CS155b: E-Commerce

Lecture 4: Jan 18, 2001

How Does the Internet Work? (continued)

Acknowledgement: J. Rexford

and Lessons Learned From Netscape
Layering in the IP Protocols

- Internet Protocol
  - Transmission Control Protocol
    - HTTP (Web)
    - Telnet
  - User Datagram Protocol
    - Domain Name Service
    - Real-Time Protocol
  - SONET
  - Ethernet
  - ATM
IP Connectionless Paradigm

• No error detection or correction for packet data
  – Higher-level protocol can provide error checking

• Successive packets may not follow the same path
  – Not a problem as long as packets reach the destination

• Packets can be delivered out-of-order
  – Receiver can put packets back in order (if necessary)

• Packets may be lost or arbitrarily delayed
  – Sender can send the packets again (if desired)

• No network congestion control (beyond “drop”)
  – Send can slow down in response to loss or delay
### IP Packet Structure

<table>
<thead>
<tr>
<th>4-bit Version</th>
<th>4-bit Header Length</th>
<th>8-bit Type of Service (TOS)</th>
<th>16-bit Total Length (Bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16-bit Identification</td>
<td>3-bit Flags</td>
<td>13-bit Fragment Offset</td>
<td></td>
</tr>
<tr>
<td>8-bit Time to Live (TTL)</td>
<td>8-bit Protocol</td>
<td>16-bit Header Checksum</td>
<td></td>
</tr>
<tr>
<td>32-bit Source IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32-bit Destination IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Options (if any)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Payload</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Main IP Header Fields

- Version number (e.g., version 4, version 6)
- Header length (number of 4-byte words)
- Header checksum (error check on header)
- Source and destination IP addresses
- Upper-level protocol (e.g., TCP, UDP)
- Length in bytes (up to 65,535 bytes)
- IP options (security, routing, timestamping, etc.)
Time-to-Live Field

- Potential robustness problem
  - What happens if a packet gets stuck in a routing loop?
  - What happens if the packet arrives *much* later?
- Time-to-live field in packet header
  - TTL field decremented by each router on the path
  - Packet is discarded when TTL field reaches 0
  - Discard generates “timer expired” message to source
- Expiry message exploited in `traceroute` tool
  - Generate packets with TTL of $i=1, 2, 3, 4, \ldots$
  - Extract router id from the “timer expired” message
  - Provides a way to gauge the path to destination
Type-of-Service Bits

• Initially, envisioned for type-of-service routing
  – Low-delay, high-throughput, high-reliability, etc.
  – However, current IP routing protocols are static
  – And, most routers have first-in-first-out queuing
  – So, the ToS bits are ignored in most routers today

• Now, heated debate for differentiated services
  – ToS bits used to define a small number of classes
  – Affect router packet scheduling and buffering polices
  – Arguments about consistent meaning across networks
Transmission Control Protocol (TCP)

- Byte-stream socket abstraction for applications
- Retransmission of lost or corrupted packets
- Flow-control to respond to network congestion
- Simultaneous transmission in both directions
- Multiplexing of multiple logical connections
TCP Header

- 16-bit source port number
- 16-bit destination port number
- 32-bit sequence number
- 32-bit acknowledgement number
- 16-bit TCP checksum
- Options (if any)
- Payload
Establishing a TCP Connection

Three-way handshake to establish connection
- Host A sends a SYN (open) to the host B
- Host B returns a SYN acknowledgement (ACK)
- Host A sends an ACK to acknowledge the SYN ACK

Closing the connection
- Finish (FIN) to close and receive remaining bytes (and other host sends a FIN ACK to acknowledge)
- Reset (RST) to close and not receive remaining bytes
Lost and Corrupted Packets

- Detecting corrupted and lost packets
  - Error detection via checksum on header and data
  - Sender sends packet, sets timeout, and waits for ACK
  - Receiver sends ACKs for received packets
- Retransmission from sender
  - Sender retransmits lost/corrupted packets
  - Receiver reassembles and reorders packets
  - Receiver discards corrupted and duplicated packets

Packet loss rates are high (e.g., 10%), causing significant delay (especially for short Web transfers)!
TCP Flow Control

- Packet loss used to indicate network congestion
  - Router drop packets when buffers are (nearly) full
  - Affected TCP connection reacts by backing-off
- Window-based flow control
  - Sender limits number of outstanding bytes
  - Sender reduces window size when packets are lost
  - Initial slow-start phase to learn a good window size
- TCP flow-control header fields
  - Window size (maximum # of outstanding bytes)
  - Sequence number (byte offset from starting #)
  - Acknowledgement number (cumulative bytes)
User Datagram Protocol (UDP)

- Some applications do not want or need TCP
  - Don’t need recovery from lost or corrupted packets
  - Don’t want flow control to respond to loss/congestion
- Amount of UDP packets is rapidly increasing
  - Commonly used for multimedia applications
  - UDP traffic interferes with TCP performance
  - But, many firewalls do not accept UDP packets
- Dealing with the growth in UDP traffic
  - Pressure for applications to apply flow control
  - Future routers may enforce “TCP-like” behavior
  - Need better mathematical models of TCP behavior
Classless Inter-Domain Routing (CIDR)

- IP addresses are all 32 bits in length
  - “Dotted-decimal” notation: 113.34.96.78
  - IP address has “network” part and “host” part
- Addresses used to have a natural network length
  - Class A: 8-bit network and 24-bit host part
  - Class B: 16-bit network and 16-bit host part
  - Class C: 24-bit network and 8-bit host part
- Now any division of the 32 bits is fine
  - Arbitrary division into prefix and mask
  - E.g.: 113.34.96.0/24 for mask of 255.255.255.0
Getting an IP Packet From A to B

- Host must know at least three IP addresses
  - Host IP address (to use as its own source address)
  - Domain Name Service (to map names to addresses)
  - Default router to reach other hosts (e.g., gateway)

- Simple customer/company
  - Connected to a single service provider
  - Has just one router connecting to the provider
  - Has a set of IP addresses allocated in advance
  - Does not run an Internet routing protocol
Open Shortest-Path First (OSPF) Routing

- Network is a graph with routers and links
  - Each unidirectional link has a weight (1-63,535)
  - Shortest-path routes from sum of link weights
- Weights are assigned statically (configuration file)
  - Weights based on capacity, distance, and traffic
  - Flooding of info about weights and IP addresses
- Large networks can be divided in multiple domains
Example Network and Shortest Path

OSPF domain

link

router

6.8.9.0/24, 7.0.0.0/8

5.5.5.0/24

12.34.0.0/16

1.2.3.0/24, 4.5.0.0/16
IP Routing in OSPF

- Each router has a complete view of the topology
  - Each router transmits information about its links
  - Reliable flooding to all routers in the domain
  - Updates periodically or on link failure/installation
- Each router computes shortest path(s)
  - Maintenance of a complete link-state database
  - Execution of Dijkstra’s shortest-path algorithm
- Each router constructs a forwarding table
  - Forwarding table with next hop for each destination
  - Hop-by-hop routing independently by each router
Routing Software

• Routing protocol software
  – Checking connection with neighboring routers (“hello”)
  – Exchanging link-state information with other routers
  – Computing shortest paths and IP forwarding table
  – Handling of packets with IP options selected
  – Exchanging routing information between providers

• Router management and configuration
  – Configuration files to configure addresses, routing, etc.
  – Command-line interface to inspect/change configuration
  – Logging of statistics in management information base
  – More complex traffic measurement (e.g., NetFlow)
Connecting to Other Networks

Autonomous System: A collection of IP subnets and routers under the same administrative authority.

- Interior Routing Protocol (e.g., Open Shortest Path First)
- Exterior Routing Protocol (e.g., Border Gateway Protocol)
Connecting With Our Neighbors

- Public peering
  - Network Access Points (e.g., MAE East, MAE West)
  - Public location for connecting routers
  - Routers exchange data and routing information

- Private peering
  - Private connections between two peers (e.g., MCI)
  - Private peers exchange direct traffic (no transit)
  - Private peers must exchange similar traffic volumes

- Transit networks
  - Provider pays another for transit service (e.g., BBN)
  - Improve performance and reach more addresses
HTTP

- Standard protocol for web transfer

- Request-response interaction

- Request methods: GET, HEAD, PUT, POST, DELETE, …

- Response: Status line + additional info (e.g., a web page)
HTML

- The language in which web pages are written
- Contains formatting commands
- Tells browser what to display & how to display

\(<\text{HEAD} >\text{ Welcome to Yale } \text{ </HEAD}>\)
- The head of this page is “Welcome to Yale”

\(<B>\text{ Great News! } \text{ </B}>\)
- Set “Great News!” in boldface

\(<A \text{ HREF= ”http://www.cs.yale.edu/index.html”}>\text{ Yale Computer Science Department } \text{ </A}>\)
- with the text: “Yale Computer Science Department” displayed.
What does “http://www.cs.yale.edu/index.html” mean?

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Host domain name</th>
<th>Local file</th>
</tr>
</thead>
<tbody>
<tr>
<td>http</td>
<td><a href="http://www.cs.yale.edu">www.cs.yale.edu</a></td>
<td>index.html</td>
</tr>
</tbody>
</table>
• **Late 1990:** WWW, HTTP, HTML, “Browser” invented by Tim Berners-Lee

• **Mid-1994:** Mosaic Communications founded (later renamed to Netscape Communications)

• **Summer of 1995:** Market share 80%+

• **August 1995:** Windows 95 released with Internet Explorer

• **January 1998:** Netscape announced that its browser would thereafter be **free**; the development of the browser would move to an **open-source** process
Estimated Market Share of Netscape

NOTE: data are from different sources and not exact

Nov 1998:
AOL buys Netscape
Perfectly Captures the *Essence* of Internet Business

- Enormous power of Internet architecture and ethos (*e.g.*, layering, “stupid network,” open standards)
- **Must** bring new technology to market quickly to build market share
- Internet *is* the distribution channel
  - First via FTP, then via HTTP (using Netscape!)
  - Downloadable version available free and CD version sold
Uses Many “Internet Business Models”
(esp. those that involve making money by “giving away” an information product)

Complementary products (esp. server code)
• Bundling
  – Communicator includes browser, email tool, collaboration tool, calendar and scheduling tool, etc. One “learning curve,” integration, compatibility, etc.
• Usage monitoring
  – Datamining, strategic alliances
  – “Installed base” ≠ “Active installed base”
Browser as “Soul of the Internet”

- “New layer” (Note Internet architectural triumph!)
- Portal business
  - Early “electronic marketplace”
  - Necessity of strategic alliances
  - “Positive transfers” to customers
- (Temporarily?) Killed R&D efforts in user interfaces
Pluses and Minuses of Network Effects

+ Initial “Metcalf’s Law”- based boom
+ Initial boom accelerated by bundling, complementary products, etc.

- Market share ≠ lock in
  high market cap ≠ high switching costs

- Network effects strong for “browser” but weak for any particular browser
Exposed the True Nature of Microsoft

- 1995: Navigator released, MS rushes IE to market
- 1996: Version 3.0 of IE no longer technically inferior (“Openness” and standardization begets commoditization)
- MS exploits advantage with strategic allies (Windows!)
  - Contracts with ISPs to make IE the default
  - Incents OEMs not to load Netscape products
  - Exclusive access to premium content (from, e.g., Star Trek)
- 1998: MS halts browser-based version of these “strategies” under DoJ scrutiny of its contracts with ISPs.
Internet-ERA Anti-Trust Questions are Still Open

• Can consumers benefit from full integration of browser and OS?

• How to prevent “pre-emptive strikes” on potential competitors in the Windows-monopoly universe?
  – (“post-desktop era” technical Solution?)

• Remember: DoJ case is not about protecting Netscape!