Mobile IP
- Defines the addressing schemes for all TCP/IP devices
- Each TCP/IP network interface requires a unique IP addr
- IPv4: 32 bits long \( (2^{32} = \text{over 4B}) \)

Internet → whole world’s backbone network
→ IPv4 addressing scheme is gradually running out of gas.
IPv4 → organizes the networked world into a simple two-level hierarchy
   - Network numbers
   - Host numbers

→ **Globally unique address**
Globally Unique Addresses

Assigns unique network number to requesting organizations

By IANA: Internet Assigned Numbers Authority

Assigns unique host numbers to its attached devices

By local authorities (cooperate network administrators)
IPv4 maintains a hierarchical structure

- Class A (0, 7bits) → 0.0.0.0 ~ 127.255.255.255
- Class B (10, 14bits) → 128.0.0.0 ~ 191.255.255.255
- Class C (110, 21bits) → 192.0.0.0 ~ 223.255.255.255
- Class D (1110) → 224.0.0.0 ~ 239.255.255.255
  · There is no relevance to network and host portions in multicast operations
- Class E (1111) → 240.0.0.0 ~ 255.255.255.255
  · reserved for future use
### IP datagram format

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP protocol version</td>
<td>Number of IP protocol version</td>
</tr>
<tr>
<td>header length</td>
<td>Number of bytes in the header</td>
</tr>
<tr>
<td>“type” of data</td>
<td>16-bit identifier</td>
</tr>
<tr>
<td>max number remaining</td>
<td>Number of hops remaining (decremented at each router)</td>
</tr>
<tr>
<td>remaining hops</td>
<td>8-bit field for fragmentation/reassembly</td>
</tr>
<tr>
<td>upper layer protocol</td>
<td>32-bit source IP address</td>
</tr>
<tr>
<td>to deliver payload</td>
<td>32-bit destination IP address</td>
</tr>
<tr>
<td>Options (if any)</td>
<td>E.g. timestamp, record route taken, specify list of routers to visit.</td>
</tr>
<tr>
<td>data</td>
<td>(variable length, typically a TCP or UDP segment)</td>
</tr>
</tbody>
</table>

**Header Format**

- **version (ver)**: 4 bits
- **header length (len)**: 4 bits
- **type of service (type)**: 8 bits
- **length**
- **16-bit identifier (id)**: 16 bits
- **flags (flgs)**
- **fragment offset (offset)**
- **time to live (ttl)**
- **upper layer (layer)**
- **header checksum (chksum)**
- **total datagram length (bytes)** for fragmentation/reassembly
IP Fragmentation & Reassembly

- network links have MTU (max.transfer size) - largest possible link-level frame.
  - different link types, different MTUs
- large IP datagram divided ("fragmented") within net
  - one datagram becomes several datagrams
  - "reassembled" only at final destination
  - IP header bits used to identify, order related fragments

fragmentation:
in: one large datagram
out: 3 smaller datagrams
IP Fragmentation and Reassembly

Example
- 4000 byte datagram
- MTU = 1500 bytes

One large datagram becomes several smaller datagrams

4000 byte datagram

MTU = 1500 bytes

1480 bytes in data field

offset = 1480/8
IP addresses: how to get one?

**Q:** How does a *host* get IP address?

- hard-coded by system admin in a file
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config

- **DHCP:** Dynamic Host Configuration Protocol: dynamically get address from a server
  - “plug-and-play”
ICMP: Internet Control Message Protocol

- used by hosts & routers to communicate network-level information
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- network-layer “above” IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

<table>
<thead>
<tr>
<th>Type</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>echo reply (ping)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>dest. network unreachable</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>dest host unreachable</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>dest protocol unreachable</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>dest port unreachable</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>dest network unknown</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>dest host unknown</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>source quench (congestion control - not used)</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>echo request (ping)</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>route advertisement</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>router discovery</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>TTL expired</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>bad IP header</td>
</tr>
</tbody>
</table>
Motivation for Mobile 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 topological correct address (standard IP) or needs special entries in the routing tables

Specific routes to end-systems?
- change of 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 to long time
- TCP connections break, security problems
Requirements to 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
The Goal of a Mobile IP

Supporting end-system mobility while maintaining scalability, efficiency, and compatibility in all respects with existing applications and Internet protocols
Terminology

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)
- communication partner
Example network

HA (home network)

Mobile end-system

Internet

FA (foreign network)

CN (end-system)

(physical home network for the MN)

(current physical network for the MN)
Data transfer to the mobile system

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
Data transfer from the mobile system

1. Sender sends to the IP address of the receiver as usual, FA works as default router
Overview
Network integration

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
- MN reads a COA from the FA advertisement messages
Agent advertisement

<table>
<thead>
<tr>
<th>type</th>
<th>code</th>
<th>checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>23</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24</td>
</tr>
<tr>
<td></td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>#addresses</td>
<td>addr. size</td>
<td>lifetime</td>
</tr>
<tr>
<td>router address 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>preference level 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>router address 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>preference level 2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

... 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
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
Registration

MN → FA
registration request
registration reply

FA → HA
registration request
registration reply

MN → HA
registration request
registration reply

t
Mobile IP registration request

<table>
<thead>
<tr>
<th>0</th>
<th>7</th>
<th>8</th>
<th>15</th>
<th>16</th>
<th>23</th>
<th>24</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>type = 1</td>
<td>S</td>
<td>B</td>
<td>D</td>
<td>M</td>
<td>G</td>
<td>r</td>
<td>T</td>
</tr>
<tr>
<td>home address</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>home agent</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>identification</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>extensions . .</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

S: simultaneous bindings
B: broadcast datagrams
D: decapsulation by MN
M minimal encapsulation
G: GRE encapsulation
r: =0, ignored
T: reverse tunneling requested
x: =0, ignored
Mobile IP registration reply

Example codes:

registration successful
- 0 registration accepted
- 1 registration accepted, but simultaneous mobility bindings unsupported

registration denied by FA
- 65 administratively prohibited
- 66 insufficient resources
- 67 mobile node failed authentication
- 68 home agent failed authentication
- 69 requested Lifetime too long

registration denied by HA
- 129 administratively prohibited
- 131 mobile node failed authentication
- 133 registration Identification mismatch
- 135 too many simultaneous mobility bindings
Encapsulation

original IP header | original data
new IP header | new data
outer header | inner header | original data
Encapsulation I

Encapsulation of one packet into another as payload
- e.g. IPv6 in IPv4 (6Bone), Multicast in Unicast (Mbone)
- here: e.g. IP-in-IP-encapsulation, minimal encapsulation or GRE (Generic Routing Encapsulation)

**IP-in-IP-encapsulation (mandatory, RFC 2003)**
- tunnel between HA and COA

<table>
<thead>
<tr>
<th>Field</th>
<th>ver.</th>
<th>IHL</th>
<th>DS (TOS)</th>
<th>length</th>
<th>IP identification</th>
<th>flags</th>
<th>fragment offset</th>
<th>TTL</th>
<th>IP-in-IP</th>
<th>length</th>
<th>ip checksum</th>
<th>IP address of HA</th>
<th>Care-of address COA</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>of MN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>of CN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>of HA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>of COA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TCP/UDP/ ... payload
Encapsulation II

Minimal encapsulation (optional)

- avoids repetition of identical fields
- e.g. TTL, IHL, version, DS (RFC 2474, old: TOS)
- only applicable for unfragmented packets, no space left for fragment identification

<table>
<thead>
<tr>
<th>ver.</th>
<th>IHL</th>
<th>DS (TOS)</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP identification</td>
<td>flags</td>
<td>fragment offset</td>
<td></td>
</tr>
<tr>
<td>TTL</td>
<td>min. encap.</td>
<td>IP checksum</td>
<td></td>
</tr>
<tr>
<td>IP address of HA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>care-of address COA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>lay. 4 protoc.</td>
<td>S</td>
<td>reserved</td>
<td>IP checksum</td>
</tr>
<tr>
<td>IP address of MN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>original sender IP address (if S=1)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TCP/UDP/ ... payload
# Generic Routing Encapsulation

**RFC 1701**

<table>
<thead>
<tr>
<th>ver.</th>
<th>IHL</th>
<th>DS (TOS)</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP identification</td>
<td>flags</td>
<td>fragment offset</td>
<td></td>
</tr>
<tr>
<td>TTL</td>
<td>GRE</td>
<td>IP checksum</td>
<td></td>
</tr>
</tbody>
</table>

**IP address of HA**

**Care-of address COA**

<table>
<thead>
<tr>
<th>CRK</th>
<th>S</th>
<th>rec.</th>
<th>rsv.</th>
<th>ver.</th>
<th>protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>checksum (optional)</td>
<td>offset (optional)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>key (optional)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sequence number (optional)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>routing (optional)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**RFC 2784**

<table>
<thead>
<tr>
<th>C</th>
<th>reserved0</th>
<th>ver.</th>
<th>protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>checksum (optional)</td>
<td>reserved1 (=0)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Mobile IP
Optimization of packet forwarding

**Triangular Routing**
- sender sends all packets via HA to MN
- higher latency and network load

“Solutions”
- sender learns the current location of MN
- direct tunneling to this location
- HA informs a sender about the location of MN
- big security problems!

**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
Change of foreign agent
Reverse tunneling (RFC 3024, was: 2344)

1. MN sends to FA
2. FA tunnels packets to HA by encapsulation
3. HA forwards the packet to the receiver (standard case)
Mobile IP with reverse tunneling

Router accept often only “topological correct“ addresses (firewall!)
- a packet from the MN encapsulated by the FA is now topological correct
- furthermore multicast and TTL problems solved (TTL in the home network correct, but MN is to far away from the receiver)

Reverse tunneling does not solve
- 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 (double triangular routing)

The standard is backwards compatible
- the extensions can be implemented easily and cooperate with current implementations without these extensions
- Agent Advertisements can carry requests for reverse tunneling
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

Security, firewalls, QoS etc. are topics of current research and discussions!