PART VI

INTERNET PROTOCOL:
CONNECTIONLESS DATAGRAM DELIVERY
Internet Protocol

- One of two major protocols in TCP/IP suite
- Major goals
  - Hide heterogeneity
  - Provide the illusion of a single large network
  - Virtualize access
IP allows a user to think of an internet as a single virtual network that interconnects all hosts, and through which communication is possible; its underlying architecture is both hidden and irrelevant.
Internet Services
And Architecture
Of Protocol Software

- Design has proved especially robust
IP Characteristics

- Provides connectionless packet delivery service
- Defines three important items
  - Internet addressing scheme
  - Format of packets for the (virtual) Internet
  - Packet forwarding
Internet Packet

- Analogous to physical network packet
- Known as *IP datagram*
## IP Datagram Layout

<table>
<thead>
<tr>
<th>DATAGRAM HEADER</th>
<th>DATAGRAM DATA AREA</th>
</tr>
</thead>
</table>

- Header contains
  - Source Internet address
  - Destination Internet address
  - Datagram type field
- Payload contains data being carried
## Datagram Header Format

| Field             | Octet 0 | Octet 1 | Octet 2 | Octet 3 | Octet 4 | Octet 5 | Octet 6 | Octet 7 | Octet 8 | Octet 9 | Octet 10 | Octet 11 | Octet 12 | Octet 13 | Octet 14 | Octet 15 | Octet 16 | Octet 17 | Octet 18 | Octet 19 | Octet 20 | Octet 21 | Octet 22 | Octet 23 | Octet 24 | Octet 25 | Octet 26 | Octet 27 | Octet 28 | Octet 29 | Octet 30 | Octet 31 |
|-------------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Version (VERS)    |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Header Length (HLEN) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Type of Service (TYPE OF SERVICE) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Total Length (TOTAL LENGTH) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Identification (IDENT) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Flags and Fragment Offset (FLAGS | FRAGMENT OFFSET) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Time to Live (TTL) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Protocol (TYPE)   |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Header Checksum (HEADER CHECKSUM) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Source IP Address |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Destination IP Address |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| IP Options (MAY BE OMITTED) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Padding |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |
| Beginning of Payload (DATA) |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |         |

The Datagram Header Format is used as the protocol-level header for the Internet Protocol (IP) version 4 (IPv4). It contains information about the IP address, protocol type, header length, and options among other things.
Addresses In The Header

- SOURCE is the address of original source
- DESTINATION is the address of ultimate destination
IP Versions

- Version field in header defines version of datagram
- Internet currently uses version 4 of IP, IPv4
- Preceding figure is the IPv4 datagram format
- IPv6 discussed later in the course
Datagram Encapsulation

- Datagram *encapsulated* in network frame
- Network hardware treats datagram as data
- Frame type field identifies contents as datagram
  - Set by sending computer
  - Tested by receiving computer
 Datagram Encapsulation For Ethernet

- Ethernet header contains Ethernet hardware addresses
- Ethernet type field set to 0x0800
Ethernet Frame Format

<table>
<thead>
<tr>
<th>Preamble</th>
<th>Destination Address</th>
<th>Source Address</th>
<th>Frame Type</th>
<th>Frame Data</th>
<th>CRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 octets</td>
<td>6 octets</td>
<td>6 octets</td>
<td>2 octets</td>
<td>46–1500 octets</td>
<td>4 octets</td>
</tr>
</tbody>
</table>

- Header format fixed (Destination, Source, Type fields)
- Frame data size can vary from packet to packet
  - Maximum 1500 octets
  - Minimum 46 octets
- Preamble and CRC removed by framer hardware before frame stored in computer’s memory
Datagram Encapsulated In Ethernet Frame

- 20-octet IP header follows Ethernet header
- IP source: 128.10.2.3 (800a0203)
- IP destination: 128.10.2.8 (800a0208)
- IP type: 01 (ICMP)
Standards For Encapsulation

- TCP/IP protocols define encapsulation for each possible type of network hardware
  - Ethernet
  - Frame Relay
  - Others
Encapsulation Over Serial Networks

- Serial hardware transfers stream of octets
  - Leased serial data line
  - Dialup telephone connection
- Encapsulation of IP on serial network
  - Implemented by software
  - Both ends must agree
- Most common standards: Point to Point Protocol (PPP)
Encapsulation For Avian Carriers (RFC 1149)

- Characteristics of avian carrier
  - Low throughput
  - High delay
  - Low altitude
  - Point-to-point communication
  - Intrinsic collision avoidance

- Encapsulation
  - Write in hexadecimal on scroll of paper
  - Attach to bird’s leg with duct tape

- For an implementation see

  http://www.blug.linux.no/rfc1149
A Potential Problem

- A datagram can contain up to 65535 total octets (including header)

- Network hardware limits maximum size of frame (e.g., Ethernet limited to 1500 octets)
  - Known as the network Maximum Transmission Unit (MTU)

- Question: how is encapsulation handled if datagram exceeds network MTU?
Possible Ways To Accommodate Networks With Differing MTUs

- Force datagram to be less than smallest possible MTU
  - Inefficient
  - Cannot know minimum MTU
- Hide the network MTU and accommodate arbitrary datagram size
Accommodating Large Datagrams

- Cannot send large datagram in single frame
- Solution
  - Divide datagram into pieces
  - Send each piece in a frame
  - Called *datagram fragmentation*
Illustration Of When Fragmentation Needed

- Hosts A and B send datagrams of up to 1500 octets
- Router $R_1$ fragments large datagrams from Host A before sending over Net 2
- Router $R_2$ fragments large datagrams from Host B before sending over Net 2
Datagram Fragmentation

- Performed by routers
- Divides datagram into several, smaller datagrams called fragments
- Fragment uses same header format as datagram
- Each fragment forwarded independently
Illustration Of Fragmentation

Original datagram

<table>
<thead>
<tr>
<th>Header</th>
<th>data_1</th>
<th>data_2</th>
<th>data_3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>600 bytes</td>
<td>600 bytes</td>
<td>200 bytes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Header_1</th>
<th>data_1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fragment #1 (offset of 0)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Header_2</th>
<th>data_2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fragment #2 (offset of 600)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Header_3</th>
<th>data_3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fragment #3 (offset of 1200)</td>
</tr>
</tbody>
</table>

- Offset specifies where data belongs in original datagram
- Offset actually stored as multiples of 8 octets
- MORE FRAGMENTS bit turned off in header of fragment #3
Fragmenting A Fragment

- Fragment can be further fragmented
- Occurs when fragment reaches an even-smaller MTU
- Discussion: which fields of the datagram header are used, and what is the algorithm?
Reassembly

- Ultimate destination puts fragments back together
  - Key concept!
  - Needed in a connectionless Internet
- Known as reassembly
- No need to reassemble subfragments first
- Timer used to ensure all fragments arrive
  - Timer started when first fragment arrives
  - If timer expires, entire datagram discarded
**Time To Live**

- TTL field of datagram header decremented at each hop (i.e., each router)
- If TTL reaches zero, datagram discarded
- Prevents datagrams from looping indefinitely (in case forwarding error introduces loop)
- IETF recommends initial value of 255 (max)
Checksum Field In Datagram Header

- 16-bit 1’s complement checksum
- Over IP header only!
- Recomputed at each hop
IP Options

- Seldom used
- Primarily for debugging
- Only *some* options copied into fragments
- Are variable length
- Note: padding needed because header length measured in 32-bit multiples
- Option starts with option code octet
Option Code Octet

<table>
<thead>
<tr>
<th>Option Class</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Datagram or network control</td>
</tr>
<tr>
<td>1</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>2</td>
<td>Debugging and measurement</td>
</tr>
<tr>
<td>3</td>
<td>Reserved for future use</td>
</tr>
</tbody>
</table>
**IP Semantics**

- IP uses best-effort delivery
  - Makes an attempt to deliver
  - Does not guarantee delivery

- In the Internet, routers become overrun or change routes, meaning that:
  - Datagrams can be lost
  - Datagrams can be duplicated
  - Datagrams can arrive out of order or scrambled

- Motivation: allow IP to operate over the widest possible variety of physical networks
Output From PING Program

PING venera.isi.edu (128.9.0.32): 64 data bytes at 1.0000 second intervals

72 bytes from 128.9.0.32: icmp_seq=0. time=170. ms
72 bytes from 128.9.0.32: icmp_seq=1. time=150. ms
72 bytes from 128.9.0.32: icmp_seq=1. time=160. ms
72 bytes from 128.9.0.32: icmp_seq=2. time=160. ms
72 bytes from 128.9.0.32: icmp_seq=3. time=160. ms

--- venera.isi.edu PING Statistics ---
4 packets transmitted, 5 packets received,
  25% packet loss
round-trip (ms) min/avg/max = 150/160/170

• Shows actual case of duplication
Summary

- Internet Protocol provides basic connectionless delivery service for the Internet
- IP defines *IP datagram* to be the format of packets on the Internet
- Datagram header
  - Has fixed fields
  - Specifies source, destination, and type
  - Allows options
- Datagram encapsulated in network frame for transmission
Summary
(continued)

- Fragmentation
  - Needed when datagram larger than MTU
  - Usually performed by routers
  - Divides datagram into fragments
- Reassembly
  - Performed by ultimate destination
  - If some fragment(s) do not arrive, datagram discarded
- To accommodate all possible network hardware, IP does not require reliability (best-effort semantics)