

五、系統相關檔案與資訊

取得/etc/passwd的單一筆資料

```
#include <pwd.h>
#include <sys/types.h>
struct passwd *getpwnam(const char *name);
struct passwd *getpwuid(uid_t uid);
```

```
struct passwd {
    char *pw_name;      /* user name */
    char *pw_passwd;    /* user password */
    uid_t pw_uid;       /* user id */
    gid_t pw_gid;       /* group id */
    char *pw_gecos;     /* real name */
    char *pw_dir;       /* home directory */
    char *pw_shell;     /* shell program */
};
```

- getpwnam與getpwuid函式用來取得/etc/passwd中的資料，前者使用login name，後者使用uid

瀏覽整個/etc/passwd

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

struct passwd *getpwent(void);
void setpwent(void);
void endpwent(void);
```

- setpwent函式用來開啟/etc/passwd，並將讀取位置移到第一筆資料
- getpwent函式傳回下一筆資料
- endpwent函式用來關閉/etc/passwd

取得/etc/group的資料

```
#include <grp.h>
#include <sys/types.h>
struct group *getgrnam(const char *name);
struct group *getgrgid(gid_t gid);
```

```
struct group {
    char *gr_name; /* group name */
    char *gr_passwd; /* group password */
    gid_t gr_gid; /* group id */
    char **gr_mem; /* group members */
};
```

- getgrnam與getgrgid函式用來取得/etc/group中的單一筆資料，前者使用group name，後者使用gid。如果要瀏覽整個/etc/group，可以用下列函式，其用法與上一頁相同

```
#include <grp.h>
#include <sys/types.h>
struct group *getgrent(void);
void setgrent(void);
void endgrent(void);
```

取得系統的版本資訊

```
#include <sys/utsname.h>
int uname(struct utsname *buf);
```

```
struct utsname {
    char sysname[];
    char nodename[];
    char release[];
    char version[];
    char machine[];
};
```

- 可以用uname函式取得系統的版本資訊
 - sysname為作業系統的名稱(例如Linux) ,
 - nodename為host的名稱(例如a.cis.nctu.edu.tw)
 - release為kernel的版本編號(例如2.4.18-14)
 - version為此release的版本編號(例如#1 Wed Sep 4 13:35:50 EDT 2002)
 - Machine為機器的架構(例如i686)

時間與日期函式(1/2)

```
#include <time.h>
time_t time(time_t *t);
struct tm *gmtime(const time_t *timep);    //UTC時間
struct tm *localtime(const time_t *timep);  //當地時間
time_t mktime(struct tm *tm);
char *asctime(const struct tm *tm);
char *ctime(const time_t *timep);
```

```
struct tm {
    int    tm_sec;        /* seconds */
    int    tm_min;        /* minutes */
    int    tm_hour;       /* hours */
    int    tm_mday;       /* day of the month */
    int    tm_mon;        /* month */
    int    tm_year;       /* year */
    int    tm_wday;       /* day of the week */
    int    tm_yday;       /* day in the year */
    int    tm_isdst;      /* daylight saving time */
};
```

時間與日期函式(2/2)

