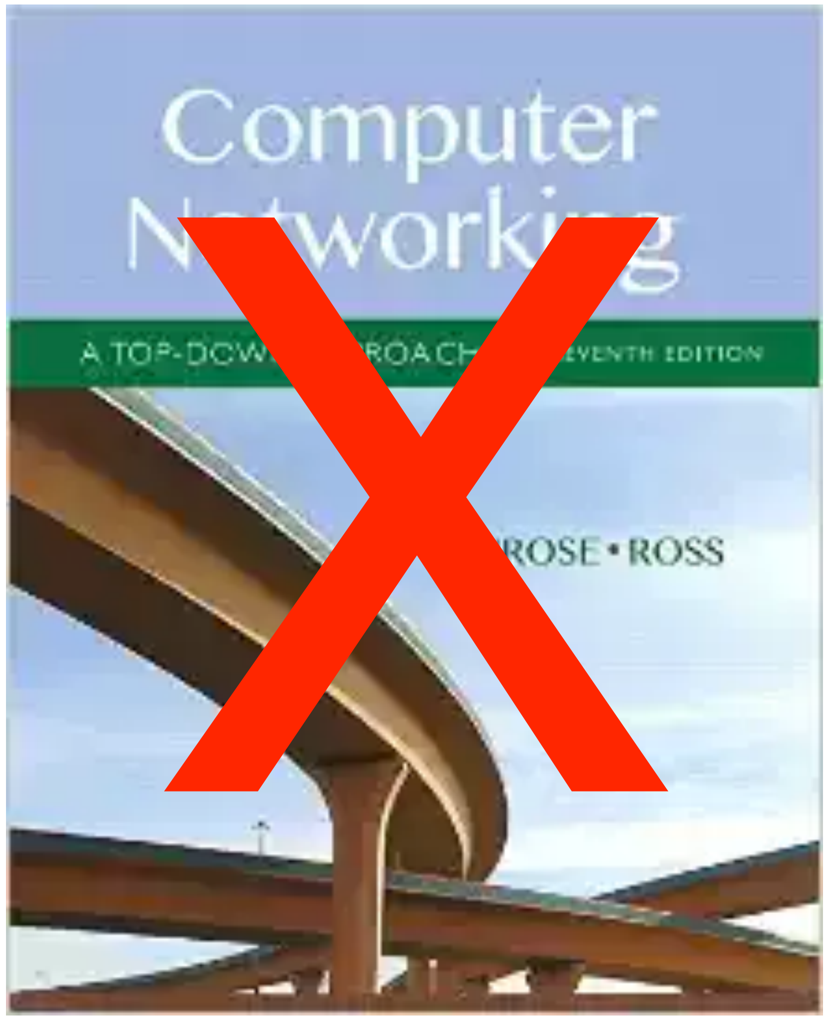
Application Layer
Part 7

Mark Allman
Case / ICSI

EECS 325/425
Fall 2018

"We're the bad news, we're the young guns,
We're the ones they told you to run from"

Reading Along ...

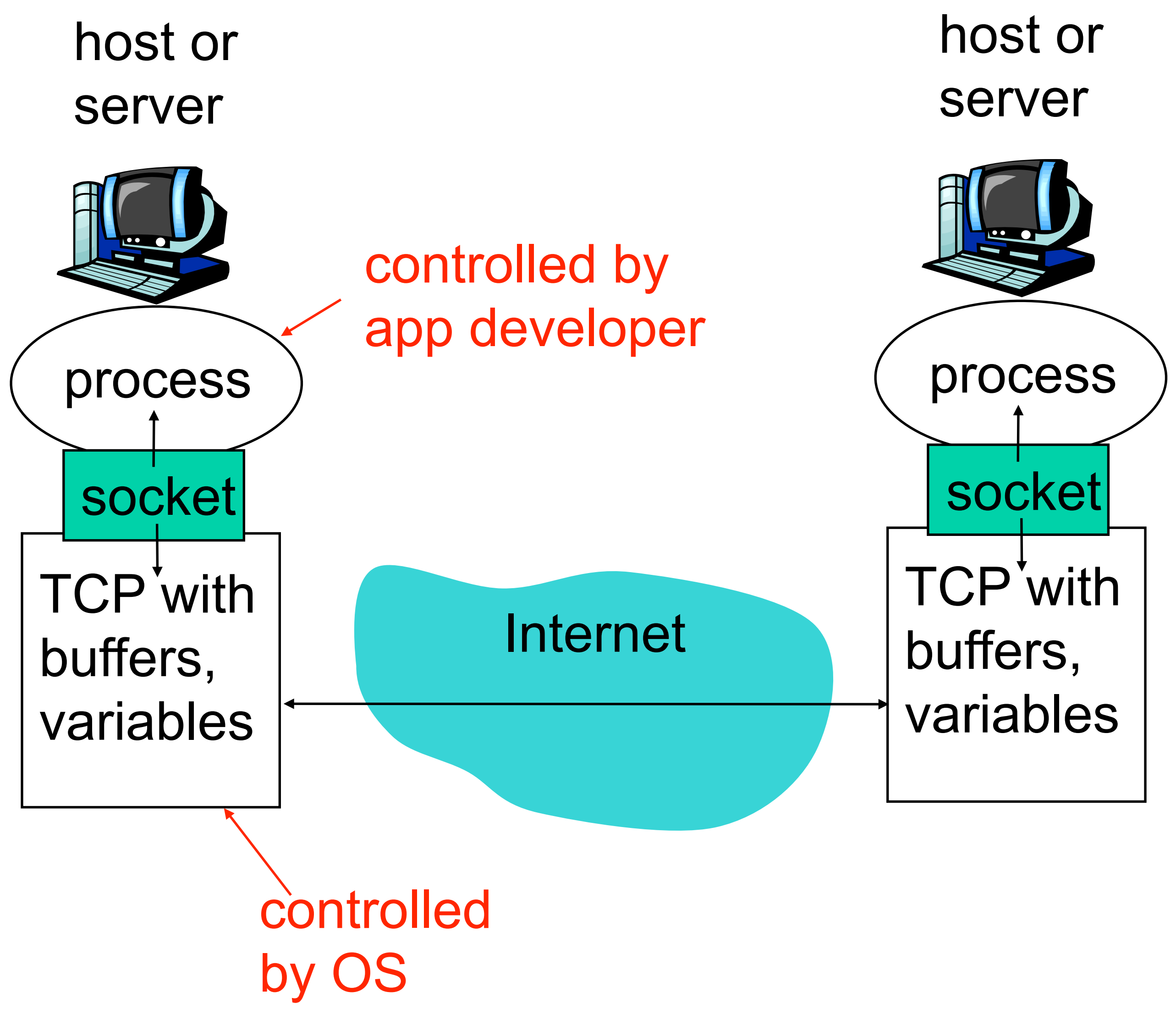


- Sockets programming (server side)

Sockets

Recall that sockets sit between the application process and the transport protocol

Sockets form the glue that allows processes to *interface* with transports (and hence all lower layers)


host or server
controlled by app developer
host or server
process
process
socket
socket
TCP with buffers, variables
Internet
TCP with buffers, variables
controlled by OS

Example TCP Server

```
 write (sd2,msg,strlen (msg));
```

Example TCP Server

```
write (sd2,msg,strlen (msg));
```

Example TCP Server

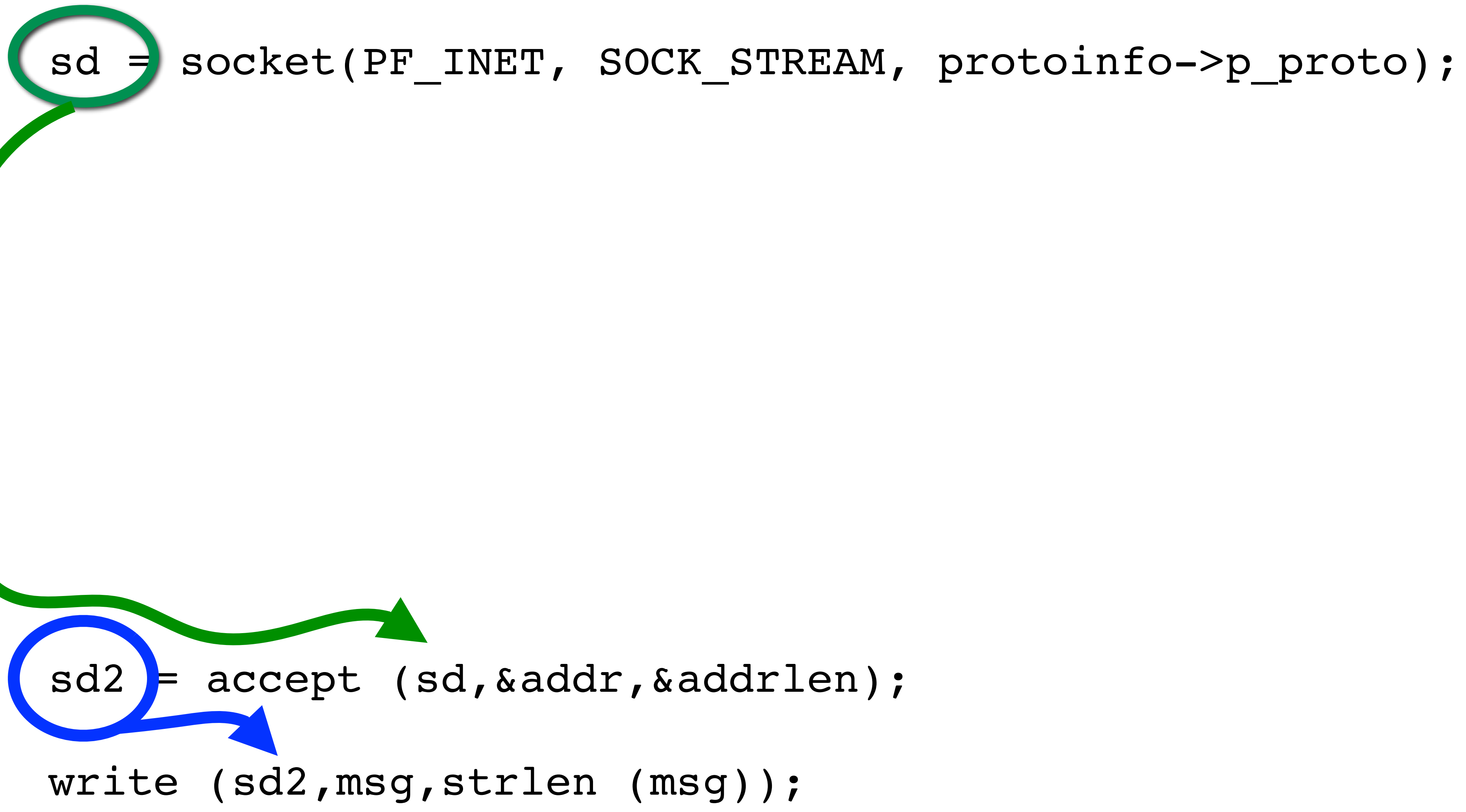
```
sd2 = accept (sd,&addr,&addrlen);

write (sd2,msg,strlen (msg));
```

Example TCP Server

```
sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server



```
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);







sd2 = accept (sd,&addr,&addrlen);

write (sd2,msg,strlen (msg));
```

Example TCP Server

```
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);




sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

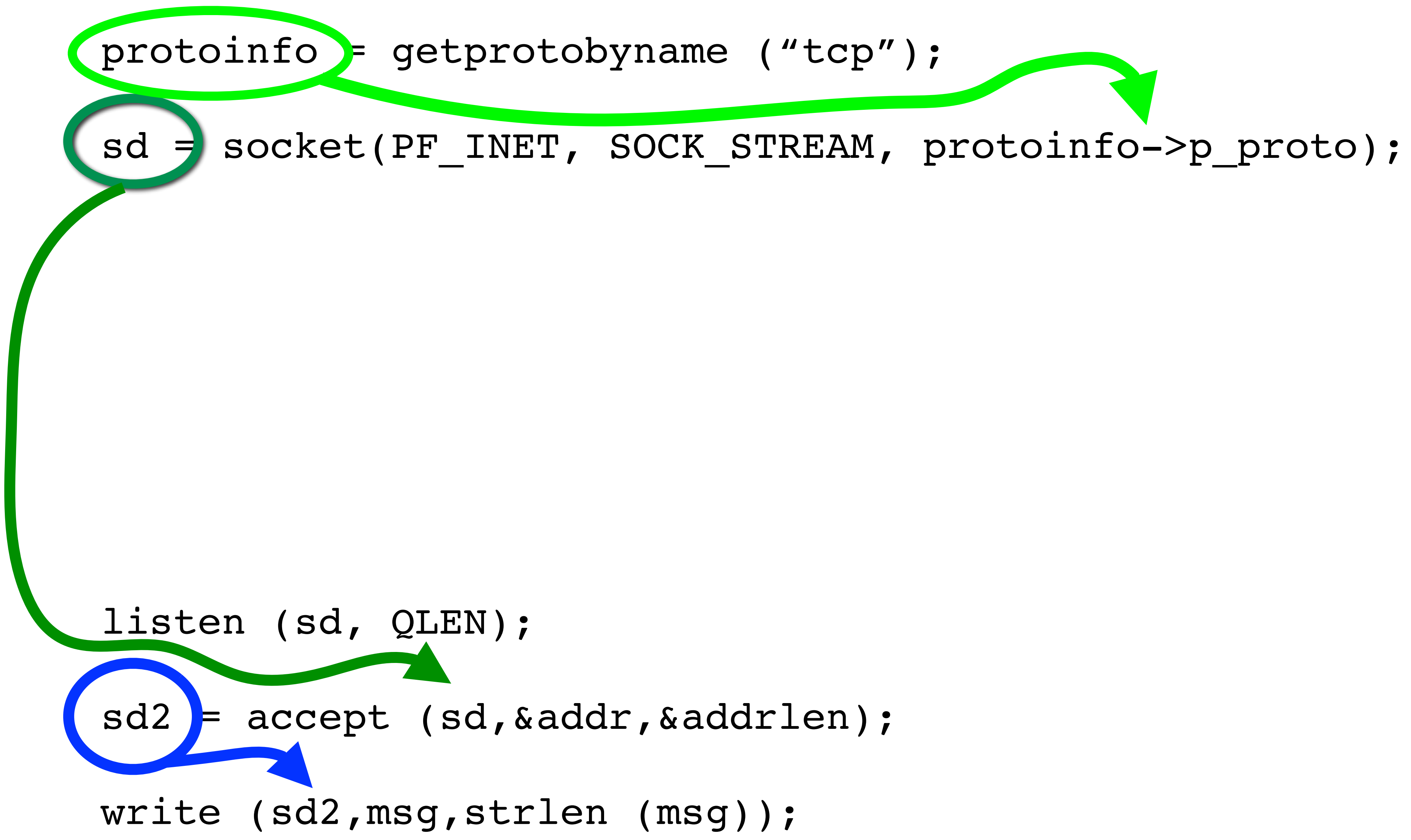
Example TCP Server

```
protoinfo = getprotobyname (“tcp”);
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);



sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server



```
protoinfo = getprotobyname (“tcp”);
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);




listen (sd, QLEN);
sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname (“tcp”);
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);





listen (sd, QLEN);
sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname (“tcp”);
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);




bind (sd, (struct sockaddr *)&sin, sizeof(sin));
listen (sd, QLEN);
sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname (“tcp”);

sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);




bind (sd, (struct sockaddr *)&sin, sizeof(sin));

listen (sd, QLEN);

sd2 = accept (sd,&addr,&addrlen);

write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname (“tcp”);
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);




bind (sd, (struct sockaddr *)&sin, sizeof(sin));
listen (sd, QLEN);
sd2 = accept (sd,&addr,&addrlen);
write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname (“tcp”);

sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);

/* setup endpoint info */
memset ((char *)&sin,0x0,sizeof (sin));
sin.sin_family = AF_INET;
sin.sin_addr.s_addr = INADDR_ANY;
sin.sin_port = htons ((u_short) atoi (argv [PORT_POS]));

bind (sd, (struct sockaddr *)&sin, sizeof(sin));

listen (sd, QLEN);

sd2 = accept (sd,&addr,&addrlen);

write (sd2,msg,strlen (msg));
```

Example TCP Server

```
protoinfo = getprotobyname ("tc
sd = socket(PF_INET, SOCK_S                    ->p_proto);
/* setup endpoint info *
memset ((char *)&sin,0
sin.sin_family = AF
sin.sin_addr.s_ad
sin.sin_port = h                 i (argv [PORT_POS]));

bind (sd, (s                  n, sizeof(sin));

listen (

sd2 = accept               addrlen);

write (sd2,msg,s        en (msg));
```

BAD CODING PRACTICE!

Error Checking

```
sd = socket(PF_INET, SOCK_STREAM, protoinfo->p_proto);
if (sd < 0)
    errexit ("cannot create socket");

if (bind (sd, (struct sockaddr *)&sin, sizeof(sin)) < 0)
    errexit ("cannot bind to port");
```

Socket Example

Socket Example

Server:

```
% hostname
eecslab-5
% ./socketsd 1947 "hello world"
```

Socket Example

Server:

```
% hostname
eecslab-5
% ./socketsd 1947 "hello world"
```

Client:

```
% hostname
eecslab-6
% ./sockets eecslab-5.case.edu 1947
hello world
```