

# 实验报告

生信2001 张子栋 2020317210101

[输出重定向](#)

[null 文件](#)

[tail 的跟踪用法](#)

[输入重定向](#)

[管道, tee](#)

[查看过去一个月shell相关用户的登录IP](#)

[顺序执行命令](#)

[条件执行命令](#)

[命令组](#)

[文件类型转换](#)

[创建带有特殊字符的文件](#)

```
-  
#  
;  
$  
*  
  
" or "
```

## 输出重定向

> 将输出内容覆盖到文件

>> 将输出内容追加到文件

```
ls -l / /filenotexist >out.txt 2>err.txt
```

第一个 > 将命令的正确输出内容输出到 `out.txt` 文件中，第二个 > 将命令的错误输出内容输出到 `err.txt` 文件中

```
ubuntu@VM-0-6-ubuntu:~$ ls  
a.txt  err.txt  lsman.txt  out.txt  
ubuntu@VM-0-6-ubuntu:~$ ls -l file >out.txt 2>err.txt  
ubuntu@VM-0-6-ubuntu:~$ cat out.txt  
ubuntu@VM-0-6-ubuntu:~$ cat err.txt  
ls: cannot access 'file': No such file or directory
```

## null 文件

Linux中的 `/dev/null` 是空设备文件。这将丢弃写入其中的所有内容，并在读取时返回 `EOF`

如果使用 `cat` 命令读取文件结束符( `EOF` )

可以通过重定向到 `/dev/null` 来丢弃脚本使用的任何输出

```
ubuntu@VM-0-6-ubuntu:~$ ls >/dev/null
ubuntu@VM-0-6-ubuntu:~$
```

如果是错误输出，需要写为 `2>/dev/null`

## tail 的跟踪用法

使用参数 `-f` 跟踪文件的内容更改

## 输入重定向

将 `err.txt` 中的内容输入到 `cat` 中然后输出到标准输出

```
ubuntu@VM-0-6-ubuntu:~$ cat < err.txt
```

▼ 查看文件内容可以直接 `cat [filename]`

还可以与输出重定向同时使用向文件中添加内容

```
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ ls
a.txt err.txt lsman.txt out.txt test.txt
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ cat a.txt
hello world
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ cat <<EOF >>a.txt
> append line 01
> apoen line 02
> EOF
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ cat a.txt
hello world
append line 01
apoen line 02
```

▼ `cat` 接受 `EOF` ( `<<` 后的字符串,可以为任意字符串)字符串之前的内容,然后输出

`<<<` 可以跟字符串或变量,将字符串和变量输入给 `cat`

```
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ cat <<<"this is a string"
this is a string
```

## 管道, tee

**tee** 命令主要被用来向stdout(标准输出流, 通常是命令执行窗口)输出的同时也将内容输出到文件(覆盖)

```
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ touch pinglog.txt
ubuntu@VM-0-6-ubuntu:~/linuxCourse$ ping localhost | tee pinglog.txt
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=2 ttl=64 time=0.025 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=4 ttl=64 time=0.030 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=5 ttl=64 time=0.035 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=6 ttl=64 time=0.029 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=7 ttl=64 time=0.027 ms
^Cubuntu@VM-0-6-ubuntu:~/linuxCourse$ cat pinglog.txt
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=2 ttl=64 time=0.025 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=4 ttl=64 time=0.030 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=5 ttl=64 time=0.035 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=6 ttl=64 time=0.029 ms
64 bytes from localhost.localdomain (127.0.0.1): icmp_seq=7 ttl=64 time=0.027 ms
```

▼ 使用 **tee -a** 参数,向文件中追加内容

## 查看过去一个月shell相关用户的登录IP

```
ubuntu@VM-0-6-ubuntu:~$ last
ubuntu pts/0 223.104.122.41 Mon May 17 09:56 still logged in
ubuntu pts/0 223.104.122.170 Mon May 10 10:54 - 10:56 (00:02)
ubuntu pts/0 223.104.122.170 Mon May 10 10:54 - 10:54 (00:00)
ubuntu pts/0 223.104.122.170 Mon May 10 10:16 - 10:16 (00:00)
ubuntu pts/0 223.104.122.170 Mon May 10 10:04 - 10:05 (00:00)
ubuntu pts/0 223.104.122.170 Mon May 10 09:57 - 10:02 (00:04)
ubuntu pts/3 58.48.165.79 Sat May 8 16:36 - 21:20 (04:43)
ubuntu pts/2 223.104.122.161 Sat May 8 16:33 - 18:47 (02:13)
blur pts/1 223.104.122.161 Sat May 8 16:33 - 18:46 (02:13)
ubuntu pts/1 47.108.152.104 Sat May 8 16:05 - 16:08 (00:02)
ubuntu pts/1 47.108.152.104 Sat May 8 16:01 - 16:04 (00:03)
ubuntu pts/1 47.108.152.104 Sat May 8 15:54 - 15:56 (00:02)
```

```
ubuntu pts/1 47.108.152.104 Sat May 8 15:46 - 15:47 (00:01)
ubuntu pts/1 47.108.152.104 Sat May 8 15:41 - 15:45 (00:03)
ubuntu pts/1 223.104.122.161 Sat May 8 15:40 - 15:40 (00:00)
ubuntu pts/1 223.104.122.161 Sat May 8 15:34 - 15:38 (00:04)
ubuntu pts/1 223.104.122.161 Sat May 8 15:30 - 15:33 (00:03)
ubuntu pts/0 223.104.122.161 Sat May 8 15:27 - 17:40 (02:13)
ubuntu pts/0 223.104.122.161 Sat May 8 15:19 - 15:21 (00:01)
ubuntu pts/0 223.104.122.161 Sat May 8 15:19 - 15:19 (00:00)
ubuntu pts/0 223.104.122.161 Sat May 8 14:58 - 15:17 (00:19)
ubuntu pts/0 223.104.122.161 Sat May 8 14:27 - 14:42 (00:14)
ubuntu pts/0 223.104.122.161 Sat May 8 14:26 - 14:26 (00:00)
ubuntu pts/0 223.104.122.161 Sat May 8 14:19 - 14:21 (00:02)
ubuntu pts/0 223.104.122.161 Sat May 8 14:10 - 14:16 (00:06)
ubuntu pts/0 61.242.149.93 Sun May 2 17:43 - 17:44 (00:00)
ubuntu pts/0 61.242.130.206 Sun May 2 16:18 - 16:19 (00:00)
ubuntu pts/0 61.242.130.206 Sun May 2 15:45 - 15:45 (00:00)
ubuntu pts/0 61.242.130.206 Sun May 2 15:08 - 15:09 (00:01)
```

```
wtmp begins Sun May 2 15:08:13 2021
```

## 顺序执行命令

```
pwd;uname;whoami;date;cal
```

命令之间使用 `;` 隔开

## 条件执行命令

```
false && echo yes || echo no
```

条件成立则执行 `&&` 后的命令

条件不成立则执行 `||` 后的命令

| 类似三目运算符

## 命令组

```
pwd; (cd /; pwd;); pwd
pwd; { cd /; pwd;}; pwd
```

使用 `()` 成组,左括号不需要用空格隔开

使用 `{ }` 成组,左括号需要用空格隔开

## ## tr字符串转换

- 大小写转换

```
tr 'a-z' 'A-Z'
```

- 压缩重复字母

```
tr -s 'a-zA-Z'
```

`-s` → squeeze

- 删除字符

```
tr -d '[o]' 删除字母 o
```

`-d` → delete

- 选择补集

```
tr -d -c '[o]' 删除非 o 字母
```

- 替换字符,并压缩

```
tr -s 'a' 'b'
```

---

## 文件类型转换

不同系统下的换行符不同:

Mac OS下: `\r`

Linux下: `\n`

DOS\Windows下: `\r\n`

可以使用 `unix2dos` / `dos2unix` 进行文件类型的转换

▼ 使用 `tr` 命令可以达到同样的效果

---

## 创建带有特殊字符的文件

-

直接创建带有 `-` 的文件,shell会将 `-` 后的内容当作参数

方法1:使用路径创建文件

```
touch ./-test.txt
```

方法2:指定参数

```
touch -- -test.txt
```

#

shell会将#后的内容当作注释忽略

方法1:使用路径创建文件

```
touch ./#test.txt
```

方法2:使用''

```
touch '#test.txt'
touch test1.txt '#test2.txt' #创建两个文件
```

;

shell中;起分隔命令的作用

使用''

```
touch ';test.txt'
```

\$

```
touch '$test.txt'
```

\*

\转义或使用''


```
touch '*t1.txt'
touch \'*t2.txt'
```

' or ''

交替使用'和''

同时出现使用\转义

```
touch 'te"st.txt'  
touch "te'st.txt"
```

包含空格使用  转义

```
touch test\ 01\ .txt
```