概念1：TCP/IP只是一个协议栈，就像操作系统的运行机制一样，必须要具体实现，同时还要提供对外的操作接口。就像操作系统会提供标准的编程接口，比如Win32编程接口一样，TCP/IP也必须对外提供编程接口，这就是Socket编程接口

概念2：socket跟TCP/IP并没有必然的联系。Socket编程接口在设计的时候，就希望也能适应其他的网络协议。所以，socket的出现只是可以更方 便的使用TCP/IP协议栈而已，其对TCP/IP进行了抽象，形成了几个最基本的函数接口。比如 create，listen，accept，connect，read和write等等。

概念3：HTTP协议即超文本传送协议(Hypertext Transfer Protocol )，是Web联网的基础，也是手机联网常用的协议之一，HTTP协议是建立在TCP协议之上的一种应用。

概念4：绑定的是Listen侦听，Accept是来建立通信的套接字，这样多个客户端建立通信，就多个套接字来分别跟这些客户端通信  
套接字可以理解为一个用来通信的系统句柄

问题1：TCP/IP的协议栈

Telnet实现：

1. 需要用到的报文格式

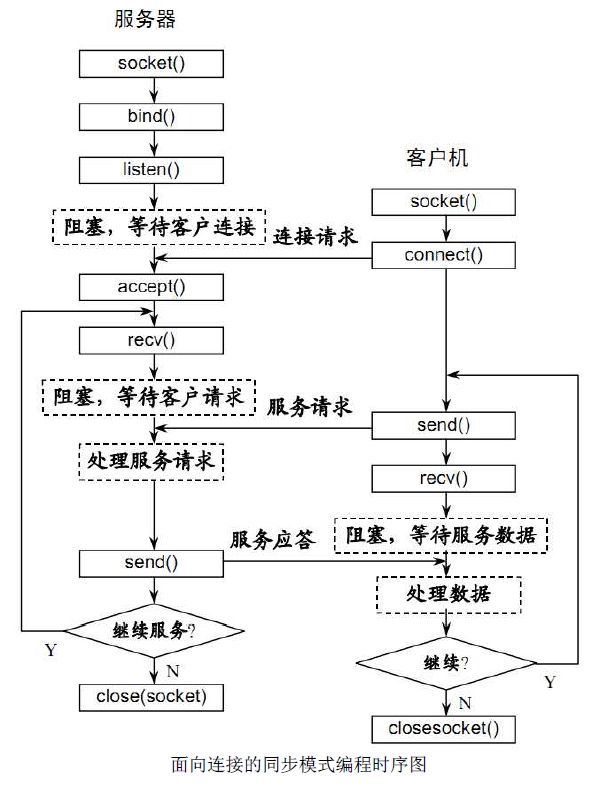
IAC

1. 报文意义
2. 连接使用socket 或 封装完成的fsockopent

Telnet流程：

在连接成功后，服务器会跟客户端进行“协商”，这个过程以ASCII的255开始。协商结束后，才会进行数据的交换。

业务流程



服务器端代码

一，创建监听服务相关

/\*\*

\* 创建socket

\*

\* @param AF\_INET IP4协议

\* @param SOCK\_STREAM 基于网络的通信

\* @param SOL\_TCP TCP协议

\*\*/

$scoket = socket\_create(AF\_INET, SOCK\_STREAM, SOL\_TCP);

/\*\*

\* 设置参数，可设置超时等参数，设置不对有可能获取不到数据，但是不报错。

\* @param SOL\_SOCKET

\* @param SO\_REUSEADDR

\* ------

\*/

//1表示接受所有的数据包

socket\_set\_option($scoket, SOL\_SOCKET, SO\_REUSEADDR, 1);

/\*\*

\* 绑定端口

\* @param socket $server socket套接字

\* @param string $address IP地址

\* @param int $port 端口

\*/

socket\_bind($server, $address, $port);

/\*\*

\* 监听服务

\*/

socket\_listen($server);

-------------------->等待客户端的链接<-----------

1. 获取数据

这里需要用到死循环，因为要不停的获取数据,循环内部

/\*\*

\* 这个函数是同时接受多个连接的关键，我的理解它是为了阻塞程序继续往下

\* 执行。

\*

\* $write,$except可以都为空

\* socket\_select ($sockets, $write = NULL, $except = NULL, NULL);

\* $sockets可以理解为一个数组，这个数组中存放的是文件描述符。当它有变化（就是有新消息到或者有客户端连接/断开）时，socket\_select函数才会返回，继续往下执行。

\* $write是监听是否有客户端写数据，传入NULL是不关心是否有写变化。

\* $except是$sockets里面要被排除的元素，传入NULL是”监听”全部。

\* 最后一个参数是超时时间。

如果为0：则立即结束。

如果为n>1: 则最多在n秒后结束。

如遇某一个连接有新动态，则提前返回，如果为null：如遇某一个连接有新动态，则返回

\*\*/

socket\_select($scoket, $write, $except, NULL);

/\*\*

\* 接收一个socket连接

\*\*/

$client=socket\_accept($scoket);

//读取该socket的信息，注意：第二个参数是引用传参即接收数据，第三个参数是接收数据的长度

$l=socket\_recv($sock,$buf,1000,0);

Socket\_read();

Socket\_recvfrom（）