Introduction to routing in the Internet

Internet architecture
IPv4, ICMP, ARP
Addressing, routing principles

(Chapters 2–3 in Huitema)

Internet Architecture Principles
End-to-end principle by Dave Clark

- Hop-by-hop control vs. End-to-end control
  - In X.25
  - Error and flow control on each hop
  - In IP
  - Error and flow control in end stations

- The network can not be trusted
- The user must in any case check for errors
  → Network control is redundant
- Error checking and flow control by TCP in the end stations
- No state information in the network
  - The network is not aware of any connections
  - Packets routed independently
  - If a link fails, another route is used
- Same principle as in distributed systems
Internet Architecture Principles

IP over everything

by Vinton Cerf

- Internet connects different types of networks
  - Each with different framing, addressing, ...

Interconnection based on translation
- Mapping through a gateway
- Never perfect

Interconnection based on overlay
- Approach used by IP
- Single protocol over all underlying networks
- Simple to adapt to new technologies
  - Define framing or encapsulation
  - Define address resolution: IP-address \rightarrow network address
- Unique IP-address

Translation still needed in many cases
E.g. signaling interworking, IPv4 to IPv6 mapping

Internet Architecture Principles

IP over everything

HTTP, FTP, IMAP, SMTP, ...
TCP, UDP, ...
IP

IEEE-802, ATM, X.25, ...

S-38.2121 / Fall-06 / RKn, NB

Internet-4

Internet-5
Internet Architecture Principles

Connectivity is its own reward

- The value of a network increases in proportion to the square of the number of nodes on the network (Robert Metcalf's law)
- Be liberal with what you receive, conservative with what you send
  - try to make your best to understand what you receive
  - maximum adherence to standard when sending
- Snowballing effect keeps all interested in connectivity thus keeps adhering to standards

Routing is divided into interior and exterior

Autonomous System (AS) = networks operated by a single organization and having a common routing strategy

In this course we only deal with interior routing
Routing is divided into interior and exterior

- **Interior routing protocols**
  - Routing Information Protocol (RIP), RIP-2
  - Open Shortest Path First (OSPF)
  - Interior Gateway Routing Protocol (IGRP), EIGRP
  - Intermediate System-to-Intermediate System (IS-IS)

- **Exterior routing protocols**
  - External Gateway Protocol (EGP) *(historical)*
  - Border Gateway Protocol version 4 (BGP-4)

By connecting Ethernet segments with routers the traffic of the segments can be separated

A router
- operates on the network layer
- can interconnect networks of different technology

Host 2 does not receive packets sent by Host 1 to other hosts on Ethernet 1
Two functions of a router:
1. Packet forwarding

On which interface should this packet be forwarded?

Which is the following destination on that network?

Look in the routing table!

Two functions of a router:
2. Construction and maintenance of the routing table

- Routers exchange routing information with routing protocols (e.g. RIP, OSPF, BGP)
Internet routing is based on routing protocols, which collect information

- Routing is completely automatic
- No offline route planning
- Only dimensioning is made offline
- The routers communicate with a routing protocol
- The routing algorithm finds the shortest (cheapest) route to every destination

Routing in the Internet is generally dynamic, but static routing is used in some cases

- Dynamic routing is based on routing protocols which create and maintain the routing tables automatically
  - examples of routing protocols are RIP, OSPF, BGP...
  - E.g. to connect an organization with multiple links to the Internet
- Static routing is based on manually configured routing tables.
  - Static routing is used when e.g. two peer providers do not trust each other
  - To connect an organization to a service provider with a single connection
  - Static routing is difficult to maintain
The IP address defines the interface
(not the host)

Every interface also has a media specific
MAC address
Internet layer model – hosts and routers

Layers and message forwarding
IPv4 address formats

- Originally a two-level (network, host) hierarchy

32 bits

<table>
<thead>
<tr>
<th>MSB</th>
<th>Network</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>7 bits</td>
<td>24 bits</td>
</tr>
<tr>
<td>10</td>
<td>14 bits</td>
<td>16 bits</td>
</tr>
<tr>
<td>110</td>
<td>21 bits</td>
<td>8 bits</td>
</tr>
<tr>
<td>1110</td>
<td>28 bits - multicast address</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td>For experimental and future use</td>
<td></td>
</tr>
</tbody>
</table>

Class

- A new level for easier network administration

Example:

<table>
<thead>
<tr>
<th>Network</th>
<th>Subnet</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>00001010</td>
<td>00100010 1010011010 01110101</td>
<td></td>
</tr>
<tr>
<td>11111111</td>
<td>11111110 00000000 00000000</td>
<td></td>
</tr>
</tbody>
</table>

Network: first bit “0” = 10
Subnet: address* AND mask = 9 (36)
Host: address AND NOT mask = 2.154.117

address* = address with network part zeroed

Also written as 10.38.154.117/14
IPv4 address formats

• Examples:

<table>
<thead>
<tr>
<th>Mask</th>
<th>IP address</th>
<th>Network</th>
<th>Subnet</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xFFFF0000</td>
<td>10.27.32.100</td>
<td>A: 10</td>
<td>27</td>
<td>32.100</td>
</tr>
<tr>
<td>0xFFFFFE00</td>
<td>136.27.33.100</td>
<td>B: 136.27</td>
<td>16 (32)</td>
<td>1.100</td>
</tr>
<tr>
<td>0xFFFFFFC0</td>
<td>136.27.34.141</td>
<td>C: 193.27.32</td>
<td>17 (34)</td>
<td>0.141</td>
</tr>
<tr>
<td>0xFFFFFFFC0</td>
<td>193.27.32.197</td>
<td></td>
<td>3 (192)</td>
<td>5</td>
</tr>
</tbody>
</table>

High order bits:
- 0 .... 0 - 127. → A-class
- 10.... 128. - 191. → B-class
- 110...192. - 223. → C-class

Without right zeroes (and with right zeroes)

Later updated by CIDR
(discussed later)

Routers maintain routes to networks (not to hosts)

• Example

Network 10.29.0.0 / 16
- 10.29.0.1
- 10.29.0.2

Network 10.30.0.0 / 16
- 10.30.0.1
- 10.30.0.2

Network 10.31.0.0 / 16
- 10.31.0.30
- 10.31.0.35

Router

Host 1

Host 2

Host 3

Later updated by CIDR
(discussed later)
Aggregation describes several addresses in a single entry to reduce size of routing tables

- **Example**

```
  Router
   10.29.0.0 / 16
     10.29.0.1
     Network 10.31.0.0 / 16
       10.31.0.1
       10.31.0.30
       10.31.0.35
     10.30.0.2
   10.30.0.0 / 16
     10.30.0.1
     10.30.0.254
     Network 10.0.0.0 / 8
   Host 2
   Host 3
```

Special purpose addresses

- An unknown network is replaced by 0
  - Only used as source address (e.g. a booting host)
  - `0.0.0.0` = "this host in this network"
  - `0.X.Y.Z` = "host X.Y.Z in this network"
- Limited broadcast address `255.255.255.255`
  - To all hosts in the local network
- Directed broadcast addresses `A.255.255.255`, `B.B.255.255`, `C.C.C.255`
  - To all hosts in a specified network
- Loopback-address `127.X.X.X` (usually `127.0.0.1`)
  - Internal in one host
- Multicast-addresses (e.g. `224.0.0.2` = all routers on this subnet)
IPv4 packet header

We assume that the sender knows its own IP address. If not: self configuration protocols such as RARP, BOOTP, DHCP (dynamic host configuration protocol) are used
### IPv4 packet header

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>The fourth bit indicates the version of IPv4 (version 4)</td>
</tr>
<tr>
<td>IHL</td>
<td>The length of the header in 32-bit units</td>
</tr>
<tr>
<td>Type of service</td>
<td>The protocol field indicates the type of service being used, e.g., 6=TCP, 17=UDP, 1=ICMP, 89=OSPF</td>
</tr>
<tr>
<td>Total length</td>
<td>The length of the entire packet, including the header and data</td>
</tr>
<tr>
<td>Identification</td>
<td>Unique identifier assigned to each packet by the source</td>
</tr>
<tr>
<td>Flag</td>
<td>Flags indicating various packet characteristics, e.g., fragmenting</td>
</tr>
<tr>
<td>Fragment offset</td>
<td>The offset of the fragment relative to the original packet</td>
</tr>
<tr>
<td>Time-to-live (TTL)</td>
<td>The value is decremented with an integer representing the quality of the network on each router along the path of the packet. The packet is deleted when TTL reaches 0.</td>
</tr>
<tr>
<td>Protocol</td>
<td>The protocol field indicates the type of service being used, e.g., 6=TCP, 17=UDP, 1=ICMP, 89=OSPF</td>
</tr>
<tr>
<td>Header checksum</td>
<td>The header checksum is calculated as a 16-bit one's complement sum.</td>
</tr>
<tr>
<td>Source IP Address</td>
<td>The IP address of the sender of the packet</td>
</tr>
<tr>
<td>Destination IP Address</td>
<td>The IP address of the receiver of the packet</td>
</tr>
<tr>
<td>Optional</td>
<td>Used for special types of information or “tricks”. One packet can carry many option fields.</td>
</tr>
</tbody>
</table>

The header checksum is calculated as a 16-bit one’s complement sum.
The most important fields in routing are the destination address and the time-to-live

<table>
<thead>
<tr>
<th>Version</th>
<th>IHL</th>
<th>Type of service</th>
<th>Total length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Identification</td>
<td>Flag</td>
</tr>
<tr>
<td>Time-to-live (TTL)</td>
<td>Protocol</td>
<td>Header checksum</td>
<td></td>
</tr>
<tr>
<td>Source IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Options</td>
<td>Padding</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Every router decrements the TTL → must calculate new checksum
- Options (e.g. source routing, record route, timestamp)
  - rarely/never used in practice.

Type of service

<table>
<thead>
<tr>
<th>Precedence</th>
<th>D</th>
<th>T</th>
<th>R</th>
<th>C</th>
</tr>
</thead>
</table>

- Route selection criteria
  - D – minimization of delay
  - T – maximization of transmission capacity
  - R – maximization of reliability
  - C – minimization of cost
  - Only one can be selected.
- Precedence
  - Packet with the highest precedence is first taken from the queue to be routed.
- In practice, these are not used
- DiffServ uses the field in another way
Source routing

- Implemented with the “source routing” option
  - **Loose source routing** (type 131, \texttt{10000011})
    - The packet is sent to the next address in the list using normal routing.
  - **Strict source routing** (type 137, \texttt{10001001})
    - The packet is sent to the next address in the list. If there is no direct link to the address, the packet is destroyed.
- Slow $\rightarrow$ Rarely used
  - Can be replaced by encapsulation:

<table>
<thead>
<tr>
<th>Type</th>
<th>Length</th>
<th>Pointer</th>
<th>Address 1</th>
<th>Address 2</th>
<th>...</th>
<th>Address N</th>
</tr>
</thead>
</table>

ICMP – Internet Control Message Protocol

- Gives feedback about the network operation.
- ICMP packet is sent backwards if e.g.
  - The destination is unreachable
  - The router destroys a packet
  - TTL expires
- All hosts and routers must support ICMP.
- ICMP messages are transported in IP packets
- If a ICMP message is dropped, a new one is not generated
  - to avoid the “snowballing effect”.

| A$\rightarrow$C, IP-IP | A$\rightarrow$B, TCP | TCP | Data |
ICMP messages

<table>
<thead>
<tr>
<th>Type</th>
<th>Code</th>
<th>Header checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - Echo reply (used for “ping”)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 - Destination unreachable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 - Source quench (=“slow down”) (dropped from recommendations)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 - Redirect</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 - Echo (used for “ping”)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9 - Router advertisement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 - Router solicitation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 - Time exceeded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 - Parameter problem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 - Timestamp</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 - Timestamp reply</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 - Information request</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 - Information reply</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Packet sending – how to determine the next hop

- The sender checks if the destination address is in the same sub-network by comparing the masked values of the source and destination address.
  - If same, the destination is in the same subnet (next hop=destination).
  - Otherwise, the packet must be sent to a router (next hop=router).
- It then obtains the media address (MAC-address) of the next hop using the ARP-protocol.
  - The destination recognizes its own address

- The media address is stored in the cache.
  - Note: All hosts in the same subnet stores the address in their cache.
ARP – Address Resolution Protocol

- ARP maps IP to the underlying protocol
- IP-address \(\rightarrow\) MAC-address
- Each network technology requires its own ARP adaptation.
  - Easy if the network supports broadcast or multicast.
    - E.g. Ethernet, Token Ring, FDDI
  - ATM requires a special ARP-server
  - Manually defined address for point-to-point links
    - E.g. X.25, ISDN, Frame-Relay
- Works on top of Ethernet (not on top of IP)

Router discovery

- How to know the address of the router?
  - Configure manually – “default gateway”
  - Obtain with DHCP
    - Configured by administrator, still needs manual work
  - Listen to routing protocols
    - Uses resources of the host, too many routing protocols \(\rightarrow\) not used today
  - Automatic router discovery with ICMP
ICMP router discovery (1)

- The routers send *router advertisements* to all hosts periodically (e.g. in 7 minute intervals)

The advertisement contains
- a list of the router's addresses.
- the preference of the addresses, which are used to identify the normal, reserve, etc. router or router address (the preference of the default router is highest)
- lifetime of the information (e.g. 30 min)

ICMP router discovery (2)

- The host would have to wait up to 7 minutes before it can send packets outside its sub-network.
- Using a *router solicitation*, the host gets the advertisement immediately
ICMP router discovery (3)

- The host chooses the router with the highest priority as its default router.
- All packets for destinations outside the sub-network are then sent to the default router.
- Any advertisement from a router outside the sub-network is discarded.

A network may have many routers, the closest to the destination must be found

- A packet sent through the default router reaches the destination, but may waste resources.
A network may have many routers, the closest to the destination must be found

- The router can send a redirect to indicate a shorter route to the destination

```
<table>
<thead>
<tr>
<th>Type</th>
<th>Code</th>
<th>Header checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>IP address → router=Y</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IP header + 8 octets of the original datagram</td>
</tr>
</tbody>
</table>
```

Host must have feedback from the first router to avoid sending to a “black hole”

Feedback may be
- TCP acknowledgements
- Router advertisements
- ARP-replies
- ICMP echo reply (ping)

Between routers, routing protocols provide similar feedback and help in detecting failed router neighbors.
DNS – Domain Name Service

• Host name → IP address
• Why DNS?
  – Easier to remember names than addresses
  – Allows address changes without changing the name
  – Several addresses per host
  – Extensions: service location, ENUM
• DNS does not affect routing, routers only deal with IP addresses

Routing algorithms
Routing algorithms
Proactive vs. reactive

• **Proactive**
  – The router creates and maintains routes to all destinations
    → The routes are available in advance
  – The routing algorithms in the Internet are proactive

• **Reactive**
  – Routes are created only when they are needed
  – Used in e.g. ad hoc networks (discussed later in this course)

Routing algorithms
Distance vector vs. link state

• **Distance vector**
  – Distance vectors are sent, until the state of the network is stable
  – The routers cooperate to generate the routes
  – Example: RIP

• **Link state**
  – Topology descriptions are sent periodically and nodes generate a map over the network
  – Every router generates the routes independently of the other routers
  – Example: OSPF
**Routing algorithms**

**Distance vector vs. link state**

<table>
<thead>
<tr>
<th>Distance vector</th>
<th>Link state</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Simple and lightweight</td>
<td>+ Complex and heavy</td>
</tr>
<tr>
<td>▼ Slow convergence</td>
<td>+ Fast convergence</td>
</tr>
<tr>
<td>▼ Only one route per destination</td>
<td>+ Several routes per destination</td>
</tr>
<tr>
<td>▼ Only one metric</td>
<td>+ Supports different metrics</td>
</tr>
</tbody>
</table>