IP Mobility

- Host addressing
  - DHCP and NAT
    - Ok for client, not server
    - Provides nomadicty
    - Not mobility
  - Change IP address, but not hostname
    - DNS system can’t handle it now
    - Could be fixed
    - TCP servers can’t handle IP addr change
      - All connections must terminate
Mobile IP

- RFC 2002-2006
  - Mobile note
  - Home Agent
  - Foreign Agent
Mobile IP

- Home/Foreign agents broadcast advertisements
- Mobile host determines location and registers
- Home agent advertises mobile node reachability
- Tunnels packets to foreign agents
- Reply packets are routed directly
Admission Control

- Choke Packet
  - voluntary host reduction
- Fair queuing
  - Individual queues
- Weighted fair queueing
- Load Shedding
  - wine and milk
- Fragmentation
  - Reassembly (attacks)
Hardware flow control

- RTS/CTS
- 802.X
  - Pause control
- VLANs
  - VLAN tags
    - 4 bytes between Eth Source and type/length
      - user priority 3 bits
      - Canonical Format Indicator 2
      - VID 12 bits
Other Wireless Solutions

- Wireless Application Protocol (WAP)
  - TCP developed mostly over wired connections
  - Argued that other protocols are needed
    - When bandwidth is expensive
      - TCP is too chatty
WAP stack elements

- WDP - Wireless Datagram Protocol-like UDP
- WTLS - W Transport Layer Security
  - Similar to RFC-2246
- WTP - wireless Transport Protocol
  - non-reliable one-way, reliable one-way, reliable two way
- WSP - Wireless Session Protocol
  - Datagram and VC
- WAE - Wireless Application Environment
Cell Phones and WAP

- XML
- WML an XML derivative
  - `<!DOCTYPE wml PUBLIC “-//WAPFORUM//DTD WML 1.1//EN”`
- Java Servlet (JSDK)
- WML script
  - Client, e.g. phone, execution
WML hello world

• <wml>
  • <card>
    • <p> Hello World. </p>
  • </card>
• </wml>
  • Softkeys
ARP

- IP addr to Ethernet address mapping
  - ARP request
  - Proxy ARP
  - Static ARP
  - Reverse ARP
Domain Name System (DNS)

- Replaced “hosts.txt” file
- Name-number-name mapping
  - Hierarchy
    - Full specification or default host
  - Caching
  - Reverse lookups
Transport Layer

• Provides reliable, efficient, data server to application layer
  • End point of the efforts of the other layers
  • Applications can use standard set of interfaces
  • Underlying layers can be unreliable.
    • Some duplication of error handling of network layer
    • Better efficiency if network layer handles problems
      • But better to duplicate in Transport layer than having every application implement reliability features
Implementing Quality of Service (QoS)

- QoS Parameters
  - Connection initiation delay
  - Connection initiation failure probability
  - Throughput
  - Transit delay
  - Error ratio
  - Protection
  - Priority
  - Resilience
Transport Layer Primitives

- Listen - wait for connection
- Connect -- establish connection
- Send -- send data
- Receive -- receive data
- Disconnect -- end connection
Berkeley Sockets

- Socket--create socket
- Bind--attach local address to socket
- Listen--accept connections, Q size
- Accept--block until data received
- Connect--establish connection
- Send--send data
- Receive--receive data
- Close
Establishing connections

- Sounds simple
  - Request, ack, ack
- Problems
  - Lost packets are not bad
  - Delayed duplicate packets are bad
    - Restrict subnet to time out packets
    - Set a low hop count limit
    - Timestamp each packet
Aside: ICMP Trace Route

- Sets incremental hop count limits
- Provides path of each packet by collecting ICMP errors as packet hop counts expire.
- Valuable tool for Internet problems
  - Available for all platforms
Tomlinson packet lifetime bounding

- Consider sequence numbers which track clock
  - Can’t send packets faster than clock
  - Sending too slow also causes problems.
  - Resynchronize sequence numbers
Establishing Connections

- Three way connect handshake
- Syn attacks
  - Spoofed address
  - Router defenses
    - TCP-Intercept mode
    - tcp-watch mode
      - tcp intercept watch mode
      - tcp intercept watch-timeout
        - Default 30 seconds
  - After 1100 incomplete connections
    - Aggressive mode 15 seconds
Aside: Connection Hijacking

- Initial Sequence number attacks
  - First Sequence number should be random
  - Blind Spoofing attack
    - Computer random numbers are predictable
    - Can not try all $2^{32}$
  - John Morris, 1985, weakness in Berkeley ISNs
- RFC 1948
  - Use IP addr, ports, plus secret key
- Interesting phase space analysis of ISNs
  - razor.bindview.com/publish/papers/tcpseq.html
Closing Connections

• Again, sounds simple
  • Two army problem
• Easily proven that no protocol exists that works
  • Proof by contradiction
• Accept some risk and use three way handshake
  • Timers release connection after no answer
Disconnect

- Disconnect Request
  - timer
- Dis. Response
  - timer
- Release
  - send ack
  - release
TCP Header Segment

- source port, dest port
- seq number
- ack number
- header len & flags, window size
  - urgent flag
  - ack valid flag
  - push, rst, syn, fin
- checksum, urgent pointer
  - checksum includes addresses, proto, segment len
- options (variable number of words)
- data
TCP Header Options

- Window size negotiation-RFC1323
  - left shift 16 window size
    - allow 14 bit shift, hence $2^{30}$ bytes
- Selective repeat
  - use of NAK
TCP state management

- Closed
- listen
- Syn Rec--request arrived
- Syn sent--open started
- Established
- Fin Wait 1--app finished
- Fin Wait 2--other side agreed
- Timed wait--wait for packets to die
- closing--both sides closing
- close wait--other side initiated
- last ack--wait for packets to die
Tomlinson packet lifetime bounding

- Consider sequence numbers which track clock
  - Can’t send packets faster than clock
  - Sending too slow also causes problems.
    - Resynchronize sequence numbers
TCP Implementation

- TCP Tuning
  - Retransmission timer and backoff
  - Window based flow control
  - Max segment size
  - Congestion Avoidance
  - Round Trip Estimate
Retransmission Timer and backoff

- RTT for each connection
  - RTT = RTT + x*error
    - Use mean deviation
      - Adjust fast to varying conditions
- Karn’s algorithm
  - Don’t measure RTT of retrans.
    - TCP_Maxretries
      - Double time-out for small value
Window Management

- Advertise window size
  - Piggyback on acks
    - Otherwise 40 bytes of header wasted on each
      - Nagel’s algorithm.
        - send one byte and buffer the rest
  - On zero
    - Urgent data still sent
- Silly window syndrome
  - Persist state
Segment size

- Want as big as possible, without fragmentation
  - Local connections
    - Use data link level size
  - Gateway connections
    - Use min gateway size 536 octets
    - advertise in Syn packet
TCP Adaptive Retransmission

- Multiplicative decrease
  - Reduce congestion window on each retrans
  - Trans min of congestion window or recipients window
- Slow start on new connections
  - Send one MSS
    - Add one more MSS to window for each success
- Slow after threshold
  - Add smaller increments, one MSS per window