

10.6 读取文件元数据

应用程序能够通过调用 `stat` 和 `fstat` 函数，检索到关于文件的信息（有时也称为文件的元数据(metadata)）。

```
#include <unistd.h>
#include <sys/stat.h>

int stat(const char *filename, struct stat *buf);
int fstat(int fd, struct stat *buf);
```

返回：若成功则为 0，若出错则为 -1。

`stat` 函数以一个文件名作为输入，并填写如图 10-9 所示的一个 `stat` 数据结构中的各个成员。`fstat` 函数是相似的，只不过是以文件描述符而不是文件名作为输入。当我们在 11.5 节中讨论 Web 服务器时，会需要 `stat` 数据结构中的 `st_mode` 和 `st_size` 成员，其他成员则不在我们的讨论之列。

```
----- statbuf.h (included by sys/stat.h)

/* Metadata returned by the stat and fstat functions */
struct stat {
    dev_t      st_dev;      /* Device */
    ino_t      st_ino;      /* inode */
    mode_t     st_mode;     /* Protection and file type */
    nlink_t    st_nlink;    /* Number of hard links */
    uid_t      st_uid;      /* User ID of owner */
    gid_t      st_gid;      /* Group ID of owner */
    dev_t      st_rdev;     /* Device type (if inode device) */
    off_t      st_size;     /* Total size, in bytes */
    unsigned long st_blksize; /* Block size for filesystem I/O */
    unsigned long st_blocks; /* Number of blocks allocated */
    time_t     st_atime;    /* Time of last access */
    time_t     st_mtime;    /* Time of last modification */
    time_t     st_ctime;    /* Time of last change */
};

----- statbuf.h (included by sys/stat.h)
```

图 10-9 `stat` 数据结构

`st_size` 成员包含了文件的字节数大小。`st_mode` 成员则编码了文件访问许可位(图 10-2)和文件类型(10.2 节)。Linux 在 `sys/stat.h` 中定义了宏谓词来确定 `st_mode` 成员的文件类型：

`S_ISREG(m)`。这是一个普通文件吗？

`S_ISDIR(m)`。这是一个目录文件吗？

`S_ISSOCK(m)`。这是一个网络套接字吗？

图 10-10 展示了我们会如何使用这些宏和 `stat` 函数来读取和解释一个文件的 `st_mode` 位。