

北京科技大学实验报告

学院： 计算机与通信工程学院

专业： 计算机科学与技术

班级： 计 184

姓名： 王丹琳

学号： 41824179

实验日期： 2021 年 3 月 15 日

实验名称：

实验一 Linux 操作系统安装及初步使用

实验目的：

- (1) 独立安装 Linux 系统（Ubuntu, Fedora, RedHat 三选一）
- (2) 正确认识 Linux 启动过程
- (3) 熟练使用 15 条基本命令

实验仪器：

ThinkPad T480

实验环境：

VMware 虚拟机

Ubuntu 64 位

实验原理：

Linux，全称 GNU/Linux，是一种免费使用和自由传播的类 UNIX 操作系统，其内核由林纳斯·本纳第克特·托瓦兹于 1991 年 10 月 5 日首次发布，它主要受到 Minix 和 Unix 思想的启发，是一个基于 POSIX 的多用户、多任务、支持多线程和多 CPU 的操作系统。它能运行主要的 Unix 工具软件、应用程序和网络协议。它支持 32 位和 64 位硬件。Linux 继承了 Unix 以网络为核心的设计思想，是一个性能稳定的多用户网络操作系统。

实验内容与步骤：

1) 熟悉安装关键步骤

- (1) 在复习第 1,2 讲课件的基础上，查阅网上 Linux 系统相应发行版本(Ubuntu, Fedora, RedHat 三选一) 安装攻略，选择写得好的帖子，一步一步领会并实践。

例如：U 盘安装 Windows 10 和 Ubuntu Linux 双系统图解教程：

<http://www.linuxdiyf.com/linux/13140.html>

(2) 梳理出相关要点，消化理解。

2) 15 条常用命令的熟悉和使用

(1) Basename

(2) cd

(3) chgrp

(4) chmod

(5) chown

(6) cp

(7) dd

(8) file

(9) find

(10) ln

(11) locate

(12) ls

(13) mkdir

(14) mv

(15) pwd

实验数据及处理：

(1) Ubuntu 安装相关要点

1) VMWare 安装包和 **ubuntu** 镜像文件准备

2) VMWare 安装并创建虚拟机

3) 为虚拟机安装 ubuntu 操作系统

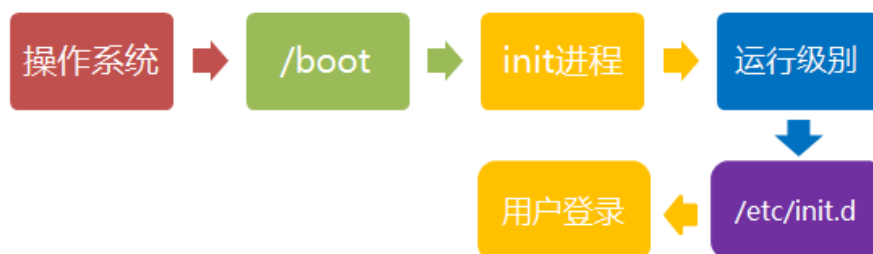
安装攻略：<https://www.bilibili.com/video/av50963336?p=2>教程

(2) Linux 启动过程的认识

1)内核的引导

2)运行 init

- 3)系统初始化
- 4)建立终端
- 5)用户登录系统



(3) 15 条基本指令的使用

由于之前在虚拟机中安装过 Ubuntu，故使用原先装的环境进行实验

- **basename**

用于获取路径中的文件名或路径名，还可以删除文件名扩展名。（路径不一定真实存在于该计算机中）

```
loongson@loongson-VirtualBox:~$ basename --help
用法: basename 名称 [后缀]
或: basename 选项
Print NAME with any leading directory components removed.
If specified, also remove a trailing SUFFIX.
```

basename 名称: 获取末尾的文件名或路径名（未删除扩展名）

```
loongson@loongson-VirtualBox:~$ basename /home/func/Readme_first.txt
Readme_first.txt
```

basename 名称[后缀]: 获取末尾的文件名或路径名（删除扩展名）

```
loongson@loongson-VirtualBox:~$ basename /home/func/Readme_first.txt .txt
Readme_first
```

basename -a/--multiple 多个路径，获得多个路径的末尾的文件名（未删除扩展名）;

- **cd**

切换当前工作目录至 dirName(目录参数)。其中 dirName 表示法可为绝对路径或相对路径。若目录名称省略，则变换至使用者的 home 目录（也就是刚 login 时所在的目录）。另外，"~" 也表示为 home 目录的意思，"." 则是表示目前所在的目录，".." 则表示目前目录位置的

上一层目录。

```
loongson@loongson-VirtualBox:~$ cd --help
bash: cd: --: 无效选项
cd: 用法: cd [-L|[-P [-e]] [-@]] [dir]
```

cd 绝对路径，切换到对应目录：

```
loongson@loongson-VirtualBox:~$ cd /home
loongson@loongson-VirtualBox:/home$
```

cd 相对路径，切换到对应目录：

4 个相对路径的表示方法：当前目录 .

父目录 ..

某用户的根目录 ~user

自己的根目录 ~

cd ..，返回上级目录：

```
loongson@loongson-VirtualBox:/home$ cd ..
loongson@loongson-VirtualBox:/$
```

cd ~，进入用户主目录：

```
loongson@loongson-VirtualBox:/home$ cd ~
loongson@loongson-VirtualBox:~$
```

cd -，返回进入此目录之前所在的目录：

```
loongson@loongson-VirtualBox:/home$ cd -
/home/loongson
```

- chgrp

变更文件或目录的所属群组。群组是 Linux 的一个概念，多个用户可以构成一个群组。--reference 参数可以根据指定文件改变文件的群组属性。

```

loongson@loongson-VirtualBox:~$ chgrp --help
用法: chgrp [选项]... 用户组 文件...
或: chgrp [选项]... --reference=参考文件 文件...
Change the group of each FILE to GROUP.
With --reference, change the group of each FILE to that of RFI
LE.

    -c, --changes           like verbose but report only when a c
change is made
    -f, --silent, --quiet  suppress most error messages
    -v, --verbose           output a diagnostic for every file pr
ocessed
        --dereference      affect the referent of each symbolic
link (this is
                           the default), rather than the symboli
c link itself
    -h, --no-dereference  affect symbolic links instead of any
referenced file
                           (useful only on systems that can chan
ge the
                           ownership of a symlink)
        --no-preserve-root do not treat '/' specially (the defa

```

-c 或 - changes 效果类似” -v” 参数，但仅回报更改的部分。

-f 或 - quiet 或 - silent 不显示错误信息。

-h 或 - no-dereference 只对符号连接的文件作修改，而不更动其他任何相关文件。

-R 或 - recursive 递归处理，将改变指定目录及其下的所有子目录和文件的属组。

-v 或 - verbose 显示指令执行过程。

- reference=<参考文件或目录> 把指定文件或目录的所属群组全部设成和参考文件或目录的所属群组相同。

该命令改变指定指定文件所属的用户组。其中 group 可以是用户组 ID，也可以是/etc/group 文件中用户组的组名。文件名是以空格分开的要改变属组的文件列表，支持通配符。如果用户不是该文件的属主或超级用户，则不能改变该文件的组。

● chmod

Linux 的文件调用权限分为三级：文件拥有者、群组、其他。利用 chmod 可以藉以控制文件如何被他人所调用。其中 mode 是权限设定字符串，u 表示该 文件的拥有者，g 表示与该文件的拥有者属于

同一个群体(group)者，o 表示其 他以外的人，a 表示这三者皆是。
+ 表示增加权限、- 表示取消权限、= 表示唯一设定权限。r 表示可
读取，w 表示可写入，x 表示可执行，X 表示只有当该文件是个子
目录或者该文件已经被设定过为可执行。

```
loongson@loongson-VirtualBox:~$ chmod --help
用法: chmod [选项]... 模式[,模式]... 文件...
或: chmod [选项]... 八进制模式 文件...
或: chmod [选项]... --reference=参考文件 文件...
Change the mode of each FILE to MODE.
With --reference, change the mode of each FILE to that of RFILE.
E.
```

将文件 test.s 设为所有人可以读取，并不让其他人写入。

```
loongson@loongson-VirtualBox:~/test$ chmod ugo+r test.s
loongson@loongson-VirtualBox:~/test$ chmod o-w test.s
loongson@loongson-VirtualBox:~/test$
```

- chown

利用 chown 将指定文件的拥有者改为指定的用户或组，用户可以是
用户名或者用户 ID；组可以是组名或者组 ID；文件是以空格分开
的要改变权限的文件 列表，支持通配符。

```
loongson@loongson-VirtualBox:~$ chown --help
用法: chown [选项]... [所有者][:[组]] 文件...
或: chown [选项]... --reference=参考文件 文件...
Change the owner and/or group of each FILE to OWNER and/or GRO
UP.
With --reference, change the owner and group of each FILE to t
hose of RFILE.
```

将 test.s 文件的所有者改为 loongson，必须在管理员权限下进行。

```
loongson@loongson-VirtualBox:~/test$ chown loongson test.s
loongson@loongson-VirtualBox:~/test$ sudo chown loongson test.
s
[sudo] loongson 的密码:
loongson@loongson-VirtualBox:~/test$
```

- cp

用于复制文件或目录，如果使用该指令复制目录时，必须使用参数
"-r"或者 "-R"。参数-d 可以保留超链接的指向(类似 Windows 的快
捷方式)；-p 参数还把修改时间和访问权限也复制到新文件中。

```
loongson@loongson-VirtualBox:~/test$ cp --help
用法: cp [选项]... [-T] 源文件 目标文件
或: cp [选项]... 源文件... 目录
或: cp [选项]... -t 目录 源文件...
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
必选参数对长短选项同时适用。
```

把 test.s 复制到 test1.s，把 test.s 复制到桌面。

```
loongson@loongson-VirtualBox:~/test$ cp test.s /home/loongson/桌面
loongson@loongson-VirtualBox:~/test$ cp test.s test1.s
```



● dd

用于读取、转换并输出数据。dd 可从标准输入或文件中读取数据，根据指定的格式来转换数据，再输出到文件、设备或标准输出。if、of 参数位输入文件名和输出文件名，bs 参数指定一次读入多少字节，conv 参数可以指定格式，有转化为大写、转化为小写，交换每两个字节、转化为 ascii、转化为 ebcdic 等等功能。

```
loongson@loongson-VirtualBox:~/test$ dd --help
用法: dd [操作数] ...
或: dd 选项
Copy a file, converting and formatting according to the operands.

bs=BYTES      read and write up to BYTES bytes at a time
cbs=BYTES      convert BYTES bytes at a time
conv=CONVS     convert the file as per the comma separated
symbol list
count=N        copy only N input blocks
ibs=BYTES      read up to BYTES bytes at a time (default: 512)
if=FILE        read from FILE instead of stdin
iflag=FLAGS     read as per the comma separated symbol list
obs=BYTES      write BYTES bytes at a time (default: 512)
of=FILE        write to FILE instead of stdout
oflag=FLAGS     write as per the comma separated symbol list
seek=N         skip N obs-sized blocks at start of output
```


将 D.txt 中的英文转化为小写并输出到 d.txt 中。

```
loongson@loongson-VirtualBox:~/桌面$ dd if=D.txt of=d.txt bs=1  
k conv=lcas  
记录了0+1 的读入  
记录了0+1 的写出  
4 bytes copied, 0.000341643 s, 11.7 kB/s
```



- file

用于辨识文件类型，参数-b 不显示文件名称只显示结果，参数-c 显示详细信息以便排查错误，参数-z 尝试去解读压缩文件内容。

```
loongson@loongson-VirtualBox:~/桌面$ file --help  
Usage: file [OPTION...] [FILE...]  
Determine type of FILES.
```

显示文件类型，test.s 为汇编文件，D.txt 为 txt 文件。

```
loongson@loongson-VirtualBox:~/桌面$ file test.s  
test.s: assembler source, ASCII text  
loongson@loongson-VirtualBox:~/桌面$ file D.txt  
D.txt: ASCII text
```

- find

用来在指定目录下查找文件。任何位于参数之前的字符串都将被视为欲查找的目录名。如果使用该命令时，不设置任何参数，则 find 命令将在当前目录下 查找子目录与文件。并且将查找到的子目录和文件全部进行显示。expression 可用的选项有很多，常用的有-name: 根据名称查找，-ctime: 在过去的 n 天内被修改过，-atime: 在过去的 n 天被读取过，-type: 根据文件类型查找等等。


```

loongson@loongson-VirtualBox:~/桌面$ find --help
用法: find [-H] [-L] [-P] [-Olevel] [-D help|tree|search|stat|
rates|opt|exec|time] [path...] [expression]

默认路径为当前目录；默认表达式为 -print
表达式可能由下列成份组成：操作符、选项、测试表达式以及动作：

操作符（优先级递减；未做任何指定时默认使用 -and）:
      ( EXPR )   ! EXPR   -not EXPR   EXPR1 -a EXPR2   EXPR1 -
and EXPR2
      EXPR1 -o EXPR2   EXPR1 -or EXPR2   EXPR1 , EXPR2

位置选项（总是真）: -daystart -follow -regextype

普通选项（总是真，在其它表达式前指定）:
      -depth --help -maxdepth LEVELS -mindepth LEVELS -mount -
noleaf
      --version -xdev -ignore_readdir_race -noignore_readdir_r
ace

测试(N可以是 +N 或 -N 或 N):-amin N -anewer FILE -atime N -cmin

```

在桌面这个当前目录下找到所有以.txt 结尾的文件。

```

loongson@loongson-VirtualBox:~/桌面$ find -name '*.txt'
./func/Readme_first.txt
./func2/Readme_first.txt
./D.txt
./d.txt

```

- ln

为某一个文件在另外一个位置建立一个同步的链接。链接可以分为两种：软连接和硬链接，软连接可以跨文件系统，硬链接不行。参数-s 可以进行软链接。

```

loongson@loongson-VirtualBox:~/桌面$ ln --help
用法: ln [选项]... [-T] 目标 链接名      (第一种格式)
或: ln [选项]... 目标                    (第二种格式)
或: ln [选项]... 目标... 目录            (第三种格式)
或: ln [选项]... -t 目录 目标...        (第四种格式)
In the 1st form, create a link to TARGET with the name LINK_NAME.
In the 2nd form, create a link to TARGET in the current directory.
In the 3rd and 4th forms, create links to each TARGET in DIRECTORY.
Create hard links by default, symbolic links with --symbolic.
By default, each destination (name of new link) should not already exist.
When creating hard links, each TARGET must exist. Symbolic links
can hold arbitrary text; if later resolved, a relative link is
interpreted in relation to its parent directory.

必选参数对长短选项同时适用。

```

在/home 创建 D.txt 的软链接。

```

loongson@loongson-VirtualBox:~/桌面$ ln -s D.txt /home/loongson/D

```



● locate

用于查找符合条件的文档，他会去保存文档和目录名称的数据库内，查找合乎范本样式条件的文档或目录。其与 find 指令不同的是，find 是去硬盘找，locate 只在/var/lib/slocate 资料库中找。locate 的速度比 find 快，它并不是真的查找，而是查数据库，一般文件数据库在 /var/lib/slocate/slocate.db 中，所以 locate 的查找并不是实时的，而是以数据库的更新为准。

```

loongson@loongson-VirtualBox:~/桌面$ locate --help
Usage: locate [OPTION]... [PATTERN]...
Search for entries in a mlocate database.

  -A, --all                only print entries that match all pat
terns
  -b, --basename          match only the base name of path name
s
  -c, --count             only print number of found entries
  -d, --database DBPATH  use DBPATH instead of default databas
e (which is
                        /var/lib/mlocate/mlocate.db)
  -e, --existing          only print entries for currently exis
ting files
  -L, --follow            follow trailing symbolic links when c
hecking file
                        existence (default)
  -h, --help             print this help
  -i, --ignore-case       ignore case distinctions when matchin
g patterns
  -l, --limit, -n LIMIT  limit output (or counting) to LIMIT e
ntries
  -m, --mmap              ignored, for backward compatibility

```

查找所有包含字符串 'ABC' 的文件。

```

loongson@loongson-VirtualBox:~/桌面$ locate ABC
/home/loongson/.cache/mozilla/firefox/y9rwddn.default/cache2/
entries/7ABCD0D91859D8DC6C8856A8B45E956504AF9BB6
/home/loongson/.cache/mozilla/firefox/y9rwddn.default/cache2/
entries/B45E5CC20D29A15853C9AE0ABC8EC2A0A51AAC54

```

● ls

用于显示指定工作目录下之内容。参数-l 显示文件详细状态。参数-t 将文件依建立时间之先后次序列出，参数-F 在列出的文件名称后加一符号；例如可执行档则加 "*"，目录则加 "/" 等等。

```

loongson@loongson-VirtualBox:~/桌面$ ls --help
用法: ls [选项]... [文件]...
List information about the FILES (the current directory by def
ault).
Sort entries alphabetically if none of -cftuvSUX nor --sort is
specified.

必选参数对长短选项同时适用。
  -a, --all                不隐藏任何以 . 开始的项目
  -A, --almost-all       列出除 . 及 .. 以外的任何项目
                        --author 与 -l 同时使用时列出每个文件的
作者
  -b, --escape            以八进制溢出序列表示不可打印的
字符
                        --block-size=SIZE  scale sizes by SIZE before printi
ng them; e.g.,
                        '--block-size=M' prints sizes i
n units of

```

显示当前目录下的文件

```
loongson@loongson-VirtualBox:~/桌面$ ls
d.txt D.txt func func2 test.s
```

显示隐藏文件

```
loongson@loongson-VirtualBox:~/桌面$ ls -a
.  ..  d.txt D.txt func func2 test.s
```

显示文件详细信息

```
loongson@loongson-VirtualBox:~/桌面$ ls -l
总用量 20
-rw-rw-r-- 1 loongson loongson  4 3月 15 20:11 d.txt
-rw-rw-r-- 1 loongson loongson  4 3月 15 20:09 D.txt
drwx----- 4 loongson loongson 4096 2月 28 19:15 func
drwx----- 5 loongson loongson 4096 3月  4 14:11 func2
-rw-rw-r-- 1 loongson loongson  69 3月 15 20:01 test.s
```

按最后打开时间显示文件

```
loongson@loongson-VirtualBox:~/桌面$ ls -t
d.txt D.txt test.s func2 func
```

在每一个文件的末尾加上一个字符说明该文件的类型。"@"表示符号链接、"|"表示 FIFOs、"/"表示目录、"="表示套接字。

```
loongson@loongson-VirtualBox:~/桌面$ ls -f
test.s func func2 D.txt . d.txt ..
```

ls -s 在每个文件的后面打印出文件的大小。 size(大小)

```
loongson@loongson-VirtualBox:~/桌面$ ls -s
总用量 20
4 d.txt  4 D.txt  4 func  4 func2  4 test.s
```

- mkdir

用于新建一个子目录。参数-p 确保目录名称存在,不存在的就建一个。

```
loongson@loongson-VirtualBox:~/桌面$ mkdir --help
用法: mkdir [选项]... 目录...
Create the DIRECTORY(ies), if they do not already exist.

必选参数对长短选项同时适用。
  -m, --mode=MODE  set file mode (as in chmod), not a=rwx - u
mask
  -p, --parents      no error if existing, make parent director
ies as needed
  -v, --verbose      print a message for each created directory
  -Z                set SELinux security context of each cr
eated directory
                    to the default type
  --context[=CTX]   like -Z, or if CTX is specified then se
t the SELinux
                    or SMACK security context to CTX
  --help            显示此帮助信息并退出
  --version         显示版本信息并退出
```

创建一个名为”aa”的目录。

```
loongson@loongson-VirtualBox:~/桌面$ mkdir aa
```



- mv

用来为文件或目录改名、或将文件或目录移入其它位置。参数-i 询问是否覆盖旧文件若指定目录已有同名文件，而参数-f 则在覆盖时不给任何提示。

```
loongson@loongson-VirtualBox:~/桌面$ mv --help
用法: mv [选项]... [-T] 源文件 目标文件
或: mv [选项]... 源文件... 目录
或: mv [选项]... -t 目录 源文件...
Rename SOURCE to DEST, or move SOURCE(s) to DIRECTORY.

必选参数对长短选项同时适用。
  --backup[=CONTROL] 为每个已存在的目标文件创建备份
  -b                  类似--backup 但不接受参数
  -f, --force          覆盖前不询问
  -i, --interactive    覆盖前询问
  -n, --no-clobber     不覆盖已存在文件
如果您指定了-i、-f、-n 中的多个，仅最后一个生效。
  --strip-trailing-slashes 去掉每个源文件参数尾部的斜线
  -S, --suffix=SUFFIX  替换常用的备份文件后缀
  -t, --target-directory=DIRECTORY move all SOURCE arguments
into DIRECTORY
  -T, --no-target-directory treat DEST as a normal file
  -u, --update          move only when the SOURCE file
is newer                than the destination file or
                        when the
                        destination file is missing
  -v, --verbose         explain what is being done
```

将 D.txt 改名为 D1.txt

```
loongson@loongson-VirtualBox:~/桌面$ mv D.txt D1.txt
```



- pwd

用于显示工作目录，执行 `pwd` 指令可立刻得知您目前所在的工作目录的绝对路径名称。

```
loongson@loongson-VirtualBox:~/桌面$ pwd --help
bash: pwd: --: 无效选项
pwd: 用法: pwd [-LP]
```

显示当前所在的位置。

```
loongson@loongson-VirtualBox:~/桌面$ pwd
/home/loongson/桌面
```

实验结果与分析:

本次实验由于之前在虚拟机中安装过 Ubuntu，故未重新进行 Linux 系统的安装。在完成实验报告时，对 Linux 启动过程进行了系统性地回顾。Linux 启动首先是 BIOS 自检，操作系统接管硬件后，读入 `/boot` 目录下的内核文件。然后启动 `init` 进程，`init` 程序首先是需要读取配置文件 `/etc/inittab`，然后检验运行级别，进行系统初始化。之后就进入了登录界面，Linux 启动完成。

Linux 提供了丰富的系统指令，用户可以在终端下使用这些指令完成对计算机的操作。在本次实验中，我通过熟悉 `basename`、`cd`、`mv`、`pwd` 等 15 条基本命令的使用方法，对 Linux 指令进行了初步的了解。但要真正熟练使用 Linux 指令，还要在实际操作中多去使用。