

2024-04-18-2021219113-2021213595-沈尉林

1.

Step 1: 运行 `1.sh`

在终端中输入，并将输出重定向到 `1.txt` 中：

```
Bash ▾  
1 ./1.sh > 1.txt  
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# ./1.sh > 1.txt
```

Step 2: 然后将其悬停

键入 `ctrl + z`

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# ./1.sh > 1.txt  
^Z  
[1]+  Stopped                  ./1.sh > 1.txt
```

Step 3: 再运行 `2.sh`

在终端中输入：

```
Bash ▾  
1 ./2.sh
```

```
OpenSSH SSH client
hello 69816
hello 69817
hello 69818
hello 69819
hello 69820
hello 69821
hello 69822
hello 69823
hello 69824
hello 69825
hello 69826
hello 69827
hello 69828
hello 69829
hello 69830
hello 69831
hello 69832
hello 69833
hello 69834
hello 69835
hello 69836
hello 69837
hello 69838
hello 69839
hello 69840
hello 69841
hello 69842
hello 69843
hello 69844
hello 69845
```

Step 4: 然后将其悬停

同上

```
OpenSSH SSH client
hello 648201
hello 648202
hello 648203
hello 648204
hello 648205
hello 648206
hello 648207
hello 648208
hello 648209
hello 648210
hello 648211
hello 648212
hello 648213
hello 648214
hello 648215
hello 648216
hello 648217
hello 648218
hello 648219
hello 648220
hello 648221
hello 648222
hello 648223
hello 648224
hello 648225
hello 648226
hello 648227
^Z
[2]+  Stopped                  ./2.sh
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen#
```

Step 5: 看看有哪些程序处于悬停中

在终端中输入:

```
Bash
1 jobs
```

```
root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# jobs
[1]-  Stopped                  ./1.sh
[2]+  Stopped                  ./2.sh
root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen#
```

Step 6: 将 `1.sh` 在后台恢复运行

在终端中输入:

```
Bash
1  bg %1

root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# bg %1
[1]+  ./1.sh > 1.txt &
```

Step 7: 将 `2.sh` 在前台恢复运行

在终端中输入:

```
Bash
1  fg %2
```

Step 8: 然后终止 `2.sh`

键入 `ctrl + c`

```
OpenSSH SSH client
hello 108589
hello 108590
hello 108591
hello 108592
hello 108593
hello 108594
hello 108595
hello 108596
hello 108597
hello 108598
hello 108599
hello 108600
hello 108601
hello 108602
hello 108603
hello 108604
hello 108605
hello 108606
hello 108607
hello 108608
hello 108609
hello 108610
hello 108611
hello 108612
hello 108613
hello 108614
hello 108615
hello 108616
hello 108617
^Croot@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen#
```

2. 写一个每周6凌晨3:00(或者你正在写该作业的时间), 使得在屏幕上自动打印“hehe”

在终端中输入，编辑 `cron` 任务：

```
Bash ▾  
1  crontab -e
```

在编辑器中，添加下面代码：

```
Bash ▾  
1  24 22 * * * echo "hehe" | wall
```

这行的意思是会在每天的晚上10点24分输出"hehe"到终端。

```
OpenSSH SSH client  ×  +  ▾  
GNU nano 4.8 /tmp/crontab.A1wstN/crontab Modified  
# Edit this file to introduce tasks to be run by cron.  
#  
# Each task to run has to be defined through a single line  
# indicating with different fields when the task will be run  
# and what command to run for the task  
#  
# To define the time you can provide concrete values for  
# minute (m), hour (h), day of month (dom), month (mon),  
# and day of week (dow) or use '*' in these fields (for 'any').  
#  
# Notice that tasks will be started based on the cron's system  
# daemon's notion of time and timezones.  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow  command  
24 22 * * * echo "hehe" | wall  
  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo  M-A Mark Text  
^X Exit      ^R Read File  ^\ Replace  ^U Paste Text ^T To Spell  ^_ Go To Line M-E Redo  M-6 Copy Text
```

到点执行结果：

```
Broadcast message from root@iZ2ze0olgnf08nfb73yrokZ (somewhere) (Mon Apr 22 22:  
  
hehe  
  
|
```

3. 在目录里创建一些文件和目录，然后实现

Step 1: 删除所有三个字符的文件

删除前：

```
root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# ls  
111 222 333 debug_part_aa debug_part_ad dir1 file1 file.list parent.sh  
1.sh 2.sh child.sh debug_part_ab debug_part_ae dir2 file2 nohup.out  
1.txt 2.txt debug2.txt debug_part_ac debug.txt file file-cmb output.txt
```

在终端中输入:

```
Bash
1 rm ???

root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# ls
1.sh  2.sh  child.sh  debug_part_aa  debug_part_ac  debug_part_ae  dir1  file  file2  file.list  output.txt
1.txt 2.txt  debug2.txt  debug_part_ab  debug_part_ad  debug.txt      dir2  file1  file-cmb  nohup.out  parent.sh
```

Step 2: 删除所有以数字开头, 以小写字母结尾的目录

删除前:

```
root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# ls
1.sh  2.sh  child.sh  debug_part_aa  debug_part_ac  debug_part_ae  dir1  file  file2  file.list  output.txt
1.txt 2.txt  debug2.txt  debug_part_ab  debug_part_ad  debug.txt      dir2  file1  file-cmb  nohup.out  parent.sh
```

在终端中输入:

```
Bash
1 rm -rf [0-9]*[a-z]

root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# ls
child.sh  debug_part_aa  debug_part_ac  debug_part_ae  dir1  file  file2  file.list  output.txt
debug2.txt  debug_part_ab  debug_part_ad  debug.txt      dir2  file1  file-cmb  nohup.out  parent.sh
```

4.

Step 1: 定义一个变量 `temp=3`

在终端中输入:

```
Bash
1 temp=3

root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen# temp=3
root@iZ2ze0olgnf08nfb73yrokZ:/home/alanshen#
```

Step 2: 定义一个变量 `cont` 为 `temp` 变量加 `is content of` 加文件 `1.txt` 的内容, 并显示该变量

在终端中输入:

```
Bash
1 cont="$temp is content of $(cat 1.txt)"
2 echo "$cont"
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# cont="$temp is content of $(cat 1.txt)"
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# echo "$cont"
3 is content of this is 1.txt
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen#
```

Step 3: 定义一个变量 `dd` 为 `this is content of dir1` 加目录 `dir1` 的内容,并打印该变量值

在终端中输入:

```
Bash
1 dd="this is content of $(ls dir1)"
2 echo "$dd"

root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# dd="this is content of $(ls dir1)"
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# echo "$dd"
this is content of 1.txt
```

Step 4: 验证一下, 如果双引号改为单引号的话, 变量 `dd` 就是一个常量了

在终端中输入:

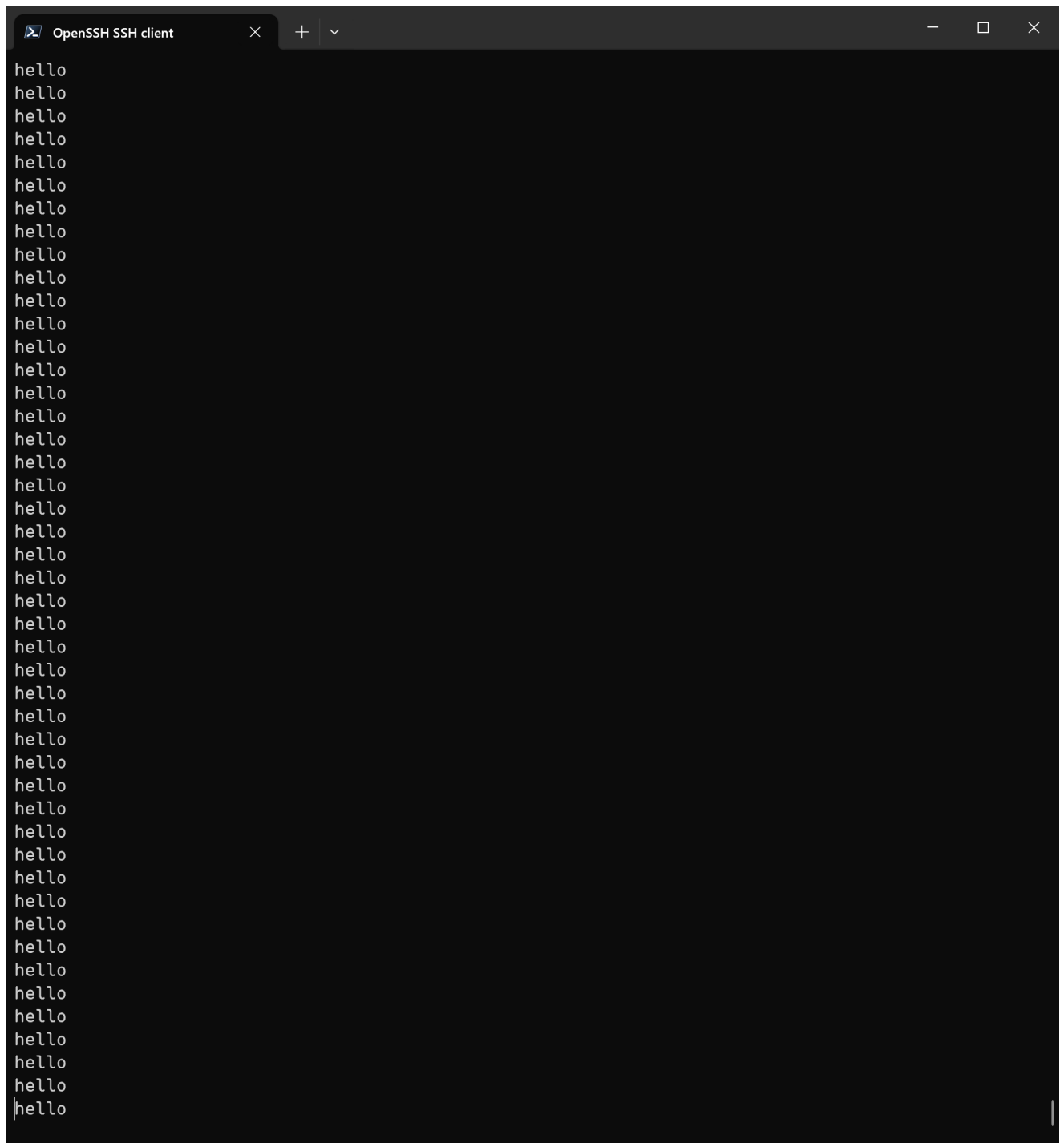
```
Bash
1 dd='this is content of $(ls dir1)'
2 echo "$dd"

root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# dd='this is content of $(ls dir1)'
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen# echo "$dd"
this is content of $(ls dir1)
```

5. 运行无限循环程序 `1.sh`, 将输出重定向到 `debug.txt` 中, 将错误重定向到 `error.txt` 中

在终端中输入:

```
Shell
1 nohup ./1.sh >debug.txt 2>error.txt &
2 tail -f debug.txt
3 tail -f error.txt
```



在终端中输入：

```
Bash
1 sort /etc/passwd > ./passwd.sorted
```

```
OpenSSH SSH client
apt:x:105:65534::/nonexistent:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
_chrony:x:110:121:Chrony daemon,,,:/var/lib/chrony:/usr/sbin/nologin
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
games:x:5:60:games:/usr/games:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
Hyr1sky:x:1001:1001::/home/Hyr1sky:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
lisi:x:1003:1006::/home/lisi:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
root:x:0:0:root:/root:/bin/bash
sshd:x:109:65534::/run/sshd:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
tcpdump:x:107:113::/nonexistent:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
"./passwd.sorted" 32L, 1689C
1,1 Top
```

Step 3: 对文件 **3.txt**

```
Txt
1 101
2 2
3 2
4 4
5 56
6 32
```

由小到大进行排序后，压缩重复行，并保存到 **3.txt.sorted.uniq** 文件中

在终端中输入：

```
Bash
1 sort -u -n 3.txt >3.txt.sorted.uniq
```


Bash ▾

```
1 grep -n -e temp -e Temp *
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen/123# grep -n -e temp -e Temp *
file_with_temp.txt:1:This file contains the word temp.
file_with_Temp.txt:1:This file contains the word Temp.
temp_file.txt:1:This is a temporary file.
```

Step 4: 搜索所有 `temp` 的文件，并忽略大小写

在终端中输入:

Bash ▾

```
1 grep -i temp *
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen/123# grep -i temp *
file_with_temp.txt:This file contains the word temp.
file_with_Temp.txt:This file contains the word Temp.
temp_file.txt:This is a temporary file.
```

Step 5: 搜索所有含单词 `temp` 的文件

在终端中输入:

Bash ▾

```
1 grep -w temp *
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen/123# grep -w temp *
file_with_temp.txt:This file contains the word temp.
```

Step 6: 搜索所以含字符串 `i am a boy` 的文件

在终端中输入:

Bash ▾

```
1 grep "i am a boy" *
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen/123# grep "i am a boy" *
file_with_phrase.txt:This file contains the phrase 'i am a boy'.
```

Step 7: 搜索所有以 `t` 或者 `T` 开头，以数字为中间，以大写字母为结尾的字符串

在终端中输入:

Bash ▾

```
1 grep [tT][0-9][A-Z] *
```

```
root@iZ2ze0lgnf08nfb73yrokZ:/home/alanshen/123# grep [tT][0-9][A-Z] *  
new.txt:T1Z
```