

《Linux 操作系统与 shell 编程实验》

实验 1

专业：计算机科学与技术 班级：2021级
姓名：杜 学号：202101010101

一、 实验内容：who 命令和 C 语言实现

二、 实验步骤

1. 以普通账户登录，输入命令 who, 理解名称输出的内容
2. 按 Ctrl + Alt +F2, 以 root 账户登录
3. 按 Ctrl + Alt +F7, 再次输入命令 who, 观看输出的变化
4. who 命令是一个应用程序，如何用 C 语言实现这样一个应用？
5. 账户登录信息保存在文件 /var/run/utmp、/var/log/wtmp 中，这是二进制文件，使用到了 utmp 结构体，这个结构体的定义在文件 utmp.h 中
6. 以普通账户身份，使用 find 命令，

```
find / -name utmp.h
```

从根/开始搜索文件 utmp.h，搜索的时候出现很多标准错误输出，为什么？

答：

如果以 root 账户执行这条命令，会不会出现标准错误输出？为什么？

答：

7. 如果不想看到太多的标准错误输出刷屏，这样做

```
find / -name utmp.h 2>/dev/null
```

8. 用 more 命令打开文件 utmp.h，应该能看到下面的定义

```

struct utmp {
    short    ut_type;           /* Type of record */
    pid_t    ut_pid;           /* PID of login process */
    char      ut_line[UT_LINESIZE]; /* Device name of tty - "/dev/" */
    char      ut_id[4];         /* Terminal name suffix,
                                or inittab(5) ID */

    char      ut_user[UT_NAMESIZE]; /* Username */
    char      ut_host[UT_HOSTSIZE]; /* Hostname for remote login, or
                                kernel version for run-level
                                messages */

    struct    exit_status ut_exit; /* Exit status of a process
                                marked as DEAD_PROCESS; not
                                used by Linux init(8) */

    /* The ut_session and ut_tv fields must be the same size when
       compiled 32- and 64-bit. This allows data files and shared
       memory to be shared between 32- and 64-bit applications. */
    #if __WORDSIZE == 64 && defined __WORDSIZE_COMPAT32
        int32_t ut_session;      /* Session ID (getsid(2)),
                                used for windowing */

        struct {
            int32_t tv_sec;      /* Seconds */
            int32_t tv_usec;    /* Microseconds */
        } ut_tv;                /* Time entry was made */
    #else
        long    ut_session;      /* Session ID */
        struct  timeval ut_tv;    /* Time entry was made */
    #endif
};

```

现在基本上可以知道 who 命令的原理：打开文件 utmp，逐条读出记录 struct utmp，提取相关信息输出

9. 打开 gedit 这个文本编辑器，或者使用其他文本编辑器，输入下面代码

```

#include    <stdio.h>

#include    <utmp.h>

#include    <fcntl.h>

#include    <unistd.h>

int main()
{
    struct utmp current_record;

    int      utmpfd;

    int      reclen = sizeof(current_record);

    if ( (utmpfd = open(UTMP_FILE, O_RDONLY)) == -1 ){

```

```

        perror( UTMP_FILE );
        exit(1);
    }
    while ( read(utmpfd, &current_record, reclen) == reclen )
        show_info(&current_record);
    close(utmpfd);
    return 0;
}

show_info( struct utmp *utbufp )
{
    printf("%-8.8s", utbufp->ut_name);
    printf(" ");
    printf("%-8.8s", utbufp->ut_line);
    printf(" ");
    printf("%10ld", utbufp->ut_time); 注意 d 前面那个是字母 l,不是数字
    printf(" ");
    printf("\n");
}

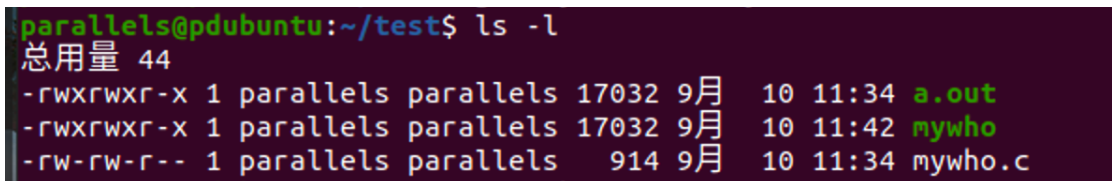
```

文件保存为 mywho.c 注意不要从 word 里面拷贝代码，自己输入，否则会出错。

10. 编译程序

gcc mywho.c -o mywho 这是一个小写字母 o，把 C 程序编译成为二进制目标(object)可执行代码

11. 执行命令 ls -l，是否有可执行文件 mywho？（此处插入截图）



```

parallels@pdubuntu:~/test$ ls -l
总用量 44
-rwxrwxr-x 1 parallels parallels 17032 9月 10 11:34 a.out
-rwxrwxr-x 1 parallels parallels 17032 9月 10 11:42 mywho
-rw-rw-r-- 1 parallels parallels 914 9月 10 11:34 mywho.c

```

12. 输入 `./mywho` 执行，观察输出结果（此处插入截图）

```
parallels@pdubuntu:~/test$ ./mywho
reboot    ~           14116261615ld
runlevel  ~           14116261631ld
          :0          14116265322ld
parallel  :1          14116401042ld
```

13. 对比 `who` 和 `mywho` 的输出，时间的输出格式存在问题，修改 `mywho.c`，解决这个问题。提示：`ctime()` 函数
（此处插入截图）

```
parallels@pdubuntu:~/test$ gcc mywho.c -o mywho
parallels@pdubuntu:~/test$ ls -l
总用量 44
-rwxrwxr-x 1 parallels parallels 17032 9月 10 11:34 a.out
-rwxrwxr-x 1 parallels parallels 17032 9月 10 11:42 mywho
-rw-rw-r-- 1 parallels parallels  914 9月 10 11:34 mywho.c
parallels@pdubuntu:~/test$ ./mywho
reboot    ~           14116261615ld
runlevel  ~           14116261631ld
          :0          14116265322ld
parallel  :1          14116401042ld
```

```
parallels@pdubuntu:~/test$ gcc mywho.c -o mywho
parallels@pdubuntu:~/test$ ls -l
总用量 44
-rwxrwxr-x 1 parallels parallels 17032 9月 10 11:34 a.out
-rwxrwxr-x 1 parallels parallels 17112 9月 10 11:51 mywho
-rw-rw-r-- 1 parallels parallels  877 9月 10 11:51 mywho.c
parallels@pdubuntu:~/test$ ./mywho
reboot    ~           Thu Sep  9 09:29:49
runlevel  ~           Thu Sep  9 09:30:01
          :0          Thu Sep  9 10:00:50
parallel  :1          Thu Sep  9 20:46:26
parallels@pdubuntu:~/test$
```