IP Mobility

- Application level mobility
  - Home and Foreign agents
  - Mobitext network
- Link Level Mobility
  - CDPD, Mobitext
  - 802.11b
- Host address routing
  - Routing uses longest prefix
  - Host based routing can work
    - Requires every router in path to be updated!
IP Mobility

- Host addressing
  - DHCP and NAT
    - Ok for client, not server
    - Provides nomadicity
    - 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
Remote Procedure Calls

- Distributed OS
- Application oriented design
  - Vs. Communication oriented design
- Model for distributing procedure calls over network
  - Conventional Procedure
    - Single thread of control
    - Call and return
  - RPC model
    - Call and wait
    - Data transfer
    - No shared pointers or I/O descriptors
RPC Development

- RPC Politics
- XML RPC
  - Many languages
- SOAP RPC
  - “Web Services”
SUN ONC RPC

- Open Network Computing (ONC)
  - Remote Program and procedures
    - Combine multiple procedures in single program
  - Remote Program identification
    - 0x00000000-0x1fffffff  SUN
    - 0x20000000-0x3fffffff  Local site
    - 0x40000000-0x5fffffff  Temporary
    - 0x60000000-0xffffffff  Reserved
ONC RPC

- Multiple versions of a program supported
  - RPC call specifies: (prog, vers, proc)
  - Easy migration to new component versions
- Mutual exclusion for procedures in remote program
  - ONC RPC specifies on one execution at a time
- Choose TCP or UDP
  - UDP At least once semantics
    - Receive a reply
  - UDP zero or more semantics
    - No reply
    - Calls should be idempotent
RPC Retransmission

- ONC RPC library supports simple retransmission
  - Simple retransmission does not guarantee reliability
    - Programmer can adjust retrans time
    - Eventually retry limit exceeded
Port Mapping

- ONC RPC programs have 32 bit number
  - TCP ports have 16 bit number
    - Can’t directly map RPC to TCP ports

- Dynamic Port Mapping
  - RPC port mapper
    - RPC program 100000, tcp port 111
    - One per machine
    - Programs register themselves
      - (RPC prog number, TCP port, version)
        - Registrations only from local machine
        - Request for lookup from anywhere
RPC Data Transfer

• Different processor architectures
  • Little- and big-endian
  • Convert to dest machine
    • \((N^2 - N)/2\) problem
    • Need Network standard order
• TCP does this with symmetric conversion of headers
• XDR - eXternal Data Representation
  • XDR defacto standard
ONC XDR

- XDR libraries provide conversion to network order from local order
  - One message at a time
  - Buffer conversion
    - xdrmem_create
      - char buf[bufsize];
      - xdrmem_create(xdrs, buf, bufsize, XDR_ENCODE)
    - xdr_int()
      - int i;
      - i = 10;
      - xdr_int(xdrs, &i);
ONC XDR

• Streams interface works well for TCP
  • TCP sockets use streams as well
• Alternative Record interface for UDP
  • inproc, outproc
    • input and output procedure
RPCGEN

- Remote Procedure Generation
  - Q.x
    - Q_clnt.c
  - Q.h
  - Q_xdr.c
  - Q_svr.c
  - You provide
    - Q_svr_proc.c
    - Q_clnt_proc.c (or Q.c)
RPCGEN

- sample
  - const XXX
  - struct example {
    - int field
    - char field
  }
  - program Q
    - version X
    - int proc1(string)
    - ....
  }
  = version
Distributed system generation

1) Write procedure program and test
2) Divide the program into remote and local
3) Write rpcgen specification
4) Run rpcgen to test config and generate header and client stubs
5) Write stub interface routines for client and server
6) Compile Client
   - original application
   - rpcgen client stub
   - client interface stub
   - rpcgen xdr procedures
Dist system cont.

- 7) Compile Server program
  - original server code
  - rpcgen server stub
  - server side interface stub
  - rpcgen xdr procedures
- 8) start and run server
Congestion Prevention

- Transport
  - retrans policy
  - out-of-order caching
  - ack policy
  - flow control
- Network
  - vc vs datagram
  - routing
  - packet discard
Congestion Prevention (cont)

- Data Link
  - retrans
  - out of order caching
  - ack policy
  - flow control
Flow Control

- Leaky bucket
  - Bursty traffic
- Token bucket
  - Credit for idle time
  - packet vs byte
Flow Specifications

• characteristics of input
  • Max packet size
  • Rate
• Service desired
  • Packet loss rate
  • Loss interval
  • Minimum delay
  • Mac delay
  • Quality guarantee
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
Internetworking

• Bridges
• Multiprotocol routers
• Transport Gateways
• Level-3 switching
• Fragmentation
Firewalls

- Firewalls inspect and allow or disallow each packet according to:
  - Sender IP address
  - Receiver IP address
  - Protocol type (by port number)
  - Direction of origination request,
    - e.g. allow telnet connections to originate from internal network, but only allow replies from external network
- Fairly universal deployment in corporations