【Linux Programming】Day16

Tags	
Date	@June 8, 2022
Summary	Errors and /proc

【Ch3】Work with Files

3.8 Errors

When many of the system calls and functions fail, they indicate the reason for their failure by setting the value of the external variable `errno`.

The values and meanings of the errors are listed in the header file `errno.h`. They include

- `EPERM`: Operation not permitted
- `ENOENT`: No such file or directory
- `EINTR`: Interrupted system call
- `EIO`: I/O Error
- `EBUSY`: Device or resource busy
- `EEXIST`: File exists
- `EINVAL`: Invalid argument
- `EMFILE`: Too many open files
- `ENODEV`: No such device
- `EISDIR`: Is a directory
- `ENOTDIR`: Isn't a directory

3.8.1 strerror

The `strerror` function maps an error number into a string describing the type of error that has occurred.

```
#include <string.h>

char *strerror(int errnum);
```

3.8.2 perror

The `perror` function also maps the current error, as reported in `errno`, into a string and prints it on the standard error stream.

It's preceded by the message given in the string s(if not NULL), followed by a colon and a space.

```
#include <stdio.h>

void perror(const char *s);
```

For example,

```
perror("Program");
```

might give the following on the standard error input:

```
program: Too many open files.
```

3.9 The /proc File System

Linux provides a special file system, `procfs`, that is usually made available as the directory `/proc`. It contains many special files that allow higher-level access to driver and kernel information.

In many cases, the files can just be read and will give status information. For example, `/proc/cpuinfo` gives details of the processors available;

```
$ cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Pentium(R) 4 CPU 2.66GHz
stepping        : 8
cpu MHz         : 2665.923
cache size      : 512 KB
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 2
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat
pse36 clflush dts acpi mmx fxsr sse sse2 ss up
bogomips        : 5413.47
clflush size    : 64
```

We can find more information from specific kernel functions in subdirectories of /proc. For example, we can get network socket usage statistics from `/proc/net/sockstat`:

```
$ cat /proc/net/sockstat
sockets: used 285
TCP: inuse 4 orphan 0 tw 0 alloc 7 mem 1
UDP: inuse 3
UDPLITE: inuse 0
RAW: inuse 0
FRAG: inuse 0 memory 0
```