

opendir: 打开目录

头文件: sys/types.h dirent.h

函数定义: DIR \*opendir(const char \*name);

说明: opendir()用来打开参数name指定的目录, 并返回DIR\*形态的目录流, 和open()类似, 接下来对目录的读取和搜索都要使用此返回值. 成功则返回DIR\*型态的目录流, 打开失败则返回NULL.

错误代码:

EACCESS 权限不足

EMFILE 已达到进程可同时打开的文件数上限

ENFILE 已达到系统可同时打开的文件数上限

ENOTDIR 参数name非真正的目录

ENOENT 参数name指定的目录不存在, 或是参数name为一空字符串

ENOMEM 核心内存不足

readdir: 读取目录

头文件: sys/types.h dirent.h

定义函数: struct dirent \*readdir(DIR \*dir);

说明: readdir()返回参数dir目录流的下个目录进入点. 结构dirent定义如下:

```
struct dirent{
```

```
ino_t d_ino;
```

```
ff_t d_off;
```

```
signed short int d_reclen;
```

```
unsigned char d_type;
```

```
char d_name[256];
```

```
};
```

d\_ino 此目录进入点的inode

d\_off 目录文件开头至此目录进入点的位移

d\_reclen \_name的长度, 不包含NULL字符

d\_type d\_name所指的文件类型

d\_name 文件名

成功则返回下个目录进入点, 有错误发生或读取到目录文件尾则返回NULL. EBADF参数dir为无效的目录流.

应用举例:

```
#include <stdio.h>
```

```
#include <unistd.h>
```

```
#include <sys/types.h>
```

```
#include <dirent.h>
```

```
int main(void)
```

```
{
```

```
DIR *dir;
```

```
struct dirent *ptr;
int i;
dir = opendir("/etc/rc.d");
while((ptr = readdir(dir)) != NULL)
{
printf("d_name: %s\n", ptr->d_name);
}
return 0;
}
```

运行结果:

```
d_name: rc1.d
d_name: rc3.d
d_name: rc
d_name: rc2.d
d_name: rc0.d
d_name: rc.sysinit
d_name: ..
d_name: rc6.d
d_name: rc4.d
d_name: init.d
d_name: rc.local
d_name: rc5.d
d_name:
```