

第六章 系统数据文件和信息

1. 口令文件

Wuhan University

- /etc/passwd

说 明	struct passwd成员	POSIX.1
用户名	char *pw_name	•
加密口令	char *pw_passwd	
数值用户ID	uid_t pw_uid	•
数值组ID	gid_t pw_gid	•
注释字段	char *pw_gecos	
初始工作目录	char *pw_dir	•
初始shell (用户程序)	char *pw_shell	•

1. 口令文件

Wuhan University

- /etc/passwd

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

```
#include <pwd.h>
```

```
struct passwd *getpwuid(uid_t uid);
```

```
struct passwd *getpwnam(const char *name);
```

1. 口令文件

- /etc/passwd

```
#include <sys/types.h>
#include <pwd.h>
struct passwd *getpwent(void);
void setpwent(void);
void endpwent(void);
```

```
struct passwd *ptr;

setpwent();
while ( (ptr = getpwent()) != NULL) {
    if (strcmp(name, ptr->pw_name) == 0)
        break;      /* found a match */
}
endpwent();
```


2. 阴影口令

Wuhan University

- `/etc/shadow`

3. 组文件

Wuhan University

- /etc/group

说 明	struct group成员	POSIX.1
组名	char *gr_name	•
加密口令	char *gr_passwd	
数字组ID	int gr_gid	•
指向各用户名指针的数组	char **gr_mem	•

3. 组文件

- /etc/group

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

```
#include <grp.h>
```

```
struct group *getgrgid(gid_t gid);
```

```
struct group *getgrnam(const char *gname);
```

```
struct group *getgrent(void);
```

```
void setgrent(void);
```

```
void endgrent(void);
```

4. 添加组ID

Wuhan University

- /etc/group

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

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

```
int getgroups(int gidsetsize, gid_t grouplist[]);
```

```
int setgroups(int ngroups, const gid_t grouplist[]);
```

```
int initgroups(const char *username, gid_t basegid);
```


5. 其他数据文件

Wuhan University

- /etc/services
- /etc/protocols
- /etc/networks

说 明	数据文件	头 文 件	结 构	附加的关键字搜索函数
口令	/etc/passwd	<pwd.h>	passwd	getpwnam, getpwuid
组	/etc/group	<grp.h>	group	getgrnam, getgrgid
主机	/etc/hosts	<netdb.h>	hostent	gethostbyname, gethostbyaddr
网络	/etc/networks	<netdb.h>	netent	getnetbyname, getnetbyaddr
协议	/etc/protocols	<netdb.h>	protoent	getprotobyname, getprotobynumber
服务	/etc/services	<netdb.h>	servent	getservbyname, getservbyport

6. 登录会计

Wuhan University

- utmp文件，记录当前登录系统的用户
- wtmp文件，跟踪登录和注销事件

```
struct utmp {  
    char    ut_line[8];    /* tty line: "ttyh0", "ttyd0", "ttyp0", ... */  
    char    ut_name[8];    /* login name */  
    long    ut_time;       /* seconds since Epoch */  
};
```

7. 系统标识

- 系统标识

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

```
int uname(struct utsname *name);
```

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

```
int gethostname(char *name, int namelen);
```

```
struct utsname {  
    char sysname[9];    /* name of the operating system */  
    char nodename[9];   /* name of this node */  
    char release[9];    /* current release of operating system */  
    char version[9];    /* current version of this release */  
    char machine[9];    /* name of hardware type */  
};
```

8. 时间和日期例程

Wuhan University

- 日期

```
#include <time.h>
```

```
time_t time(time_t *calptr);
```

```
struct tm *gmtime(const time_t *calptr);
```

```
struct tm *localtime(const time_t *calptr);
```

```
time_t mktime(struct tm *tmptr);
```

```
char *asctime(const struct tm *tmptr);
```

```
char *ctime(const time_t *calptr);
```


8. 时间和日期例程

Wuhan University

- 日期

```
#include <time.h>
size_t strftime(char *buf, size_t maxsize, const char *format,
                const struct tm *tm_ptr);
```

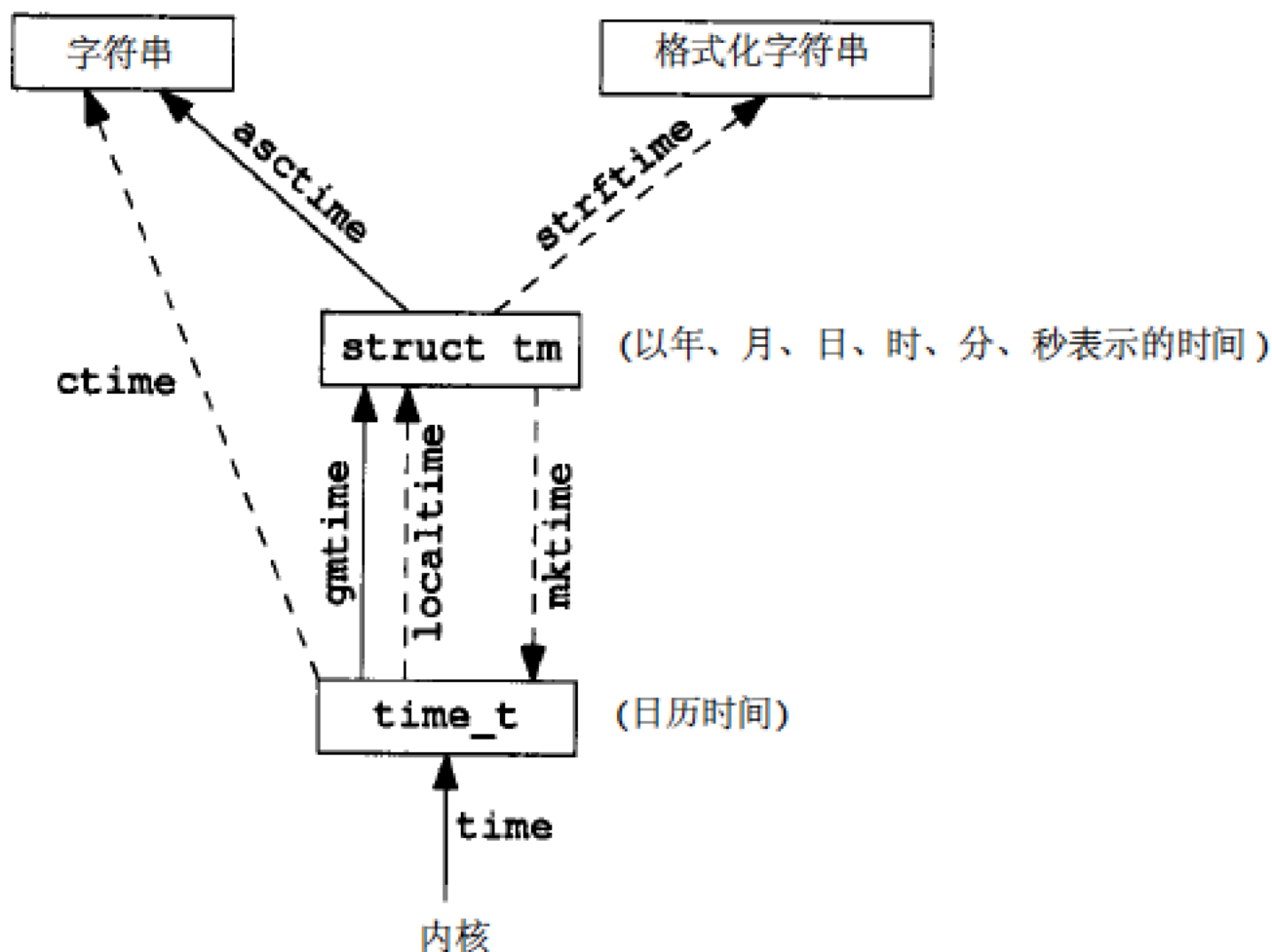
8. 时间和日期例程

Wuhan University

格 式	说 明	例 子
%a	缩写的周日名	Tue
%A	全周日名	Tuesday
%b	缩写的月名	Jan
%B	月全名	January
%c	日期和时间	Tue Jan 14 19:40:30 1992
%d	月日: [01, 31]	14
%H	小时 (每天24小时): [00, 23]	19
%I	小时 (上、下午各12小时): [01, 12]	07
%j	年日: [001, 366]	014
%m	月: [01, 12]	01
%M	分: [00, 59]	40
%p	AM/PM	PM
%S	秒: [00, 61]	30
%U	星期日周数: [00, 53]	02
%w	周日: [0=星期日, 6]	2
%W	星期一周数: [00, 53]	02
%x	日期	01/14/92
%X	时间	19:40:30
%y	不带公元的年: [00, 991]	92
%Y	带公元的年	1992
%Z	时区名	MST

8. 时间和日期例程

Wuhan University



8. 时间和日期例程

Wuhan University

- 日期

```
struct tm { /* a broken-down time */
    int  tm_sec; /* seconds after the minute: [0, 61] */
    int  tm_min; /* minutes after the hour: [0, 59] */
    int  tm_hour; /* hours after midnight: [0, 23] */
    int  tm_mday; /* day of the month: [1, 31] */
    int  tm_mon; /* month of the year: [0, 11] */
    int  tm_yday; /* days since January 1: [0, 365] */
    int  tm_isdst; /* daylight saving time flag: <0, 0, >0 */
};
```