Client-Server Architecture
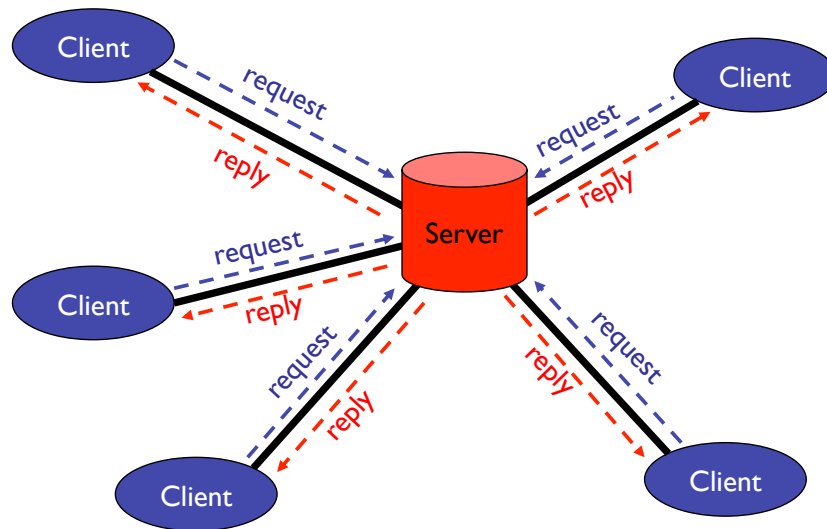

File I/O

```
int open(char* path, int flags, ...)

int read(int fd, void* buf, int nbytes)

int write(int fd, void* buf, int nbytes)

int close(int fd)
```

Socket I/O: socket

```
int socket(int domain, int type,
           int protocol)
```

- **domain = AF_INET**
- **type = SOCK_STREAM**
- **protocol = 0 (or IPPROTO_TCP)**

Socket I/O: bind

```
int bind(int sock, struct sockaddr* addr,
         int addrlen)
```
- **sock = socket file descriptor**
- **addr: see below**
- **addrlen = sizeof(addr)**

```
struct sockaddr_in {
  short sin_family; // --> AF_INET
  u_short sin_port; // --> htons(portnum)
  struct in_addr sin_addr;
        // --> .s_addr = htonl(INADDR_ANY)
}
```

Socket I/O: listen

```
int listen(int sock, int backlog)
```

- **sock = socket file descriptor**
- **backlog = max # of unaccepted connections (e.g. 10)**

Socket I/O: accept

```
int accept(int sock, struct sockaddr* addr,
            int* addrlenp)
```

- **sock = listening socket**
- **addr gets filled in with client info**
- **addrlen = pointer to sizeof(struct sockaddr_in)**
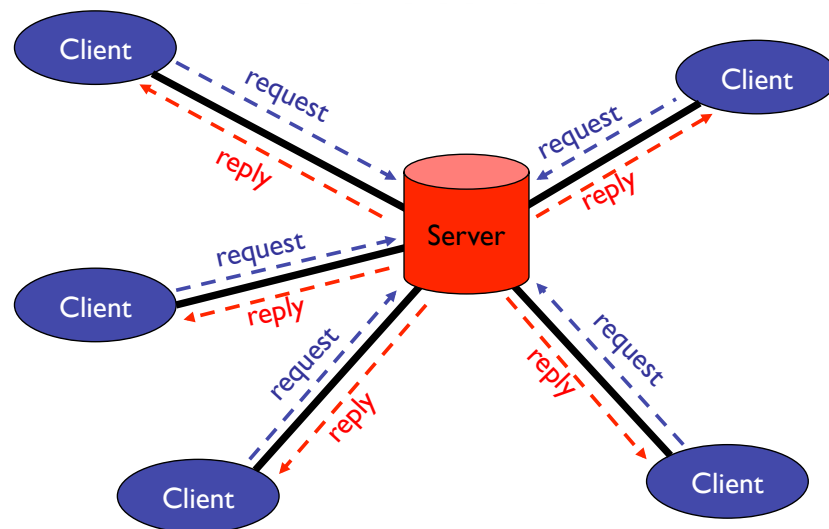
Blocking call!

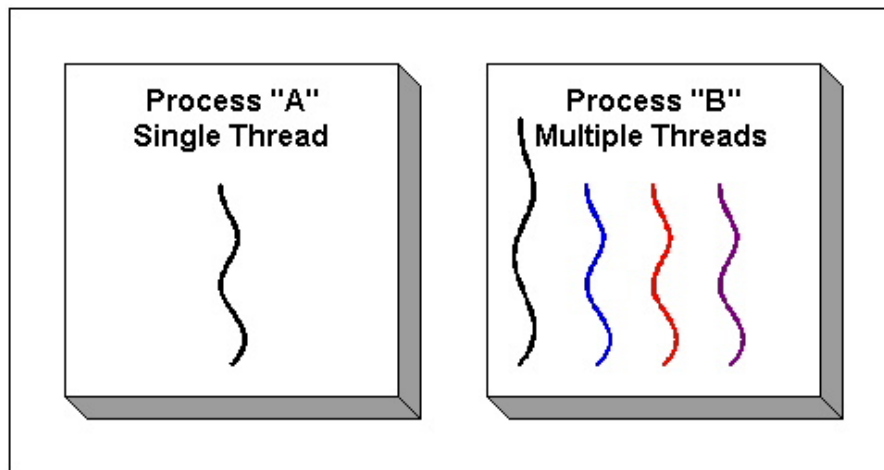# Socket I/O: send/recv

```
int send(int sock, char* msg, int msglen,
         int flags)
```

```
int recv(int sock, char* msg, int msglen,
         int flags)
```

Client-Server Architecture

Processes and Threads

Creating Processes and Threads

- Processes:
  - `int fork()`

- Threads:
```
int pthread_create(pthread_t* thr,
        const pthread_attr_t* attr,
        void* (*start_routine)(void*),
        void* arg)

        attr: usually NULL
        start_routine: function to execute
        arg: argument to function
```