Midterm Exam 2 Review

Covers Text:

• Chapter 4
  – Section 4.7 Data Link Layer Switching

• Chapter 5
  – (for exam omit sections 5.2.9-11, 5.4.3, 5.4.4, 5.4.5)

• Chapter 6
  – Through 6.5.10
Internet Layering

Level 5 -- Application Layer
   (rlogin, ftp, SMTP, POP3, IMAP, HTTP..)

Level 4 -- Transport Layer (a.k.a. Host-to-Host)
   (TCP, UDP, RTP)

Level 3 -- Network Layer (a.k.a. Internet)
   (IP, ICMP, ARP)

Level 2 -- Data Link Layer / MAC sub-layer
   (a.k.a. Network Interface or Network Access Layer)

Level 1 -- Physical Layer
General

• Any layer
  – Where does it fit in the layered architecture?
  – What services does it provide and to what layer?
  – What services does it use in providing these services and from what layer?
  – What problems does the layer address/solve or what deficiencies does the layer overcome?
  – What neat algorithms are used in this layer?
  – Where is the layer implemented (in the host, NIC, routers, etc.)?
Efficiency / Utilization

As in homework problems (and last midterm):
Be prepared to determine how a protocol may limit the utilization of a link or network to be less than its inherent capacity.
Know how to calculate the amount of packets, IPDUs etc. that are enroute on a link or in a network given its length and its bit rate (or “speed”)
Link Layer Bridging (802.x)

• Fixed Route Bridging
• Source Route Bridging
• **Transparent Bridging**
  – “Plug ‘n Play”:
    • Flooding algorithm
    • Backward Learning
  – Spanning Tree (avoids loops)
Spanning Tree Approach- Uses:

- IEEE 802.1
- Frame Forwarding
  - Each bridge maintains Forwarding Table
  - List of stations on the “side” of each port
  - Forwards to port and on to LAN corresponding to Table
    - Unless blocked
  - Floods those whose MAC address not in Table
- Address Learning
- Spanning Tree Algorithm
Address Learning

• When frame arrives on a port, table records source address as being on that port. (Backward learning)

• Timer set for each entry
  – Timer expires, entry is deleted
  – Timer reset if new frame gives same info.
Spanning Tree Algorithm

- Algorithm to avoid Loops
- (not needed if topology is a “tree”)
- Root of tree is lowest serial number (all bridges broadcast their serial number)
- Tree constructed from root to every bridge with shortest path
- Algorithm runs “continually” to detect topology changes (IEEE 801.1D)
Network Layer – Chapter 6

- Datagrams and Virtual Circuits
- Packet Routing
- Distance Vector Routing
- Link State Routing
- Hierarchical Routing
- Congestion Control
  - Choke Packets / Load shedding
- QOS (traffic shaping, buffering, overprovisioning)
- Leaky Bucket / Token Bucket
- Tunneling
Network Layer

- IP Protocol / packet (datagram)
  - Headers up to 60 bytes, total length up to 65,535 bytes
- Addressing (subnets, masks), aggregation
- NAT
- ARP / RARP (physical address $\leftrightarrow$ IP address)
- DHCP (dynamic IP address assignment)
- IGP (Interior gateway routing protocol)
  - RIP
  - OSPF (link state algorithm)
- EGP (Exterior gateway routing protocol)
  - Policies / politics / constraints
  - BGP (distance vector protocol)
Network Layer

• IPv4 vs IPv6
  – Main differences between IPv4 and IPv6
  – Reason(s) for change (see text pg. 465)
  – Reason(s) change has been slow to happen
Transport Layer

- End-to-end
- Used by applications on client and server
- Under user control (operating system / programming language calls)
- Transport Services
  - Connectionless (User Datagram Protocol UDP)
    - No flow control, no error control
  - Connection-oriented (TCP)
    - Flow and error control
  - RTP (for multimedia)
Transport Layer

- Connection Management
  - 3-way handshake in making connection
  - Sequence numbers / avoiding duplicates
  - Connection release (timer use)
- Berkeley sockets
- Flow Control and Buffering
- Multiplexing
- Crash Recovery
- End-to-End *ack*
Transport Layer

• TCP
  – Congestion Management
  – Estimating RTT (exponential weighting)
  – Timers
  – “Slow Start” / exponential back-off

• Network Performance
  – traceroute
  – ping
  – Transfer time components (vs. length, bandwidth)