**Linux Sockets – the select() function**

 **/tmp/tmp1**

connect()

**Client1**

 accept()

fd

**Client5**

**Client4**

**Client3**

**Client2**

**Server**

connect()

connect()

connect()

connect()

 master\_fd

fd

The server binds its master socket to the file /tmp/tmp1.

fd

The clients connect to the server through the file /tmp/tmp1.

The server accepts the connections from the clients through the

file /tmp/tmp1.

The server adds each connection to its active\_fd\_set using the

macro FD\_SET.

fd

fd

 **/tmp/tmp1**

**Client1**

 accept()

fd

read()/write()

**Client5**

**Client4**

**Client3**

**Client2**

**Server**

master\_fd

cl[0]

cl[4]

cl[3]

cl[1]

cl[2]

read()/write()

read()/write()

read()/write()

read()/write()

fd

fd

fd

fd

As each connection is made,

the server creates a separate file descriptor for

each client.

The server therefore has its master file descriptor bound

to /tmp/tmp1 to accept any new connections.

The server also contains a file descriptor for each client for

reading and writing.

Each client has only one file descriptor, which is used for

reading and writing.

The server uses the select() function to see if a new client is requesting a connect,

or to determine which of the existing clients has sent it a message.

If a client sends the server a message, the server searches through its active\_fd\_set

using the macro FD\_ISSET to find the client’s file descriptor.