

并发socket服务器设计

Andrew Huang <bluedrum@163.com>

课程内容:

- I 并发服务器设计
- I 基于多线程并发服务器
 - 简单文件下载服务设计
 - 多线程聊天程序
- I 基于多进程并发服务器

并发服务器设计

并发服务器

- I 网络通讯中一个重要的概念是设计并发服务器.
 - 如果设计一个同时支持多用户访问的Web 服务器
 - 支持多用户在线的游戏服务器
 - ...
- I 并发服务器是对网络开发人员最大挑战
 - 如果支持瞬间大用户在线? 在开发时如何模拟大用户量运行?
 - 如何处理并发处理多用户之类任务切换
 - 服务器往往需要7*24不停机运行, 如何长时间运行不出问题也是开发人员重要任务
 - I 内存leak,内存段错误...

多线程并发服务器设计

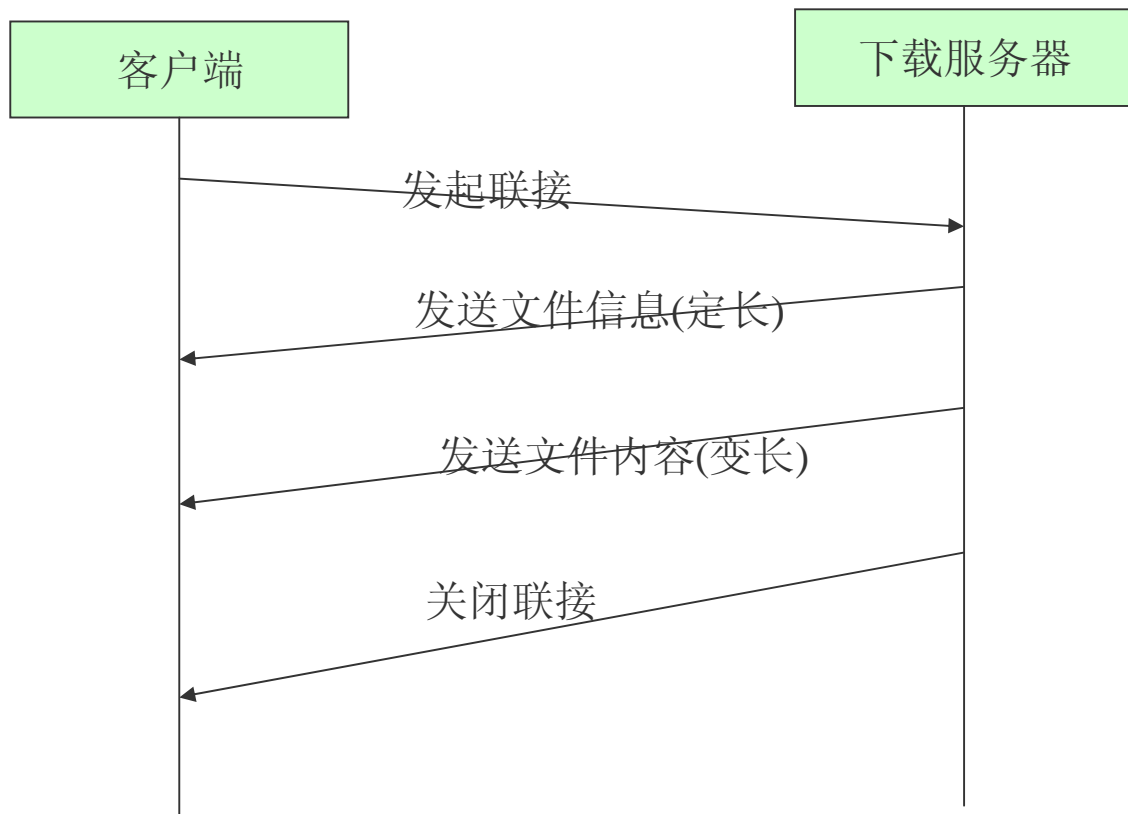
1. 多线程并发服务器

- I 多线程是实现并发一种比较简易和有效的机制. 在很多服务器中, 往往一个客户端在服务器会有专门的线程一对一进行服务.这样可以有效解决并发处理的问题
- I 多线程服务器主要问题在并发访问, 如多个线程去访问同一资源时, 往往需要加锁
- I 线程创建和销毁是非常损耗CPU和内存资源的, 复杂的服务器往往采用线程池(thread pool)机制来重用线程
 - 即一个线程服务完后并不立刻销毁, 而且休眠后加入线程池, 当有任务时, 由软件从线程池激活一个线程来执行

2. 多线程下载服务器

- I 系统需求, 由TCP的服务器去实现.当一TCP客户端向特定的端口发请求时, 服务器将会创建一个线程来处理.

I 下载协议流程如下



- 客户端发起联接
- 在这个线程里，首先会用`fstat`读取某一个文件内容，并将文件信息（最关键的是文件长度）写入一个定长的描述信息的结构里，并立即发送给客户端。
- 客户端收取结构（双方都知道结构长度），并从中读取文件长度和文件名
- 服务器依次读文件，并发往客户端。
- 客户端此时已知文件长度，因为TCP不会丢失数据，因此在本机接收这样长度表示接收完毕

3. 下载协议流程

简单文件下载服务器

- I 文件下载一般采用TCP进行设计,这样的优点程序设计相对简单,防止文件内容在下载时丢失.
- I 由于TCP是不会丢失数据,因此只有二种可能,一种是接收失败,一种是把文件完整接收下来.
- I 这个协议无需考虑客户端之间交互. 各个客户端只是不相干下载文件, 对同一文件并发处理由操作系统文件系统自行处理.

多线程聊天服务器

- I 系统需求,多个客户端通过网络登录到同一服务器上,即一个客户端的发言会通过服务器转发到其它客户端上.
- I 在实际使用聊天服务器当中,有多种实现制,如TCP,UDP等.用TCP实现,服务器监控客户

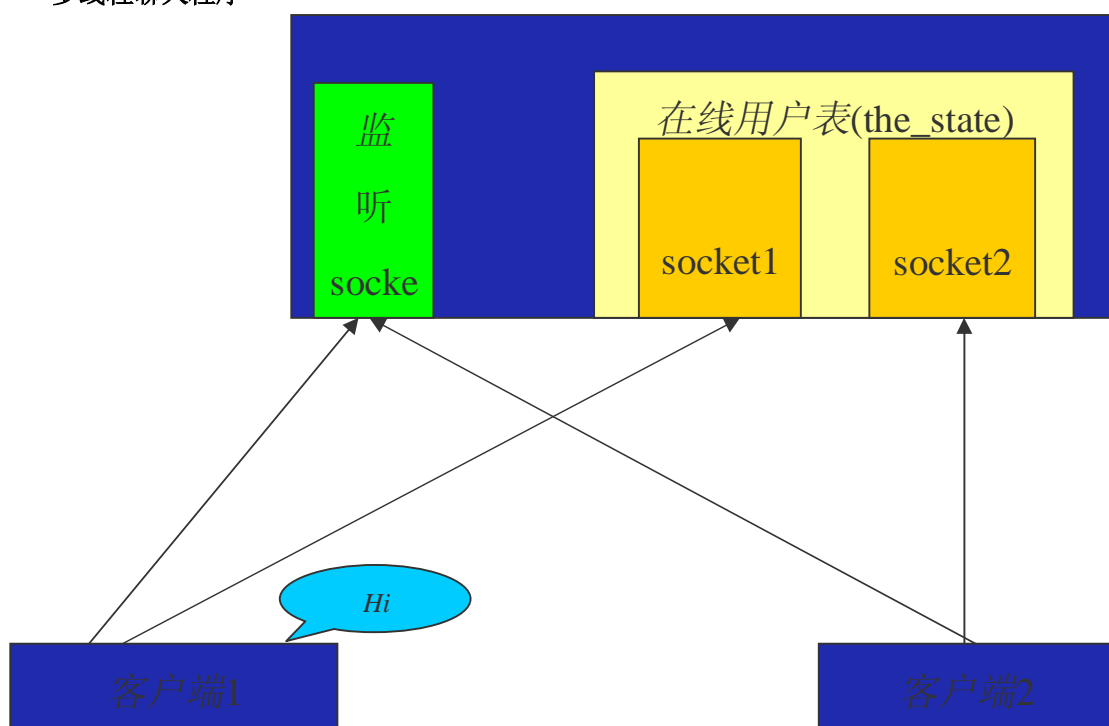
端上线下线比较方便.但TCP不能做信息广播,因此需要服务器自行处理广播.

- 当一个客户端发来信息时.服务器接收后,会把向所有客户端的socket用send发送一次,这样相当于广播,

fd_set 数据结构

- fd_set按字面意思理解是存储fd的一集合,可以理解一个32位的整数数组.它提供了一组宏来操作.
- 多线程聊天室主要采用这一个结构来保存登录的聊天客户端的socket的fd.当一个客户端上线时,在这个数据增加一项,下线时把这客户端的socket从这个数组中移走.
- 在线用户表是多线程互相竞争的资源.因此必须加锁处理.

多线程聊天程序



多进程服务器设计

1. 多进程服务器设计

- Linux下多进程服务器主要使用fork来创建新的进程来并发服务
- 多进程不合适客户端通讯比较多的场合,但合适一些重负荷的应用场合

2. 多进程telnet服务器

- 用telnet作客户端进行登录
- 每一个客户端fork一个进程来创建一个shell来服务.
- 通过重定向标准输出,标准输入和标准输出重定向到远程的socket.就可以实现一个简单

的远程shell程序

I 运行

- ./telnetd -host 192.168.0.146
- 用linux或Windows客户端去处理

dup2():重定向函数

课堂练习

I 把多线程下载改为多进程服务器

I 请将原有简单文件下载服务器,由一次下载一个文件,升级为一次可下多个文件的.

- 多个文件名可由配置文件或直接写在源码里
- 可以依次下载文件,即先发一个文件头,再发一个文件内容.然后再发一个文件头,再发送一文件内容,这样依些类推...