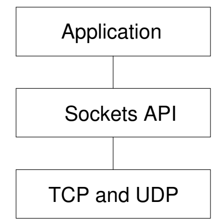
CSE and EE 461
Lecture 10

Autumn Quarter 1997

Unix Sockets

- The Socket family of system calls (initiated in Berkeley Unix and spread to other operating systems) is an API (Application Programming Interface) for TCP and UDP.



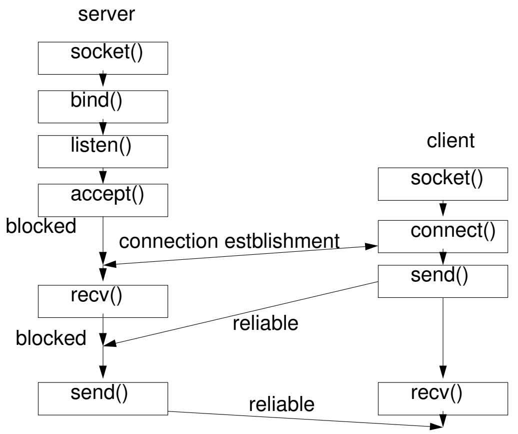
Client-Server Model

- A server is a process on one machine that reacts to messages sent to it by a client that may reside on another machine.
- A client is a process on another machine that desires that the server do something for it on a remote machine.
- The server resides at a socket address that consists of two parts
  1. host name - such as orcas.cs.washington.edu
  2. port number - such as 5001
- A client also resides at a socket address such as port number 6000 at sanjuan.cs.washington.edu.
- Once the client and server know each others socket addresses (in the socket world) then they can use the addresses to send messages to each other.

Socket Calls

- socket() : Creates a local socket endpoint.
- bind() : Used by server to attach a name to a socket endpoint.
- listen() : Used by server specify queue size (TCP).
- accept() : Used by server to block until client attempts connection (TCP).
- connect() : Used by client to attempt to connect to a server (TCP)
- send() : send a message reliably (TCP)
- recv() : receive a message reliably (TCP), blocks until message arrives.
- sendto() : send an message unreliably (UDP)
- recvfrom() : receive a message unreliably (UDP), blocks until message.
- close() : close a connection (TCP)

Connection-Oriented Protocol



Connectionless Protocol