UNIT-II

Motivation for Mobile IP

- IP Routing
  - based on IP destination address, network prefix (e.g. 129.13.42) determines physical subnet
  - change of physical subnet implies change of IP address to have a topologically correct address (standard IP) or needs special entries in the routing tables
- Specific routes to end-systems?
  - requires changing all routing table entries to forward packets to the right destination
  - does not scale with the number of mobile hosts and frequent changes in the location, security problems
- Changing the IP-address?
  - adjust the host IP address depending on the current location
  - almost impossible to find a mobile system, DNS updates take long time
  - TCP connections break, security problems

What Mobile IP does:

- Mobile IP solves the following problems:
  - **if a node moves without changing its IP address it will be unable to receive its packets,**
  - **if a node changes its IP address it will have to terminate and restart its ongoing connections everytime it moves to a new network area (new network prefix).**
- Mobile IP is a routing protocol with a very specific purpose.
- Mobile IP is a network layer solution to node mobility in the Internet.
- Mobile IP is not a complete solution to mobility, changes to the transport protocols need to be made for a better solution (i.e., the transport layers are unaware of the mobile node's point of attachment and it might be useful if, e.g., TCP knew that a wireless link was being used!).

Goals of Mobile IP:

- Transparency
  - mobile end-systems keep their IP address
  - continuation of communication after interruption of link possible
  - point of connection to the fixed network can be changed
- Compatibility
  - support of the same layer 2 protocols as IP
  - no changes to current end-systems and routers required

- mobile end-systems can communicate with fixed systems

- Security
  - authentication of all registration messages
- Efficiency and scalability
  - only little additional messages to the mobile system required (connection typically via a low bandwidth radio link)
  - world-wide support of a large number of mobile systems in the whole Internet.

Mobile IP Terminology,Entities:

- Mobile Node (MN)
  - system (node) that can change the point of connection to the network without changing its IP address
- Home Agent (HA)
  - system in the home network of the MN, typically a router
  - registers the location of the MN, tunnels IP datagrams to the COA
- Foreign Agent (FA)
  - system in the current foreign network of the MN, typically a router
  - forwards the tunneled datagrams to the MN, typically also the default router for the MN
- Care-of Address (COA)
  - address of the current tunnel end-point for the MN (at FA or MN)
  - actual location of the MN from an IP point of view
  - can be chosen, e.g., via DHCP
- Correspondent Node (CN)

  At least one communication partner. either mobile /fixed device.

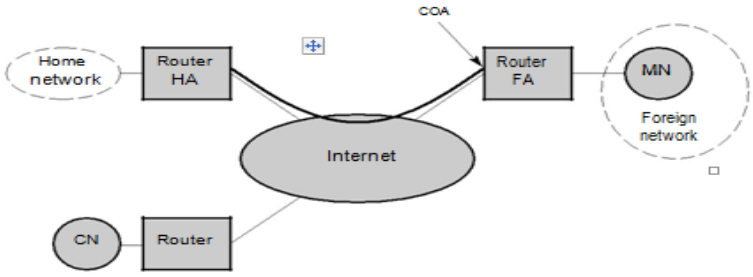


Figure 8.1
Mobile IP example network

IP packet delivery:

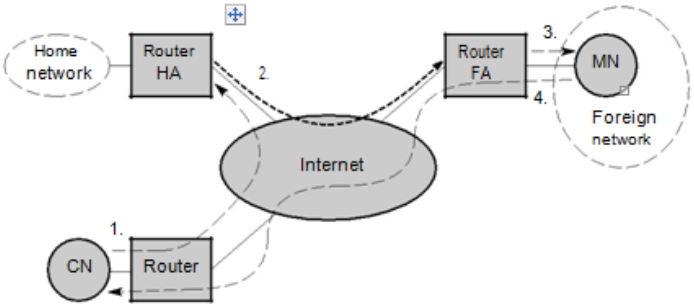


Figure 8.2
Packet delivery to and from the mobile node

- **1.** Sender sends to the IP address of MN, HA intercepts packet (proxy ARP)
- 2. HA tunnels packet to COA, here FA, by encapsulation
- 3.FA forwards the packet to the MN

Agent discovery:

- Agent Advertisement
    - HA and FA periodically send advertisement messages into their physical subnets
    - MN listens to these messages and detects, if it is in the home or a foreign network (standard case for home network)
    - MN reads a COA from the FA advertisement messages
- Agent Solicitation
- Registration (always limited lifetime!)
    - MN signals COA to the HA via the FA, HA acknowledges via FA to MN
    - these actions have to be secured by authentication
- Advertisement
    - HA advertises the IP address of the MN (as for fixed systems), i.e. standard routing information
    - routers adjust their entries, these are stable for a longer time (HA responsible for a MN over a longer period of time)
    - packets to the MN are sent to the HA,
    - independent of changes in COA/FA
- Agent advertisement

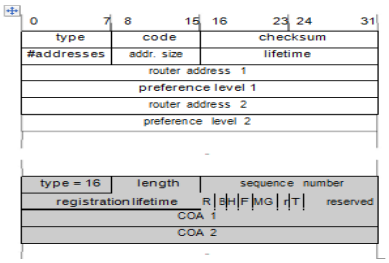


Figure 8.3
Agent advertisement packet (RFC 1256 + mobility extension)

- type = 16
- length = 6 + 4 * #COAs
- R: registration required
- B: busy, no more registrations
- H: home agent
- F: foreign agent
- M: minimal encapsulation
- G: GRE encapsulation
- r: =0, ignored (former Van Jacobson compression)
- T: FA supports reverse tunneling
- reserved: =0, ignored

- Agent Solicitation
  - If no agent advertisements are present or the inter-arrival time is too high, and an MN has not received a COA the mobile node must send agent solicitations. These solicitations are again based on RFC 1256 for router solicitations. Care must be taken to ensure that these solicitation messages do not flood the network, but basically an MN can search for an FA endlessly sending out solicitation messages.

- Registration
  - Having received a COA, the MN has to register with the HA. The main purpose of the registration is to inform the HA of the current location for correct for-warding of packets. Registration can be done in two different ways depending on the location of the COA
  - If the COA is at the FA, registration is done as illustrated in Figure 8.4 (left). The MN sends its registration request containing the COA (see Figure 8.5) to the FA which is forwarding the request to the HA. The HA now sets up a mobility binding containing the mobile node's home IP address and the cur-rent COA.

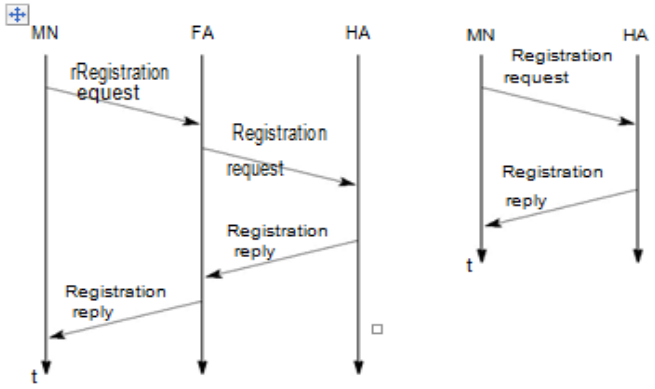


Figure 8.4 Registration of a mobile node via the FA or directly with the HA

- If the COA is co-located, registration can be simpler, as shown in Figure 8.4 (right). The MN may send the request directly to the HA and vice versa.

Mobile IP registration request:

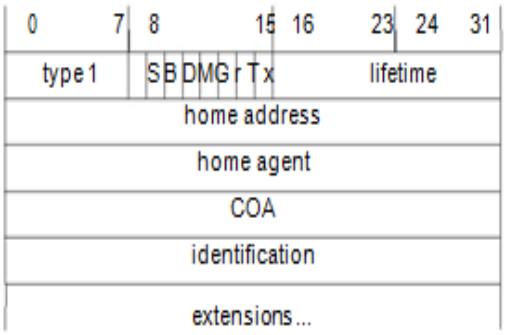


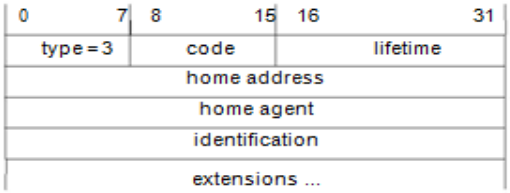
Figure 8.5
Registration request

- S: simultaneous bindings
- B: broadcast datagrams
- D: decapsulation by MN
- M mininal encapsulation
- G: GRE encapsulation
- r: =0, ignored
- T: reverse tunneling requested
- x: =0, ignored

. **Mobile IP registration reply:**

- UDP packets are used for registration requests. The IP source address of the packet is set to the interface address of the MN, the IP destination address is that of the FA or HA.

Figure 8.6
Registration reply



- **Lifetime** denotes the validity of the registration in seconds. A value of zero indicates deregistration; all bits set indicates infinity.
- The **home address** is the fixed IP address of the MN.
- **home agent** is the IP address of the HA, and **COA** represents the tunnel endpoint
- . The 64 bit **identification** is generated by the MN to identify a request and match it with registration replies. This field is used for protection against replay attacks of registrations.
- The **extensions** must at least contain parameters for authentication.

Table 8.1 Example registration reply codes

Registration	Code	Explanation
successful	0	registration accepted
	1	registration accepted, but simultaneous mobility bindings unsupported
denied by FA	65	administratively prohibited
	66	insufficient resources
	67	mobile node failed authentication
	68	home agent failed authentication
	69	requested lifetime too long
denied by HA	129	administratively prohibited
	130	insufficient resources
	131	mobile node failed authentication
	132	foreign agent failed authentication
	133	registration identification mismatch
	135	too many simultaneous mobility bindings

Tunneling and Encapsulation:

• A tunnel establishes a vir-tual pipe for data packets between a tunnel entry and a tunnel endpoint. Packets entering a tunnel are forwarded inside the tunnel and leave the tunnel unchanged. **Tunneling,** i.e., sending a packet through a tunnel, is achieved by using encapsulation.

• **Encapsulation** is the mechanism of taking a packet consisting of packet header and data and putting it into the data part of a new packet. The reverse operation, taking a packet out of the data part of another packet, is called **Decapsulation.**

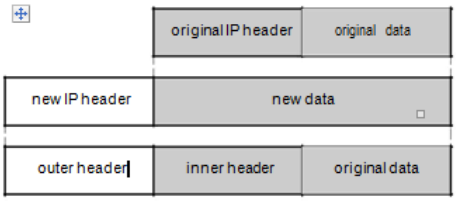
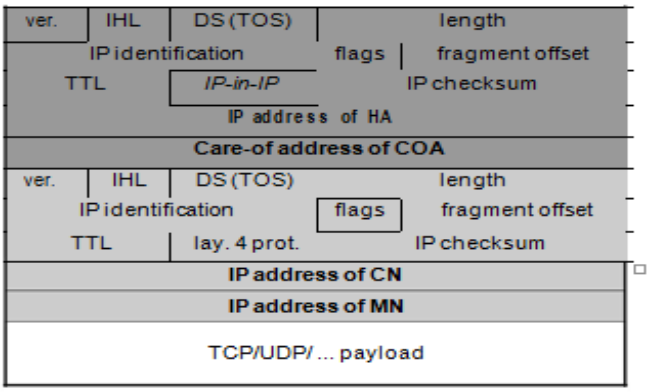


Figure 8.7
IP encapsulation

i) IP-in-IP-encapsulation:

Tunnel between HA and COA.

Figure 8.8
IP-in-IP encapsulation



- The version field **ver** is 4 for IP ver-sion 4.
- The internet header length (**IHL**) denotes the length of the outer header in 32 bit words. **DS(TOS)** is just copied from the inner header,
- The **length** field covers the complete encapsulated packet.
- **TTL** must be high enough so the packet can reach the tunnel endpoint.
- The next field, here denoted with **IP-in-IP**, is the type of the protocol used in the IP payload. This field is set to 4, the protocol type for IPv4 because again an IPv4 packet follows after this outer header.
- IP **checksum** is calculated as usual.
- The next fields are the tunnel entry as source address (the **IP address of the HA**) and the tunnel exit point as destination address (the **COA**).

ii)Minimal encapsulation:

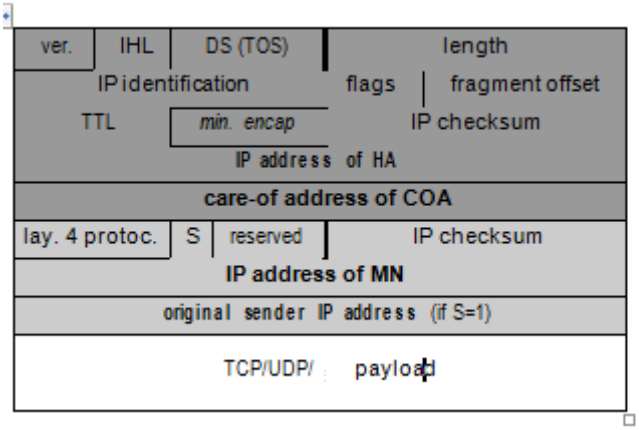


Figure 8.9
Minimal encapsulation

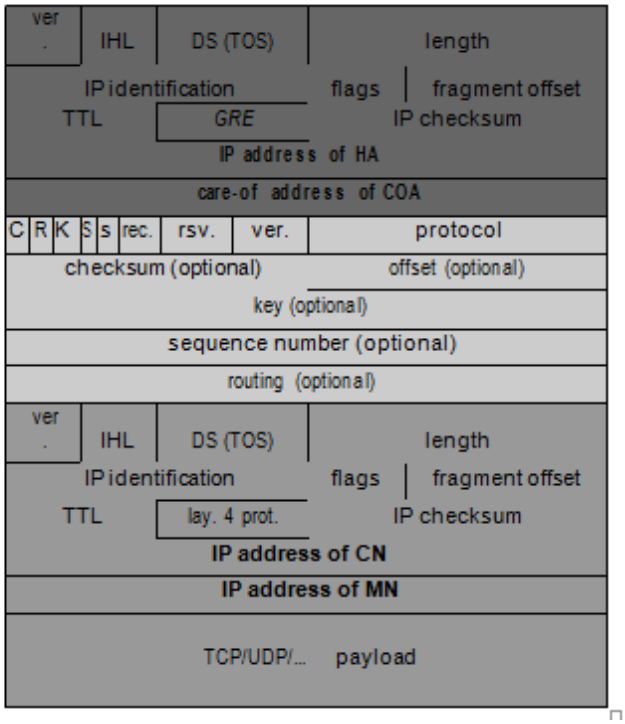
- avoids repetition of identical fields
- e.g. TTL, IHL, version, TOS

- only applicable for unfragmented packets, no space left for fragment **identification.**

iii)Generic routing encapsulation:

Figure 8.11

Protocol fields for GRE according to RFC 1701



- While IP-in-IP encapsulation and minimal encapsulation work only for IP, the following encapsulation scheme also supports other network layer protocols in addition to IP. Generic routing encapsulation (GRE) allows the encapsulation of packets of one protocol suite into the payload portion of a packet of another protocol suite.
- A minimal GRE header uses only 4 bytes; nevertheless, GRE is flexible enough to include several mechanisms in its header.
- The C bit indicates if the checksum field is present and contains valid information. If C is set, the checksum field contains a valid IP checksum of the GRE header and the pay-load.
- The R bit indicates if the offset and routing fields are present and contain valid information. The offset represents the offset in bytes for the first source routing entry. The routing field, if present, has a variable length and contains fields for source routing.
- If the C bit is set, the offset field is also present and, vice versa, if the R bit is set, the checksum field must be present. The only reason for this is to align the following fields to 4 bytes. The checksum field is valid only if C is set, and the offset field is valid only if R is set respectively.
- GRE also offers a **key** field which may be used for authentication. If this field is present, the **K** bit is set.
- . The sequence number bit **S** indicates if the **sequence** number field is present, if the s bit is set, strict source routing is used.
- **reserved** fields must be zero and are ignored on reception. The **version** field contains 0 for the GRE version.

- . The **ver-sion** field contains the value zero. The **protocol** type, again, defines the protocol of the payload following RFC 3232.

Optimizations

i) Optimization of packet forwarding

- Change of FA
- packets on-the-fly during the change can be lost
- new FA informs old FA to avoid packet loss, old FA now forwards remaining packets to new FA
- this information also enables the old FA to release resources for the MN

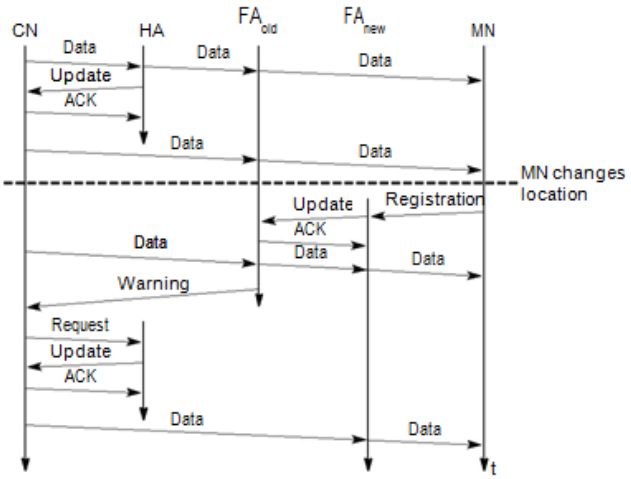


Figure 8.13
Change of the foreign agent with an optimized mobile IP

- **Triangle routing has the MN correspond directly with the CN using its home address as the SA**
  - Firewalls at the foreign network may not allow that
  - Multicasting: if a MN is to participate in a multicast group, it needs to use a reverse tunnel to maintain its association with the home network.
  - TTL: a MN might have a TTL that is suitable for communication when it is in its HM. This TTL may not be sufficient when moving around (longer routes possibly). When using a reverse tunnel, it only counts as a single hop. A MN does not want to change the TTL everytime it moves.
- **Solution: reverse tunneling**

Reverse tunneling

- Routers accept often only "topologically correct" addresses (firewall!)
- a packet from the MN encapsulated by the FA is now topologically correct
- Multicast and TTL problems solved
- Reverse tunneling does not solve
- all problems with firewalls, the reverse tunnel can be abused to circumvent security mechanisms (tunnel hijacking)
- optimization of data paths, i.e. packets will be forwarded through the tunnel via the HA to a sender (longer routes)
- The new standard is backwards compatible

- the extensions can be implemented easily

Problems with Mobile IP

➢ Security

- Authentication with FA problematic, for the FA typically belongs to another organization
- No protocol for key management and key distribution has been standardized in the Internet
- patent and export restrictions

➢ Firewalls

- Typically mobile IP cannot be used together with firewalls, special set-ups are needed (such as reverse tunneling)

➢ QoS

- Many new reservations in case of RSVP
- Tunneling makes it hard to give a flow of packets a special treatment needed for the QoS

DHCP: Dynamic Host Configuration Protocol

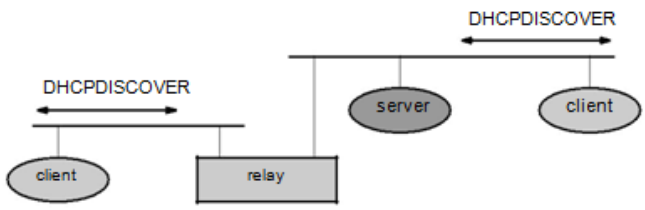
➢ Application

- Simplification of installation and maintenance of networked computers
- Supplies systems with all necessary information, such as ip address, dns server address, domain name, subnet mask, default router etc.
- Enables automatic integration of systems into an intranet or the internet, can be used to acquire a coa for mobile ip

➢ Client/Server-Model

- The client sends via a MAC broadcast a request to the DHCP server (might be via a DHCP relay)

Figure 8.17
Basic DHCP
configuration



-
- The client broadcasts a DHCPDISCOVER into the subnet.
- Servers reply to the client's request with DHCPOFFER and offer a list of configuration para-meters. The client can now choose one of the configurations

offered. The client in turn replies to the servers, accepting one of the configurations and rejecting the others using DHCPREQUEST.

- If a server receives a DHCPREQUEST with a rejection, it can free the reserved configuration for other possible clients. The server with the configuration accepted by the client now confirms the configu-ration with DHCPACK. This completes the initialization phase.
-

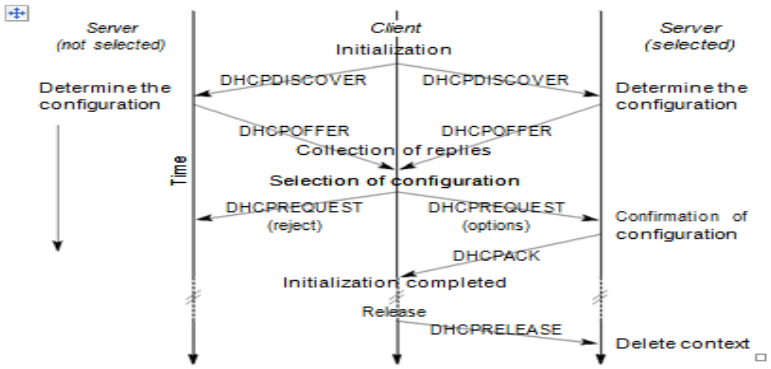


Figure 8.18 Client initialization via DHCP

Mobile Transport Layer

- TCP originally designed for
  - Fixed end-systems
  - Fixed, wired networks

TCP congestion control

- packet loss in fixed networks typically due to (temporary) overload situations
- router have to discard packets as soon as the buffers are full
- TCP recognizes congestion only indirect via missing acknowledgements, retransmissions unwise, they would only contribute to the congestion and make it even worse

- **TCP slow-start algorithm**
  - sender calculates a congestion window for a receiver
  - start with a congestion window size equal to one segment
  - exponential increase of the congestion window up to the congestion threshold, then linear increase
  - missing acknowledgement causes the reduction of the congestion threshold to one half of the current congestion window
  - congestion window starts again with one segment

- **TCP fast retransmit/fast recovery**
  - TCP sends an acknowledgement only after receiving a packet

- if a sender receives several acknowledgements for the same packet, this is due to a gap in received packets at the receiver
- however, the receiver got all packets up to the gap and is actually receiving packets
- therefore, packet loss is not due to congestion, continue with current congestion window (do not use slow-start)
- Change of foreign agent often results in packet loss
  - TCP reacts with slow-start although there is no congestion
- Forced fast retransmit
  - as soon as the mobile host has registered with a new foreign agent, the MH sends duplicated acknowledgements on purpose
  - this forces the fast retransmit mode at the communication partners
  - additionally, the TCP on the MH is forced to continue sending with the actual window size and not to go into slow-start after registration
- Advantage
  - simple changes result in significant higher performance
- Disadvantage
  - further mix of IP and TCP, no transparent approach

Transmission/time-out freezing:

- **Mobile hosts can be disconnected for a longer time**
  - no packet exchange possible, e.g., in a tunnel, disconnection due to overloaded cells or mux. with higher priority traffic
  - TCP disconnects after time-out completely
- **TCP freezing**
  - MAC layer is often able to detect interruption in advance
  - MAC can inform TCP layer of upcoming loss of connection
  - TCP stops sending, but does now not assume a congested link
  - MAC layer signals again if reconnected
- **Advantage**
  - scheme is independent of data
- **Disadvantage**
  - TCP on mobile host has to be changed, mechanism depends on MAC layer

Selective retransmission:

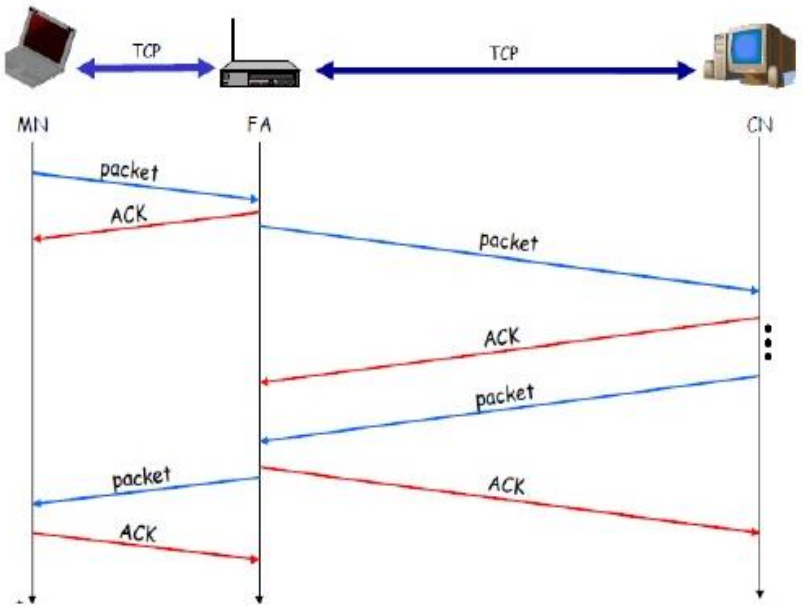
- **TCP acknowledgements are often cumulative**
  - ACK n acknowledges correct and in-sequence receipt of packets up to n
  - if single packets are missing quite often a whole packet sequence beginning at the gap has to be retransmitted (go-back-n), thus wasting bandwidth
- **Selective retransmission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

- **Advantage**
  - much higher efficiency
- **Disadvantage**
  - more complex software in a receiver, more buffer needed at the receiver.

Indirect TCP:

- Splitting of the TCP connection at, e.g., the foreign agent into 2 TCP connections, no real end-to-end connection any longer
- no changes to the TCP protocol for hosts connected to the wired Internet,.
- optimized TCP protocol for mobile hosts
- hosts in the fixed part of the node do not notice the characteristics of the wireless part



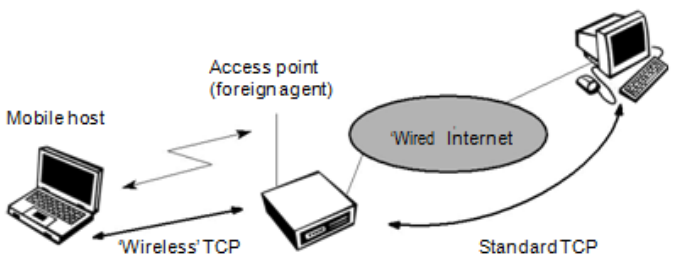


Figure 9.1
Indirect TCP segments a TCP connection into two parts

- Advantages

- no changes in the fixed network necessary, no changes for the hosts (TCP protocol) necessary, all current optimizations to TCP still work
- transmission errors on the wireless link do not propagate into the 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

Snooping TCP:

- **"Transparent" extension of TCP within the foreign agent**
  - **B**uffering 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

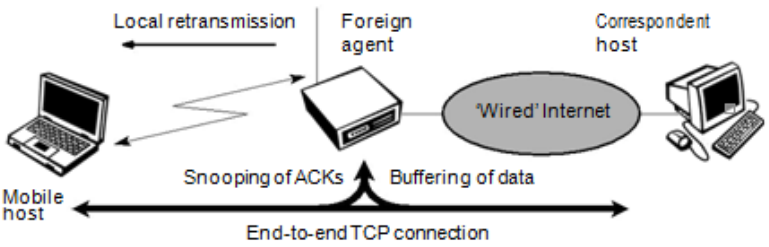
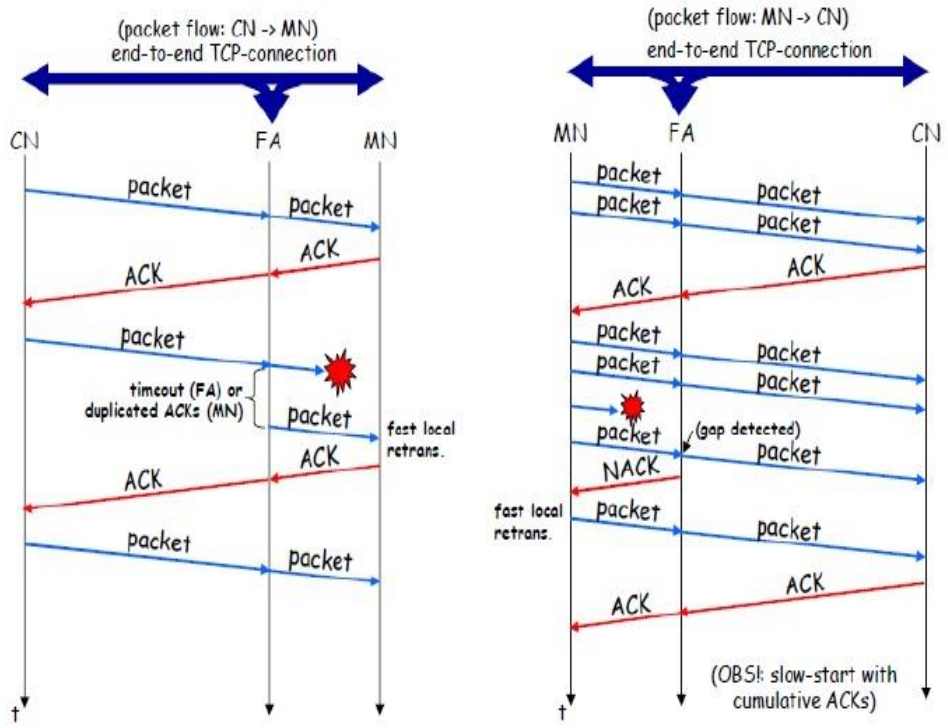


Figure 9.3 Snooping TCP as a transparent TCP extension

\

- 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 of the 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 useless depending on encryption schemes



M-TCP:

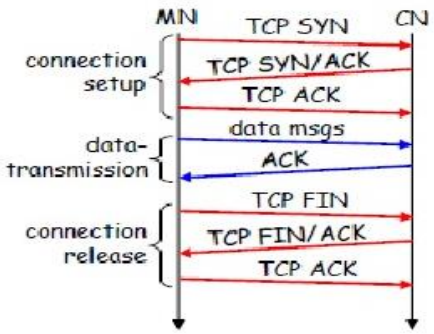
- **Special handling of lengthy and/or frequent disconnections**
- **M-TCP splits as I-TCP does**
  - unmodified TCP fixed network to supervisory host (SH)
  - optimized TCP SH to MH
- **Supervisory host**
  - no caching, no retransmission
  - monitors all packets, if disconnection detected
    - set sender window size to 0
    - sender automatically goes into persistent mode
  - old or new SH reopen the window
- **Advantages**
  - maintains semantics, supports disconnection, no buffer forwarding

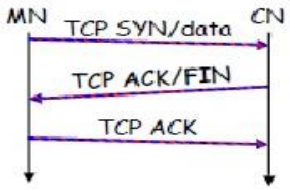
- **Disadvantages**
  - loss on wireless link propagated into fixed network
  - adapted TCP on wireless link.

Transaction oriented TCP:

- **TCP phases**
  - connection setup, data transmission, connection release
  - using 3-way-handshake needs 3 packets for setup and release, respectively
  - 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
- Advantage
  - Efficiency
- Disadvantage

  Requires changed TCP
  - Mobility not longer transparent





Comparison of "mobile" TCPs:

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