

实验二 Linux文件系统

课程名称： Unix基础与应用

实验项目名称： 实验二 Linux文件系统

专业班级： 计科1903班

姓名： 刘昱杉

学号： 201908010305

指导教师： 肖雄仁

完成时间： 2021 年 3 月 30 日

2.1 管理目录和文件

1. 实验目的

练习管理Linux系统的目录和文件，通过完成本实验，掌握下列技能：

1. 使用目录
2. 处理文件
3. 以递归方式处理文件和目录
4. 处理二进制文件

2. 实验环境

2.1 软件环境：

1. CentOS Linux release 7.9.2009
2. VMware Workstation Pro 15
3. MobaXterm v20.0

2.2 硬件环境：

1. 阿里云云服务器ECS，共享计算型实例，配置（CPU 1核\内存 2GB\带宽 1Mbps）
2. 个人笔记本，配置（建议内存>4G）

3. 实验内容和方法

3.1 管理目录

1. 登录

```
localhost login: Silvia
Password:
Last login: Sun Mar 28 06:01:56 on tty1
[Silvia@localhost ~]$
```

2. 检查所在目录

3. 将当前目录更改为根目录 (/)

4. 验证是否在根目录中，并执行简单列表和长列表

```
[Silvia@localhost ~]$ pwd
/home/Silvia
[Silvia@localhost ~]$ cd /
[Silvia@localhost /]$ pwd
/
[Silvia@localhost /]$ ls
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib  media  opt  root  sbin  sys  usr
[Silvia@localhost /]$ ls -l
total 20
lrwxrwxrwx. 1 root root 7 Mar 27 00:03 bin -> usr/bin
dr-xr-xr-x. 5 root root 4096 Mar 27 00:07 boot
drwxr-xr-x. 20 root root 3220 Mar 30 19:37 dev
drwxr-xr-x. 75 root root 8192 Mar 28 06:00 etc
drwxr-xr-x. 3 root root 20 Mar 28 05:50 home
lrwxrwxrwx. 1 root root 7 Mar 27 00:03 lib -> usr/lib
lrwxrwxrwx. 1 root root 9 Mar 27 00:03 lib64 -> usr/lib64
drwxr-xr-x. 2 root root 6 Apr 11 2018 media
drwxr-xr-x. 3 root root 18 Mar 28 04:57 mnt
drwxr-xr-x. 2 root root 6 Apr 11 2018 opt
dr-xr-xr-x. 112 root root 0 Mar 30 19:37 proc
dr-xr-xr-x. 2 root root 135 Mar 27 00:24 root
drwxr-xr-x. 24 root root 720 Mar 30 19:39 run
lrwxrwxrwx. 1 root root 8 Mar 27 00:03 sbin -> usr/sbin
drwxr-xr-x. 2 root root 6 Apr 11 2018 srv
dr-xr-xr-x. 13 root root 0 Mar 30 19:37 sys
drwxrwxrwt. 15 root root 4096 Mar 30 19:38 tmp
drwxr-xr-x. 13 root root 155 Mar 27 00:03 usr
drwxr-xr-x. 19 root root 267 Mar 27 00:08 var
[Silvia@localhost /]$
```

5. 列出当前目录中的所有文件，并列出当前目录及以下的所有文件

```
[Silvia@localhost /]$ ls -a
.  bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
.. boot  etc  lib  media  opt  root  sbin  sys  usr
[Silvia@localhost /]$ ls -R_
```

```

./proc/20/task/20/net/netfilter:
nf_log

./proc/20/task/20/net/stat:
arp_cache ndisc_cache nf_conntrack rt_cache
ls: cannot open directory ./proc/20/task/20/ns: Permission denied

./proc/21:
ls: cannot read symbolic link ./proc/21/cwd: Permission denied
ls: cannot read symbolic link ./proc/21/root: Permission denied
ls: cannot read symbolic link ./proc/21/exe: Permission denied
attr          coredump_filter gid_map   mountinfo    oom_score    sched        statm
autogroup     cpuset          io       mounts       oom_score_adj schedstat     status
auxv          cwd             limits   mountstats   pagemap      sessionid    syscall
cgroup        environ         loginuid net           patch_state  setgroups    task
clear_refs    exe            map_files ns            personality  smaps        timers
cmdline       fd             maps     numa_maps    projid_map   stack        uid_map
comm          fdinfo         mem      oom_adj      root         stat         wchan

./proc/21/attr:
current exec fscreate keycreate prev sockcreate
ls: cannot open directory ./proc/21/fd: Permission denied
ls: cannot open directory ./proc/21/fdinfo: Permission denied
ls: cannot open directory ./proc/21/map_files: Permission denied

./proc/21/net:
anycast6      igmp            ip_tables_matches nf_conntrack_expect rt_cache
arp           igmp6           ip_tables_names   packet              snmp
connector     ip6_flowlabel  ip_tables_targets protocols            snmp6
dev           ip6_mr_cache   ipv6_route        psched              sockstat
dev_mcast     ip6_mr_vif     mcfilter          ptype               sockstat6
dev_snmp6     ip6_tables_matches mcfilter6         raw                  softnet_stat
fib_trie      ip6_tables_names netfilter         raw6                 stat
fib_triestat ip6_tables_targets netlink           route                tcp
icmp          ip_mr_cache    netstat           rt6_stats            tcp6
if_inet6      ip_mr_vif      nf_conntrack      rt_acct

[Silvia@localhost ~]$

```

ctrl + c 结束此命令，结束大量输出

- 返回主目录并列出其内容，包括隐藏文件

```

[Silvia@localhost ~]$ cd ~
[Silvia@localhost ~]$ pwd
/home/Silvia
[Silvia@localhost ~]$ ls -a
.  ..  .bash_history  .bash_logout  .bash_profile  .bashrc
[Silvia@localhost ~]$ _

```

隐藏文件: ./..

- 在主目录中创建一个名为mydir的目录，查看主目录和~/mydir目录的详细列表。

```

[Silvia@localhost ~]$ mkdir mydir
[Silvia@localhost ~]$ ls -ld
drwx-----. 3 Silvia Silvia 96 Mar 30 19:52 .
[Silvia@localhost ~]$ ls -ld mydir
drwxrwxr-x. 2 Silvia Silvia 6 Mar 30 19:52 mydir
[Silvia@localhost ~]$ _

```

- 转到mydir目录，创建两个名为myfile1和myfile2的零长度文件
- 发出命令以查看mydir目录的内容的详细列表

```
[Silvia@localhost ~]$ cd mydir
[Silvia@localhost mydir]$ touch myfile1
[Silvia@localhost mydir]$ touch myfile2
[Silvia@localhost mydir]$ ls -l
total 0
-rw-rw-r--. 1 Silvia Silvia 0 Mar 30 20:01 myfile1
-rw-rw-r--. 1 Silvia Silvia 0 Mar 30 20:01 myfile2
[Silvia@localhost mydir]$ _
```

10. 返回主目录并使用ls -R命令查看目录树

```
[Silvia@localhost mydir]$ cd
[Silvia@localhost ~]$ ls -R
.:
mydir
./mydir:
myfile1 myfile2
[Silvia@localhost ~]$ _
```

11. 尝试删除mydir目录

```
[Silvia@localhost ~]$ rmdir mydir
rmdir: failed to remove 'mydir': Directory not empty
[Silvia@localhost ~]$
```

目录必须为空才能删除

12. 再次进入mydir目录，并删除该目录中的两个文件；然后回到您的主目录，然后删除mydir目录

```
[Silvia@localhost mydir]$ rm myfile1 myfile2
[Silvia@localhost mydir]$ cd ..
[Silvia@localhost ~]$ rmdir mydir
[Silvia@localhost ~]$
```

目录为空，删除成功

3.2 管理文件

1. 查看/etc/passwd文件的内容。/etc/passwd文件包含所有列表授权使用该系统的用户

```
[Silvia@localhost ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
Silvia:x:1000:1000:/:/home/Silvia:/bin/bash
```

2. 将/etc/passwd文件复制到您的主目录，然后将其重命名为usersfile

```
[Silvia@localhost ~]$ mv passwd usersfile
[Silvia@localhost ~]$ ls -l
total 4
-rw-r--r--. 1 Silvia Silvia 841 Mar 30 20:17 usersfile
```

3. 将usersfile拆分为多个较小的文件，每个文件200字节
4. 列出您主目录中的所有文件

```
[Silvia@localhost ~]$ split -b 200 usersfile usersfile
[Silvia@localhost ~]$ ls -l ~
total 24
-rw-r--r--. 1 Silvia Silvia 841 Mar 30 20:17 usersfile
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileaa
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileab
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileac
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfilead
-rw-rw-r--. 1 Silvia Silvia 41 Mar 30 20:20 usersfileae
[Silvia@localhost ~]$ _
```

3.3 递归管理文件和目录

1. 创建目录sub1，并在sub1中创建目录sub2。使用一个命令即可完成所有操作

```
[Silvia@localhost ~]$ mkdir -p sub1/sub2
[Silvia@localhost ~]$ ls -R
.:
sub1 usersfile usersfileaa usersfileab usersfileac usersfilead usersfileae
./sub1:
sub2
./sub1/sub2:
```

2. 转到sub2目录并创建一个名为myfile的文件

```
[Silvia@localhost ~]$ cd sub1/sub2
[Silvia@localhost sub2]$ touch myfile
[Silvia@localhost sub2]$ ls -l
total 0
-rw-rw-r--. 1 Silvia Silvia 0 Mar 30 20:26 myfile
```

3. 返回主目录，复制整个sub1目录树到tree1，并递归列出sub1和tree1中的所有文件和目录

```
[Silvia@localhost sub2]$ cd
[Silvia@localhost ~]$ cp -R sub1 tree1
[Silvia@localhost ~]$ ls -R sub1
sub1:
sub2
sub1/sub2:
myfile
[Silvia@localhost ~]$ ls -R tree1
tree1:
sub2
tree1/sub2:
myfile
```

4. 移动目录树tree1进入sub1子目录
5. 列出主目录的内容，递归列出所有文件和目录sub1目录

```
[Silvia@localhost ~]$ mv tree1 sub1
[Silvia@localhost ~]$ ls -l
total 24
drwxrwxr-x. 4 Silvia Silvia 31 Mar 30 20:29 sub1
-rw-r--r--. 1 Silvia Silvia 841 Mar 30 20:17 usersfile
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileaa
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileab
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfileac
-rw-rw-r--. 1 Silvia Silvia 200 Mar 30 20:20 usersfilead
-rw-rw-r--. 1 Silvia Silvia 41 Mar 30 20:20 usersfileae
[Silvia@localhost ~]$ ls -R sub1
sub1:
sub2 tree1

sub1/sub2:
myfile

sub1/tree1:
sub2

sub1/tree1/sub2:
myfile
```

3.4 管理二进制文件

1. 使用od和hexdump列出文件/bin/ls的内容

```
0023200 022777 105762 000041 046150 000000 164400 175440 177777
0023220 022777 105752 000041 046550 000000 164400 175420 177777
0023240 022777 105742 000041 047150 000000 164400 175400 177777
0023260 022777 105732 000041 047550 000000 164400 175360 177777
0023300 022777 105722 000041 050150 000000 164400 175340 177777
0023320 022777 105712 000041 050550 000000 164400 175320 177777
0023340 022777 105702 000041 051150 000000 164400 175300 177777
```

```
[Silvia@localhost ~]$ od /bin/ls_
```

```
000e010 5441 f631 5355 8948 48fb ec83 4820 6c8d
000e020 1024 4864 048b 2825 0000 4800 4489 1824
000e030 c031 8948 e8ea 46c6 ffff c085 477e 8d4c
000e040 2464 eb0c 0f2a 001f 448b 0c24 f883 7401
000e050 830a 04f8 0574 f883 7520 4845 ea89 01be
```

```
[Silvia@localhost ~]$ hexdump /bin/ls_
```

2. 列出/bin/ls程序中的所有字符串

```
.dynamic
.got
.got.plt
.data
.bss
.gnu_debuglink
.gnu_debugdata
[Silvia@localhost ~]$ strings /bin/ls
```

2.2 文件和目录权限

1. 实验目的

练习控制文件和目录权限，通过完成本实验，掌握下列技能：

修改文件和目录权限

2. 实验环境

2.1 软件环境：

1. CentOS Linux release 7.9.2009
2. VMware Workstation Pro 15
3. MobaXterm v20.0

2.2 硬件环境：

1. 阿里云云服务器ECS，共享计算型实例，配置（CPU 1核\内存 2GB\带宽 1Mbps）
 2. 个人笔记本，配置（建议内存>4G）
-

3. 实验步骤和方法

3.1 创建用户账户

1. 在本地虚拟机的tty4上，以root用户身份Login

```
CentOS Linux 7 (Core)
Kernel 3.10.0-1160.el7.x86_64 on an x86_64

localhost login: root
Password:
Last login: Sun Mar 28 06:05:37 on tty4
[root@localhost ~]# tty
/dev/tty4
[root@localhost ~]# _
```

2. 执行一下一系列命令

```
#groupadd penguins
#useradd -m -g penguins -c "penguins penguins (1)" tux1
#useradd -m -g penguins -c "penguins penguins (2)" tux2
#passwd tux1
New password:penguin1
Retype new password:penguin1
#passwd tux2
New password:penguin2
Retype new password:penguin2
```

```

[root@localhost ~]# groupadd penguins
[root@localhost ~]# useradd -m -g penguins -c "penguins penguins(1)" tux1
[root@localhost ~]# useradd -m -g penguins -c "penguins penguins(2)" tux2
[root@localhost ~]# passwd tux1
Changing password for user tux1.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]# passwd tux2
Changing password for user tux2.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]# _

```

3. 在第一个基于文本的虚拟终端 (tty2) 上, 使用密码penguin1以tux1登录, 然后在下一个 (tty3) 上, 使用密码penguin2以tux2登录

```

localhost login: tux1
Password:
[tux1@localhost ~]$ tty
/dev/tty2
[tux1@localhost ~]$

```

```

localhost login: tux2
Password:
[tux2@localhost ~]$ tty
/dev/tty3
[tux2@localhost ~]$

```

3.2 文件和目录权限

1. 切换到以tux1身份登录的虚拟终端, 然后在主目录中查看权限

```

[tux1@localhost ~]$ ls -ld /home/tux1
drwx-----. 2 tux1 penguins 62 Mar 30 20:59 /home/tux1
[tux1@localhost ~]$ tty
/dev/tty2
[tux1@localhost ~]$ _

```

2. 切换到以tux2登录的虚拟终端, 尝试到tux1的目录, 或读取tux1的主目录的内容

```

[tux2@localhost ~]$ tty
/dev/tty3
[tux2@localhost ~]$ cd /home/tux1
-bash: cd: /home/tux1: Permission denied
[tux2@localhost ~]$ ls /home/tux1
ls: cannot open directory /home/tux1: Permission denied

```

/home/tux1的权限为rwx-----, tux2没有权限查看和读取/home/tux1

3. 切换到tux1虚拟终端, 更改目录/home/tux1的权限, 以便允许其他用户读取和访问它。然后切换至tux2再次尝试访问该目录

```

[tux1@localhost ~]$ tty
/dev/tty2
[tux1@localhost ~]$ chmod 755 /home/tux1
[tux1@localhost ~]$ _

```

```

[tux2@localhost ~]$ tty
/dev/tty3
[tux2@localhost ~]$ cd /home/tux1
[tux2@localhost tux1]$ ls /home/tux1
[tux2@localhost tux1]$

```


755: rwxr-xr-x, 组和其他用户可以查看和执行

4. 作为tux2, 请尝试在tux1s主目录中创建和删除文件

```
[tux2@localhost tux1]$ touch tesefile
touch: cannot touch 'tesefile': Permission denied
[tux2@localhost tux1]$ _
```

755: 组和其他用户没有write的权利

5. 再次切换到tux1的虚拟终端。创建一个bin目录, 然后将文件/bin/ls复制到该文件中, 并将其重命名为my_ls

```
[tux1@localhost ~]$ tty
/dev/tty2
[tux1@localhost ~]$ chmod 755 /home/tux1
[tux1@localhost ~]$ chvt 3
[tux1@localhost ~]$ mkdir /home/tux1/bin
[tux1@localhost ~]$ cp /bin/ls /home/tux1/bin/my_ls
[tux1@localhost ~]$ ls -l /home/tux1/bin
total 116
-rwxr-xr-x. 1 tux1 penguins 117608 Mar 30 21:22 my_ls
[tux1@localhost ~]$
```

6. 将my_ls的权限设置为rw-r -----, 然后尝试以tux1和tux2的身份执行它

```
[tux1@localhost ~]$ chmod 640 /home/tux1/bin/my_ls
[tux1@localhost ~]$ tty
/dev/tty2
[tux1@localhost ~]$
```

```
[tux2@localhost ~]$ tty
/dev/tty3
[tux2@localhost ~]$ /home/tux1/bin/my_ls
-bash: /home/tux1/bin/my_ls: Permission denied
[tux2@localhost ~]$ _
```

```
[tux1@localhost ~]$ /home/tux1/bin/my_ls
-bash: /home/tux1/bin/my_ls: Permission denied
```

7. 将my_ls的权限设置为rwxr-xr-x, 然后尝试再次执行一次, 两者均以tux1和tux2

```
[tux1@localhost ~]$ chmod 755 /home/tux1/bin/my_ls
[tux1@localhost ~]$ /home/tux1/bin/my_ls
bin
[tux1@localhost ~]$
```

```
[tux2@localhost ~]$ /home/tux1/bin/my_ls
[tux2@localhost ~]$ _
```

8. 尝试将my_ls作为tux1, tux2和您自己的账户执行, 但现在具有rw -----, rw-rw ----, rwx -----, rwx--x ---以及rwx--x--x的权限, tux1执行my_ls至少需要什么权限? tux2需要什么权限? 您自己的用户帐户需要什么权限?

	rw -----	rw-rw ----	rwx -----	rwx--x ---	rwx--x--x
tux1	x	x	√	√	√
tux2	x	x	x	√	√
Silvia	x	x	x	x	√

至少需要x (execute) 权限

实验总结

1. 很奇怪的是虚拟终端快捷键`ctrl+alt+fn`只有f4可以切换
所以一直用`chvt + n`切换
2. 不明白为什么 `/home/tux1/bin/my_ls`执行时每个用户的执行情况都不一样 (?)