



## Computer Networks

Sockets

## Sockets and the OS

- ◆ An end-point for Internet connection
  - What the application “plugs into”
  - OS provides Application Programming Interface (API)

User  
Socket  
Operating System  
Network



## Socket Basics

- ◆ User sees “descriptor”, integer index
  - like: FILE \*, or file index
- ◆ End point determined by two things:
  - Host addr: Internet address of machine (street)
  - Port num: Where application receives (number)
- ◆ Two end-points determine a connection: socket pair
  - ex: reno.wpi.edu,p21 + bert.wpi.edu,p1500
  - ex: bill.microsoft.com,p21 + linus.linux.org,p1499



## Ports

- ◆ Numbers:
  - 0-1024 “reserved”, must be root
  - 1024 - 5000 “ephemeral”
  - however, many systems allow > 3977 ports
    - ◆ (50,000 is correct number)
- ◆ /etc/services:
 

```
ftp 21
telnet 23
finger 79
httpd 80
```



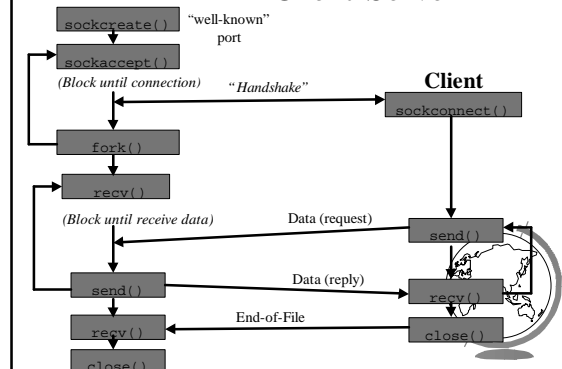
## Internet Protocols

- ◆ TCP: Transmission Control Protocol
  - reliable (in order, all arrive, no duplicates)
  - flow control
  - connection
  - duplex
- ◆ UDP: User Datagram Protocol
  - no acknowledgements
  - no retransmissions
  - out of order, duplicate possible
  - connectionless



### Server

### Client-Server



## Samples



## fcntl()

- ◆ 'File control' but used for sockets, too
- ◆ Set socket non-blocking

```
flags = fcntl(sockfd, F_GETFL, 0);
flags |= O_NONBLOCK;
fcntl(sockfd, F_SETFL, flags);
```

- ◆ Beware not getting flags before setting



## select()

- ◆ Wait for any in set of descriptors to be ready
  - data, error, closed

```
int select(int max, fd_set *readset, fd_set
          *writeset, fd_set *exceptset, timeval *timeout);
```

- ◆ check for reading, writing, exceptions

- ◆ fd\_set contains set of descriptors (bits in array)

- FD\_ZERO() - clear all bits
- FD\_SET() - turn on specific fd
- FD\_CLR() - turn off specific fd
- FD\_ISSET() - check if fd bit is set

- ◆ Ex: fd\_set rset:

- FD\_ZERO(&rset); /\* clear bits \*/
- FD\_SET(1, &rset); /\* turn on bit for fd 1 \*/



## Select between stdin and socket

```
FD_ZERO(&rset);
while (1) {
    FD_SET(fileno(stdin), &rset);
    FD_SET(sockfd, &rset);
    max = max(fileno(stdin), sockfd) + 1;
    select(max, &rset, NULL, NULL, NULL);
    if (FD_ISSET(sockfd, &rset))
        /* do socket stuff */;
    if (FD_ISSET(fileno(stdin), &rset))
        /* do stdin stuff */;
}
```

