<https://blog.csdn.net/21aspnet/article/details/6740110>

当linux中的C api函数发生异常时,一般会将errno变量(需include errno.h)赋一个整数值,不同的值表示不同的含义,可以通过查看该值推测出错的原因，在实际编程中用这一招解决了不少原本看来莫名其妙的问题。但是errno是一个数字，代表的具体含义还要到errno.h中去阅读宏定义，而每次查阅是一件很繁琐的事情。有下面几种方法可以方便的得到错误信息  
  (1)void perror(const char \*s)  
函数说明  
perror ( )用来将上一个函数发生错误的原因输出到标准错误(stderr),参数s 所指的字符串会先打印出,后面再加上错误原因 字符串。此错误原因依照全局变量 errno 的值来决定要输出的字符串。  
  (2) char \*strerror(int errno)  
将错误代码转换为字符串错误信息,可以将该字符串和其它的信息组合输出到用户界面例如  
fprintf(stderr,"error in CreateProcess %s, Process ID %d ",strerror(errno),processID)  
注：假设processID是一个已经获取了的整形ID  
  (3)printf("%m", errno);  
另外不是所有的地方发生错误的时候都可以通过error获取错误代码，例如下面的代码段  
/\*注：下面的头文件使用""而没有直接使用尖括号是因为博客大巴中尖括号当作html符号，所以其内部的头文件名字会被直接忽略\*/  
#include"stdio.h"  
#include "stdlib.h"  
#include "errno.h"  
#include "netdb.h"  
#include "sys/types.h"  
#include "netinet/in.h"  
int main (int argc, char \*argv[])  
{  
struct hostent \*h;  
if (argc != 2)  
{  
fprintf (stderr ,"usage: getip address\n");  
exit(1);  
}  
/\* 取得主机信息 \*/  
if((h=gethostbyname(argv[1])) == NULL)  
{  
/\* 如果gethostbyname 失败，则给出错误信息 \*/  
herror(“gethostbyname”);  
exit(1);  
}  
/\* 列印程序取得的信息 \*/  
printf(“Host name : %s\n”, h->h\_name);  
printf(“IP Address : %s\n”, inet\_ntoa (\*((struct in\_addr \*)h->h\_addr))）;  
return 0;  
}  
  
/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
  通过上面的代码可以看到：使用gethostbyname()函数，你不能使用perror()来输出错误信息（因为错误代码存储在 h\_errno 中而不是errno 中。所以，你需要调用herror()函数。  
你简单的传给gethostbyname() 一个机器名（“bbs.tsinghua.edu.cn”），然后就从返回的结构struct hostent 中得到了IP 等其他信息．程序中输出IP 地址的程序需要解释一下：h->h\_addr 是一个char\*，但是inet\_ntoa()函数需要传递的是一个struct in\_addr 结构。所以上面将h->h\_addr 强制转换为struct in\_addr\*，然后通过它得到了所有数据。

errno.h中定义的错误代码值如下：

查 看错误代码errno是调试程序的一个重要方法。当linuc C api函数发生异常时,一般会将errno变量(需include errno.h)赋一个整数值,不同的值表示不同的含义,可以通过查看该值推测出错的原因。在实际编程中用这一招解决了不少原本看来莫名其妙的问题。比较 麻烦的是每次都要去linux源代码里面查找错误代码的含义，现在把它贴出来，以后需要查时就来这里看了。  
以下来自linux 2.4.20-18的内核代码中的/usr/include/asm/errno.h  
#ifndef \_I386\_ERRNO\_H  
#define \_I386\_ERRNO\_H  
#define EPERM   1 /\* Operation not permitted \*/  
#define ENOENT   2 /\* No such file or directory \*/  
#define ESRCH   3 /\* No such process \*/  
#define EINTR   4 /\* Interrupted system call \*/  
#define EIO       5 /\* I/O error \*/  
#define ENXIO   6 /\* No such device or address \*/  
#define E2BIG   7 /\* Arg list too long \*/  
#define ENOEXEC   8 /\* Exec format error \*/  
#define EBADF   9 /\* Bad file number \*/  
#define ECHILD 10 /\* No child processes \*/  
#define EAGAIN 11 /\* Try again \*/  
#define ENOMEM 12 /\* Out of memory \*/  
#define EACCES 13 /\* Permission denied \*/  
#define EFAULT 14 /\* Bad address \*/  
#define ENOTBLK 15 /\* Block device required \*/  
#define EBUSY 16 /\* Device or resource busy \*/  
#define EEXIST 17 /\* File exists \*/  
#define EXDEV 18 /\* Cross-device link \*/  
#define ENODEV 19 /\* No such device \*/  
#define ENOTDIR 20 /\* Not a directory \*/  
#define EISDIR 21 /\* Is a directory \*/  
#define EINVAL 22 /\* Invalid argument \*/  
#define ENFILE 23 /\* File table overflow \*/  
#define EMFILE 24 /\* Too many open files \*/  
#define ENOTTY 25 /\* Not a typewriter \*/  
#define ETXTBSY 26 /\* Text file busy \*/  
#define EFBIG 27 /\* File too large \*/  
#define ENOSPC 28 /\* No space left on device \*/  
#define ESPIPE 29 /\* Illegal seek \*/  
#define EROFS 30 /\* Read-only file system \*/  
#define EMLINK 31 /\* Too many links \*/  
#define EPIPE 32 /\* Broken pipe \*/  
#define EDOM 33 /\* Math argument out of domain of func \*/  
#define ERANGE 34 /\* Math result not representable \*/  
#define EDEADLK 35 /\* Resource deadlock would occur \*/  
#define ENAMETOOLONG 36 /\* File name too long \*/  
#define ENOLCK 37 /\* No record locks available \*/  
#define ENOSYS 38 /\* Function not implemented \*/  
#define ENOTEMPTY 39 /\* Directory not empty \*/  
#define ELOOP 40 /\* Too many symbolic links encountered \*/  
#define EWOULDBLOCK EAGAIN /\* Operation would block \*/  
#define ENOMSG 42 /\* No message of desired type \*/  
#define EIDRM 43 /\* Identifier removed \*/  
#define ECHRNG 44 /\* Channel number out of range \*/  
#define EL2NSYNC 45 /\* Level 2 not synchronized \*/  
#define EL3HLT 46 /\* Level 3 halted \*/  
#define EL3RST 47 /\* Level 3 reset \*/  
#define ELNRNG 48 /\* Link number out of range \*/  
#define EUNATCH 49 /\* Protocol driver not attached \*/  
#define ENOCSI 50 /\* No CSI structure available \*/  
#define EL2HLT 51 /\* Level 2 halted \*/  
#define EBADE 52 /\* Invalid exchange \*/  
#define EBADR 53 /\* Invalid request descriptor \*/  
#define EXFULL 54 /\* Exchange full \*/  
#define ENOANO 55 /\* No anode \*/  
#define EBADRQC 56 /\* Invalid request code \*/  
#define EBADSLT 57 /\* Invalid slot \*/  
#define EDEADLOCK EDEADLK  
#define EBFONT 59 /\* Bad font file format \*/  
#define ENOSTR 60 /\* Device not a stream \*/  
#define ENODATA 61 /\* No data available \*/  
#define ETIME 62 /\* Timer expired \*/  
#define ENOSR 63 /\* Out of streams resources \*/  
#define ENONET 64 /\* Machine is not on the network \*/  
#define ENOPKG 65 /\* Package not installed \*/  
#define EREMOTE 66 /\* Object is remote \*/  
#define ENOLINK 67 /\* Link has been severed \*/  
#define EADV 68 /\* Advertise error \*/  
#define ESRMNT 69 /\* Srmount error \*/  
#define ECOMM 70 /\* Communication error on send \*/  
#define EPROTO 71 /\* Protocol error \*/  
#define EMULTIHOP 72 /\* Multihop attempted \*/  
#define EDOTDOT 73 /\* RFS specific error \*/  
#define EBADMSG 74 /\* Not a data message \*/  
#define EOVERFLOW 75 /\* Value too large for defined data type \*/  
#define ENOTUNIQ 76 /\* Name not unique on network \*/  
#define EBADFD 77 /\* File descriptor in bad state \*/  
#define EREMCHG 78 /\* Remote address changed \*/  
#define ELIBACC 79 /\* Can not access a needed shared library \*/  
#define ELIBBAD 80 /\* Accessing a corrupted shared library \*/  
#define ELIBSCN 81 /\* .lib section in a.out corrupted \*/  
#define ELIBMAX 82 /\* Attempting to link in too many shared libraries \*/  
#define ELIBEXEC 83 /\* Cannot exec a shared library directly \*/  
#define EILSEQ 84 /\* Illegal byte sequence \*/  
#define ERESTART 85 /\* Interrupted system call should be restarted \*/  
#define ESTRPIPE 86 /\* Streams pipe error \*/  
#define EUSERS 87 /\* Too many users \*/  
#define ENOTSOCK 88 /\* Socket operation on non-socket \*/  
#define EDESTADDRREQ 89 /\* Destination address required \*/  
#define EMSGSIZE 90 /\* Message too long \*/  
#define EPROTOTYPE 91 /\* Protocol wrong type for socket \*/  
#define ENOPROTOOPT 92 /\* Protocol not available \*/  
#define EPROTONOSUPPORT 93 /\* Protocol not supported \*/  
#define ESOCKTNOSUPPORT 94 /\* Socket type not supported \*/  
#define EOPNOTSUPP 95 /\* Operation not supported on transport endpoint \*/  
#define EPFNOSUPPORT 96 /\* Protocol family not supported \*/  
#define EAFNOSUPPORT 97 /\* Address family not supported by protocol \*/  
#define EADDRINUSE 98 /\* Address already in use \*/  
#define EADDRNOTAVAIL 99 /\* Cannot assign requested address \*/  
#define ENETDOWN 100 /\* Network is down \*/  
#define ENETUNREACH 101 /\* Network is unreachable \*/  
#define ENETRESET 102 /\* Network dropped connection because of reset \*/  
#define ECONNABORTED 103 /\* Software caused connection abort \*/  
#define ECONNRESET 104 /\* Connection reset by peer \*/  
#define ENOBUFS 105 /\* No buffer space available \*/  
#define EISCONN 106 /\* Transport endpoint is already connected \*/  
#define ENOTCONN 107 /\* Transport endpoint is not connected \*/  
#define ESHUTDOWN 108 /\* Cannot send after transport endpoint shutdown \*/  
#define ETOOMANYREFS 109 /\* Too many references: cannot splice \*/  
#define ETIMEDOUT 110 /\* Connection timed out \*/  
#define ECONNREFUSED 111 /\* Connection refused \*/  
#define EHOSTDOWN 112 /\* Host is down \*/  
#define EHOSTUNREACH 113 /\* No route to host \*/  
#define EALREADY 114 /\* Operation already in progress \*/  
#define EINPROGRESS 115 /\* Operation now in progress \*/  
#define ESTALE 116 /\* Stale NFS file handle \*/  
#define EUCLEAN 117 /\* Structure needs cleaning \*/  
#define ENOTNAM 118 /\* Not a XENIX named type file \*/  
#define ENAVAIL 119 /\* No XENIX semaphores available \*/  
#define EISNAM 120 /\* Is a named type file \*/  
#define EREMOTEIO 121 /\* Remote I/O error \*/  
#define EDQUOT 122 /\* Quota exceeded \*/  
#define ENOMEDIUM 123 /\* No medium found \*/  
#define EMEDIUMTYPE 124 /\* Wrong medium type \*/  
#endif

同时也可以使用strerror（）来自己翻译  
如：  
#include <stdio.h>  
#include <string.h>  
#include <errno.h>   
int main(void)  
{  
int   fd;  
extern int errno;

if((fd = open("/dev/dsp",O\_WRONLY)) < 0)  
{  
printf("errno=%d\n",errno);   
char \* mesg = strerror(errno);  
printf("Mesg:%s\n",mesg);   
}  
  
exit(0);  
}  
dsp设备忙的话将输出如下：  
errno=16  
Mesg:Device or resource busy

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/21aspnet/article/details/6740110