

一、Linux简介

1. Linux起源与发展

Linux操作系统诞生于1991年10月，创始人Linus Benedict Torvalds。1991年初，Linus在学习期间，使用一台386sx兼容微机上学习minix操作系统，由于觉得该系统不好用，便着手编写自己的操作系统。编写完成后，Linus将代码发布到网络上，供大家下载和修改。



2. Linux特点

1) 开源

Linux内核是一款免费、开源的操作系统，用户可以通过网络或其它途径免费获得，并可以任意修改其源代码。正是由于这一点，来自全世界的无数程序员参与了Linux的修改、编写工作，程序员可以根据自己的兴趣和灵感对其进行改变，这让Linux吸收了无数程序员的精华，不断壮大。

2) 多用户、多任务

多用户：一台机器可以提供给多个用户同时使用

多任务：每个用户可以运行多个程序

3) 性能优异

Linux内核设计优异，资源消耗低，具有良好的稳定性，常用作服务器

4) 支持多种平台

Linux支持x86、680x0、SPARC、Alpha等处理器的平台。此外Linux还是一种嵌入式操作系统，可以运行在手机、机顶盒或游戏机上。能在大到大规模服务器、小到电子表的硬件系统上运行。



3. Linux主要发行版

Linux严格来说是内核的名称，众多厂商在该内核的基础上，进行了包装，采用了不同的名称，这些我们称之为Linux发行版。常见发行版有：

1) Ubuntu

中文翻译做“乌班图”，最易使用的Linux，良好的图形界面，方便的在线软件包管理。



2) RedHat

中文翻译做“红帽子”，性能稳定，主要用于企业级服务器、云计算。



3) Debian

非常稳定;卓越的质量控制；方便的在线软件包管理。

4) openSUSE

精美的界面，全面而直观的配置工具；大型软件包库。

二、Linux系统管理

1. 初识Linux

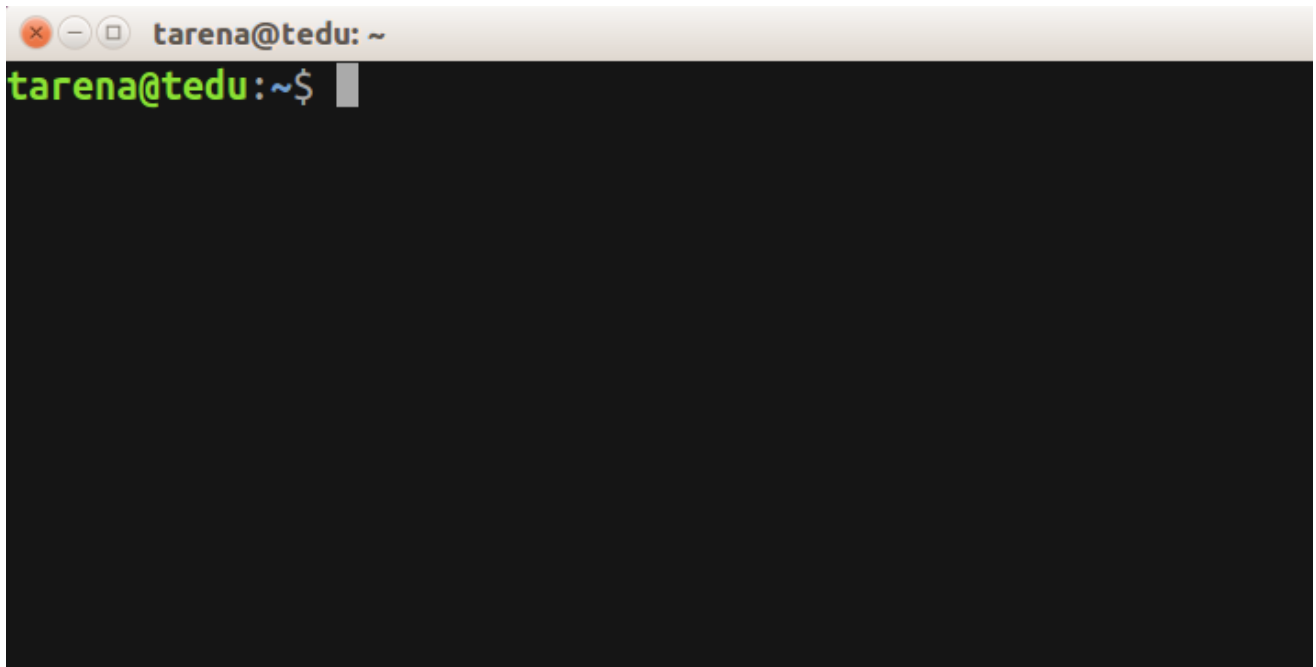
1) 主界面

Ubuntu操作系统图形界面及主要应用如下图所示，ubuntu提供了丰富的图形界面。



2) 终端

但作为专业用户，主要在字符界面（即终端）下使用系统。终端主要用来执行用户输入的操作（这些操作称之为命令），并显示执行结果。



3) 根目录

Linux中，根（root）目录是文件系统起始位置(类似于Windows中计算机)，使用一个斜线来表示（/）。

4) Linux命令格式

Linux命令基本格式为：

```
1 命令名 选项 参数
```

例如：

```
1  ls test -lrt
```

ls：命令名称，表示查看目录中所有文件

test：目录名称，表示要查看的目录是test

-lrt：查看目录中内容所带的参数

2. 文件及目录操作

文件操作是Linux最基本的操作，在日常使用中频率最高，需要熟练掌握。文件操作主要有目录操作、单文件操作、文件压缩解压、文件权限管理等。

1) 目录操作

- 查看当前目录：pwd

```
1 pwd
2 /home/tarena
```

说明：第1行代码为命令，第2行代码为执行结果

- 显示目录内容：ls

作用：ls命令用来查看目录内有哪些文件及子目录

参数：ls -l 以长格式显示(long)

ls -a 显示所有(包含隐藏)all

ls -t 按照文件最后修改时间排序

示例（一）：查看当前目录内容

```
1 ls
```

执行结果：

```
tarena@tedu:~$ ls
1.png          databak        my_env
2.jpg          examples.desktop pycharm-2017.1.5
anaconda3      gitrepo        PycharmProjects
core           mongo          software
```

示例（二）：查看当前目录详细内容

```
1 ls -l
```

执行结果：

```
tarena@tedu:~$ ls -lrt
总用量 5984
drwxrwxr-x 20 tarena tarena 4096 7月 17 2017 anaconda3
-rw-r--r-- 1 tarena tarena 8980 7月 17 2017 examples.desktop
drwxr-xr-x 2 tarena tarena 4096 7月 17 2017 音乐
drwxr-xr-x 2 tarena tarena 4096 7月 17 2017 文档
drwxr-xr-x 2 tarena tarena 4096 7月 17 2017 视频
drwxr-xr-x 2 tarena tarena 4096 7月 17 2017 模板
drwxr-xr-x 2 tarena tarena 4096 7月 17 2017 公共的
```

说明：执行结果从左至右每列的含义分别是文件权限、连接数或子目录数（含义暂先忽略）、文件拥有者、文件拥有者所在组、文件大小、最后修改时间、文件名称

- 切换目录：cd

作用：用于从一个目录切换到另一个目录

参数：cd 目录名称

示例：

```
1 cd /home/tarena
```

说明：Linux中，有几个特殊目录，如下表所示

符号	说明	含义
.	英文点号	当前目录
..	两个英文点号	上级目录
~	英文波浪线	用户主目录
-	英文横线	上一次所在目录

- 创建目录：mkdir
作用：mkdir (make director的简写) 用来创建一个空目录
参数：make -p 如果父目录不存在，同时创建父目录

示例（一）：在当前目录下创建study目录

```
1 mkdir study
2 ls
```

说明：第1行命令，执行创建study目录；第2行命令，查看当前目录内容（如果创建成功，可以看到study目录）。执行结果：

```
tarena@tedu:~$ mkdir study
tarena@tedu:~$ ls
1.png          databak        my_env         static
2.jpg          examples.desktop pycharm-2017.1.5 study
anaconda3      gitrepo        PycharmProjects test
core           mongo          software       公共的
```

示例（二）：创建目录，如果被创建目录的上级目录不存在，则自动创建

```
1 mkdir -p /tmp/demo/test
```

```
1 说明：该示例中，在mkdir后面加了-p选项，会自动创建test的上级目录demo
2
3 执行结果：
```

```
tarena@tedu:~$ mkdir -p /tmp/demo/test
tarena@tedu:~$ ls /tmp/demo/test
tarena@tedu:~$
```

2) 文件操作

- 创建空文件：touch

格式：touch 文件1 文件2 文件3

示例：

```
1 cd ~/study/
2 touch test1.txt test2.txt
```

执行结果：

```
tarena@tedu:~/study$ touch test1.txt test2.txt
tarena@tedu:~/study$ ls -l
总用量 0
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test1.txt
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test2.txt
```

- 向文件写入内容：echo

格式：echo > 文件名

示例：

```
1 echo 'abc' > test1.txt
```

说明：将abc字符串写入test1.txt，会覆盖原来的内容；如果不想覆盖原来的内容，使用两个大于符号>>

- 查看文件内容：cat

格式：cat 文件名

示例：

```
1 cat test1.txt
```

执行结果：

```
tarena@tedu:~/study$ cat test1.txt
abc
```

- 查看文件前几行：head

格式：head -n 行数 文件名

说明：-n 后跟行数，表示显示前面多少行

示例：

```
1 head -n 3 test1.txt
```

执行结果：

```
tarena@tedu:~/study$ head -n 3 test1.txt
abc
bbb
ccc
```

- 查看文件最后几行：tail

格式：tail -n 行数 文件名

说明：-n 后跟行数，表示显示后面多少行

示例：

```
1 tail -n 2 test1.txt
```

执行结果：

```
tarena@tedu:~/study$ tail -n 2 test1.txt
ddd
eee
```

- 文件查找：find

格式：find 查找路径 -name 文件名 | 目录名

示例：

```
1 find . -name "*.txt"
```

说明：查找当前目录下以txt结尾的所有文件

执行结果：

```
tarena@tedu:~/study$ find . -name "*.txt"
./test1.txt
./test2.txt
```

- 文件复制：cp

格式：cp 原文件 新文件

示例：

```
1 cp test1.txt new.txt
```

执行结果：

```
tarena@tedu:~/study$ cp test1.txt new.txt
tarena@tedu:~/study$ ls
new.txt  test1.txt  test2.txt
```

- 文件移动（重命名）：mv

格式：mv 原文件名 新文件名

示例：

```
1 mv new.txt new2.txt
```

执行结果：

```
tarena@tedu:~/study$ mv new.txt new2.txt
tarena@tedu:~/study$ ls
new2.txt  test1.txt  test2.txt
```

- 文件删除：rm

格式：rm [选项] 文件名|目录名

选项：-f force 的意思，强制删除（慎用！）

-i 互动模式，在删除前询问

-r 递归删除，删除包含的子目录（慎用！）

示例：

```
1 rm -i new2.txt
```

执行结果：

```
tarena@tedu:~/study$ rm -i new2.txt
rm: 是否删除普通文件 'new2.txt'? y
tarena@tedu:~/study$ ls
test1.txt  test2.txt
```

3) 文件压缩与解压

Linux中常使用tar命令进行打包、压缩/解压操作。

- 压缩打包

格式：tar -zcvf 压缩包名字.tar.gz 文件1 文件2 目录3

参数：

z 使用gzip方式压缩

c 创建包 create

v 显示操作文件的细节

f 要操作的文件名 file

示例：

```
1 tar -zcvf tmp.tar.gz test1.txt test2.txt
```

说明：上述命令将test1.txt, test2.txt打包成一个文件，并进行压缩，打包压缩后的文件名为tmp.tar.gz（gz后缀名表示进行了压缩，tar后缀名表示进行了打包，加后缀名是为了清晰、易于理解，而不是必须的）

执行结果：

```
tarena@tedu:~/study$ tar -zcvf tmp.tar.gz test1.txt test2.txt
test1.txt
test2.txt
tarena@tedu:~/study$ ls -l
总用量 8
-rw-rw-r-- 1 tarena tarena 20 5月 21 18:24 test1.txt
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test2.txt
-rw-rw-r-- 1 tarena tarena 155 5月 21 19:43 tmp.tar.gz
```

- 解压

格式：tar -zxvf 压缩包名字.tar.gz

参数：x 解压缩

其它参数和打包压缩含义一致

示例：

```
1 rm test1.txt test2.txt
2 ls
3 tar -zxvf tmp.tar.gz
```

说明：第1行代码，删除test1.txt和test2.txt两个文件；第2行代码，查看当前目录内容，确定文件已删除；第3行代码，对tmp.tar.gz进行解压缩

执行结果：

```
tarena@tedu:~/study$ rm test1.txt test2.txt
tarena@tedu:~/study$ ls
tmp.tar.gz
tarena@tedu:~/study$ tar -zxvf tmp.tar.gz
test1.txt
test2.txt
```

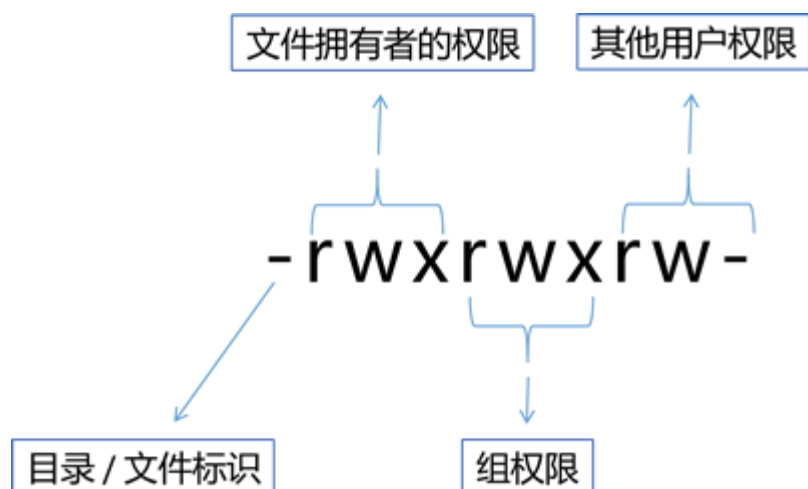
4) 文件权限管理

Linux系统上对文件的权限有着严格的控制，用于如果相对某个文件执行某种操作，必须具有对应的权限方可执行成功。

权限类型：权限类型一般包括读，写，执行，对应字母为 r、w、x

权限粒度：文件拥有者（创建文件的用户）、组（创建文件用户所在的组）、其它组三

所以，一个文件的权限表示为9中状态，分别指拥有这、组、其它用户对该文件是否具有读、写、执行的权限



在rwx三个权限中，从右至左依次用值1,2,4来表示，所以拥有某个文件的读、写、执行权限也可以用三个权限值累加的方式来表示，如下表所示：

权限	含义	对应值
rwx	可读、可写、可执行	4+2+1=7
rw-	可读、可写、不可执行	4+2=6
r-x	可读、不可写、可执行	4+0+1=5
r--	可读、不可写、不可执行	4+0+0=4
-wx	不可读、可写、可执行	0+2+1=3
-w-	不可读、可写、不可执行	0+2+0=2
--x	不可读、不可写、可执行	0+0+1=1
---	不可读、不可写、不可执行	0+0+0=0

- 修改文件权限：chmod
格式：chmod {权限值} {文件/目录名}

示例（一）：给文件test1.txt添加可执行权限

```
1 | chmod +x test1.txt
```

执行结果：

```
tarena@tedu:~/study$ chmod +x test1.txt
tarena@tedu:~/study$ ls -l
总用量 8
-rwxrwxr-x 1 mysql  tarena    0 5月  22 10:01 aaa
-rwxrwxr-x 1 tarena tarena   20 5月  21 18:24 test1.txt
-rw-rw-r-- 1 tarena tarena    0 5月  21 18:14 test2.txt
-rw-rw-r-- 1 tarena tarena  155 5月  21 19:43 tmp.tar.gz
```

示例（二）：设置test2.txt文件，拥有者有读、写、执行权限，其他用户没有访问权限

```
1 | chmod 700 test1.txt
```

执行结果：

```
tarena@tedu:~/study$ chmod 700 test1.txt
tarena@tedu:~/study$ ls -l
总用量 8
-rwxrwxr-x 1 mysql tarena 0 5月 22 10:01 aaa
-rwx----- 1 tarena tarena 20 5月 21 18:24 test1.txt
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test2.txt
-rw-rw-r-- 1 tarena tarena 155 5月 21 19:43 tmp.tar.gz
```

示例（三）：创建test3.txt文件，设置拥有者有读、写、执行权限，其他用户有只读权限

```
1 touch test3.txt
2 chmod 400 test3.txt
```

执行结果：

```
tarena@tedu:~/study$ touch test3.txt
tarena@tedu:~/study$ chmod 700 test3.txt
tarena@tedu:~/study$ ls -l
总用量 8
-rwx----- 1 tarena tarena 20 5月 21 18:24 test1.txt
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test2.txt
-rwx----- 1 tarena tarena 0 5月 22 11:11 test3.txt
-rw-rw-r-- 1 tarena tarena 155 5月 21 19:43 tmp.tar.gz
```

- 示例（四）：取消test3.txt的可执行权限

```
1 chmod -x test3.txt
```

执行结果：

```
tarena@tedu:~/study$ chmod -x test3.txt
tarena@tedu:~/study$ ls -l
总用量 8
-rwx----- 1 tarena tarena 20 5月 21 18:24 test1.txt
-rw-rw-r-- 1 tarena tarena 0 5月 21 18:14 test2.txt
-rw----- 1 tarena tarena 0 5月 22 11:11 test3.txt
-rw-rw-r-- 1 tarena tarena 155 5月 21 19:43 tmp.tar.gz
```

3. 网络操作

- 查看本机网络地址：ifconfig

作用：查看本机IP地址，该地址可以理解为本机在网络上的门牌号，该编号是唯一的

示例：

```
1 ifconfig
```

执行结果：

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 172.17.38.137 netmask 255.255.240.0 broadcast 172.17.47.255
      ether 00:16:3e:08:31:9a txqueuelen 1000 (Ethernet)
      RX packets 4420241 bytes 1104943424 (1.0 GiB)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 3239576 bytes 337248711 (321.6 MiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
     inet 127.0.0.1 netmask 255.0.0.0
     loop txqueuelen 1 (Local Loopback)
     RX packets 1275 bytes 114306 (111.6 KiB)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 1275 bytes 114306 (111.6 KiB)
     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

说明：上图中，红色框中的部分就是本机的IP地址，IP地址可能不只一个，其中127.0.0.1特指本机地址

- 测试网络连通性：ping

格式：ping [对方主机地址]

功能：ping命令主通过网络主动向对方连续发送数据包，对方收到后，应答相同数量的数据包，本机通过对方是否应答来判断彼此是否连通

示例：

```
1 ping www.baidu.com
```

执行结果：

```
[root@ ~ ]#ping www.baidu.com
PING www.a.shifen.com (220.181.38.149) 56(84) bytes of data.
64 bytes from 220.181.38.149 (220.181.38.149): icmp_seq=1 ttl=52 time=4.42 ms
64 bytes from 220.181.38.149 (220.181.38.149): icmp_seq=2 ttl=52 time=4.37 ms
64 bytes from 220.181.38.149 (220.181.38.149): icmp_seq=3 ttl=52 time=4.34 ms
64 bytes from 220.181.38.149 (220.181.38.149): icmp_seq=4 ttl=52 time=4.41 ms
```

说明：通过ping命令，向www.baidu.com主机持续发送数据，并且收到了对方应答的数据包（from 220.181.38.149），其中www.baidu.com称为域名，方便用户记忆；220.181.38.149是IP地址，用于计算机之间通信，在发送数据之前，实际上对域名和IP进行了换行，通过IP地址将数据发给了对方。