Lecture 9: TCP/IP

CS 5516
Computer Architecture
Networks

VA Tech

Prof. Roy M. Wnek
Traffic and Overload Control in Telephone Networks

• *Dynamic* aspects of multiplexing information flows into a single line

• Concentration involves the sharing of a number of trunks by a set of users
  – Ensuring sufficient trunks to provide high availability (low probability of blocking)
  – Very efficient usage of network resources if the volume of traffic is sufficiently large

  ![Concentration Diagram]

• The number of trunks in use varies randomly over time but is typically much smaller than the total number of lines
  – Multiplexers concentrate the connection requests to a smaller number of trunks
  – Blocking occurs when no trunks are available
  – Typically a maximum acceptable probability of blocking is specified.
Traffic and Overload Control 2

• Utilization of a set of seven trunks over some time periods
  – Number of trunks in use, $N(t)$, as a function of at time $t$
  – Blocking state occurs when $N(t) = 7$

• The users require trunk connections in a sporadic and unscheduled manner
  – Statistical behavior of the users can be characterized as a Poisson process with connection request rate $\lambda$ calls/sec
• The time that a user maintains a connection (holding time), \( X \)
  – A random variable with an expected value, \( E[X] \)
  – \( E[X] \) is the amount of "work" that the system has to do for a typical user
  – Typical voice conversations have a mean holding time of several minutes

• The offered load is the rate at which user traffic is offered to the system

\[
a = \lambda \text{ calls/second} \times E[X] \text{ seconds/call (Erlang)}
\]

• One Erlang corresponds to an offered load that would occupy a single trunk 100% of the time
  – e.g. \( \lambda = 1 \text{ call/sec} \) and \( E[X] = 1 \text{ sec/call} \)

• Typically telephone systems are designed to provide a certain grade of service given the offered load during busy hours
  – Measurements reveal clear and relatively stable patterns of normal call activity
Traffic and Overload Control 4

• $P_b$, the blocking probability for a system with $c$ trunks and offered load $a$ is given by the Erlang B formula

$$P_b = \frac{a^c}{c!} \sum_{k=0}^{c} \frac{a^k}{k!}$$

Where $K!=1,2,3…(K-1),K$

• The blocking probability as a function of offered load and number of trunks
  – $P_b$ decreases with the number of trunks
  – $P_b$ of 1% is typical in trunk system design
  – 4 trunks are required to achieve this $P_b$ requirement when the offered load is 1 Erlang
  – 16 trunks are required for an offered load of 9 Erlangs
Trunk Utilization

- Systems become more efficient as the size of the offered load in the system increases
  - Efficiency is measured by Trunk Utilization
  - Average number of trunks in use / the total number of trunks

\[
\text{utilization} = \frac{\lambda(1 - P_b)E[X]}{c} = (1 - P_b)\frac{\lambda}{c}
\]

- Utilization is relatively low for small loads
  - Extra trunks are required to deal with surges in connection requests
  - A 2 Erlang load requires 7 trunks

- Utilization increases as the offered load of the systems increases
  - A 6 Erlang load only requires 13 trunks
  - High utilization is possible when the offered loads are large (50 and 100 Erlangs)

- Sharing of network resources becomes more efficient as the scale or size of the system is increased
  - “Multiplexing Gain” is the performance improvement resulting from aggregating traffic flow
Trunk Utilization 2

- Trunk utilization for various offered loads (in Erlangs) and $P_b=0.01$

<table>
<thead>
<tr>
<th>Load</th>
<th>trunks@1%</th>
<th>utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>0.20</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>0.29</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>0.38</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>0.40</td>
</tr>
<tr>
<td>5</td>
<td>11</td>
<td>0.45</td>
</tr>
<tr>
<td>6</td>
<td>13</td>
<td>0.46</td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td>0.50</td>
</tr>
<tr>
<td>8</td>
<td>15</td>
<td>0.53</td>
</tr>
<tr>
<td>9</td>
<td>17</td>
<td>0.53</td>
</tr>
<tr>
<td>10</td>
<td>18</td>
<td>0.56</td>
</tr>
<tr>
<td>30</td>
<td>42</td>
<td>0.71</td>
</tr>
<tr>
<td>50</td>
<td>64</td>
<td>0.78</td>
</tr>
<tr>
<td>60</td>
<td>75</td>
<td>0.80</td>
</tr>
<tr>
<td>90</td>
<td>106</td>
<td>0.85</td>
</tr>
<tr>
<td>100</td>
<td>117</td>
<td>0.85</td>
</tr>
</tbody>
</table>
Routing Control are the procedures for assigning paths in a network to connections
- Connections should follow the most direct route, to use the fewest network resources
- However, when traffic is low, the required resources to provide high availability will be used inefficiently
- Economic considerations lead to provide:
  - Direct trunks between switches that have large traffic flows between them
  - Indirect paths through tandem switches for smaller flows

A hierarchical approach to routing when the traffic between switches is small
- Entails aggregating traffic flows onto paths that are shared by multiple switches

10 Erlangs between each pair
90 Erlangs
Routing Control 2

- Each pair of switches would require 18 long distance trunks to handle the 10 Erlangs of traffic with \( P_b = 1\% \) blocking probability
  - \( 9 \times 18 = 162 \) trunks
- Concentrating the traffic flows through the tandem switches, the number of long distance trunks required to handle the combined 90 Erlangs of traffic is only 106
  - Requires the use of local trunks to the tandem switch
- Choice depends on the relative costs of local and long distance trunks

- Increasing trunk utilization efficiency by handling larger offered loads presents a sensitivity problem to traffic overloads
  - Higher efficiency means a smaller number of spare circuits are required to meet 1\% \( P_b \)
  - However, the smaller number of spare circuits makes the system more sensitive to traffic overload conditions
    - E.g. if each of the direct trunks is subjected to a 10\% overload (11 Erlangs) on the trunks, \( P_b \) increases to 2.45\%
    - A 10\% overload on the tandem trunks (99 Erlangs) increases \( P_b \) to 9.5\%
  - Trunk designs must provide a margin for some percentage of overload
• A typical approach for routing connections between two switches with significant traffic volume between them
  – Provide a number of direct trunks between the switches, selected to have a high usage and a $P_b$ higher than 1%, e.g. 10%
  – Provide number of trunks in an alternate path through a tandem switch, selected so that overall $P_b$ is 1%
    • $P_b = 10\%$ on the alternate path is sufficient to bring the overall blocking probability to 1%
  – Connection request first attempts to engage a direct trunk, if unavailable attempt an alternate route

• The Erlang formula cannot be applied directly in the calculation of $P_b$ on the alternate route
  – $\lambda_{tandem\ switch}$ is not a poison process, not independent, but dependent on blocking on the direct trunks
More realistically, tandem switches handle:

- Direct traffic between switches that have small volumes of traffic between them
  - Traffic between switches A and D must be provided with a $P_b = 1\%$
- Overflow traffic from some high-usage trunks
  - $P_b = 10\%$ is sufficient for the other switch pairs
- Direct traffic from A to D must receive some priority in accessing the trunks between the tandem switches

Typical Routing Scenario
• Traffic flows vary according to the time of day and the day of the week and year
  – Determining the state of network links and switches provides an opportunity to assign routes in more dynamic fashion
  – E.g., time/day differences between the east coast and the west coast allow the network resources at one coast to provide alternate routes for traffic on the other coast during certain times of the day

• Dynamic Non-Hierarchical Routing (DNHR) is a dynamic approach to routing
  – The first route attempt is a direct route
  – Tandem switches provide alternate routes
    • The order in which tandem switches are attempted is determined dynamically according to the state of the network

![Routing Control 5 Diagram]
**Overload Controls**

- Traffic and Routing Control deals with traffic flows during normal predictable network conditions
  - Network traffic increases or decreases with the traffic that is offered to it

- Overload Control deals with traffic flows during unexpected or unusual conditions
  - E.g. holidays, catastrophes, or equipment failures

- Overload Conditions result in traffic levels for which the network has not been provisioned
  - As offered traffic approaches network capacity, it is possible for the carried traffic to fall
  - Network resources become scarce, many call attempts seize only partial resources, and ultimately end up uncompleted
Overload Controls 2

- Proper handling of overload conditions avoids service degradation to all users
  - Ensure that a maximum number of calls are completed so carried load can approach the network capacity

- Network monitoring is required to identify overload conditions
  - Traffic loading measured at links and switches
  - Monitor the answer/bid ratio of call attempts to a given destination
  - The combination of measurements is useful in diagnosing fault conditions
    - Increased traffic load indicates some problems
    - Answer/bid ratio identifies switch failure
  - Alarms set by the monitoring system
    - Processed by network monitoring software that is used to diagnose network problems

- Once an overload condition is identified, several types of actions can be taken.
  - Allocating additional resources from back-up redundant capacity
    - Interconnecting SONET rings using digital cross-connects
    - Dynamic alternate routing between high traffic areas
Overload Controls 3

- Maximize the efficiency of available resources
  - E.g., in network-wide congestion, try to meet all call attempts using direct routes
  -Disallow Alternate routes because they require more resources to complete calls

- Overload controls dealing with extreme levels of traffic as resulting from a natural disaster
  - One approach is allowing only outbound traffic to seize the available trunks
  - Complementary control of distant switches
    - “Code Block” all calls to the affected area
    - Limit the rate of call request acceptance to the affected area

- Overload controls make extensive use of the signaling system
  - Another potential source of serious problem conditions
  - E.g. faulty signaling software can result in abnormal levels of signaling traffic which in turn can incapacitate the network
  - Clearly overload control mechanisms are also essential for the signaling system
Routing Protocols

- Routing functions
  - Address translation
  - Maintenance of a “routing” or “forwarding” table
  - Determine the “best” way to reach an address

- Routing algorithms exchange network topology and status data among the nodes
  - Convergence is when the network has gone from an “unstable state” to a “stable state”
    - when all nodes have a common understanding about the state of the network
    - Packets are not misrouted due to inaccurate routing tables

- Two types of routing algorithms
  - Distance Vector algorithms
    - Simple
    - Based exclusively on neighbor-to-neighbor data exchange
  - Link State algorithms
    - More complex
    - Faster convergence
    - Based on initial exchange of tables with neighbors and flood broadcasting of updates
    - More memory & CPU intensive at the router
Distance Vector Routing Protocols

- Internet Routing Information Protocol (RIP)
  - Example of a Distance Vector routing algorithm
  - Simple
  - Message size grows with the size of the network
  - Message size may be moderated depending upon the degree of summarization
  - Operates at the Application layer, in a connectionless mode
  - Interfaces with UDP at the Transport layer
  - Routing decision depend only on hop count

- Other Routing Protocols
  - Interior Gateway Routing Protocol
    - Proprietary to CISCO routers

- Type of Network routing capabilities
  - Point-to-Point
  - Broadcast
  - Non-broadcast Multi-Access
    - (e.g. X.25)
Distance Vector Routing Protocols

• Distance Vector Routing Table Exchange

Node A → Node C → Node D → Node F

Node B ← Node E ← Node G

• RIP Routing Example

Node A → T1 → Node B → T1 → Node C → T1 → Node E

Node A ← 500kb file ← Node B ← T1 ← Node C ← T1 ← Node E

Node D ← 9600 bps ← Node B ← T1 ← Node C ← T1 ← Node E

Node D ← 9600 bps ← Node E
Link State Routing Protocols

• Internet’s open Shortest Path First (OSPF)
  – Example of a Link State routing algorithm
  – Operates only with TCP/IP
  – Evaluates multiple factors
    • Shortest path
    • Bandwidth
    • Cost
    • Traffic loading
    • Expected delay
  – Ranks potential paths with a single number value

• Other Routing Protocols
  – OSI’s Intermediate System-to-Intermediate System (IS-IS)
    • Equivalent to IP
    • End System-to-Intermediate System (ES-IS) is equivalent to ARP
  – Novell Link Services Protocol (NLSP)
History of TCP/IP
(Transmission Control Protocol/Internet Protocol)

- **1960** - RAND Corp. paper on survivable communications in the event of a nuclear exchange using packet switching

- **1960s** - ARPA (Advanced Research Projects Agency) work on sharing technical data among university, military, and defense contractors

- **1972** - ARPANET connects UCLA, University of California at Santa Barbara, University of Utah, and Stanford Research Institute
  - First packet switching network
  - Uses Network Control Protocol (NCP) and Interface Message Processor (IMP) protocol
  - ARPA Internetting Project begins study to link various packet networks
    - Basis for development of TCP/IP

- **1975** - Management of the Internet is transferred to the Defense Communications Agency (DCA)
  - BBN opens TELNET, first commercial packet switched network
  - TCP defined
### History of TCP/IP

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1979</td>
<td>USENET begins offering its worldwide broadcast conferencing system of newsgroups</td>
</tr>
</tbody>
</table>
| 1981 | BITNET (Because It's Time Network) begins offering e-mail and list servers for information distribution  
|      | CSNET (Computer Science Network) offers dial-up service for e-mail |
| 1982 | TCP/IP becomes the standard for network communications on ARPANET |
| 1983 | ARPANET began using TCP/IP  
|      | MILNET and DDN (Defense Data Network) split off from ARPANET  
|      | ARPANET was the nonmilitary research network and the origin of the Internet |
| 1984 | The Domain Name Server (DNS) system is introduced |
## History of TCP/IP

- **1986** - The National Science Foundation (NSF) funded the construction of a 56 kbps network, NSFNET, connecting its six new supercomputer centers.
- **1988** - "Worm" released into the Internet
  - NSFNET upgraded to DS1
- **1989** - ARPANET ceases and joins to NSFNET
  - NSFNET backbone, T1 level (1.544 Mb/s)
- **1990** - CERN (the European Particle Physics Laboratory in Switzerland) develops the World Wide Web
  - NSF begins upgrading the entire Internet backbone to DS3 speeds (45 Mbps) for supercomputer interconnection
  - ARPANET ceases
History of TCP/IP

• 1991 - NSF lifts the restrictions against commercial use of the Internet

• 1994 - Digital video and audio transmit over the Internet
  – NSF began upgrade of the Internet backbone for supercomputer communication to OC 3 speeds (150 Mbps)
    • Major Internet Service Providers (ISPs) also upgrading to OC 3
  – Internet standards body, the Internet Engineering Task Force (IETF) formed

• 1995 - NSFNET ceases, non-US networks over 50%

• 2000 – TCP/IP are the most widely used protocol in the world
Transmission Control protocol (TCP) / Internet Protocol (IP)

- Many network and transport protocols work closely together

- User Datagram Protocol (UDP) and TCP interface directly with IP
  - Allow the host to distinguish among multiple applications through port numbers
  - TCP provides adaptive flow control, segmentation, and reassembly, and prioritized data flows
  - TCP provides a reliable sequenced delivery of data to applications
  - UDP only provides unacknowledged datagram capability

<table>
<thead>
<tr>
<th>OSI Reference Model</th>
<th>Internet Architecture</th>
<th>Example Protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 Application</td>
<td>Application</td>
<td>FTP</td>
</tr>
<tr>
<td>6 Presentation</td>
<td>Transport</td>
<td>TELNET</td>
</tr>
<tr>
<td>5 Session</td>
<td>Internet</td>
<td>SMTP</td>
</tr>
<tr>
<td>4 Transport</td>
<td>Network Interface</td>
<td>SNMP</td>
</tr>
<tr>
<td>3 Network</td>
<td>Hardware</td>
<td>TFTP</td>
</tr>
<tr>
<td>2 Data Link</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>1 Physical</td>
<td></td>
<td>IP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>802 LANs, Frame Relay, ATM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UTP, Coax, Fiber</td>
</tr>
</tbody>
</table>
Service Aspects of TCP/IP

- TCP/IP implementations typically constitute a router, TCP/IP workstation and server software, and network management.

- TCP/IP protocol implementations span UNIX, DOS, VM, and MVS environments.

- Many Network Operating System (NOS) have integral TCP/IP
  - Novell NetWare, Banyan Vines, and Windows 95/98

- IP operates of over a number of network, data link, and physical layer services
  - Network layer: IP can operate over X.25 and SMDS
  - At the data link layer, IP operates over frame relay, Ethernet, Token Ring, FDDI, and ATM
  - At the physical layer: IP operates over circuit switched and dedicated lines
Related Protocols

- Some other protocols interfacing with TCP
  - Address Resolution Protocol (ARP) maps a physical address (e.g., an Ethernet MAC address) to an IP address
    - Interfaces directly to the Data Link Layer
  - Domain Name Services (DNS) provide a centralized name service
  - File Transfer Protocol (FTP) provides security log in, directory manipulation, and file transfers
  - Trivial FTP (TFTP) protocol provides a simplified version of FTP
  - Internet Control Message Protocol (ICMP) allows routers to send error and control messages to other routers and allows users to “ping” an IP addressed host
    - Verify connectivity by sending an echo packet
  - Remote Procedure Call (RPC) and Network File Server (NFS) capabilities allow applications to interact over IP
  - Simple Network Management Protocol (SNMP) supports configuration setting, data retrieval, and alarms
  - TELNET provides a remote terminal log in capability
TCP/IP Functions

• TCP and IP assume the underlying network is a connectionless
  – Packets may be delivered out of order or have duplicates delivered

• IP provides a connectionless datagram delivery service to the transport layer
  – Does not provide:
    • End to end reliable delivery
    • Error control
    • Retransmission
    • End to end flow control
  – Relies on TCP to provide these functions.

• Functions of IP as a routing protocol
  – Provides the means for devices to discover the topology of the network
  – Detect changes of state in nodes, links, and hosts
  – Route packets along available paths
    • Route around points of failure
    • No notion of reserving bandwidth

• TCP is a connection-oriented protocol
  – Has specific messages for an application to request a distant connection
  – Has messages for a destination to identify that it is ready to receive incoming connection requests
IP Packet Formats

<table>
<thead>
<tr>
<th>Version</th>
<th>Header Length</th>
<th>Type of Service</th>
<th>Total Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Identification</th>
<th>Flags</th>
<th>Fragment Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time</th>
<th>Protocol</th>
<th>Header Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Destination IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Options</th>
<th>Padding</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

- Version specifies the IP protocol version
- Header length specifies length in units of 32 bit words
- Meaning of the Type Of Service (TOS) varies across the industry
- Total length specifies the total IP datagram length for the header and the user data
- Identification, flags, and fragment offset control segmentation and re-assembly of IP datagrams
- Time To Live (TTL) specifies how many routers the packet can pass through before it is declared "dead"
  - Intermediate nodes or routers decrement TTL, and when it reaches zero, the packet is discarded
- Protocol identifies the higher level protocol type (e.g., TCP or UDP), which identifies the format of the data field
- Header checksum ensures integrity of the header fields through a calculation that is easy to implement in software
- Source and destination IP addresses are required
- Options and padding are not mandatory and can specify routing and timestamp information
- Data field carries user data

---

Copyright 2000 © Roy M. Wnek. All rights reserved.
TCP Frame Format

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>4</th>
<th>10</th>
<th>16</th>
<th>24</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Port</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination Port</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequence Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acknowledgement Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HLEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reserved</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code bits</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Window</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Urgent Pointer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Options</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Padding</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Source and destination TCP port numbers identify a specific application programs
- Sequence number identifies the position of the sender's byte stream in the data field
- Acknowledgment number identifies the number of the next octet to be received
- HLEN provides the length of the header
- Code bits determines the use of the segment contents (e.g., SYN for synchronize sequence numbers and RST for reset connection)
- Window tells the amount of data the application is willing to accept
- Checksum is applied across the TCP header and the user data and is used to detect errors
- Urgent pointer specifies the position in the data segment where urgent data begins if the code bits indicate that this segment contains urgent data
- Options and padding fields are not mandatory
- Data field carries user data
TCP/IP Operation

- TCP works over IP to achieve end to end reliable transmission of data across network

- TCP handles message segmentation and reassembly using the sequence number in the TCP header
  - IP does this using the fragment control fields in the IP header
  - Either method, or both, may be used

- TCP/IP of acceptance of datagrams out of order gives ability to operate over an unreliable underlying network
  - Makes it quite robust
  - None of the other standard modern data communication protocols has this attribute
  - IPX/SPX approximates it
Traffic and Congestion Control

- TCP flow control uses a sliding window flow control protocol
  - Like X.25, but with a variable size window

- The sender starts with a window size equal to that of one TCP segment
  - The IP datagrams are delivered to the destination workstation
  - The delivered TCP segment is acknowledged.

- The sender then increments the window size and transmits more segments
  - Until the network has become congested and the third segment is lost (not acknowledged)
  - The sender detects this by starting a timer whenever a segment is sent
  - If the timer expires, then
    - The segment is resent
    - The sender reduces window size to one segment
    - The process is repeated

- Specific implementations will vary
  - Fine tuning of the algorithms
  - The TCP/IP architecture permits the sender and receiver to use different algorithms
TCP Dynamic Windowing Flow Control

Window

1. \( S(0) \) → \( S(0) \)
2. \( S(1) \) → \( S(1) \)
   \( S(2) \) → \( S(2) \)
   \( S(2) \rightarrow \) ACK(1)
   \( S(2) \rightarrow \) ACK(2)
3. \( S(3) \) → \( S(3) \)
   \( S(4) \) → \( S(4) \)
   \( S(5) \) → \( S(4) \)
   \( S(5) \rightarrow \) ACK(3)
   \( S(5) \rightarrow \) ACK(4)

Timeout

1. \( S(5) \) → \( S(5) \)
   \( S(5) \rightarrow \) ACK(5)

\( S \) = TCP Segment
ACK = Acknowledgement
\( (N) \) = Segment Number
Internet Addresses

• Computers
  – host.institution.domain
    e.g. lucia.lib.lawrence.edu

• Users
  – username@host.institution.domain
    e.g. wnekr@erols.com

• Uniform Resource Locator (URL)
  – protocol://host.institution.domain[/directory/filename]
    e.g. http://cwis.lawrence.edu/www/lib/learn/5.htm
Domain Naming Service 1

• The Internet Network Information Center (InterNIC) is the government-sponsored activity that assigns all US domain names
  – charges $100 for the first two years of domain name service, plus $50 every subsequent year
  – currently, the domain name registrar is Network Solutions Inc. but 4 other companies are being transitioned into this role

• Sites outside the US use different organizations to assigns domain names using their own standards
  – E.g. domain names .co.uk or .ac.uk, indicate a company or academic institution in the United Kingdom

• Domain Naming Service (DNS) is an Internet protocol and distributed database
  – Used to translate between domain names and IP addresses
    • pointing a browser at www.asite.com is far easier than remembering the site's numeric IP address
  – Used to control Internet email delivery
• At the top of the DNS database tree are Root Name Servers, which contain pointers to master name servers for each of the top-level domains (e.g. .com).
  – E.g. to find the IP address of www.asite.com, a DNS server would ask the root name server for the address of the master name server for the .com domain.

• The Master Name Servers for each of the top-level domains contain a record and name-server address of each domain name

• The individual name servers for each domain name contain detailed address information for the hosts in that domain.

DNS server ➔ Root Name Servers

Master Name Servers

.com .org .edu .gov ...

Individual Name Servers

Domain name IP address

Copyright 2000 ©Roy M. Wnek. All rights reserved.
• Providing DNS is an important part of connecting to the Internet

• There are two basic ways to use DNS
  – Use your ISP's DNS server
  – Set up a DNS server on your own network

• With a small network-support staff and simple needs, having an ISP provide DNS services is the better option
  1. ISP informs InterNIC that it is providing both primary and secondary DNS services for your organization
  2. ISP gives you the numeric IP addresses of the primary and secondary DNS servers, which you’ll need to configure your users’ TCP/IP stacks by entering this information manually either at the desktop or at your Dynamic Host Configuration Protocol (DHCP) server
  3. you inform the ISP about the DNS records that you wish to publish to allow outside users to interact with your network
To receive E-mail from the Internet, need a Mail Exchange (MX) Record for your domain in the ISP's DNS database
  - MX refers to a machine that accepts E-mail connections for your domain.

An MX Record has three parts:
  - your domain name,
  - the name of the machine that will accept mail for the domain,
  - and a preference value
    - The preference value lets you build some fault tolerance into your mail setup.

A domain can have multiple MX Records:

```
  Acme.com mail.acme.com 0
  Acme.com mail2.acme.com 10
  Acme.com mail.isp.net 100
```
  - First, mail delivery will be attempt to mail.acme.com because it has the lowest preference value.
  - If delivery fails, mail2.acme.com will be tried
  - If mail2 is also down, mail will be sent to mail.isp.net, which in this case is at Acme's ISP
• If using an ISP's DNS server, need the ISP set up some ‘A Records’
  – ‘A Records’ which associate IP addresses with computer names
  – Each of the computers mentioned in your MX records needs an ‘A Record’ to associate them with an IP address

• May want to set up ‘A Records’ for each of your workstations if your users need to use ftp (File Transfer Protocol) to download software from the Internet
  – Because some ftp sites perform a lookup to get the DNS name of the machine from which they receive download requests
  – If the machine has no name, the sites deny the request

• Need ‘A Records’ for any public access servers, “Web Servers”
  – e.g need the ISP set up an ‘A Record’ linking the name www.acme.com to the IP address of your Web server

www.acme.com
Domain Naming Service 6

- InterNIC requires two name servers--a primary and a secondary before they will grant a domain name
  - unless there are at least two DNS servers on the Internet with information about that domain
  - also a second server provides fault tolerance

- Some sites use an onsite DNS server as well as their ISP's
  - Because maintenance of the domain names is done at the primary name server, choosing which one is primary is important
  - If you administer the primary name server, keep you must maintain the DNS records
  - If you have only a secondary name server, your ISP will do all of the work, and your secondary name server will simply download the data about your domain from the primary server periodically
• Reasons for having a DNS server on a LAN
  – If running IP network-based applications inside the network that require users to connect to internal machines by name, it is not a great idea to advertise the names and addresses of these machines
    • DNS can give hackers a map of your network
    • Setting up an internal DNS server does not publish information to the world
  – A DNS server inside the network allows changes, additions, and deletions on your own schedule
  – Finally, name resolution will be faster for users because the DNS server is probably not as heavily loaded as an ISP's server
IP Addressing

• IP addressing is based on hosts and networks
  – Hosts or nodes are any device capable of receiving and transmitting IP packets e.g. workstations and routers
  – Hosts are connected together by one or more networks

• Host IP addresses are composed of two parts: a network address and a host address within the network
  – Unlike IPX addresses
  – An IP address is 32 bits wide
  – The portion of the address is used to specify the network and the portion used to specify the host varies from network to network

• IP addresses are, by convention, expressed as four decimal numbers separated by periods e.g. "200.1.2.3"
IP Address Classes

- Valid IP addresses thus range from 0.0.0.0 to 255.255.255.255,
  - A total of about 4.3 billion addresses

- The first few bits of the address indicate the Class that the address belongs to:

<table>
<thead>
<tr>
<th>Class</th>
<th>Prefix</th>
<th>Network Number</th>
<th>Host Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>Bits 1-7</td>
<td>Bits 8-31</td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td>Bits 2-15</td>
<td>Bits 16-31</td>
</tr>
<tr>
<td>C</td>
<td>110</td>
<td>Bits 3-24</td>
<td>Bits 25-31</td>
</tr>
<tr>
<td>D</td>
<td>1110</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>1111</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

- Class D addresses are multicast, and Class E are reserved

- The range of network numbers and host numbers may then be derived:

<table>
<thead>
<tr>
<th>Class</th>
<th>Range of Net Numbers</th>
<th>Range of Host Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0 to 126</td>
<td>0.0.1 to 255.255.254</td>
</tr>
<tr>
<td>B</td>
<td>128.0 to 191.255</td>
<td>0.1 to 255.254</td>
</tr>
<tr>
<td>C</td>
<td>192.0.0 to 254.255.255</td>
<td>1 to 254</td>
</tr>
</tbody>
</table>
Classed IP Addressing and the ARP

• Reserved addresses
  – Addresses starting with 127 is a loopback address
    • Never be used for addressing outside the host
  – A host number of all binary 1's indicates a directed broadcast over the specific network
    • E.g. 200.1.2.255 = a broadcast over network 200.1.2
  – A host number of 0 indicates "this host"
  – A network number of 0 indicates "this network"

• Reserved addresses severely reduce the available IP addresses from the theoretical maximum of 4.3 billion

• Most Internet users will be assigned addresses within Class C
  – Many networks with relatively fewer nodes
  – Address space is becoming very limited
  – Primary motivation for IPv6, which will have 128 bits of address space
Internal Network Addressing

• A small internal TCP/IP network consisting of one Ethernet segment and three nodes
  – The IP network number is 200.1.2.
  – The host numbers are 1, 2, and 3
  – These are Class C addresses (max of 254 nodes on this network segment)

• Each node has a corresponding six byte Ethernet address
  – Normally written in hexadecimal form separated by dashes, e.g. 02-FE-87-4A-8C-A9
Address Resolution Protocol (ARP)

• Node A needs to send a packet to node C
  – A knows C's IP address
  – The Address Resolution Protocol (ARP) is used for the dynamic discovery of C's Ethernet address

• ARP keeps an internal table of IP address and corresponding Ethernet address
  – The ARP module at node A does a lookup in its table on C's IP address
  – If this is the first transmission, the ARP will discover no entry
    • ARP will then broadcast a special request packet over the Ethernet segment
    • If the receiving node (node C) has the specified IP address, it will return its Ethernet address in a reply packet back to node A
    • Once node A receives this reply packet, the ARP module updates its table
    • Node A uses the Ethernet address to direct the packet to node C
    • ARP table entries
      • Stored statically
      • Stored with a time stamp so that old table entries are periodically flushed
Two Connected Ethernets

- Separate Ethernet networks that are joined by node C
  - Device C acts as an IP router between these two networks
  - A router chooses different paths for packets, based on IP addressing
  - The router will have more than one address as it is part of different network.
Two Connected Ethernets

• Each Ethernet segment has its own Class C network number
  – The router must know which network interface to use to reach a specific node
  – Each interface is assigned a network number.

• If A wants to send a packet to E, it must first send it to C who can then forward the packet to E
  – Accomplished by having A use C's Ethernet address, but E's IP address
  – C will receive a packet destined to E and will then forward it using E's Ethernet address

• If E was assigned the same network number as A
  – A would try to reach E by sending an ARP request and hoping for a reply
  – Since E is on a different segment, it will never see the ARP request and never reply

• By specifying that E is on a different network
  – The IP module in A will know that E cannot be reached directly
  – Requires forwarding by some node on the same network
Direct vs. Indirect Routing

• Direct routing
  – Both the source and destination addresses have the same network number
  – Packets do not need forwarding

• Indirect routing
  – The network numbers of the source and destination are not the same
  – Packets must be forwarded by a node that knows how to reach the destination

• A "gateway" node or router
  – Node on both networks
  – Must have an IP address for each network
  – Sends routing data to other nodes
    • route add [destination_ip] [gateway] [metric]
  – Metric value = number of hops to the destination
  e.g.
    – route add 200.1.3.2 200.1.2.3 1
      • tells A to use C as the gateway to reach E
    – route add 200.1.2.1 200.1.3.10 1
      • tells E to use C as the gateway to reach A

• Node C’s routing module will indicate which network interface to use for forwarding IP packets
  – Normally sufficient to set up C as the default gateway for all other nodes on both networks
Static vs. Dynamic Routing

- The default gateway is the node to send all packets that are not addressed to a node on the directly-connected network
- The routing table within the default gateway will properly forward packets

- **Static routing**
  - Performed using a preconfigured routing table which is only changed manually
  - Most basic form of routing
  - Requires that all machines remain on their networks and have statically configured addresses
    - Otherwise, the routing tables must be manually altered

- **Dynamic routing**
  - Uses routing protocols to automatically update the routing table with routes known by peer routers
  - Protocols are grouped according to ability to distribute routing information in an Autonomous System (AS)
    - An AS is a set of routers inside the domain administered by one authority
    - Interior Gateway Protocols (IGPs) = intra-AS
      - e.g. OSPF and RIP
    - Exterior Gateway Protocols = inter-AS
      - e.g. EGP and BGP
The Netmask

• A netmask must be specified when setting up each node with its IP address

• Netmasks are used to specify which part of the address is the network number and which is the host number
  – Accomplished by a logical bitwise-AND between the netmask and the IP address
  – The result specifies the network number
  – For Class C addresses, the netmask = 255.255.255.0
  – For Class B, the netmask = 255.255.0.0
  – For Class A, the netmask = 255.0.0.0

• When node A sent a packet to E, it compared
  – AND (255.255.255.0 and 200.1.3.2) = 200.1.3
  – A's own network number 200.1.2
  – Therefore, A knew that E wasn't on the same network segment

• The netmask becomes more complicated with "classless" addressing
Hierarchical Sub-Allocation of Class C Addresses

• Class C are subnetted hierarchically from the Internet Service Provider to user organizations
  – Make more efficient use of addresses
  – Users are allocated bitmask-oriented subsets of the provider's address space
  – These are classless addresses

• Service provider allocated several different Class C addresses to be used for its clients
  – Example organization has been allocated the network number 210.20.30
  – The gateway address at the provider end is 210.20.30.254

• If the organization has only one computer, C
  – The entire Class C address is available for use
  – C’s IP address may be anything in the range 210.20.30.1 to 210.20.30.253
  – Its default gateway would be 210.20.30.254 with netmask 255.255.255.0.

• With three networks, the Class C address must be subnetted
Subnetting

- Accomplished by using one or more of the bits that are normally allocated to the host number to extend the size of the network number
  - In this example, 210.20.30 has been extended to include four networks (3 in use, 1 reserve)
  - Two additional bits used for the network number in the IP address
  - Netmask has changed to 255.255.255.192
  - in binary = 11111111.11111111.11111111.11000000

- Since the organization is allocated all of 210.20.30, it has the use of the four following network numbers (in binary):

<table>
<thead>
<tr>
<th>Net#</th>
<th>IP Network Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>11010010.00010100.00011110.00</td>
</tr>
<tr>
<td>1</td>
<td>11010010.00010100.00011110.01</td>
</tr>
<tr>
<td>2</td>
<td>11010010.00010100.00011110.10</td>
</tr>
<tr>
<td>3</td>
<td>11010010/00010100.00011110.11</td>
</tr>
</tbody>
</table>