lowers bandwidth requirements and is extremely helpful in slow wireless links.
  - The gain in efficiency is not restricted to wireless links and mobile environments.
  - Using selective retransmission is also beneficial in all other networks.
- However, there might be the minor disadvantage of more complex software on the receiver side, because now more buffers is necessary to re-sequence data and to wait for gaps to be filled.
- But while memory sizes and CPU performance permanently increase, the bandwidth of the air interface remains almost the same.
- Therefore, the higher complexity is no real disadvantage any longer as it was in the early days of TCP.

7. Transaction Oriented TCP:

- Assume an application running on the mobile host that sends a short request to a server from time to time, which responds with a short message.
- If the application requires reliable transport of the packets, it may use TCP.
- Using TCP now requires several packets over the wireless link.
- First, TCP uses a three-way handshake to establish the connection.
- At least one additional packet is usually needed for transmission of the request, and requires three more packets to close the connection via a three-way handshake.
- Assuming connections with a lot of traffic or with a long duration, this overhead is minimal.
- But in an example of only one data packet, TCP may need seven packets altogether.
- Figure 9.4 shows an example for the overhead introduced by using TCP over GPRS in a web scenario.
- Web services are based on HTTP which requires a reliable transport system. In the internet, TCP is used for this purpose.

- Before a HTTP request can be transmitted the TCP connection has to be established.
- This already requires three messages. If GPRS is used as wide area transport system, one-way delays of 500 ms and more are quite common.
- The setup of a TCP connection already takes far more than a second.

Figure 9.4
Example TCP connection setup overhead



- The obvious advantage for certain applications is the reduction in the overhead which standard TCP has for connection setup and connection release.
- However, T/TCP is not the original TCP anymore, so it requires changes in the mobile host and all correspondent hosts, which is a major disadvantage.
- This solution no longer hides mobility. Furthermore, T/TCP exhibits several security problems

Approach	Mechanism	Advantages	Disadvantages
Indirect TCP	Splits TCP connection into two connections	Isolation of wireless link, simple	Loss of TCP semantics, higher latency at handover, security problems
Snooping TCP	Snoops data and acknowledgements, local retransmission	Transparent for end-to-end connection, MAC integration possible	Insufficient isolation of wireless link, security problems
M-TCP	Splits TCP connection, chokes sender via window size	Maintains end-to-end semantics, handles long term and frequent disconnections	Bad isolation of wireless link, processing overhead due to bandwidth management, security problems
Fast retransmit/ fast recovery	Avoids slow-start after roaming	Simple and efficient	Mixed layers, not transparent
Transmission/ time-out freezing	Freezes TCP state at disconnection, resumes after reconnection	Independent of content, works for longer interruptions	Changes in TCP required, MAC dependent
Selective retransmission	Retransmits only lost data	Very efficient	Slightly more complex receiver software, more buffer space needed
Transaction-oriented TCP	Combines connection setup/release and data transmission	Efficient for certain applications	Changes in TCP required, not transparent, security problems

Table 9.1 Overview of classical enhancements to TCP for mobility

TCP over 2.5 / 3G wireless Networks:

- The current internet draft for TCP over 2.5G/3G wireless networks describes a profile for optimizing TCP over todays and tomorrows wireless WANs such as GSM/GPRS, UMTS, or CDMA.
- The configuration optimizations recommended in this draft can be found in most of today's TCP implementations so this draft does not require an update of millions of TCPstacks.
- The focus on 2.5G/3G for transport of internet data is important as already more than 1 billion people use mobile phones and it is obvious that the mobile phone systems will also be used to transport arbitrary internet data.

The following characteristics have to be considered when deploying applications over 2.5G/3G wireless links:

- **Data rates:**
  - While typical data rates of todays 2.5G systems are 10 –20 kbit/s uplink and 20–50 kbit/s downlink, 3G and future 2.5G systems will initially offer data rates around 64 kbit/s uplink and 115–384 kbit/s downlink.
  - Typically, data rates are asymmetric as it is expected that users will download more data compared to uploading. Uploading is limited by the limited battery power.

- In cellular networks, asymmetry does not exceed 3–6 times, however, considering broadcast systems as additional distribution media(digital radio, satellite systems), asymmetry may reach a factor of 1,000.

- **Latency:**
  - All wireless systems comprise elaborated algorithms for error correction and protection, such as forward error correction (FEC), check summing, and interleaving.
  - FEC and interleaving let the round trip time(RTT) grow to several hundred milliseconds up to some seconds.
- **Jitter:**
  - Wireless systems suffer from large delay variations or delay spikes.
  - Reasons for sudden increase in the latency are: link outages due to temporal loss of radio coverage, blocking due to high-priority traffic, or handovers.
- **Packet loss:**
  - Packets might be lost during handovers or due to corruption. o Thanks to link-level retransmissions the loss rates of 2.5G/3G systems due to corruption are relatively low (but still orders of magnitude higher than, e.g., fiber connections!).
- **Large windows:**
  - TCP should support large enough window sizes based on the bandwidth delay product experienced in wireless systems. With the help of the windows scale option and larger buffer sizes this can be accomplished.
- **Limited transmit:**
  - This mechanism, defined in RFC 3042 is an extension of Fast Retransmission/Fast Recovery and is particularly useful when small amounts of data are to be transmitted.
- **Large MTU:**
  - The larger the MTU (Maximum Transfer Unit) the faster TCP increases the congestion window.
  - Link layers fragment PDUs for transmission anyway according to their needs and large MTUs may be used to increase performance.
- **Selective Acknowledgement (SACK):**
  - SACK (RFC 2018) allows the selective retransmission of packets and is almost always beneficial compared to the standard cumulative scheme.
- **Explicit Congestion Notification (ECN):**

- ECN as defined in RFC 3168 allows a receiver to inform a sender of congestion in the network by setting the ECN-Echo flag on receiving an IP packet that has experienced congestion.
- This mechanism makes it easier to distinguish packet loss due to transmission errors from packet loss due to congestion.
- However, this can only be achieved when ECN capable routers are deployed in the network.

- **Timestamp:**
  - TCP connections with large windows may benefit from more frequent RTT samples provided with timestamps by adapting quicker to changing network conditions.
  - With the help of timestamps higher delay spikes can be tolerated by TCP without experiencing a spurious timeout.
  - The effect of bandwidth oscillation is also reduced.
- **No header compression:**
  - As the TCP header compression mechanism according to RFC 1144 does not perform well in the presence of packet losses this mechanism should not be used.
  - Header compression according to RFC 2507 or RFC 1144 is not compatible with TCP options such as SACK or timestamps.