Housekeeping

- Defer discussion of performance to next week
- Questions?
- Lecture Slides on class web site – now PDF format
- EECS accounts: mail info to jding1@eecs.wsu.edu
- Need the access code for ETRL 301? See the EECS systems staff in Sloan 358
- Agenda for today:
  - Implementation structure for protocol end-nodes
  - Feedback/Assessment

Lab Info

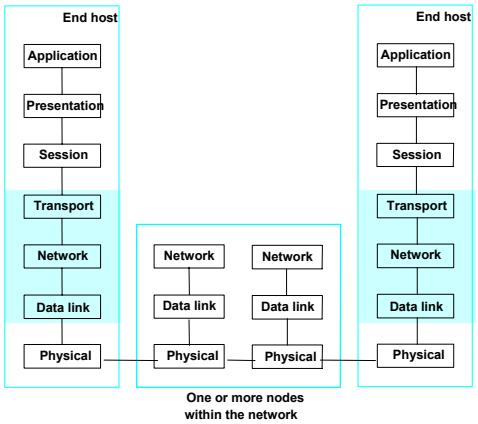
- Lab machines are: `nif-c10 to nif-c25, nif-c6, nif-c7, nif-c33, nif-c39`
- Need ssh version 2 to access remotely
  - www.ssh.com, www.openssh.org, `www.appgate.com/products/mindterm/`
- Or .. Use ssh v1 to connect to nif-s3 and then use the ssh version 2 installed there to get to the other machines

Elements of a Protocol Implementation

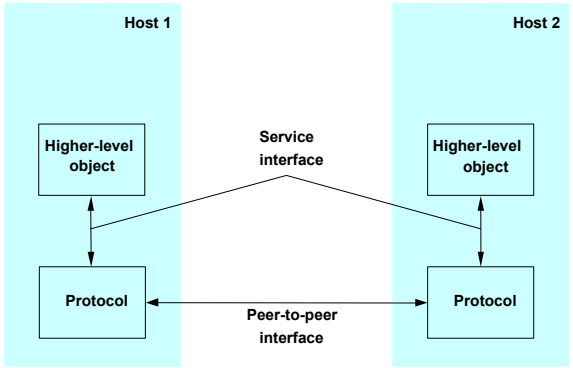
Outline

- Service Interface
- Process Model
- Common Subroutines
- Example Protocol

OSI Architecture



Interfaces



Socket API

- Creating a socket

  int socket(int domain, int type, int protocol)
  - domain = PF_INET, PF_UNIX
  - type = SOCK_STREAM, SOCK_DGRAM

- Passive Open (on server)

  int bind(int socket, struct sockaddr *addr, int addr_len)

  int listen(int socket, int backlog)

  int accept(int socket, struct sockaddr *addr, int addr_len)

Sockets (cont)

- Active Open (on client)

  int connect(int socket, struct sockaddr *addr,
  int addr_len)

- Sending/Receiving Messages

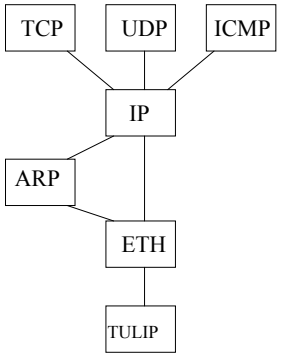
  int send(int socket, char *msg, int mlen, int flags)

  int recv(int socket, char *buf, int blen, int flags)

Protocol-to-Protocol Interface

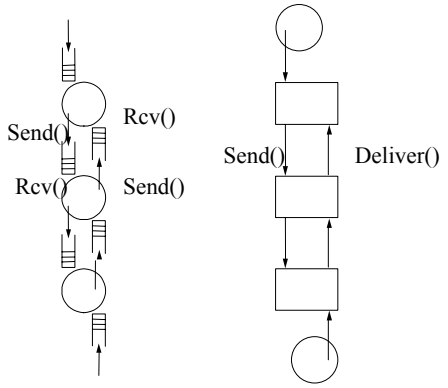
- Configure multiple layers
  - static versus extensible
- Process Model
  - avoid context switches
- Buffer Model
  - avoid data copies

PPI Configuration




```
name=tulip;
name=eth llp=tulip;
name=arp llp=eth;
name=ip llp=eth,arp;
name=icmp llp=ip;
name=udp llp=ip;
name=tcp llp=ip;
```

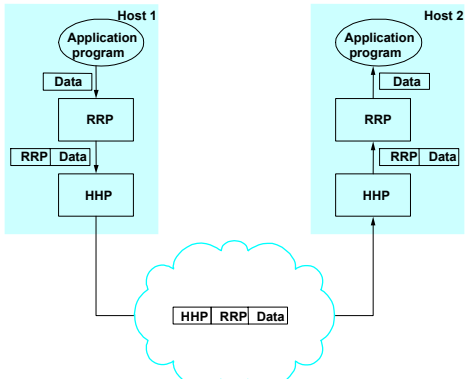

PPI Process Model



Process-per-Protocol Process-per-Message

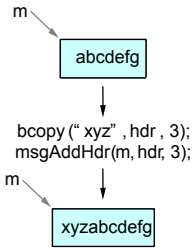
PPI Machinery

- Multiplexing and Demultiplexing (demux key)
- Encapsulation (header/body)

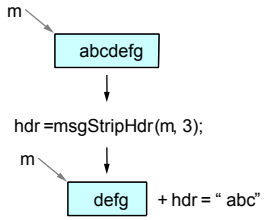


Message Library

- Add header

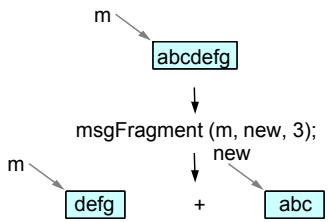


- Strip header

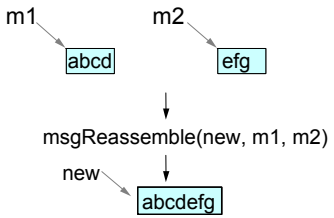


Message Library (cont)

- Fragment message



- Reassemble messages



Message Library Discussion

- Data Structures for messages
- Memory management
- Concurrency

Map Library

- Demultiplex Packets
- Operations

```
Map mapCreate(int number,int size)

Binding mapBind(Map map,
                void *key,
                void *id)

XkReturn mapResolve(Map map,
                    void *key,void **id)
```


Map Library Discussion

- Data structures
- Memory management
- Concurrency

Event Library

- Scheduling Timeouts & Book-keeping activity
- Operations

```
Event evSchedule(EvFunc function,
                 void *argument,
                 int time)

EvCancelReturn evCancel(Event event)
```