快速通道

文档分类
- 软件作品 <=
- 文档作品
- 代码作品

推荐作品
- HGod 0.51
- HScan v1.20
- trtool2.4.exe for windows,VC6.0 下编译
- 多线程包转发稳定版 HTran V1.00
- NTscan v1.0
- HDoor V0.41

listentty

提交作品： bkbll
提交日期： 2003-07-28
作品属性： 推荐
文档类别： 软件作品
浏览次数： 今3次/总1555次

---

cntoltty的专用客户端, 比使用nc好的地方就是支持vi等,因为他屏蔽了回显,更重要的是支持ctrl+c,ctrl+d.而nc是不可能支持的.

```
/*
    cntoltty's client soft.
*/


#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
#include <fcntl.h>
#include <termios.h>
#include <netdb.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <errno.h>
#include <sys/ioctl.h>
#include <signal.h>

#define BUF     1024
#define INT_KEY 0x05
#define SUS_KEY 0x06
#define QUT_KEY 0x07
int    sock,scli,slen,count,on=1;

int    usage(char *s)
{
    printf("use: %s [port] [-e]\n", s);
    printf("%s -h show this message\r\n",s);
    return 1;
}



int    main(int argc, char *argv[])
{
```

```
    int        ifecho=0,port = 34567;
    struct    sockaddr_in serv,cli;

    unsigned char    buf[BUF];
    fd_set        fds;
    struct    termios    old, new;
    /* input checks */
    if (argc > 1)
    {
        if(strcmp("-h",argv[1])==0)
        {
            usage(argv[0]);
            exit(0);
        }
        port=atoi(argv[1]);
        if(port == 0)
        {
            printf("port must >0\r\n");
            exit(0);
        }
    }
    if ((argc == 3) && (strcmp("-e",argv[2])==0))
            ifecho=1;

    /* create server */

    sock = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
    if (sock < 0)
    {
        perror("socket");
        return 1;
    }
    bzero((char *) &serv, sizeof(serv));
    serv.sin_family = AF_INET;
    serv.sin_addr.s_addr = htonl(INADDR_ANY);
    serv.sin_port = htons(port);
    if(setsockopt(sock,SOL_SOCKET,SO_REUSEADDR,&on,sizeof(on))){perror("set
sockopt");close(sock);exit(0);}
    printf("Listening %d...", port);
    fflush(stdout);
    if (bind(sock, (struct sockaddr *) &serv, sizeof(serv)) < 0) {
        perror("bind");
        return 1;
    }
    if (listen(sock, 1) < 0) {
        perror("listen");
        return 1;
    }

    printf("ok\r\n");
    slen = sizeof(cli);
    scli = accept(sock, (struct sockaddr *) &cli, &slen);
    printf("Accept connect from %s:%d\r\n",inet_ntoa(cli.sin_addr),ntohs(cli.sin_port));
    tcgetattr(0, &old);
    if(ifecho == 0)
    {
        new = old;
        new.c_lflag &= ~(ICANON | ECHO | ISIG |IEXTEN);
        tcsetattr(0, TCSANOW, &new);
        printf("Close echo on this term\r\n");
    }
    else
    {

        new = old;
        new.c_lflag &= ~(ICANON |ISIG);
        /*
        new.c_cc[VINTR]=INT_KEY;
        new.c_cc[VQUIT]=QUT_KEY;
        new.c_cc[VSUSP]=SUS_KEY;
        */
        tcsetattr(0, TCSANOW, &new);
```

```
        printf("Open echo on this term\r\n");
    }
    //signal(SIGINT,SIG_IGN);
    //signal(SIGSTOP,SIG_IGN);
    //signal(SIGTSTP,SIG_IGN);
    while(1)
    {
        FD_ZERO(&fds);
        FD_SET(0, &fds);
        FD_SET(scli, &fds);

        if (select(scli+1, &fds, NULL, NULL, NULL) < 0)
        {
            if (errno == EINTR) continue;
            break;
        }
        if (FD_ISSET(0, &fds))
        {
            count = read(0, buf, BUF);
            if (count <= 0) break;
            if (write(scli, buf, count) <= 0) break;
            memset(buf,0,BUF);
        }
        if (FD_ISSET(scli, &fds))
        {
            count = read(scli, buf, BUF);
            if (count <= 0) break;
            if (write(1, buf, count) <= 0) break;
            memset(buf,0,BUF);
        }

    }
    close(sock);
    tcsetattr(0, TCSAFLUSH, &old);
    printf("\nConnection closed.\n");
    return 0;
}
```

返回上页

Copyright © 2000-2004 **HUC** All Rights Reserved.
中国红客网络技术联盟 www.cnhonker.com
webmaster@cnhonker.net