linux错误码大全

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。

1-34号错误号是在内核源码的include/asm-generic/errno-base.h定义  
  
  
35-132则是在include/asm-generic/errno.h中定义  
  
  
剩下还有一些更大的错误号是留给内核级别的，如系统调用等，用户程序一般是看不见的这些号的，Ubuntu9.10中/usr/src/linux-headers-2.6.31-21-generic/include/linux/errno.h

#ifndef \_ASM\_GENERIC\_ERRNO\_BASE\_H

#define \_ASM\_GENERIC\_ERRNO\_BASE\_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    /＊ Argument 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 ＊/

#endif

#include <asm-generic/errno-base.h>

#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 ＊/

#define    ECANCELED    125    /＊ Operation Canceled ＊/

#define    ENOKEY        126    /＊ Required key not available ＊/

#define    EKEYEXPIRED    127    /＊ Key has expired ＊/

#define    EKEYREVOKED    128    /＊ Key has been revoked ＊/

#define    EKEYREJECTED    129    /＊ Key was rejected by service ＊/

/＊ for robust mutexes ＊/

#define    EOWNERDEAD    130    /＊ Owner died ＊/

#define    ENOTRECOVERABLE    131    /＊ State not recoverable ＊/

#define ERFKILL        132    /＊ Operation not possible due to RF-kill ＊/

#ifdef \_\_KERNEL\_\_

/＊

 ＊ These should never be seen by user programs. To return one of ERESTART＊

 ＊ codes, signal\_pending() MUST be set. Note that ptrace can observe these

 ＊ at syscall exit tracing, but they will never be left for the debugged user

 ＊ process to see.

 ＊/

#define ERESTARTSYS    512

#define ERESTARTNOINTR    513

#define ERESTARTNOHAND    514    /＊ restart if no handler.. ＊/

#define ENOIOCTLCMD    515    /＊ No ioctl command ＊/

#define ERESTART\_RESTARTBLOCK 516 /＊ restart by calling sys\_restart\_syscall ＊/

/＊ Defined for the NFSv3 protocol ＊/

#define EBADHANDLE    521    /＊ Illegal NFS file handle ＊/

#define ENOTSYNC    522    /＊ Update synchronization mismatch ＊/

#define EBADCOOKIE    523    /＊ Cookie is stale ＊/

#define ENOTSUPP    524    /＊ Operation is not supported ＊/

#define ETOOSMALL    525    /＊ Buffer or request is too small ＊/

#define ESERVERFAULT    526    /＊ An untranslatable error occurred ＊/

#define EBADTYPE    527    /＊ Type not supported by server ＊/

#define EJUKEBOX    528    /＊ Request initiated, but will not complete before timeout ＊/

#define EIOCBQUEUED    529    /＊ iocb queued, will get completion event ＊/

#define EIOCBRETRY    530    /＊ iocb queued, will trigger a retry ＊/

#endif