"Faculty Orientation Workshop on BE (E&TC)

Revised Syllabus 2019 Course"

under the aegis of Board of Studies

Electronics and Telecommunication Engg, SPPU, Pune

on subject

Mobile Computing (Elective 5).

UNIT 4: MOBILE IP AND TRANSPORT LAYER

Recourse Faculty
31 Jan 23

Dr. Vijay M. Birari_MVP'S, KBTCOE, Nashik

Unit 4: Mobile IP & Transport Layer

- **Course Objectives:**
- To study the working of Mobile IP.

- **Course Outcome:**
- Understand IP and Transport layers of Mobile Communication.

Unit 4: Mobile IP & Transport Layer

- **Reference Book**
- Jochen Schiller, “Mobile Communications”, 2nd Edition, Pearson.
- **Ch No 8 : Mobile Network Layer**
  - Mobile IP_8.1.1 to 8.1.8_Page 304 to 321
- **Ch No 9 : Mobile Transport Layer**
  - Traditional TCP_9.1_Page 352 to 354
  - Classical TCP Improvement_9.2_Page 355 to 364
- **Text Books:**
- Clint Smith, Daniel Collins, “Wireless Networks”, 3rd Edition, McGraw Hill Publications,

Unit 4: Mobile IP & Transport Layer

- **Mobile IP:**
  - **Need of mobile IP,**
  - **IP packet delivery,**
  - **Agent Discovery,**
  - **Registration,**
  - **Tunnelling and encapsulation,**
  - **Route optimization,**
  - **IP Handoff.**

- **Transport Layer:**
  - **Overview of Traditional TCP and implications of mobility control.**
  - **Improvement of TCP: Indirect TCP,**
  - **Snoop TCP,**
  - **Mobile TCP,**
  - **Fast Retransmit/fast recovery,**
  - **Time-out freezing,**
  - **Selective retransmission,**
  - **Transaction-oriented TCP.**

Unit 4: Mobile IP & Transport Layer

- **Prerequisite**
- **1. Basics of Mobile Technologies.**
- **2. Fundamental of Networking**

Goals, assumptions and requirements

- □ The internet is the network for global data communication with hundreds of millions of users. So why not simply use a mobile computer in the internet?
- □ The reason is quite simple:
- □ you will not receive a single packet as soon as you leave your home network, i.e., the network your computer is configured for, and reconnect your computer (wireless or wired) at another place (if no additional mechanisms are available).
- □ The reason for this is quite simple if you consider routing mechanisms on the internet.

Goals, assumptions and requirements

- A host sends an IP packet with the header containing a destination address with other fields. The destination address not only determines the receiver of the packet, but also the physical subnet of the receiver.
- For example, the destination address 129.13.42.99 shows that the receiver must be connected to the physical subnet with the network prefix 129.13.42 (unless CIDR is used, RFC 1519, Fuller, 1993).
- Routers in the internet now look at the destination addresses of incoming packets and forward them according to internal look-up tables.
- To avoid an explosion of routing tables, only prefixes are stored and further optimizations are applied.

Goals, assumptions and requirements

- A router would otherwise have to store the addresses of all computers in the internet, which is obviously not feasible. As long as the receiver can be reached within its physical subnet, it gets the packets; as soon as it moves outside the subnet, a packet will not reach it.
- A host needs a so-called **topologically correct address.**

Goals, assumptions and requirements

- 8.1.1.2 Requirements
- Since the quick ‘solutions’ obviously did not work, a more general architecture had to be designed.
- Many field trials and proprietary systems finally led to mobile IP as a standard to enable mobility in the internet.
- Several requirements accompanied the development of the standard:

Goals, assumptions and requirements

- **Compatibility:**
- The installed base of Internet computers, i.e., computers running TCP/IP and connected to the internet, is huge.
- People still want to use their favorite browser for www and do not want to change applications just for mobility, the same holds for operating systems.
- Mobile IP has to remain compatible with all lower layers used for the standard, non-mobile, IP.
- mobile IP implementation should still be able to communicate with fixed systems.

Goals, assumptions and requirements

- **Transparency: Mobility should remain 'invisible' for many higher layer** protocols and applications.
- Higher layers should continue to work even if the mobile computer has changed its point of attachment to the network.
- For TCP this means that the computer must keep its IP address.

Goals, assumptions and requirements

- □ **Scalability and efficiency: Introducing a new mechanism to the internet** must not jeopardize its efficiency.
- □ Enhancing IP for mobility must not generate too many new messages flooding the whole network.

Goals, assumptions and requirements

- □ **Compatibility: The installed base of Internet computers, i.e., computers** running TCP/IP and connected to the internet, is huge.
- □ A new standard cannot introduce changes for applications or network protocols already in use.
- □ Mobile IP has to be integrated into existing operating systems or at least work with them.
- □ Routers within the internet should not necessarily require other software.

Goals, assumptions and requirements

- **Security: Mobility poses many security problems. The minimum requirement** is that of all the messages related to the management of Mobile IP are authenticated.
- The IP layer must be sure that if it forwards a packet to a mobile host that this host receives the packet.

Entities and terminology

- 8.1.2 Entities and terminology
- **Mobile node (MN):**
- **A mobile node is an end-system or router that can** change its point of attachment to the internet using mobile IP.
- The MN keeps its IP address and can continuously communicate with any other system in the internet as long as link-layer connectivity is given.
- Mobile nodes are not necessarily small devices such as laptops with antennas or mobile phones; a router onboard an aircraft can be a powerful mobile node.

Entities and terminology

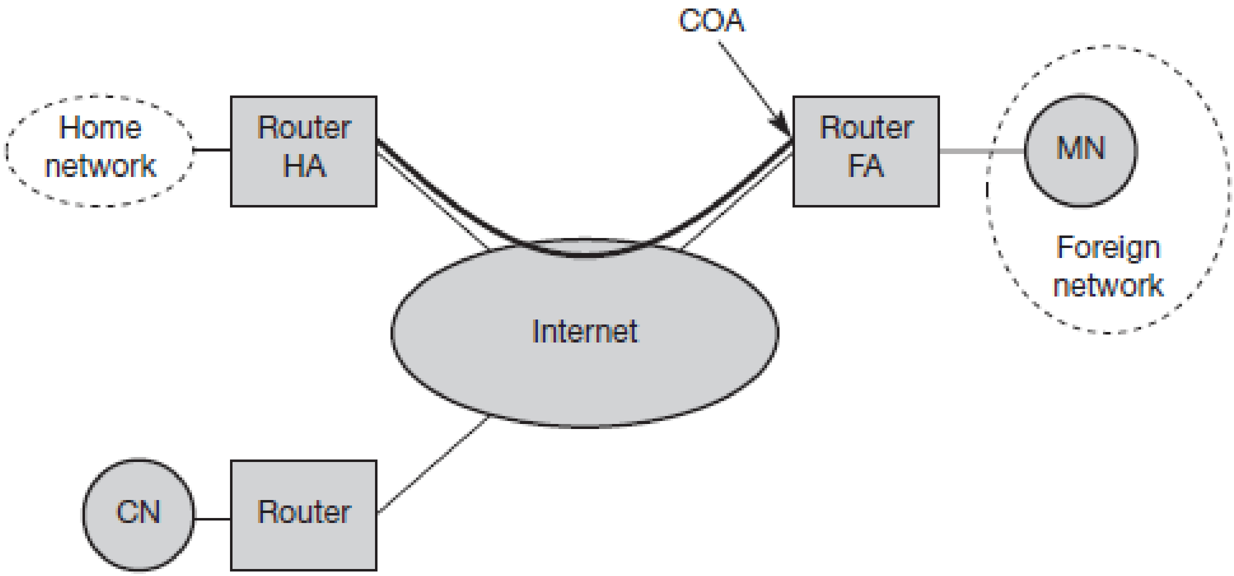


Figure 8.1
Mobile IP example network

Entities and terminology

- □ **Correspondent node (CN):**
- □ **At least one partner is needed for communication.**
- □ In the following the CN represents this partner for the MN. The CN
- □ can be a fixed or mobile node.
- □ **● Home network: The home network is the subnet the MN belongs to** with respect to its IP address. No mobile IP support is needed within the home network.
- □ **● Foreign network: The foreign network is the current subnet the MN visits** and which is not the home network.

Entities and terminology

- **Foreign agent (FA):**
- **The FA can provide several services to the MN during** its visit to the foreign network.
- The FA can have the COA (defined below), acting as tunnel endpoint and forwarding packets to the MN.
- The FA can be the default router for the MN.
- FAs can also provide security services because they belong to the foreign network as opposed to the MN which is only visiting.
- For mobile IP functioning, FAs are not necessarily needed. Typically, an FA is implemented on a router for the subnet the MN attaches to.

Entities and terminology

- **Care-of address (COA):**
- **The COA defines the current location of the MN** from an IP point of view.
- All IP packets sent to the MN are delivered to the COA, not directly to the IP address of the MN. Packet delivery toward the MN is done using a tunnel, as explained later.
- To be more precise, the COA marks the tunnel endpoint, i.e., the address where packets exit the tunnel.
- There are two different possibilities for the location of the COA:

Entities and terminology

- **Foreign agent COA:**
- **The COA could be located at the FA, i.e., the COA**
- is an IP address of the FA.
- The FA is the tunnel end-point and forwards packets to the MN.
- Many MN using the FA can share this COA as common COA.

Entities and terminology

- □ ● **Co-located COA:**
- □ **The COA is co-located if the MN temporarily acquired** an additional IP address which acts as COA.
- □ This address is now topologically correct, and the tunnel endpoint is at the MN. Co-located addresses can be acquired using services such as DHCP (see section 8.2).
- □ One problem associated with this approach is the need for additional addresses if MNs request a COA.
- □ This is not always a good idea considering the scarcity of IPv4 addresses.

Entities and terminology

- **Home agent (HA):**
- **The HA provides several services for the MN and is located** in the home network.
- The tunnel for packets toward the MN starts at the HA.
- The HA maintains a location registry, i.e., it is informed of the MN's location by the current COA.
- Three alternatives for the implementation of an HA exist.
- ● The HA can be implemented on a router that is responsible for the home network. This is obviously the best position, because without optimizations to mobile IP, all packets for the MN have to go through the router anyway.

Entities and terminology

- ● The HA can be implemented on a router that is responsible for the home network.
- This is obviously the best position, because without optimizations to mobile IP, all packets for the MN have to go through the router anyway.
- ● If changing the router’s software is not possible, the HA could also be implemented on an arbitrary node in the subnet.
- One disadvantage of this solution is the double crossing of the router by the packet if the MN is in a foreign network.
- A packet for the MN comes in via the router; the HA sends it through the tunnel which again crosses the router.

Entities and terminology

- Finally, a home network is not necessary at all.
- The HA could be again on the ‘router’ but this time only acting as a manager for MNs belonging to a virtual home network. All MNs are always in a foreign network with this solution.
- The example network in Figure 8.1 shows the following situation:
- A CN is connected via a router to the internet, as are the home network and the foreign network.
- The HA is implemented on the router connecting the home network with the internet, an FA is implemented on the router to the foreign network.
- The MN is currently in the foreign network.
- The tunnel for packets toward the MN starts at the HA and ends at the FA, for the FA has the COA in this example.

IP packet delivery

- **8.1.3 IP packet delivery**
- Figure 8.2 illustrates packet delivery to and from the MN using the example network of Figure 8.1.
- A correspondent node CN wants to send an IP packet to the MN. One of the requirements of mobile IP was to support hiding the mobility of the MN. CN does not need to know anything about the MN's current location and sends the packet as usual to the IP address of MN (step 1).
- This means that CN sends an IP packet with MN as a destination address and CN as a source address.
- The internet, not having information on the current location of MN, routes the packet to the router responsible for the home network of MN.
- This is done using the standard routing mechanisms of the internet.

IP packet delivery

- The HA now intercepts the packet, knowing that MN is currently not in its home network.
- The packet is not forwarded into the subnet as usual, but encapsulated and tunnelled to the COA.
- A new header is put in front of the old IP header showing the COA as new destination and HA as source of the encapsulated packet (step 2). (Tunneling and encapsulation is described in more detail in section 8.1.6.)
- The foreign agent now decapsulates the packet, i.e., removes the additional header, and forwards the original packet with CN as source and MN as destination to the MN (step 3). Again, for the MN mobility is not visible.
- It receives the packet with the same sender and receiver address as it would have done in the home network.

IP packet delivery

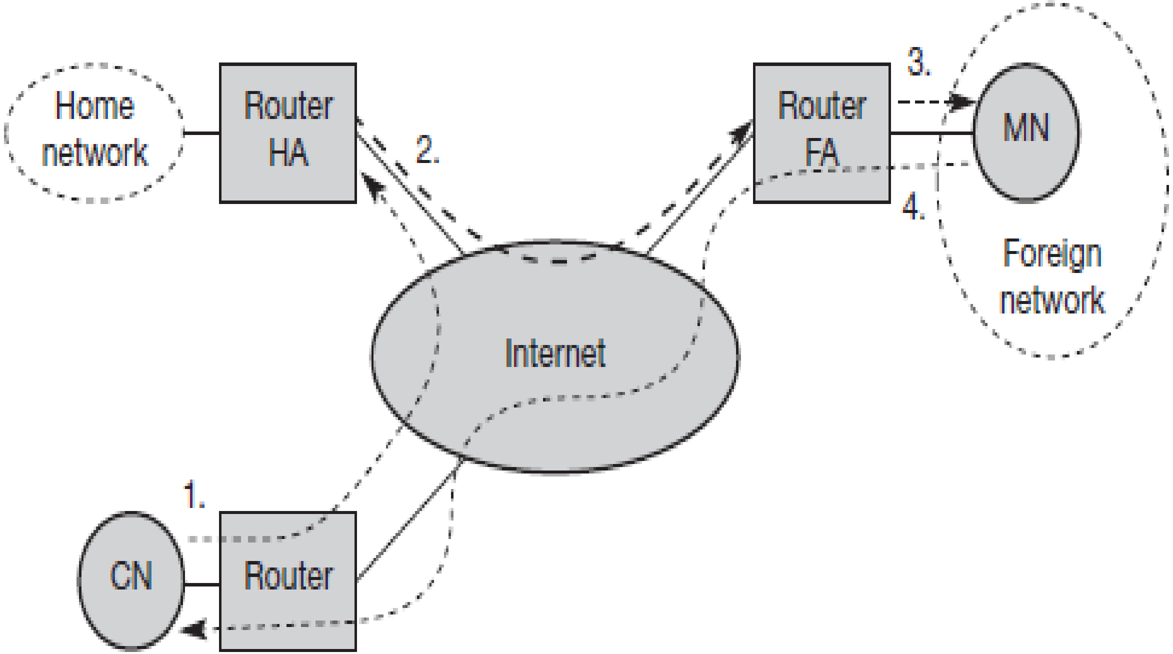


Figure 8.2
Packet delivery to and from the mobile node

IP packet delivery

- At first glance, sending packets from the MN to the CN is much simpler;
- problems are discussed in section 8.1.8.
- The MN sends the packet as usual with its own fixed IP address as source and CN's address as destination (step 4).
- The router with the FA acts as default router and forwards the packet in the same way as it would do for any other node in the foreign network.
- As long as CN is a fixed node the remainder is in the fixed internet as usual.
- If CN were also a mobile node residing in a foreign network, the same mechanisms as described
- in steps 1 through 3 would apply now in the other direction.
- The following sections present some additional mechanisms needed for mobile IP to work, some enhancements to the protocol, and some efficiency and security problems.

Agent discovery

- 8.1.4 Agent discovery
- One initial problem of an MN after moving is how to find a foreign agent.
- How does the MN discover that it has moved? For this purpose mobile IP describes two methods: agent advertisement and agent solicitation, which are in fact router discovery methods plus extensions.

Agent discovery

- 8.1.4.1 Agent advertisement
- For the first method, foreign agents and home agents advertise their presence
- periodically using special **agent advertisement messages.**
- **These advertisement** messages can be seen as a beacon broadcast into the subnet. For these advertisements Internet control message protocol (ICMP) messages according to RFC 1256 (Deering, 1991) are used with some mobility extensions.
- Routers in the fixed network implementing this standard also advertise their routing service periodically to the attached links.

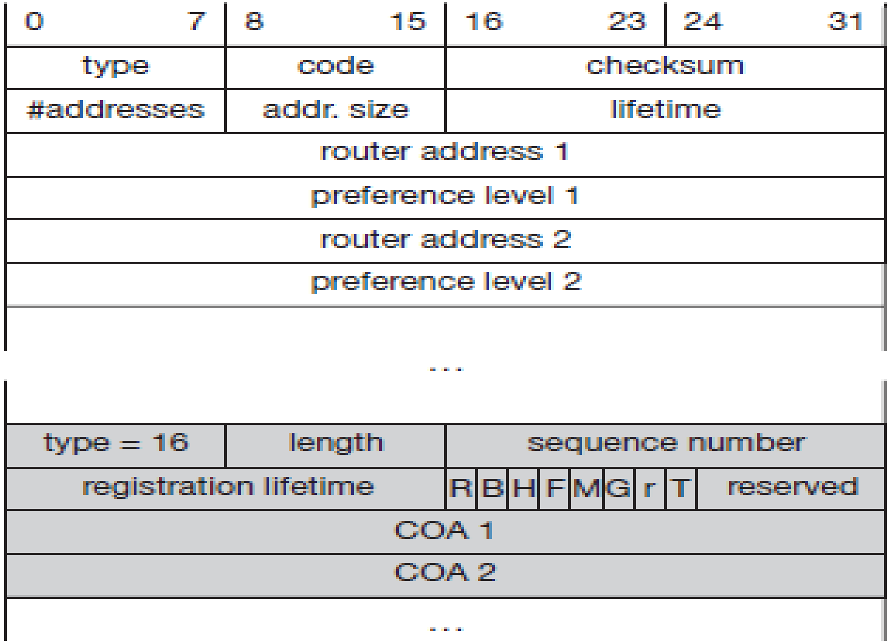
Agent discovery

- The agent advertisement packet according to RFC 1256 with the extension
- for mobility is shown in Figure 8.3.
- The upper part represents the ICMP packet while the lower part is the extension needed for mobility.
- The fields necessary on lower layers for the agent advertisement are not shown in this figure. Clearly, mobile nodes must be reached with the appropriate link layer address.
- The TTL field of the IP packet is set to 1 for all advertisements to avoid forwarding them.
- The IP destination address according to standard router advertisements can be either set to 224.0.0.1, which is the multicast address for all systems on a link (Deering, 1989), or to the broadcast address 255.255.255.255.

Agent discovery

- The fields in the ICMP part are defined as follows.
- The **type is set to 9, the code can be 0, if the agent also routes traffic from non-mobile nodes, or 16, if it** does not route anything other than mobile traffic. Foreign agents are at least required to forward packets from the mobile node.
- The number of addresses advertised with this packet is in **#addresses while the addresses themselves** follow as shown.
- **Lifetime denotes the length of time this advertisement is** valid.
- **Preference levels for each address help a node to choose the router that is** the most eager one to get a new node.

Agent discovery



0 7 | 8 15 | 16 23 | 24 31

type | code | checksum

#addresses | addr. size | lifetime

router address 1

preference level 1

router address 2

preference level 2

. . .

type = 16 | length | sequence number

registration lifetime | R | B | H | F | M | G | r | T | reserved

COA 1

COA 2

. . .

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

Agent discovery

- The difference compared with standard ICMP advertisements is what happens after the router addresses.
- This extension for mobility has the following fields defined: **type is set to 16, length depends on the number of COAs provided** with the message and equals 6 + 4*(number of addresses).
- An agent shows the total number of advertisements sent since initialization in the **sequence number.**
- **By the registration lifetime the agent can specify the maximum lifetime** in seconds a node can request during registration as explained in section 8.1.5.

Agent discovery

- The following bits specify the characteristics of an agent in detail.
- The **R** bit (registration) shows, if a registration with this agent is required even when
- using a colocated COA at the MN.
- If the agent is currently too busy to accept new registrations it can set the **B bit.**
- **The following two bits denote if the agent** offers services as
- a home agent **(H) or foreign agent (F) on the link where the** advertisement has been sent.
- Bits M and G specify the method of encapsulation used for the tunnel as explained in section 8.1.6.

Agent discovery

- While IP-in-IP encapsulation is the mandatory standard,
- **M can specify minimal encapsulation and G generic** routing encapsulation.
- In the first version of mobile IP (RFC 2002) the **V bit** specified the use of header compression according to RFC 1144 (Jacobson, 1990).
- Now the field **r at the same bit position is set to zero and must be** ignored.
- The new field **T indicates that reverse tunneling (see section 8.1.8) is** supported by the FA.
- The following fields contain the **COAs advertised.**
- **A foreign** agent setting the F bit must advertise at least one COA.
- Further details and special extensions can be found in Perkins (1997) and RFC 3220.
- A mobile node in a subnet can now receive agent advertisements from either its home agent or
- a foreign agent.
- This is one way for the MN to discover its location.

8.1.4.2 Agent solicitation

- □ **8.1.4.2 Agent solicitation**
- □ If no agent advertisements are present or the inter-arrival time is too high, and an MN has not received a COA by other means, e.g., DHCP as discussed in section 8.2, 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.
- □ Typically, a mobile node can send out three solicitations, one per second, as soon as it enters a new network.
- □ It should be noted that in highly dynamic wireless networks with moving MNs and probably with applications requiring continuous packet streams even one second intervals between solicitation messages might be too long.
- □ Before an MN even gets a new address many packets will be lost without additional mechanisms.

8.1.4.2 Agent solicitation

- new address many packets will be lost without additional mechanisms.
- If a node does not receive an answer to its solicitations it must decrease the rate
- of solicitations exponentially to avoid flooding the network until it reaches a maximum interval between solicitations (typically one minute).
- Discovering a new agent can be done anytime, not just if the MN is not connected to one.
- Consider the case that an MN is looking for a better connection while still sending via the old path.
- This is the case while moving through several cells of different wireless networks.

8.1.4.2 Agent solicitation

- After these steps of advertisements or solicitations the MN can now receive a COA, either one for an FA or a co-located COA.
- The MN knows its location (home network or foreign network) and the capabilities of the agent (if needed).
- The next step for the MN is the registration with the HA if the MN is in a foreign network as described in the following.

Registration

- **8.1.5 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 forwarding of packets.
- Registration can be done in two different ways depending on the location of the COA.

Registration

- □ 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 current** COA.
- □ Additionally, the mobility binding contains the lifetime of the registration which is negotiated during the registration process.
- □ Registration expires automatically after the lifetime and is deleted; so, an MN should reregister before expiration.
- □ This mechanism is necessary to avoid mobility bindings which are no longer used.
- □ After setting up the mobility binding, the HA sends a reply message back to the FA which forwards it to the MN.

Registration

- 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.
- This, by the way, is also the registration procedure for MNs returning to their home network.
- Here they also register directly with the HA. However, if the MN received an agent advertisement from the FA it should register via this FA if the R bit is set in the advertisement

Registration

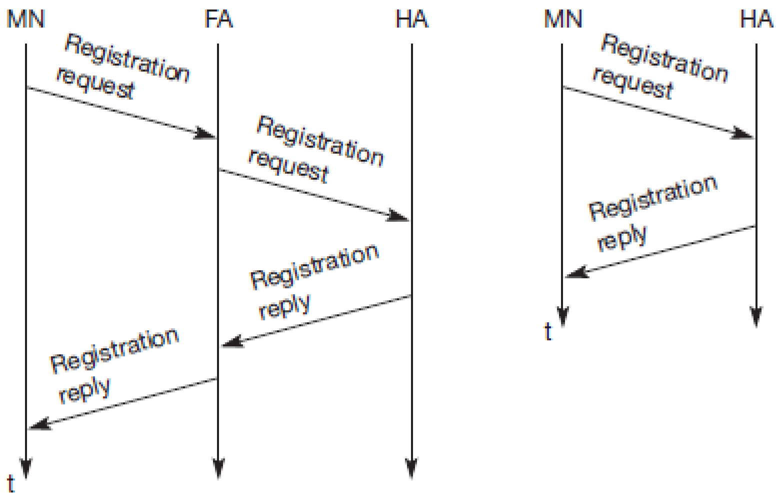
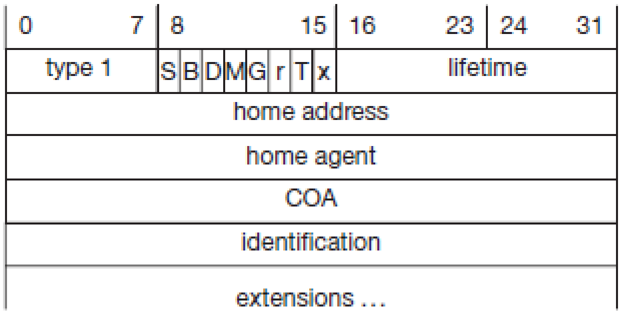


Figure 8.4 Registration of a mobile node via the FA or directly with the HA

Registration



0 7	8 15	16 23	24 31
type 1	S B D M G r T x	lifetime	
home address			
home agent			
COA			
identification			
extensions . . .			

Figure 8.5
Registration request

Registration

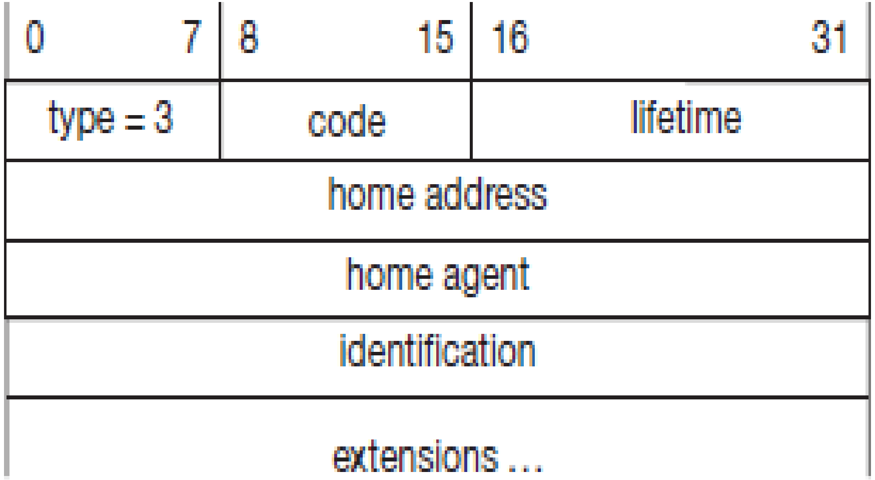
- UDP packets are used for **registration requests.**
- **The IP source address of the** packet is set to the interface address of the MN, the IP destination address is that of the FA or HA (depending on the location of the COA).
- The UDP destination port is set to 434.
- UDP is used because of low overheads and better performance compared to TCP in wireless environments (see chapter 9).
- The fields relevant for mobile IP registration requests follow as UDP data (see Figure 8.6).
- The fields are defined as follows.

Registration

- The first field **type is set to 1 for a registration request.**
- **With the S bit an MN** can specify if it wants the HA to retain prior mobility bindings.
- The following bits denote the requested behaviour for packet forwarding. Setting the **B bit generally indicates that an MN also wants to** receive the broadcast packets which have been received by the HA in the home network.
- If an MN uses a co-located COA, it also takes care of the decapsulation at the tunnel endpoint.
- The **D bit indicates** this behavior. As already defined for agent advertisements, the following bits **M and G denote the use of minimal encapsulation or generic routing encapsulation,**
- respectively.
- **T indicates reverse tunneling, r and x are set to zero.**

Registration

Figure 8.6
Registration reply



0 7	8 15	16 31
type = 3	code	lifetime
home address		
home agent		
identification		
extensions ...		

Registration

- **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.

Registration

- A **registration reply, which is conveyed in a UDP packet, contains a type** field set to 3 and a **code indicating the result of the registration request.**
- Table 8.1 gives some example codes.

Registration

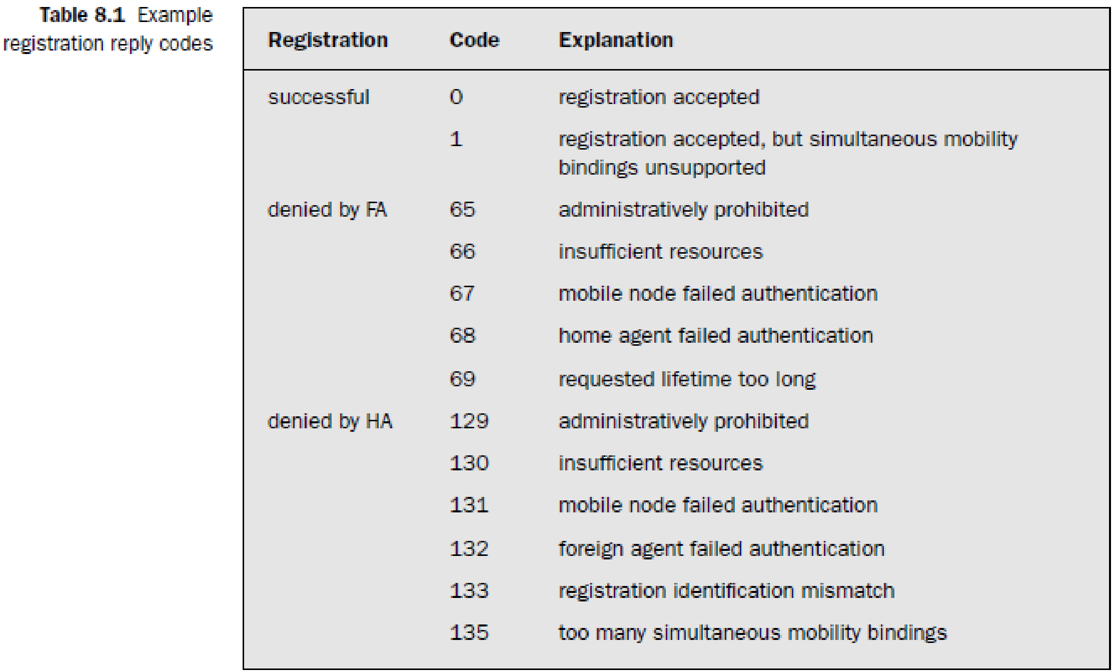


Table 8.1 Example registration reply codes

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

Registration

- The **lifetime field indicates how many seconds the registration is valid if it** was successful.
- **Home address and home agent are the addresses of the MN and** the HA, respectively.
- The 64-bit **identification is used to match registration** requests with replies.
- The value is based on the identification field from the registration and the authentication method.
- Again, the **extensions must at least** contain parameters for authentication.

Tunneling and encapsulation

- **8.1.6 Tunneling and encapsulation**
- The following describes the mechanisms used for forwarding packets between the HA and the COA, as shown in Figure 8.2, step 2.
- A **tunnel establishes a virtual** 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.

Tunneling and 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.**
- **Encapsulation and decapsulation are the operations typically** performed when a packet is transferred from a higher protocol layer to a lower layer or from a lower to a higher layer respectively.
- Here these functions are used within the same layer.

Tunneling and encapsulation

- This mechanism is shown in Figure 8.7 and describes exactly what the HA at the tunnel entry does.
- The HA takes the original packet with the MN as destination, puts it into the data part of a new packet and sets the new IP header in such a way that the packet is routed to the COA.
- The new header is also called the **outer header for obvious reasons. Additionally, there is an inner header** which can be identical to the original header as this is the case for IP-in-IP encapsulation, or the inner header can be computed during encapsulation.

Tunneling and encapsulation

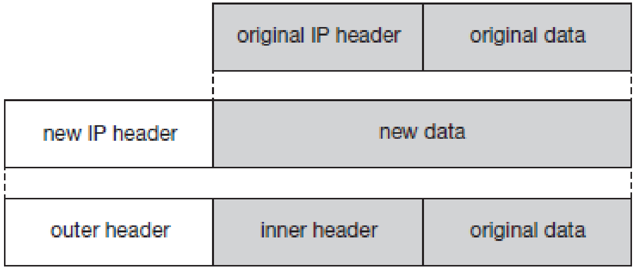
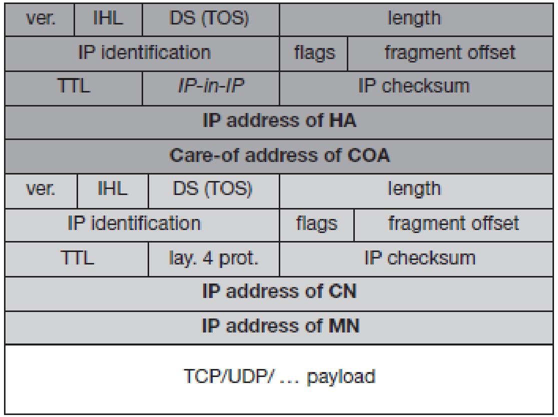


Figure 8.7
IP encapsulation

Figure 8.8
IP-in-IP encapsulation


ver. | IHL | DS (TOS) | length
IP identification | flags | fragment offset
TTL | IP-in-IP | IP checksum
IP address of HA
Care-of address of COA
ver. | IHL | DS (TOS) | length
IP identification | flags | fragment offset
TTL | lay. 4 prot. | IP checksum
IP address of CN
IP address of MN
TCP/UDP/ ... payload

Tunnelling and encapsulation

- DS field in the context of differentiated services (RFC 2474, Nichols, 1998).
- The fields of the outer header are set as follows.
- The version field **ver is 4 for IP version** 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.
- The fields up to TTL have no special meaning for mobile IP and are set according to RFC 791.

Tunnelling and encapsulation

- **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).**

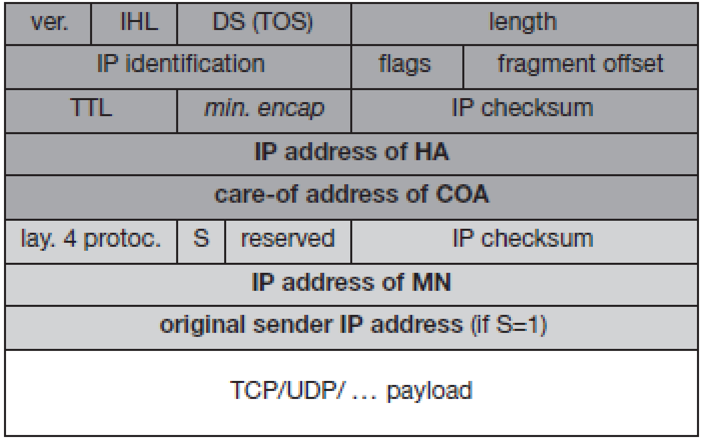
Tunneling and encapsulation

- If no options follow the outer header, the inner header starts with the same fields as just explained.
- This header remains almost unchanged during encapsulation, thus showing the original sender CN and the receiver MN of the packet.
- The only change is TTL which is decremented by 1.
- This means that the whole tunnel is considered a single hop from the original packet's point of view.
- This is a very important feature of tunneling as it allows the MN to behave as if it were attached to the home network.
- No matter how many real hops the packet has to take in the tunnel, it is just one (logical) hop away for the MN.
- Finally, the payload follows the two headers.

Tunneling and encapsulation

- **8.1.6.2 Minimal encapsulation**
- As seen with IP-in-IP encapsulation, several fields are redundant.
- For example, TOS is just copied, fragmentation is often not needed etc. Therefore, **minimal**
- **encapsulation (RFC 2004) as shown in Figure 8.9 is an optional encapsulation** method for mobile IP (Perkins, 1996c).
- The tunnel entry point and endpoint are specified.
- In this case, the field for the type of the following header contains the value 55 for the minimal encapsulation protocol.
- The inner header is different for minimal encapsulation.
- The type of the following protocol and the address of the MN are needed.
- If the **S bit is set, the original sender address of the CN is** included as omitting the source is quite often not an option.
- No field for fragmentation offset is left in the inner header and minimal encapsulation does not work with already fragmented packets.

Tunneling and encapsulation



<table>
<tr><td>ver.</td><td>IHL</td><td colspan="2">DS (TOS)</td><td colspan="2">length</td></tr>
<tr><td colspan="4">IP identification</td><td>flags</td><td>fragment offset</td></tr>
<tr><td colspan="2">TTL</td><td colspan="2"><i>min. encap</i></td><td colspan="2">IP checksum</td></tr>
<tr><td colspan="6"><b>IP address of HA</b></td></tr>
<tr><td colspan="6"><b>care-of address of COA</b></td></tr>
<tr><td colspan="2">lay. 4 protoc.</td><td>S</td><td>reserved</td><td colspan="2">IP checksum</td></tr>
<tr><td colspan="6"><b>IP address of MN</b></td></tr>
<tr><td colspan="6"><b>original sender IP address</b> (if S=1)</td></tr>
<tr><td colspan="6">TCP/UDP/ … payload</td></tr>
</table>

Figure 8.9
Minimal encapsulation

Tunneling and encapsulation

- **8.1.6.3 Generic routing encapsulation**
- 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 (Hanks, 1994). Figure 8.10 shows this procedure.
- The packet of one protocol suite with the original packet header and data is taken and a new GRE header is prepended.
- Together this forms the new data part of the new packet.
- Finally, the header of the second protocol suite is put in front.

Tunneling and encapsulation

- Figure 8.11 shows on the left side the fields of a packet inside the tunnel between home agent and COA using GRE as an encapsulation scheme according to RFC 1701.
- The outer header is the standard IP header with HA as source address and COA as destination address.
- The protocol type used in this outer IP header is 47 for GRE.
- The other fields of the outer packet, such as TTL and TOS, may be copied from the original IP header.
- However, the TTL must be decremented by 1 when the packet is decapsulated to prevent indefinite forwarding.

Tunneling and encapsulation

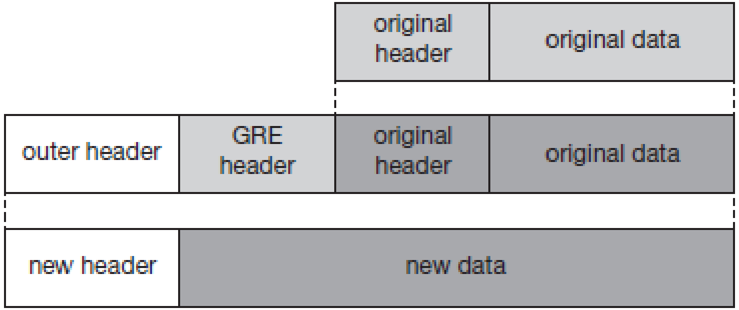
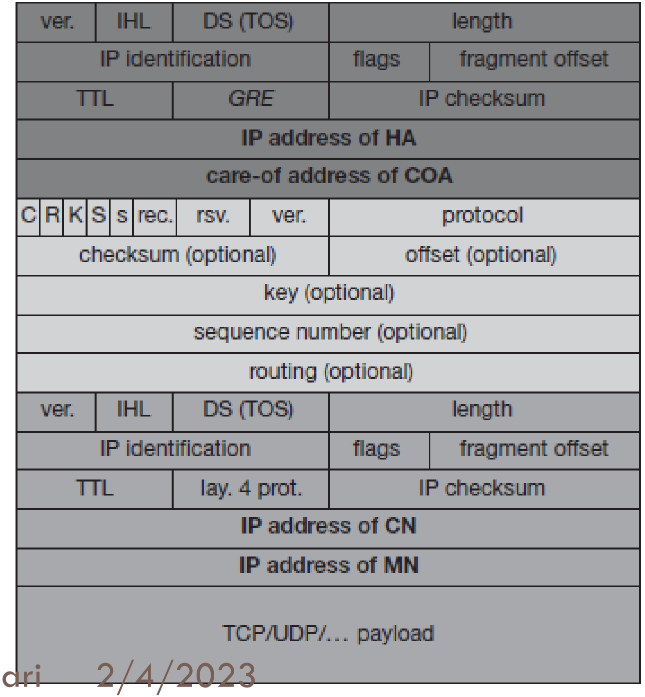


Figure 8.10
Generic routing encapsulation

Figure 8.11
Protocol fields for GRE according to RFC 1701



ver. | IHL | DS (TOS) | length
IP identification | flags | fragment offset
TTL | *GRE* | IP checksum
IP address of HA
care-of address of COA
C | R | K | S | s | rec. | rsv. | ver. | protocol
checksum (optional) | offset (optional)
key (optional)
sequence number (optional)
routing (optional)
ver. | IHL | DS (TOS) | length
IP identification | flags | fragment offset
TTL | lay. 4 prot. | IP checksum
IP address of CN
IP address of MN
TCP/UDP/… payload

Tunneling and encapsulation

- The GRE header starts with several flags indicating if certain fields are present or not.
- 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 payload.**
- The **R bit indicates if the offset and routing fields are present and contain** valid information.

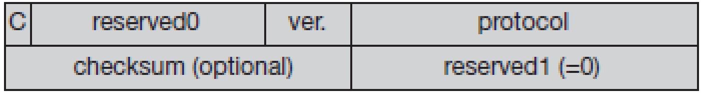
Tunneling and encapsulation

- □ 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.

Tunneling and encapsulation

- GRE also offers a **key field which may be used for authentication.**
- **If this** field is present, the **K bit is set.**
- **However, the authentication algorithms are not** further specified by GRE.
- The sequence number bit **S indicates if the sequence** number field is present, if the s bit is set, strict source routing is used.
- Sequence numbers may be used by a decapsulator to restore packet order.
- This can be important, if a protocol guaranteeing in-order transmission is encapsulated and transferred using a protocol which does not guarantee in-order delivery, e.g., IP.
- Now the decapsulator at the tunnel exit must restore the sequence to maintain the characteristic of the protocol.

Tunneling and encapsulation



<table>
<tr><td>C</td><td>reserved0</td><td>ver.</td><td>protocol</td></tr>
<tr><td colspan="3">checksum (optional)</td><td>reserved1 (=0)</td></tr>
</table>

Figure 8.12
Protocol fields for GRE according to RFC 2784

Tunneling and encapsulation

- □ The **recursion control field (rec.) is an important field that additionally distinguishes** GRE from IP-in-IP and minimal encapsulation.
- □ This field represents a counter that shows the number of allowed recursive encapsulations.
- □ As soon as a packet arrives at an encapsulator it checks whether this field equals zero.
- □ If the field is not zero, additional encapsulation is allowed – the packet is encapsulated and the field decremented by one. Otherwise the packet will most likely be discarded.
- □ This mechanism prevents indefinite recursive encapsulation which might happen with the other schemes if tunnels are set up improperly (e.g., several tunnels forming a loop).
- □ The default value of this field should be 0, thus allowing only one level of encapsulation.

Tunneling and encapsulation

- The following **reserved fields must be zero and are ignored on reception.**
- **The version field contains 0 for the GRE version.**
- **The following 2 byte protocol field** represents the protocol of the packet following the GRE header.
- Several values have been defined, e.g., 0 × 6558 for transparent Ethernet bridging using a GRE tunnel.
- In the case of a mobile IP tunnel, the protocol field contains 0 × 800 for IP.
- The standard header of the original packet follows with the source address
- of the correspondent node and the destination address of the mobile node.

Tunneling and encapsulation

- Figure 8.12 shows the simplified header of GRE following RFC 2784 (Farinacci, 2000), which is a more generalized version of GRE compared to RFC 1701.
- This version does not address mutual encapsulation and ignores several protocol-specific nuances on purpose.
- The field **C indicates again if a checksum** is present.
- The next 5 bits are set to zero, then 7 reserved bits follow.
- The **version** field contains the value zero.
- The **protocol type, again, defines the** protocol of the payload following RFC 3232 (Reynolds, 2002).
- If the flag C is set, then **checksum field and a field called reserved1 follows.**
- **The latter field is constant** zero set to zero follow. RFC 2784 deprecates several fields of RFC 1701, but can interoperate with RFC 1701-compliant implementations.

Optimizations

- **8.1.7 Optimizations**
- Imagine the following scenario. A Japanese and a German meet at a conference on Hawaii. Both want to use their laptops for exchanging data, both run mobile IP for mobility support. Now recall Figure 8.2 and think of the way the packets between both computers take.
- If the Japanese sends a packet to the German, his computer sends the data to the HA of the German, i.e., from Hawaii to Germany.
- The HA in Germany now encapsulates the packets and tunnels them to the COA of the German laptop on Hawaii.
- This means that although the computers might be only meters away, the packets have to travel around the world!
- This inefficient behaviour of a non-optimized mobile IP is called **triangular routing.**
- **The triangle is made of the** three segments, CN to HA, HA to COA/MN, and MN back to CN.

Optimizations

- With the basic mobile IP protocol all packets to the MN have to go through the HA.
- This can cause unnecessary overheads for the network between CN and HA, but also between HA and COA, depending on the current location of the MN.
- As the example shows, latency can increase dramatically.
- This is particularly unfortunate if the MNs and HAs are separated by, e.g., transatlantic links.
- One way to optimize the route is to inform the CN of the current location of the MN. The CN can learn the location by caching it in a **binding cache** which is a part of the local routing table for the CN.
- The appropriate entity to inform the CN of the location is the HA.
- The optimized mobile IP protocol needs four additional messages.

Optimizations

- **Binding request: Any node that wants to know the current location of an** MN can send a binding request to the HA.
- The HA can check if the MN has allowed dissemination of its current location.
- If the HA is allowed to reveal the location it sends back a binding update.
- **● Binding update: This message sent by the HA to CNs reveals the current** location of an MN.
- The message contains the fixed IP address of the MN and the COA. The binding update can request an acknowledgement.

Optimizations

- ● **Binding acknowledgement: If requested, a node returns this acknowledgement**
- after receiving a binding update message.
- ● **Binding warning: If a node decapsulates a packet for an MN, but it is not the** current FA for this MN, this node sends a binding warning.
- The warning contains MN's home address and a target node address, i.e., the address of the node that has tried to send the packet to this MN.
- The recipient of the warning then knows that the target node could benefit from obtaining a fresh binding for the MN.
- The recipient can be the HA, so the HA should now send a binding update to the node that obviously has a wrong COA for the MN.

Optimizations

- Figure 8.13 explains these additional four messages together with the case of an MN changing its FA.
- The CN can request the current location from the HA. If allowed by the MN, the HA returns the COA of the MN via an update message.
- The CN acknowledges this update message and stores the mobility binding.
- Now the CN can send its data directly to the current foreign agent FA old. FA old forwards the packets to the MN.
- This scenario shows a COA located at an FA.
- Encapsulation of data for tunneling to the COA is now done by the CN, not the HA.

Optimizations

- The MN might now change its location and register with a new foreign agent, FAnew.
- This registration is also forwarded to the HA to update its location database.
- Furthermore, FAnew informs FAold about the new registration of MN.
- MN's registration message contains the address of FAold for this purpose.
- Passing this information is achieved via an update message, which is acknowledged by FAold.
- Registration replies are not shown in this scenario.

Optimizations

- Without the information provided by the new FA, the old FA would not get to know anything about the new location of MN.
- In this case, CN does not know anything about the new location, so it still tunnels its packets for MN to the old FA, FAold.
- This FA now notices packets with destination MN, but also knows that it is not the current FA of MN.
- FAold might now forward these packets to the new COA of MN which is FAnew in this example.
- This forwarding of packets is another optimization of the basic Mobile IP providing **smooth handovers.**
- **Without this** optimization, all packets in transit would be lost while the MN moves from one FA to another.
- With TCP as the higher layer protocol this would result in severe performance degradation (see chapter 9).

Optimizations

- To tell CN that it has a stale binding cache, FAold sends, in this example, a binding warning message to CN.
- CN then requests a binding update. (The warning could also be directly sent to the HA triggering an update).
- The HA sends an update to inform the CN about the new location, which is acknowledged.
- Now CN can send its packets directly to FAnew, again avoiding triangular routing. Unfortunately, this optimization of mobile IP to avoid triangular routing causes several security problems (e.g., tunnel hijacking) as discussed in Montenegro (1998).
- Not all users of mobile communication systems want to reveal their current 'location' (in the sense of an IP subnet) to a communication partner.

Optimizations

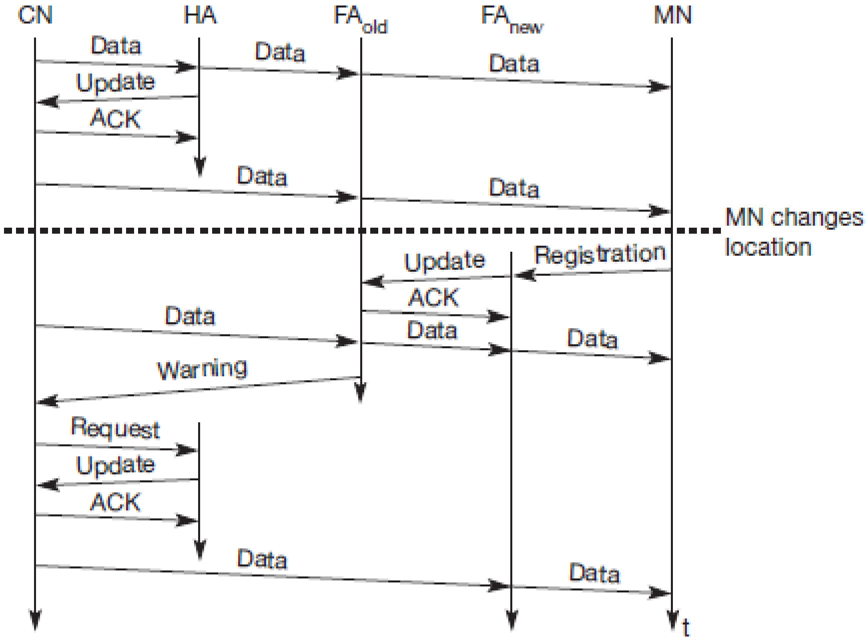


Figure 8.13
Change of the foreign agent with an optimized mobile IP

Optimizations

- **8.1.8 Reverse tunneling**
- At first glance, the return path from the MN to the CN shown in Figure 8.2 looks quite simple.
- The MN can directly send its packets to the CN as in any other standard IP situation.
- The destination address in the packets is that of CN.
- But there are several severe problems associated with this simple solution.

Optimizations

- **Firewalls:**
- **Almost all companies and many other institutions secure their** internal networks (intranet) connected to the internet with the help of a firewall.
- All data to and from the intranet must pass through the firewall.
- Besides many other functions, firewalls can be set up to filter out malicious addresses from an administrator’s point of view.
- Quite often firewalls only allow packets with topologically correct addresses to pass.
- This provides at least a first and simple protection against misconfigured systems of unknown addresses.
- However, MN still sends packets with its fixed IP address as source which is not topologically correct in a foreign network.
- Firewalls often filter packets coming from outside containing a source address from computers of the internal network.

Optimizations

- This avoids other computers that could use internal addresses and claim to be internal computers. However, this also implies that an MN cannot send a packet to a computer residing in its home network.
- Altogether, this means that not only does the destination address matter for forwarding IP packets, but also the source address due to security concerns.
- Further complications arise through the use of private addresses inside the intranet and the translation into global addresses when communicating with the internet.
- This **network address translation (NAT, network** address translator, RFC 3022, Srisuresh, 2001) is used by many companies to hide internal resources (routers, computers, printers etc.) and to use only some globally available addresses (Levkowetz, 2002, tries to solve the problems arising when using NAT together with mobile IP).

Optimizations

- **Multi-cast: Reverse tunnels are needed for the MN to participate in a multicast** group.
- While the nodes in the home network might participate in a multi-cast group, an MN in a foreign network cannot transmit multi-cast packets in a way that they emanate from its home network without a reverse tunnel.
- The foreign network might not even provide the technical infrastructure for multi-cast communication (multi-cast backbone, Mbone).

Optimizations

- □ **TTL: Consider an MN sending packets with a certain TTL while still in its** home network.
- □ The TTL might be low enough so that no packet is transmitted outside a certain region.
- □ If the MN now moves to a foreign network, this TTL might be too low for the packets to reach the same nodes as before.
- □ Mobile IP is no longer transparent if a user has to adjust the TTL while moving.
- □ A reverse tunnel is needed that represents only one hop, no matter how many hops are really needed from the foreign to the home network.

Traditional TCP

- **9.1 Traditional TCP**
- This section highlights several mechanisms of the transmission control protocol
- **9.1.1 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, fiber 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.

Traditional TCP

- Congestion may appear from time to time even in carefully designed networks.
- The packet buffers of a router are filled and the router cannot forward the packets fast enough because the sum of the input rates of packets destined for one output link is higher than the capacity of the output link.
- The only thing a router can do in this situation is to drop packets.
- A dropped packet is lost for the transmission, and the receiver notices a gap in the packet stream.
- Now the receiver does not directly tell the sender which packet is missing, but continues to acknowledge all in-sequence packets up to the missing one.

Traditional TCP

- The sender notices the missing acknowledgement for the lost packet and assumes a packet loss due to congestion. Retransmitting the missing packet and continuing at full sending rate would now be unwise, as this might only increase the congestion.
- Although it is not guaranteed that all packets of the TCP connection take the same way through the network, this assumption holds for most of the packets.
- To mitigate congestion, TCP slows down the transmission rate dramatically.
- All other TCP connections experiencing the same congestion do exactly the same so the congestion is soon resolved.
- This cooperation of TCP connections in the internet is one of the main reasons for its survival as it is today. Using UDP is not a solution, because the throughput is higher compared to a TCP connection just at the beginning.
- As soon as everyone uses UDP, this advantage disappears.
- After that, congestion is standard and data transmission quality is unpredictable.
- Even under heavy load, TCP guarantees at least sharing of the bandwidth.

Traditional TCP

- **9.1.2 Slow start**
- TCP’s reaction to a missing acknowledgement is quite drastic, but it is necessary to get rid of congestion quickly.
- The behavior TCP shows after the detection of congestion is called **slow start (Kurose, 2003).**
- 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.

Traditional TCP

- 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.**
- **As soon as the congestion window reaches the congestion threshold,** further increase of the transmission rate is only linear by adding 1 to the congestion window each time the acknowledgements come back.
- Linear increase continues until a time-out at the sender occurs due to a missing acknowledgement, or until the sender detects a gap in transmitted data because of continuous acknowledgements for the same packet.
- In either case the sender sets the congestion threshold to half of the current congestion window.
- The congestion window itself is set to one segment and the sender starts sending a single segment.
- The exponential growth (as described above) starts once more up to the new congestion threshold, then the window grows in linear fashion

Traditional TCP

- **9.1.3 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 (Kurose, 2003).**

Traditional TCP

- The receipt of acknowledgements shows that there is no congestion to justify a slow start.
- The sender can continue with the current congestion window.
- The sender performs a **fast recovery from the packet loss. This mechanism can** improve the efficiency of TCP dramatically.
- The other reason for activating slow start is a time-out due to a missing acknowledgement.
- TCP using fast retransmit/fast recovery interprets this congestion in the network and activates the slow start mechanism.

Traditional TCP

- **9.1.4 Implications on mobility**
- While slow start is one of the most useful mechanisms in fixed networks, it drastically decreases the efficiency of TCP if used together with mobile receivers or senders.
- The reason for this is the use of slow start under the wrong assumptions.
- From a missing acknowledgement, TCP concludes a congestion situation.
- While this may also happen in networks with mobile and wireless end-systems, it is not the main reason for packet loss.
- Error rates on wireless links are orders of magnitude higher compared to fixed fiber or copper links.
- Packet loss is much more common and cannot always be compensated for by layer 2 retransmissions (ARQ) or error correction (FEC).
- Trying to retransmit on layer 2 could, for example, trigger TCP retransmission if it takes too long.
- Layer 2 now faces the problem of transmitting the same packet twice over a bad link.
- Detecting these duplicates on layer 2 is not an option, because more and more connections use end-to-end encryption, making it impossible to look at the packet.

Traditional TCP

- 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.

Traditional TCP

- **9.2 Classical TCP improvements**
- Together with the introduction of WLANs in the mid-nineties several research projects were started with the goal to increase TCP's performance in wireless and mobile environments.

Overview of Traditional TCP

- **9.2.1 Indirect TCP**
- Two competing insights led to the development of indirect TCP (I-TCP) (Bakre, 1995).
- 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.

Overview of Traditional TCP

- □ 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. Even an unchanged TCP can benefit from the much shorter round trip time, starting retransmission much faster.
- □ A good place for segmenting the connection between mobile host and correspondent host is at the foreign agent of mobile IP (see chapter 8).
- □ 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 (e.g., IWF in GSM, GGSN in GPRS).

Overview of Traditional TCP

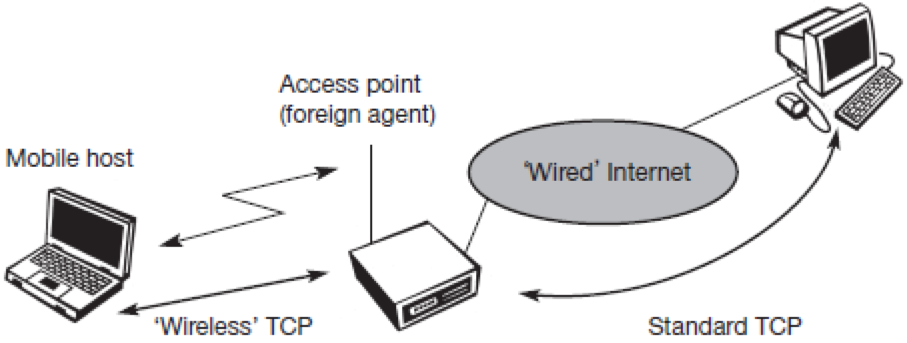


Figure 9.1
Indirect TCP segments a TCP connection into two parts

Overview of Traditional TCP

- □ 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.

Traditional TCP

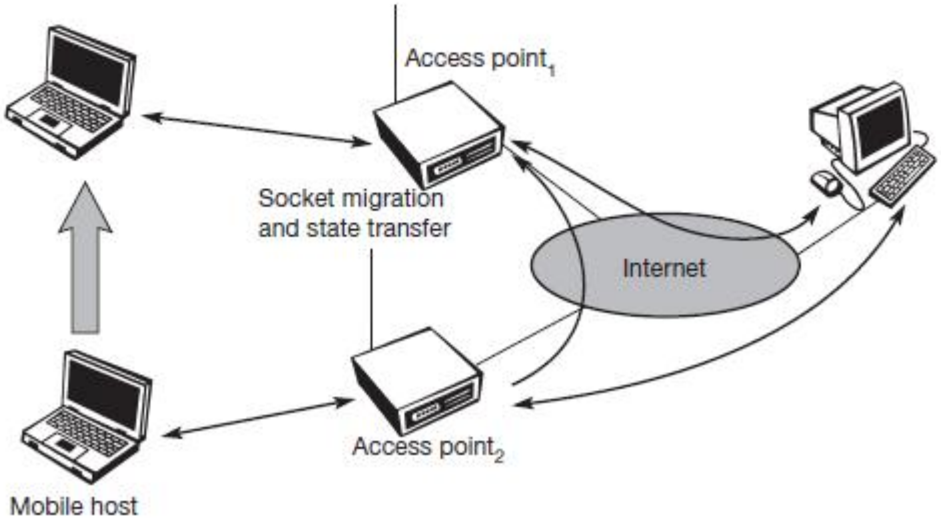
- Similarly, if the mobile host sends a packet, the foreign agent acknowledges this packet and tries to forward it to the correspondent host.
- If the packet is lost on the wireless link, the mobile hosts notice this much faster due to the lower round trip time and can directly retransmit the packet.
- Packet loss in the wired network is now handled by the foreign agent.

Traditional TCP

- 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.
- In the example shown, the access point acts as a proxy buffering packets for retransmission.
- After the handover, the old proxy must forward buffered data to the new proxy because it has already acknowledged the data.
- As explained in chapter 8, after registration with the new foreign agent, this new foreign agent can inform the old one about its location to enable packet forwarding.
- Besides buffer content, the sockets of the proxy, too, must migrate to the new foreign agent
- located in the access point.
- The socket reflects the current state of the TCP connection, i.e., sequence number, addresses, ports etc.
- No new connection may be established for the mobile host, and the correspondent host must not see any changes in connection state.

Overview of Traditional TCP

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



Overview of Traditional TCP

- □ 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.
- □ All current optimizations for TCP still work between the foreign agent and the correspondent host.
- □ ● Due to the strict partitioning into two connections, transmission errors on the wireless link, i.e., lost packets, cannot propagate into the fixed network.
- □ Without partitioning, retransmission of lost packets would take place between mobile host and correspondent host across the whole network.
- □ Now only packets in sequence, without gaps leave the foreign agent.

Overview of Traditional TCP

- It is always dangerous to introduce new mechanisms into a huge network such as the internet without knowing exactly how they will behave.
- However, new mechanisms are needed to improve TCP performance (e.g., disabling slow start under certain circumstances), but with I-TCP only between the mobile host and the foreign agent.
- Different solutions can be tested or used at the same time without jeopardizing the stability of the internet.
- Furthermore, optimizing of these new mechanisms is quite simple because they only cover one single hop.

Snoop TCP

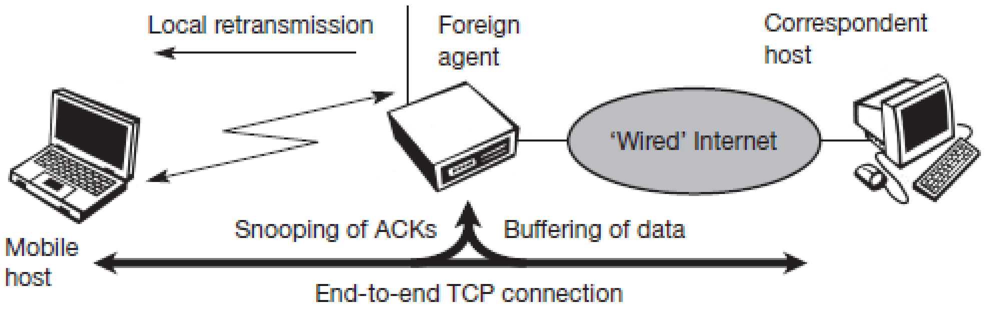
- **9.2.2 Snooping TCP**
- One of the drawbacks of I-TCP is the segmentation of the single TCP connection into two TCP connections.
- This loses the original end-to-end TCP semantic.
- The following TCP enhancement works completely transparently and leaves the TCP end-to-end connection intact.
- The main function of the enhancement is to buffer data close to the mobile host to perform fast local retransmission in case of packet loss.
- A good place for the enhancement of TCP could be the foreign agent in the Mobile IP context (see Figure 9.3).

Snoop 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 acknowledgements can be much shorter, because it reflects only the delay of one hop plus processing time.

Snoop TCP

Figure 9.3
Snooping TCP as a transparent TCP extension



Snoop TCP

- 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.

Snoop TCP

- 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.

Snoop TCP Adv.

- □ 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 (if this is the location of the buffering and snooping mechanisms), neither the correspondent host nor the mobile host have an inconsistent view of the TCP connection as is possible with I-TCP.
- □ The approach automatically falls back to standard TCP if the enhancements stop working.
- □ ● The correspondent host does not need to be changed; most of the enhancements are in the foreign agent.
- □ Supporting only the packet stream from the correspondent host to the mobile host does not even require changes in the mobile host.

Snoop TCP Adv

- □ 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.
- □ This is one of the problems of I-TCP, since the old foreign agent may have already signaled the correct receipt of data via acknowledgements to the correspondent host and now has to transfer these packets to the mobile host via the new foreign agent.

Snoop TCP Disadv.

- 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 ITCP.
- Assume, for example, that it takes some time until the foreign agent can successfully retransmit a packet from its buffer due to problems on the wireless link (congestion, interference).
- Although the time-out in the foreign agent may be much shorter than the one of the correspondent host, after a while the time-out in the correspondent host triggers a retransmission.
- The problems on the wireless link are now also visible for the correspondent host and not fully isolated.
- The quality of the isolation, which snooping TCP offers, strongly depends on the quality of the wireless link, time-out values, and further traffic characteristics.
- It is problematic that the wireless link exhibits very high delays compared to the wired link due to error correction on layer 2 (factor 10 and more higher).
- This is similar to ITCP.
- If this is the case, the timers in the foreign agent and the correspondent host are almost equal and the approach is almost ineffective.

Mobile TCP

- **9.2.3 Mobile TCP**
- Dropping packets due to a handover or higher bit error rates is not the only phenomenon of wireless links and mobility – the occurrence of lengthy and/or frequent disconnections is another problem. Quite often mobile users cannot connect at all.
- One example is islands of wireless LANs inside buildings but no coverage of the whole campus.
- What happens to standard TCP in the case of disconnection?
- A TCP sender tries to retransmit data controlled by a retransmission timer that doubles with each unsuccessful retransmission attempt, up to a maximum of one minute (the initial value depends on the round trip time).
- This means that the sender tries to retransmit an unacknowledged packet every minute and will give up after 12 retransmissions.
- What happens if connectivity is back earlier than this? No data is successfully transmitted for a period of one minute!
- The retransmission time-out is still valid and the sender has to wait.
- The sender also goes into slow-start because it assumes congestion.

Mobile TCP

- What happens in the case of I-TCP if the mobile is disconnected?
- The proxy has to buffer more and more data, so the longer the period of disconnection, the more buffer is needed. If a handover follows the disconnection, which is typical, even more state has to be transferred to the new proxy.
- The snooping approach also suffers from being disconnected.
- The mobile will not be able to send ACKs so, snooping cannot help in this situation.
- 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 (Brown, 1997).

Mobile TCP

- 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).
- 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.

Fast Retransmit/fast recovery

- □ **9.2.4 Fast retransmit/fast recovery**
- □ As described in section 9.1.4, moving to a new foreign agent can cause packet loss or time out at mobile hosts or corresponding hosts. TCP concludes congestion and goes into slow start, although there is no congestion.
- □ Section 9.1.3 showed the mechanisms of fast recovery/fast retransmit a host can use after receiving duplicate acknowledgements, thus concluding a packet loss without congestion.

Fast Retransmit/fast recovery

- The idea presented by Caceres (1995) is to artificially force the fast retransmit behavior 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.

Fast Retransmit/fast recovery

- 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 its simplicity.**
- **Only minor changes in the** mobile host's software already result in a performance increase.
- No foreign agent or correspondent host has to be changed.

Fast Retransmit/fast recovery

- 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.

Time-out freezing

- **9.2.5 Transmission/time-out 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 (however, many tunnels and subways provide connectivity via a mobile phone), or a user moving into a cell with no capacity left over.
- In this case, the mobile phone system will interrupt the connection.
- The reaction of TCP, even with the enhancements of above, would be a disconnection after a time out.

Time-out freezing

- 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.

Time-out freezing

- As soon as the MAC layer detects connectivity again, it signals TCP that it can resume operation at exactly the same point where it had been forced to stop.
- For TCP time simply does not advance, so no timers expire.
- The **advantage of this approach is that it offers a way to resume TCP connections** even 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.

Selective retransmission

- **9.2.6 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 (particularly those with a high path capacity, i.e., bandwidth delay- product).

Selective retransmission

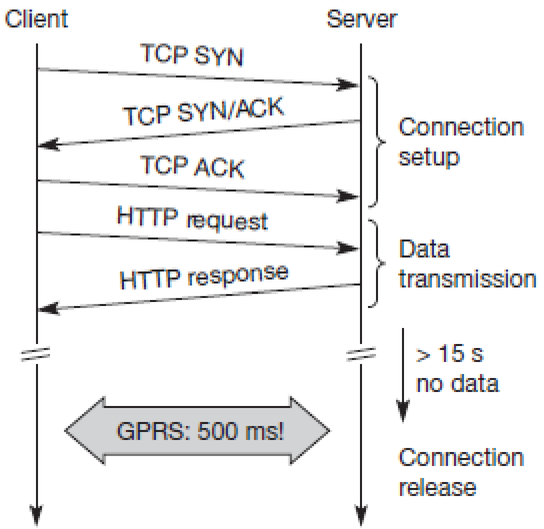
- Using RFC 2018 (Mathis, 1996), TCP can indirectly request a selective retransmission of packets.
- The receiver can acknowledge single packets, not only trains of in-sequence packets.
- The sender can now determine precisely which packet is needed and can retransmit it.
- The **advantage of this approach 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 buffer is necessary to resequence 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.

Transaction-Oriented TCP.

- **9.2.7 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 (many applications of this kind use UDP and solve reliability on a higher, application-oriented layer).
- 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.

Transaction-Oriented TCP.

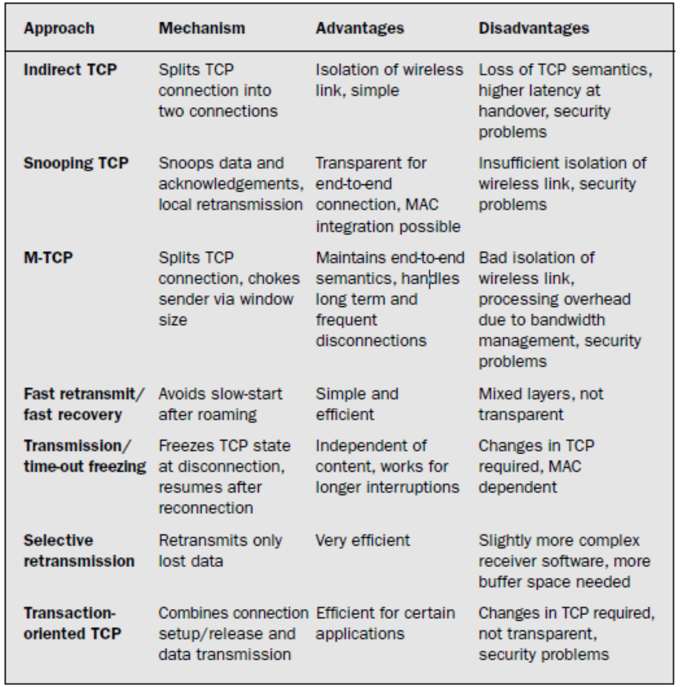
Figure 9.4
Example TCP connection setup overhead



Transaction-Oriented TCP.

- 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 (de Vivo, 1999)

Transaction-Oriented TCP.



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

Transaction-Oriented TCP.

- Table 9.1 shows an overview of the classical mechanisms presented together with some advantages and disadvantages.
- The approaches are not all exclusive, but can be combined.
- Selective retransmission, for example, can be used together with the others and can even be applied to fixed networks.
- An additional scheme that can be used to reduce TCP overhead is **header compression (Degermark, 1997).**
- **Using tunneling schemes as in mobile IP (see** section 8.1) together with TCP, results in protocol headers of 60 byte in case of IPv4 and 100 byte for IPv6 due to the larger addresses.
- Many fields in the IP and TCP header remain unchanged for every packet. Only just transmitting the differences is often sufficient. Especially delay sensitive applications like, e.g., interactive games, which have small packets benefit from small headers.
- However, header compression experiences difficulties when error rates are high due to the loss of the common context between sender and receiver.
- With the new possibilities of wireless wide area networks (WWAN) and their tremendous success, the focus of research has shifted more and more towards these 2.5G/3G networks.
- Up to now there are no final solutions to the problems arising when TCP is used in WWANs. However, some guidelines do exist.

Transaction-Oriented TCP.

- 9.3 TCP over 2.5/3G wireless networks
- The current internet draft for TCP over 2.5G/3G wireless networks (Inamura, 2002) describes a profile for optimizing TCP over today's and tomorrow's wireless WANs such as GSM/GPRS, UMTS, or cdma2000.
- 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 TCP stacks.
- 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 today's 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. Serious problems that may reduce throughput dramatically are bandwidth oscillations due to dynamic resource sharing.
- □ To support multiple users within a radio cell, a scheduler may have to repeatedly allocate and deallocate resources for each user. This may lead to a periodic allocation and release of a high-speed channel.
- □ ● **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.
- □ The current GPRS standard specifies an average delay of less than two seconds for the transport class with the highest quality (see chapter 4).

- **● 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.
- Handovers are quite often only virtually seamless with outages reaching from some 10 ms (handover in GSM systems) to several seconds (intersystem handover, e.g., from a WLAN to a cellular system using Mobile IP without using additional mechanisms such as multicasting data to multiple access points).
- **● Packet loss: Packets might be lost during handovers or due to corruption.**
- 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!).
- However, recovery at the link layer appears as jitter to the higher layers.

- □ Based on these characteristics, (Inamura, 2002) suggests the following configuration **parameters to adapt TCP to wireless environments:**
- □ **● 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 (RFC 1323) and larger buffer sizes this can be accomplished (typical buffer size settings of 16 kbyte are not enough).
- □ A larger initial window (more than the typical one segment) of 2 to 4 segments may increase performance particularly for short transmissions (a few segments in total).
- □ **● Limited transmit: This mechanism, defined in RFC 3042 (Allman, 2001) is** an extension of Fast Retransmission/Fast Recovery (Caceres, 1995) and is particularly useful when small amounts of data are to be transmitted (standard for, e.g., web service requests).
- □ **● 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. MTU path discovery according to RFC 1191 (IPv4) or RFC 1981 (IPv6) should be used to employ larger segment sizes instead of assuming the small default MTU.

- **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** (Ramakrishnan, 2001) 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.
- It is important to note that although these recommendations are still at the draft-stage, they are already used in i-mode running over FOMA as deployed in Japan and are part of the WAP 2.0 standard (aka TCP with wireless profile).

- 9.4 Performance enhancing proxies RFC 3135 ‘Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations’ lists many proxy architectures that can also be beneficial for wireless and mobile internet access (Border, 2001). Some initial proxy approaches,
- such as snooping TCP and indirect TCP have already been discussed.
- In principle, proxies can be placed on any layer in a communication system. However, the approaches discussed in RFC 3135 are located in the transport and application layer.
- One of the key features of a proxy is its transparency with respect to the end systems, the applications and the users.

- Transport layer proxies are typically used for local retransmissions, local
- acknowledgements, TCP acknowledgement filtering or acknowledgement
- handling in general.
- Application level proxies can be used for content filtering, content-aware compression, picture downscaling etc.
- Prominent examples are internet/WAP gateways making at least some of the standard web content accessible from WAP devices (see chapter 10). Figure 9.5 shows the general architecture of a wireless system connected via a proxy with the internet.

- However, all proxies share a common problem as they break the end-to-end
- semantics of a connection.
- According to RFC 3135, the most detrimental negative implication of breaking the end-to-end semantics is that it disables end-to-end use of IP security (RFC 2401).
- Using IP security with ESP (encapsulation security payload) the major part of the IP packet including the TCP header and application data is encrypted so is not accessible for a proxy.
- For any application one has to choose between using a performance enhancing proxy and using IP security.
- This is a killer criterion in any commercial environment as the only 'solution' would mean the integration of the proxy into the security association between the end systems.
- Typically this is not feasible as the proxy does not belong to the same organisation as the mobile node and the corresponding node.

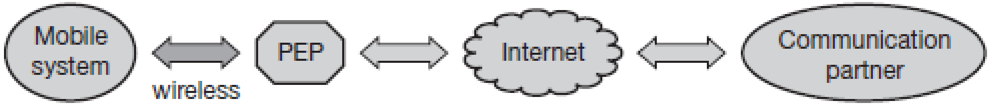


Figure 9.5
Performance enhancing proxy

Ref

- standard (RFC 2002, Perkins, 1996a) is Perkins (1997) and Solomon (1998).

Thank You