

- 1、函数复习
- 2、Linux及C高级课程安排(7天)
- 3、Linux及C高级的学习技巧
- 4、windows和ubuntu系统之间的文件的拷贝
 - 4.1 通过共享文件夹实现文件的拷贝
 - 4.1.1 创建共享文件夹
 - 4.1.2 共享文件夹在ubuntu系统的路径
 - 4.1.3 实现windows和ubuntu系统之间的文件的拷贝
 - 4.2 通过将windows中的文件拖拽的方式拷贝到ubuntu中
 - 4.3 使用右键赋值粘贴的方式，实现windows中的文件拷贝到ubuntu中
- 5、ubuntu系统的网络的配置
 - 5.1 VMware软件网络的支持
 - 5.2 配置ubuntu系统使用桥接模式连接外网
 - 5.3 配置ubuntu系统通过NAT方式连接外网
 - 5.4 ubuntu系统没有网络图标的解决办法
- 6、ubuntu系统安装软件的命令
 - 6.1 ubuntu系统离线安装软件的命令
 - 6.2 ubuntu系统在线安装软件的命令
 - 6.2.1 配置ubuntu系统的软件源
 - 6.2.2 在线软件安装相关的命令
- 7、压缩/解压缩命令
 - 7.1 压缩的命令
 - 7.2 解压缩命令
 - 7.3 归档和拆包的命令-->tar
- 8、文件操作相关的命令
 - 8.1 cat命令
 - 8.2 tail命令
 - 8.3 head命令
 - 8.4 file命令
 - 8.5 find命令
 - 8.6 grep命令
 - 8.7 |管道命令
 - 8.8 cut命令
 - 8.9 重定向的命令
- 9、通配符的使用
- 10、明日授课内容
- 11、作业

1、函数复习

- | | |
|---|---------------------------|
| 1 | 1. 函数定义 |
| 2 | 2. 函数的调用 |
| 3 | 3. 函数的形参(基本类型，数组，指针，函数指针) |
| 4 | 4. 函数的返回值(基本类型，指针类型) |
| 5 | 5. 指针函数/函数指针 |

2、Linux及C高级课程安排(7天)

- 1 1. linux命令的高级部分(1.5天)
- 2
- 3 2. shell编程(2天)
- 4 变量的定义
- 5 算数运算
- 6 控制语句:分支语句, 循环语句
- 7 函数
- 8
- 9 3. C高级部分(3天)
- 10 动态内存的分配
- 11 6种存储类型
- 12 多文件的编程
- 13 goto跳转
- 14 结构体类型
- 15 联合体类型/共用体类型
- 16 枚举类型
- 17
- 18 4. Makefile编程(0.5天)
- 19 通过make工具完成项目代码的配置和编译

3、Linux及C高级的学习技巧

- 1 1. linux命令的高级部分(1.5天)
- 2 多用
- 3 2. shell编程(2天)
- 4 理解shell的语法格式, 多写
- 5 shell语法本身不难, 难的是shell脚本文件种可以调用各种
- 6 shell命令, 难度就是shell命令的学习及使用。
- 7 要求: 掌握shell脚本文件的基本语法,
- 8 能够完成简单的脚本文件的编写,
- 9 能够看懂脚本文件即可。
- 10
- 11 3. C高级部分(3天)
- 12 记忆语法格式, 多写, 多练。
- 13
- 14 4. Makefile编程(0.5天)
- 15 掌握一个依赖的思想。
- 16 能够编写简单的Makefile文件,
- 17 能够看懂Makefile文件。

4、windows和ubuntu系统之间的文件的拷贝

4.1 通过共享文件夹实现文件的拷贝

4.1.1 创建共享文件夹

```
int a = 10000
// 3. 函数指针
// 3.1 将函数指针指向函数
result = func1(a);
printf("10000\n");
result = func2(a);
printf("10000\n");

// 3.2 将函数指针指向函数
result = (*func1)(a);
printf("20000\n");
result = (*func2)(a);
printf("20000\n");
return 0;
```

```
ubuntu:~/DC2303
ubuntu:~/DC2303
```

虚拟机(M) 选项卡(O) 帮助(H)

虚拟机设置

硬件 选项

设置 摘要 Ubuntu-18.04

常规 电源 已启用

快照 已禁用

自动保护 已禁用

客户机隔离 未加密

访问控制 未加密

VMware Tools 关闭时间同步

VNC 连接 已禁用

Unity 不受支持

设备视图 默认/默认

自动登录

高级

文件共享

共享文件夹会将您的文件显示给虚拟机中的程序。这可能为您的计算机和数据带来风险。请仅在您信任虚拟机使用您的数据时启用共享文件夹。

☐ 已禁用(D)

☒ 总是启用(E)

☐ 在下次关机或挂起前一直启用(U)

文件夹(E)

名称 主机路径

添加(A)... 移除(R) 属性(P)

确定 取消 帮助

添加windows共享ubuntu系统的文件夹及路径

添加共享文件夹向导

命名共享文件夹
如何命名此共享文件夹?

选中windows系统的共享文件夹的路径及名字

主机路径(H)

C:\Users\75043\Desktop\share

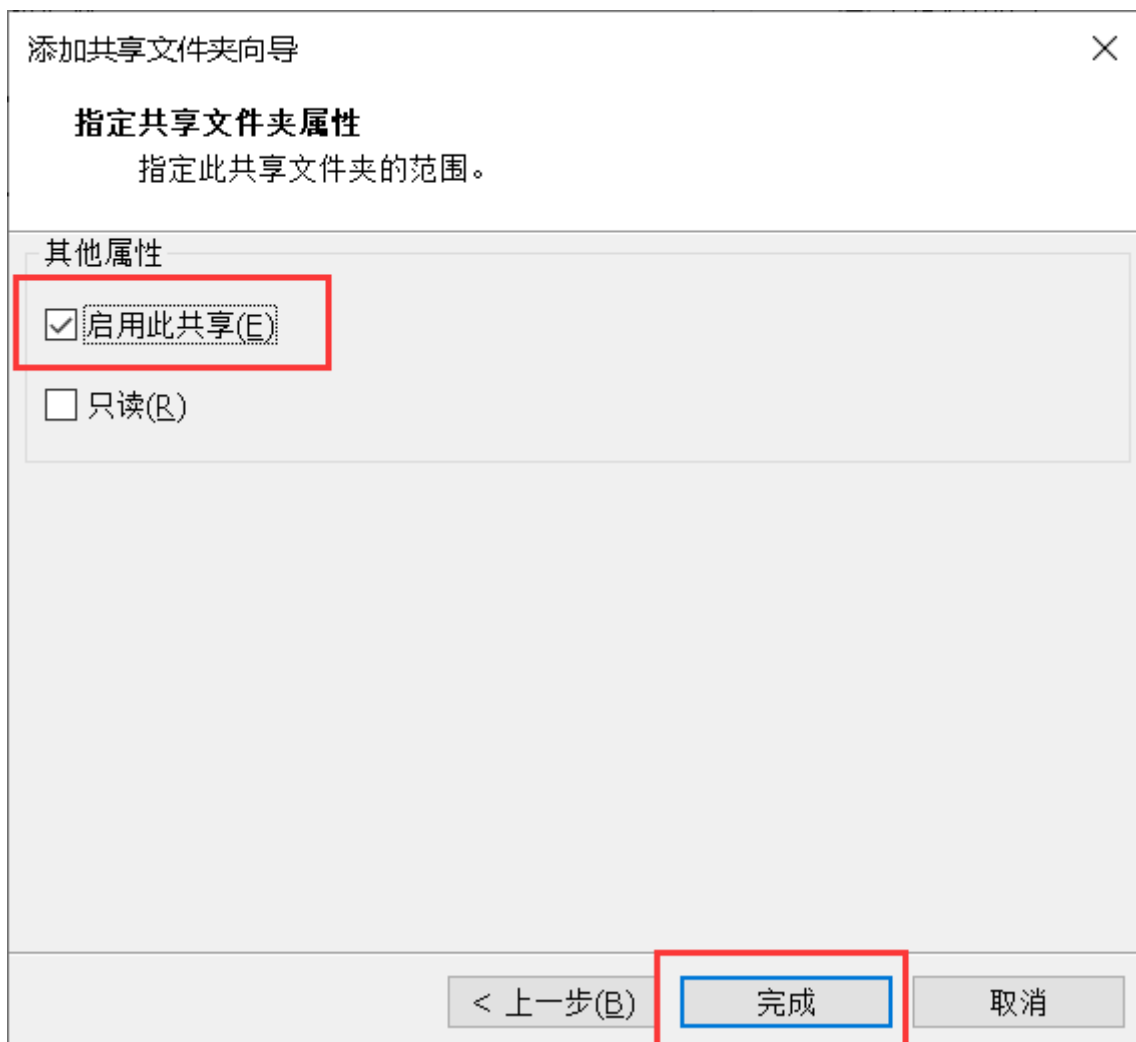
浏览(B)...

名称(A)

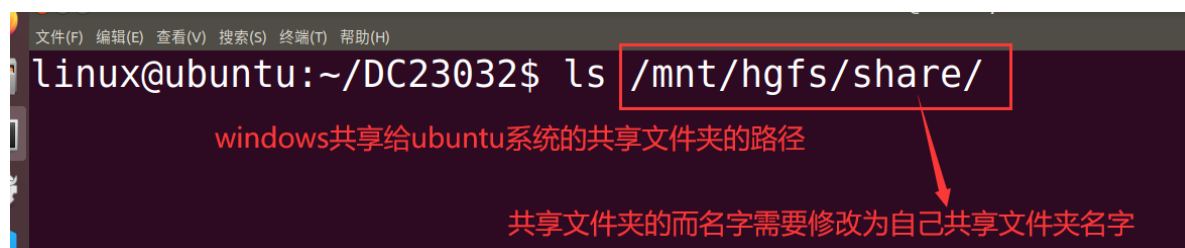
share

共享文件夹在ubuntu目录下，
显示共享文件夹的名字，
要求：写一个英文的共享文件夹名

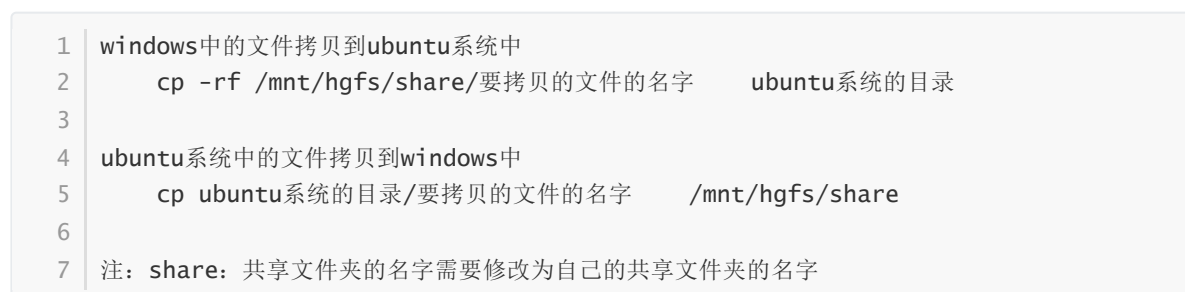
< 上一步(B) 下一步(N) > 取消



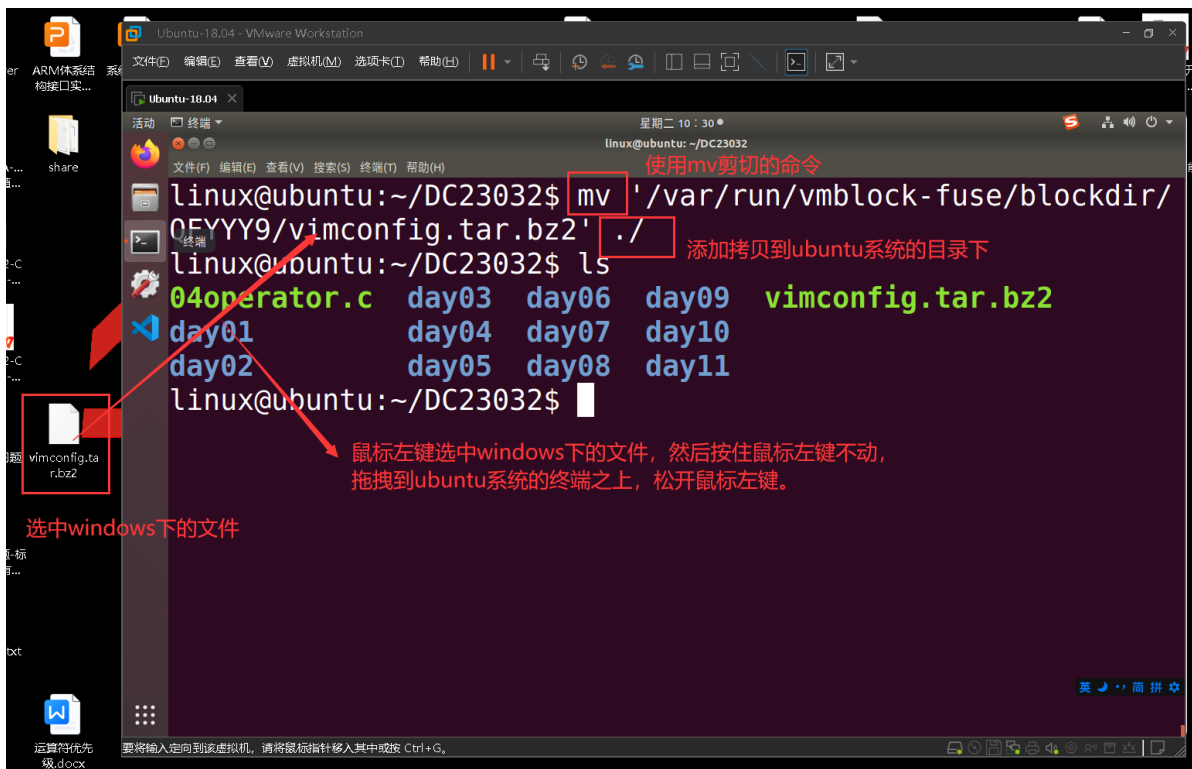
4.1.2 共享文件夹在ubuntu系统的路径



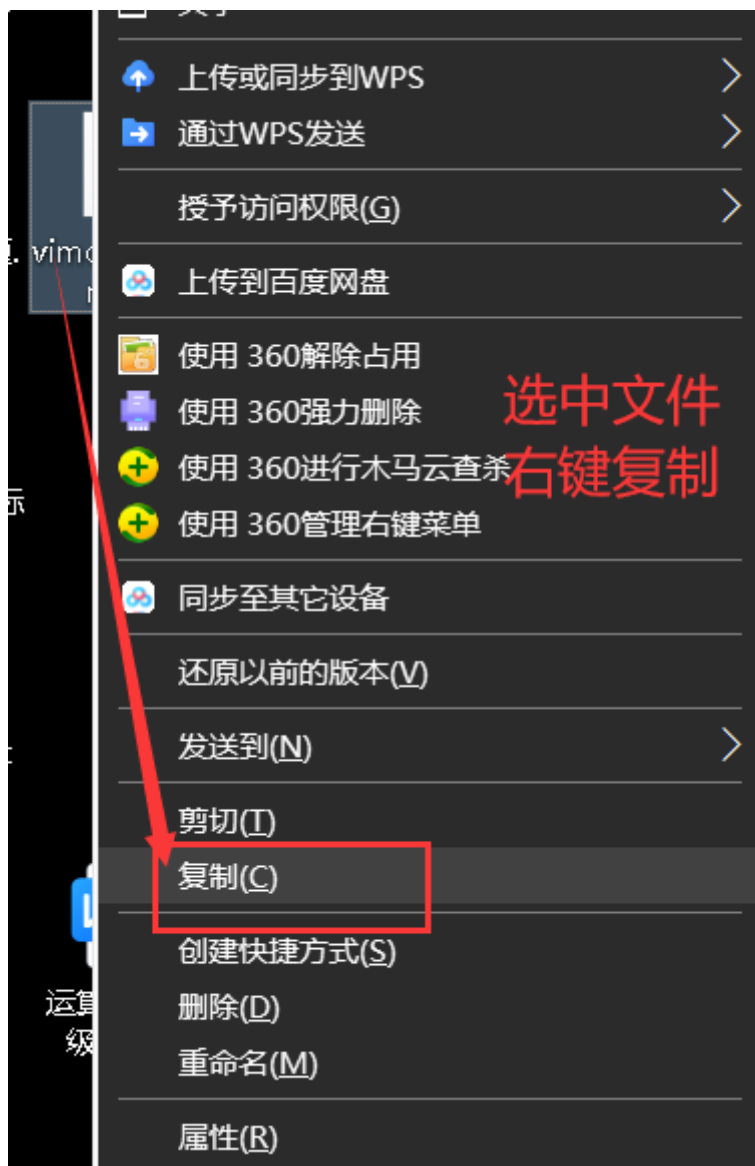
4.1.3 实现windows和ubuntu系统之间的文件的拷贝



4.2 通过将windows中的文件拖拽的方式拷贝到ubuntu中



4.3 使用右键赋值粘贴的方式，实现windows中的文件拷贝到ubuntu中

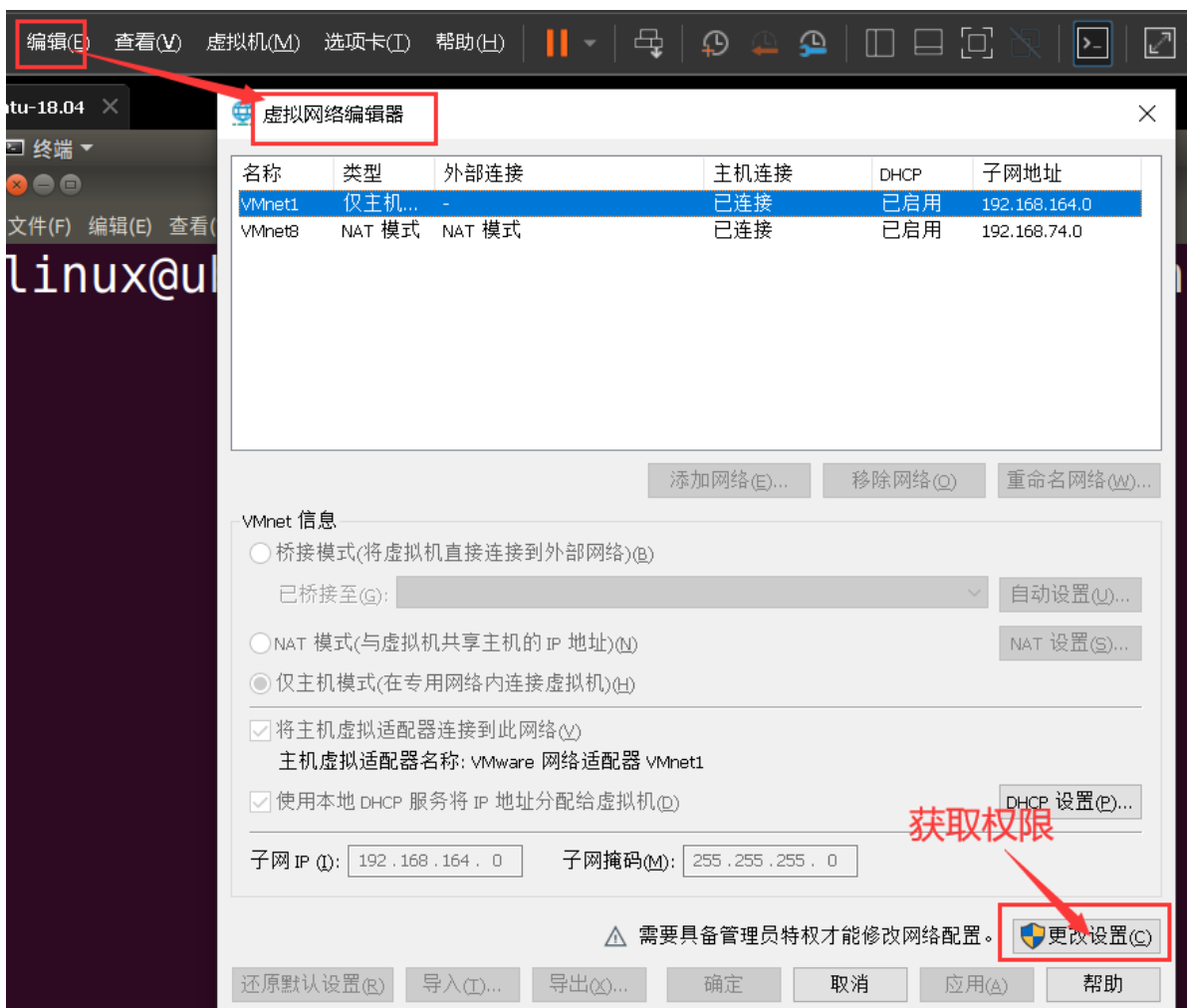




5、ubuntu系统的网络的配置

5.1 VMware软件网络的支持

- 1 桥接模式：ubuntu系统桥接到windows的网卡，
ubuntu系统和windows系统分别使用不同的IP地址连接外网。
- 2 主机模式：只能保证windows系统和ubuntu系统网络的连通。
- 3 NAT模式：ubuntu系统共享windows系统的IP地址，连接外网。

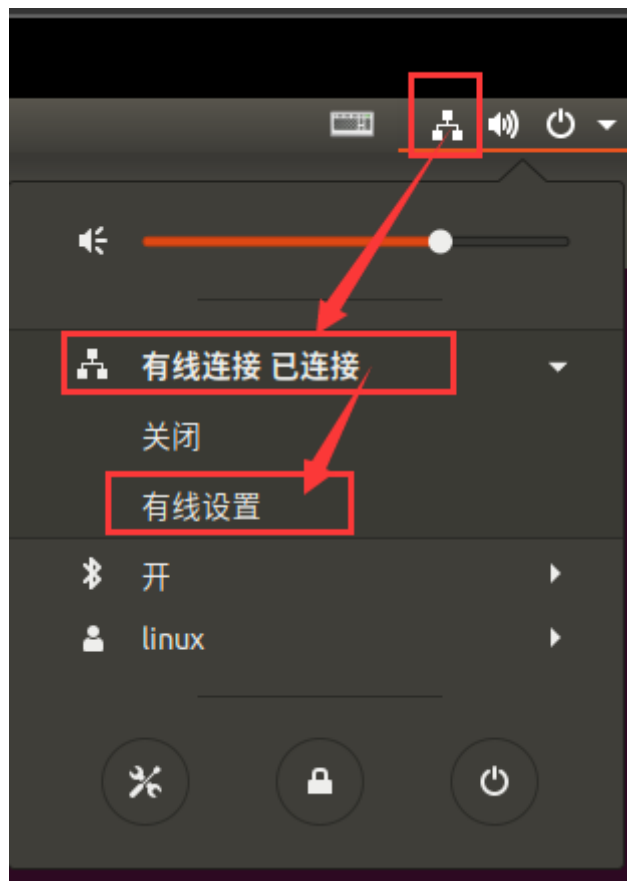
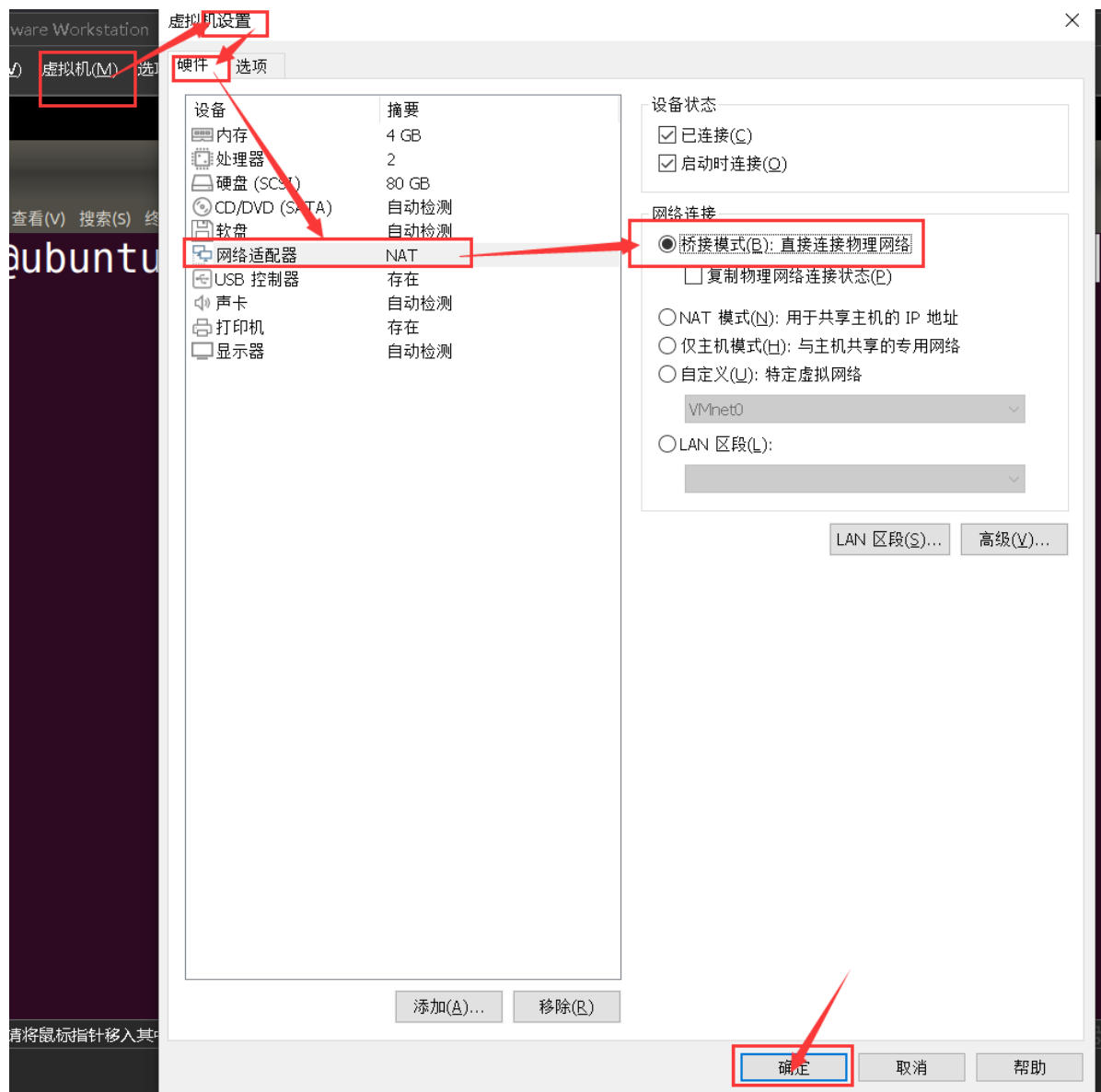


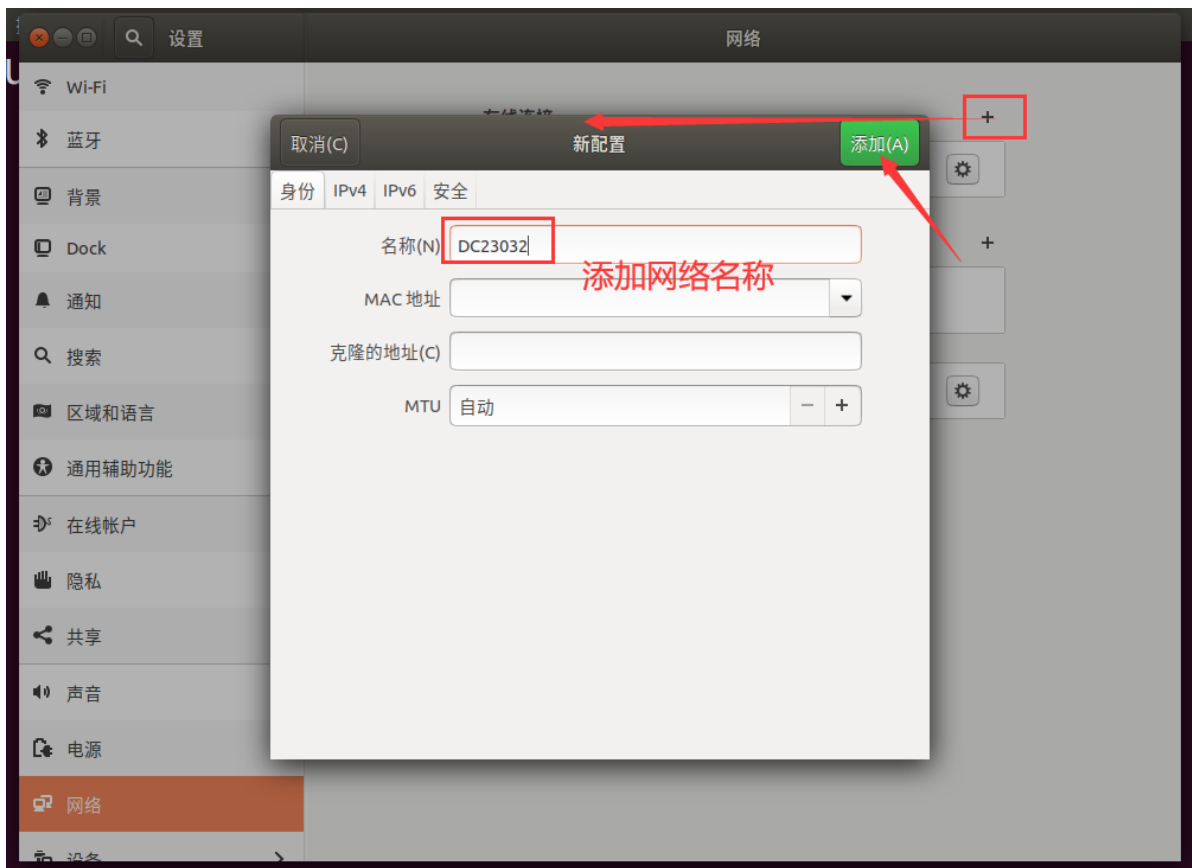


部分同学的电脑，点击更改设置之后，并没有发现：桥接模式，主机模式，NAT模式，可能缺少某种模式。

解决办法：ubuntu系统关机，然后单击“还原默认设置”，还原完成之后，就会重新添加：桥接模式，主机模式，NAT模式。

5.2 配置ubuntu系统使用桥接模式连接外网



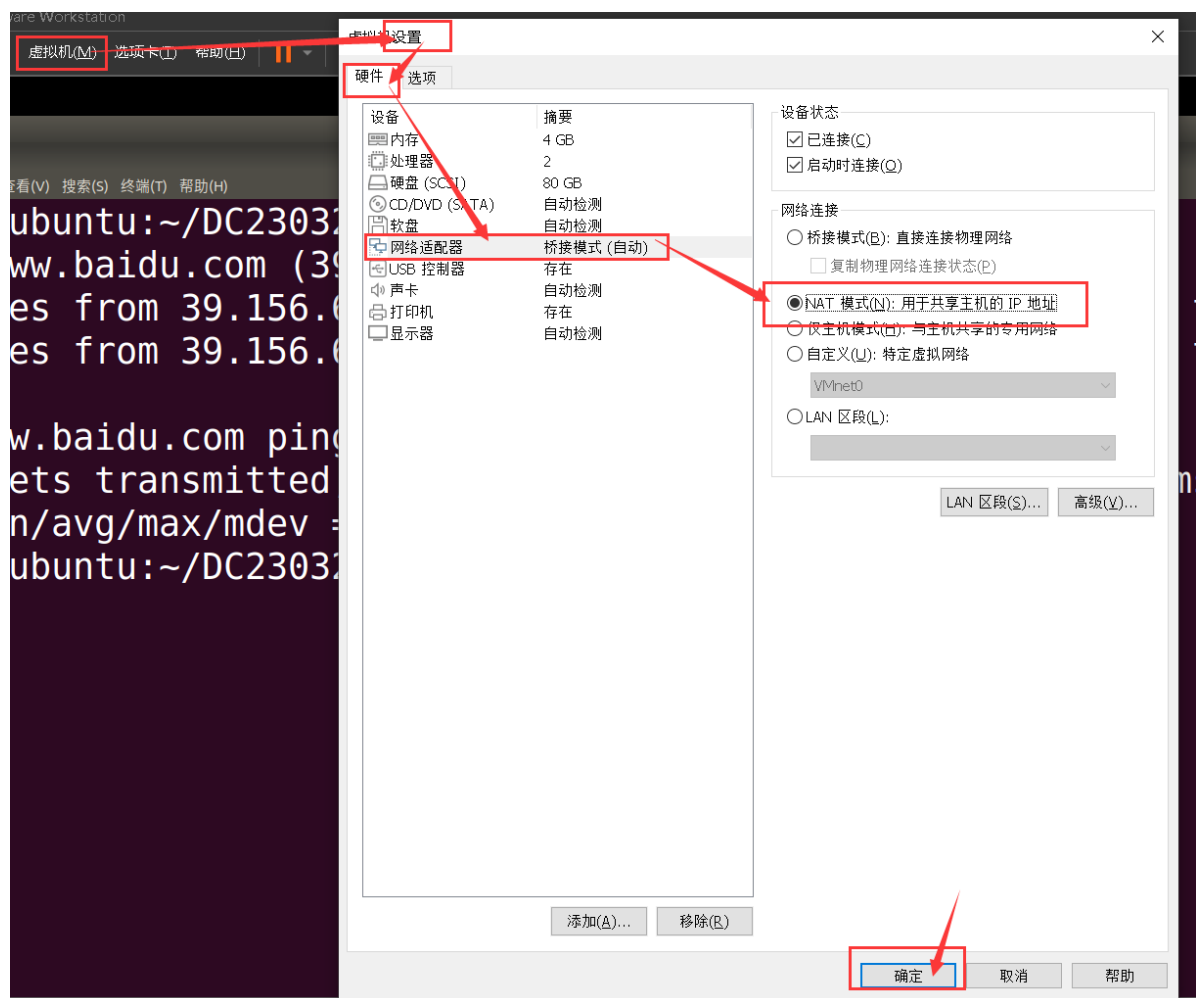


```
linux@ubuntu:~/DC23032$ ifconfig 查看ubuntu系统的IP地址
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.41 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::898f:867f:fd36:3fc6 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:a0:a6:f0 txqueuelen 1000 (以太网)
    RX packets 171937 bytes 199218433 (199.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20293 bytes 1626830 (1.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ubuntu系统通过此IP地址连接外网

```
linux@ubuntu:~/DC23032$ ping www.baidu.com 测试是否可以连接外网
PING www.baidu.com (39.156.66.18) 56(84) bytes of data.
64 bytes from 39.156.66.18 (39.156.66.18): icmp_seq=1 ttl=53 time=7.89 ms
64 bytes from 39.156.66.18 (39.156.66.18): icmp_seq=2 ttl=53 time=5.25 ms
^C
--- www.baidu.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 5.259/6.578/7.897/1.319 ms
linux@ubuntu:~/DC23032$ ctrl + c 强制退出
```

5.3 配置ubuntu系统通过NAT方式连接外网



```
linux@ubuntu:~/DC23032$ ifconfig 查看ubuntu系统的IP地址
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.74.133 netmask 255.255.255.0 broadcast 192.168.74.255
    inet6 fe80::898f:867f:fd36:3fc6 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:a0:a6:f0 txqueuelen 1000 (以太网)
    RX packets 176475 bytes 199710377 (199.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 20364 bytes 1635714 (1.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

NAT模式下
虚拟的IP地址
ubuntu系统
最终通过共享
windows的IP
地址连接外网

```

linux@ubuntu:~/DC23032$ ping www.baidu.com
PING www.baidu.com (39.156.66.18) 56(84) bytes of data.
64 bytes from 39.156.66.18 (39.156.66.18): icmp_seq=1 ttl=128 time=7.19 ms
64 bytes from 39.156.66.18 (39.156.66.18): icmp_seq=2 ttl=128 time=10.2 ms
^C
--- www.baidu.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 7.191/8.720/10.250/1.532 ms
linux@ubuntu:~/DC23032$

```

5.4 ubuntu系统没有网络图标解决办法

```

1 1. 执行关闭网络服务的命令，关闭网络服务
2   sudo service NetworkManager stop
3
4 2. 删除网络的状态文件
5   sudo rm /var/lib/NetworkManager/NetworkManager.state
6
7 3. 修改网络的配置文件
8   sudo vi /etc/NetworkManager/NetworkManager.conf
9   将以下内容：
10  managed=false
11  修改为：
12  managed=true
13 4. 重启网络服务
14   sudo service NetworkManager start
15   sudo service NetworkManager restart

```

```

1 如果经过以上操作ubuntu系统依然不可以上网，
2   ubuntu系统关机->编辑->虚拟网络编辑器->更改设置->还原默认设置
3
4 如果还原默认设置之后还是无法连接外网，卸载VMware软件，安装更高版本的VMware软件。

```

6、ubuntu系统安装软件的命令

6.1 ubuntu系统离线安装软件的命令

```

1 特点：需要下载离线的软件安装包，并且拷贝到ubuntu系统中，
2   如果离线安装的软件需要依赖其他的库，需要单独安装，
3   安装离线软件时不会自动安装所依赖的其他的库。

```

```

1 1. 离线安装软件的命令 --> dpkg
2
3 2. 离线安装的软件包的后缀
4   ubuntu: .deb后缀
5   redhat: .rpm后缀
6
7 3. 软件包的命名的方式
8   sl_3.03-17build2_amd64.deb
9   |         |         | |---> 软件包的后缀
10  |         |         | |---> 指定的处理器的架构: amd64 : 64位的操作系统

```

```
11 | | | i386 : 32位的操作系统
12 | | ---> 软件版本号
13 | ----> 软件的名字
14
15 4. 离线安装软件的命令
16 sudo dpkg -i 软件包的名字
17
18 运行软件: sl
19 oneko & ---> 后台运行程序
20
21 5. 离线卸载软件的命令
22 sudo dpkg -r 软件名
23
24 6. 查看软件的安装的信息
25 sudo dpkg -l 软件名
26
27 7. 查看软件的安装的路径信息
28 sudo dpkg -L 软件名
29
30 8. 完全卸载软件的所有内容
31 sudo dpkg -P 软件名
```

6.2 ubuntu系统在线安装软件的命令

- 1 特点：在线安装的命令要求ubuntu系统必须可以连接外网，
- 2 安装的软件默认会从对应的软件源服务器中下载并进行安装，
- 3 如果安装的软件依赖了其他的库，会先安装对应的库，
- 4 然后再安装对应的软件。

6.2.1 配置ubuntu系统的软件源

- 1 1. 常用的软件源有：阿里源，163源，网易源，清华源....
- 2
- 3 2. 本次课程使用的是清华源，清华源的网址
- 4 清华源的网址：<https://mirror.tuna.tsinghua.edu.cn/help/ubuntu/>

Ubuntu 版本: 18.04 LTS

使用官方安全更新软件源: 是

启用 proposed: 否

启用源码镜像: 否

```
# 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports main restricted universe multiverse

# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-security main restricted universe multiverse
# # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-security main restricted universe multiverse
e

deb http://security.ubuntu.com/ubuntu/ bionic-security main restricted universe multiverse
# deb-src http://security.ubuntu.com/ubuntu/ bionic-security main restricted universe multiverse

# 预发布软件源，不建议启用
# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed main restricted universe multiverse
# # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed main restricted universe multiverse
e
```

因镜像站同步有延迟，可能会导致生产环境系统不能及时检查、安装上最新的安全更新，不建议替换 security 源。

1 3. 将ubuntu系统的软件源替换为清华源

2 1> 对系统的软件源列表文件进行备份

```
3 sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak
```

5 2> 打开 sudo vi /etc/apt/sources.list文件，

6 将此文件中的内容删除之后，添加以下内容：

```
7
8 # 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
9 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main restricted
10 universe multiverse
11 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main
12 restricted universe multiverse
13 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates main
14 restricted universe multiverse
15 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates main
16 restricted universe multiverse
17 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports main
18 restricted universe multiverse
19 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports
20 main restricted universe multiverse
21
22 # deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-security main
23 restricted universe multiverse
24 # # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-security
25 main restricted universe multiverse
26
27 deb http://security.ubuntu.com/ubuntu/ bionic-security main restricted
28 universe multiverse
29 # deb-src http://security.ubuntu.com/ubuntu/ bionic-security main
30 restricted universe multiverse
31
32 # 预发布软件源，不建议启用
```

```

23 # deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed main
    restricted universe multiverse
24 # # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed
    main restricted universe multiverse
25
26 3> 执行sudo apt-get update命令
27     更新软件源，将服务器中的所有的软件列表同步到本地，
28     当安装软件时从本地检索对应的软件，如果检索到则到对应的
29     服务器中下载软件并安装。
30
31     由于软件源服务器中的软件会不定期的更新，软件版本会发生变化，
32     因此每过一段时间，再次使用在线安装的命令之前就需要执行更新软件源的命令。
33
34 4> 如果在更新源的过程中出现以下错误的解决办法：
35     sudo vi /etc/apt/sources.list文件，
36     将所有的https修改为http。
37     替换完成之后，重新执行sudo apt-get update
38

```

```

Linux 64 位
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
活动 终端
linux@ubuntu: ~
忽略:2 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-updates InRelease
命中:3 http://packages.microsoft.com/repos/code stable InRelease
忽略:4 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-backports InRelease
忽略:5 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-security InRelease
错误:6 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic Release
Certificate verification failed: The certificate is NOT trusted. The certificate chain uses expired certificate. Could not
handshake: Error in the certificate verification, [IP: 101.6.15.130 443]
错误:7 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-updates Release
Certificate verification failed: The certificate is NOT trusted. The certificate chain uses expired certificate. Could not
handshake: Error in the certificate verification, [IP: 101.6.15.130 443]
错误:8 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-backports Release
Certificate verification failed: The certificate is NOT trusted. The certificate chain uses expired certificate. Could not
handshake: Error in the certificate verification, [IP: 101.6.15.130 443]
错误:9 https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-security Release
Certificate verification failed: The certificate is NOT trusted. The certificate chain uses expired certificate. Could not
handshake: Error in the certificate verification, [IP: 101.6.15.130 443]
正在读取软件包列表... 完成
E: 仓库 "https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic Release" 不再含有 Release 文件。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
E: 仓库 "https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-updates Release" 不再含有 Release 文件。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
E: 仓库 "https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-backports Release" 不再含有 Release 文件。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
E: 仓库 "https://mirrors.tuna.tsinghua.edu.cn/ubuntu bionic-security Release" 不再含有 Release 文件。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
linux@ubuntu:~$ http://mirror

```

6.2.2 在线软件安装相关的命令

```

1 1> 在线安装软件的命令
2     sudo apt-get install 软件名
3
4     eg :
5     sudo apt-get install sl      // 小火车
6     sudo apt-get install oneko  //安装追鼠标的小猫
7     sudo apt-get install frozen-bubble //安装泡泡龙
8     sudo apt-get install cmatrix //代码雨
9     sudo apt-get install bastet //俄罗斯方块
10
11 2> 卸载软件的命令
12     sudo apt-get remove 软件名
13     sudo apt-get autoremove 软件名    ---> 清理的更干净
14
15 3> 下载软件包
16     sudo apt-get download 软件名

```

```
17
18 4> 下载软件的源码
19     sudo apt-get source 软件名
20
21     下载软件源码时，出现以下错误的信息：
22     正在读取软件包列表... 完成
23     E: 您必须在 sources.list 中指定代码源(deb-src) URI
24
25     解决办法：打开sudo vi /etc/apt/sources.list文件，
26     将deb-src开头的行取消注释，即删除行开始的“#”，
27     保存退出之后，重新执行sudo apt-get update,更新软件源。
```

7、压缩/解压缩命令

7.1 压缩的命令

```
1 1. xz压缩命令
2     xz    要压缩的文件的名字      ---> 压缩得到.xz后缀的压缩文件
3
4 2. bzip2压缩命令
5     bzip2 要压缩的文件的名字      ---> 压缩得到.bz2后缀的压缩文件
6
7 3. gzip压缩命令
8     gzip  要压缩的文件的名字      ---> 压缩得到.gz后缀的压缩文件
9
10 注意：
11     1> 只能对单个文件进行压缩，不可以对目录进行压缩
12     2> 压缩完成之后源文件消失，得到压缩文件
13     3> 压缩效率：xz > bzip2 > gzip
14     4> 压缩时间：xz < bzip2 < gzip
```

```
1 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh Image
2 -rwxrw-rw- 1 linux linux 19M 4月 18 11:10 Image
3
4 linux@ubuntu:~/DC23032/c_advanced/day01$ xz Image
5 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh Image.xz
6 -rwxrw-rw- 1 linux linux 5.4M 4月 18 11:10 Image.xz
7
8 linux@ubuntu:~/DC23032/c_advanced/day01$ bzip2 Image
9 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh Image.bz2
10 -rwxrw-rw- 1 linux linux 6.7M 4月 18 11:10 Image.bz2
11
12
13 linux@ubuntu:~/DC23032/c_advanced/day01$ gzip Image
14 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh Image.gz
15 -rwxrw-rw- 1 linux linux 7.2M 4月 18 11:10 Image.gz
16
```

7.2 解压缩命令


```
1 1. unxz解压缩命令
2     unxz 压缩包的名字.xz
3
4 2. bunzip2解压缩命令
5     bunzip2 压缩包的名字.bz2
6
7 3. gunzip解压缩命令
8     gunzip 压缩包的名字.gz
9
10 注:
11 1> 解压缩之后得到文件, 压缩文件消失。
```

```
1 linux@ubuntu:~/DC23032/c_advanced/day01$ unxz Image.xz
2
3 linux@ubuntu:~/DC23032/c_advanced/day01$ bunzip2 Image.bz2
4
5 linux@ubuntu:~/DC23032/c_advanced/day01$ gunzip Image.gz
6
```

7.3 归档和拆包的命令-->tar

```
1 归档: 将多个文件合并成一个文件, 并没有压缩。
2 拆包: 将单个文件拆分成多个文件。
3
4 1. 参数:
5     -c : 归档
6     -x : 拆包
7     -v : 回显详细的过程
8     -f : 指定归档和拆包的文件的名字
9
10 2. 归档的命令:
11     tar -vcf 归档后的文件的名字.tar 要归档的文件或目录的名字
12
13     eg: tar -vcf s1-3.03.tar s1-3.03/
14
15     drwxr-xr-x 4 root root    4096 4月  18 13:57 s1-3.03
16     -rw-rw-r-- 1 linux linux 225280 4月  18 14:46 s1-3.03.tar
17     注: 此过程只是单纯的归档, 并没有进行压缩。
18
19 3. 拆包的命令:
20     tar -vxf 要拆包的文件的名字.tar
21
22 4. 归档的同时进行压缩的参数
23     -J : 采用xz的格式进行压缩
24     -j : 采用bzip2的格式进行压缩
25     -z : 采用gzip的格式进行压缩
26
27 5. 归档的同时并进行压缩的命令
28     1> xz格式的归档压缩命令
29     tar -Jvcf 归档压缩后的文件的名字.tar.xz 要归档的文件或目录的名字
30     tar -Jvxf 要解压缩的文件的名字.tar.xz
```

```
31
32 2> bzip2格式的归档压缩命令
33 tar -jvcf  归档压缩后的文件的名字.tar.bz2  要归档的文件或目录的名字
34 tar -jvxf  要解压缩的文件的名字.tar.bz2
35
36 3> gzip格式的归档压缩命令
37 tar -zvcf  归档压缩后的文件的名字.tar.gz  要归档的文件或目录的名字
38 tar -zvxf  要解压缩的文件的名字.tar.gz
39
40 4> 万能的拆包和解压缩的命令
41 tar -vxvf  要解压缩的文件的名字.tar.xx
42 xx:需要替换为xz, gz, bz2
```

```
1 linux@ubuntu:~/DC23032/c_advanced/day01$ tar -vcf Image.tar Image
2 Image
3 linux@ubuntu:~/DC23032/c_advanced/day01$ tar -Jvcf Image.tar.xz Image
4 Image
5 linux@ubuntu:~/DC23032/c_advanced/day01$ tar -jvcf Image.tar.bz2 Image
6 Image
7 linux@ubuntu:~/DC23032/c_advanced/day01$ tar -zvcf Image.tar.gz Image
8 Image
9
10
11 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh
12 -rwxrw-rw- 1 linux linux 19M 4月 18 11:10 Image
13 -rw-rw-r-- 1 linux linux 19M 4月 18 14:57 Image.tar
14 -rw-rw-r-- 1 linux linux 6.7M 4月 18 14:57 Image.tar.bz2
15 -rw-rw-r-- 1 linux linux 7.2M 4月 18 14:57 Image.tar.gz
16 -rw-rw-r-- 1 linux linux 5.4M 4月 18 14:56 Image.tar.xz
17
18 万能的拆包和解压缩的命令
19 linux@ubuntu:~/DC23032/c_advanced/day01$ tar -vxvf Image.tar.xz
20
```

8、文件操作相关的命令

8.1 cat命令

```
1 作用：回显文件中的内容到终端
2 格式：
3     cat 文件名1 文件名2 ....
4     cat -n 文件名1 ---> 回显文件中的行号
```

8.2 tail命令

```
1 作用：默认回显文件中的内容的后10行，如果不足10行全部显示
2 格式：
3     tail  文件名
4     tail -n  文件名  --> n是一个整型的数字，回显文件的后n行
5
6 举例：
7     tail  hello.c  ---> 回显文件的后10行
8     tail -5  hello.c  ---> 回显文件的后5行
```

8.3 head命令

```
1 作用：默认回显文件中的内容的前10行，如果不足10行全部显示
2 格式：
3     head  文件名
4     head -n  文件名  --> n是一个整型的数字，回显文件的前n行
5
6 举例：
7     head  hello.c  ---> 回显文件的前10行
8     head -5  hello.c  ---> 回显文件的前5行
```

8.4 file命令

```
1 作用：查看文件的属性信息
2 格式：
3     file  文件名
4
5 举例：
6     linux@ubuntu:~/DC23032/c_advanced/day01$ file a.out
7
8     a.out: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV),
dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux
3.2.0, BuildID[sha1]=6046491338b30aa03f707ab952cc9e094cb24994, not
stripped
9
10    解释：
11    a.out:  ---> 可执行文件的名字
12    ELF 64-bit LSB shared object,  ---> 可执行文件的格式，ELF格式
13    x86-64,  ---> 处理器的架构
14    version 1 (SYSV),
15    dynamically linked,  ---> 动态链接
16    interpreter /lib64/ld-linux-x86-64.so.2,
17    not stripped --> 没有进行压缩
18
19    GNU组织中的的压缩命令：strip
20    可以对可执行文件进行压缩，但是不影响程序的运行，
21    删除可执行文件中无用的符号信息。
22
23 举例：
24    linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh a.out  --> 压缩之前
25    -rwxrwxr-x 1 linux linux 8.2K 4月 18 15:20 a.out
26    linux@ubuntu:~/DC23032/c_advanced/day01$ ./a.out
```

```

27 hello world!
28
29 linux@ubuntu:~/DC23032/c_advanced/day01$ strip a.out --> 压缩可执行文件命令
30 linux@ubuntu:~/DC23032/c_advanced/day01$ ls -lh a.out --> 压缩之后
31 -rwxrwxr-x 1 linux linux 6.0K 4月 18 15:24 a.out
32 linux@ubuntu:~/DC23032/c_advanced/day01$ ./a.out
33 hello world!
34

```

8.5 find命令

```

1 作用：按照文件的名字从某个目录下查找对应的文件
2 格式：
3     find 要查找的路径 -name 要查找的文件的名字
4
5 举例：
6     find . -name hello.c ---> 从当前目录下查找hello.c文件
7     find ../day01 -name hello.c ---> 从../day01目录下查找hello.c文件
8
9 使用场合：
10    比如从内核源码目录下的6万多文件中查找某一个文件时，
11    可以使用find命令，如果找到对应的文件，会回显文件所在的路径。
12
13 eg:
14 linux@ubuntu:~/DC23032/c_advanced/day01$ find ~/DC23032/ -name 01hello.c
15 /home/linux/DC23032/c_basic/day02/01hello.c
16

```

8.6 grep命令

```

1 作用：从文件中查找某个字符串的命令
2 格式：
3     grep -参数 "要查找的字符串" 文件名/目录名
4
5 参数：
6     -n : 显示查找到的内容的行号
7         vi 文件名 +行号 ---> 打开文件，并将光标定位到对应的行
8     -R : 如果从目录中的文件中查找对应的字符串，需要加-R参数，
9         表示递归从所有的文件中查找符合要求的字符串。
10    -i : 查找的字符串不区分大小写。
11    -w : 按照单词进行字符串的查找。
12
13 案例：
14     grep -n "main" hello.c
15         ---> 从hello.c文件中查找main字符串，并显示行号
16     grep -nR "main" ./
17         ---> 从当前目录下所有的文件中查找main字符串，并显示行号
18     grep -niR "MAIN" ./
19         ---> 从当前目录下所有的文件中查找MAIN字符串，不区分大小写，并显示行号
20     grep -nwR "int" ./
21         ---> 从当前目录下按照单词查找int字符串，并显示行号
22     grep -nR "^int" ./

```

```

23      ---> 查找文件每行的开头为int的行，并显示行号
24      grep -nR "int$" ./
25      ---> 查找文件中每行的结尾为int的行，并显示行号
26      grep -nR "^int$" ./
27      ---> 查找文件的每行的开头为int,结尾为int的行，并显示行号
28
29 使用场合：
30      在调试代码时，当运行代码出错时，会调用printf函数打印一个字符串，
31      报"***失败"，此时就可以使用grep搜索字符串，查看代码的哪个位置调用了
32      printf打印了此字符串。
33
34      retVal = putchar('\n');
35      if (retVal == EOF)
36      {
37          printf("send char failed\n");
38          return -1;
39      }
40

```

8.7 | 管道命令

```

1 作用： | 管道符， 将前一个命令的输出作为后一个命令的输入
2 格式：
3      shell命令1 | shell命令2 | shell命令3 | .....
4
5 举例：
6      使用管道的方式查看hello.c文件是否存在？
7      ls | grep "hello.c"
8
9      使用cat/head/tail输出/etc/passwd文件的第41行，并显示行号。
10     cat -n /etc/passwd | head -41 | tail -1
11     cat -n /etc/passwd | tail -4 | head -1
12     tail -4 /etc/passwd | head -1 ---> 不会显示行号
13
14 使用场合：
15     可以用来过滤信息，缩小查找信息的范围。

```

8.8 cut命令

```

1 作用：从文件中截取部分域对应的字符串
2 格式：
3      cut -d "分隔符" -f 指定剪切的域 文件名
4      |      |      |      |      |      | ---> 从哪个文件中剪切字符串
5      |      |      |      |      |      | ----> 截取文件中每一行对应的域
6      |      |      |      |      |      | ----> 如果截取的域连续，使用-隔开，比如：1-3
7      |      |      |      |      |      | ----> 如果截取的域不连续，使用,隔开，比如：1,3
8      |      |      |      |      |      | ----> 指定截取文件每一行对应的域的参数
9      |      |      |      |      |      | ----> 每个域之间的分隔符
10     |      |      |      |      |      | ----> 指定分隔符的参数
11     |      |      |      |      |      | ----> 字符串剪切的命令

```

```

12
13 /etc/passwd文件中的内容的解析:
14 linux:x:1000:1000:linux,,,:/home/linux:/bin/bash
15 | | | | | | |-> 使用bash解析器解析linux命令
16 | | | | | | |--> 家目录
17 | | | | | |----> 用户信息
18 | | | |----> 组ID
19 | | |----> 用户ID
20 | |----> 密码
21 |----> 用户名
22
23 分隔符: ":"
24 linux:x:1000:1000:linux,,,:/home/linux:/bin/bash
25 1 2 3 4 5 6 7
26
27 41:linux:x:1000:1000:linux,,,:/home/linux:/bin/bash
28 1 2 3 4 5 6 7 8
29
30 举例:
31 从/etc/passwd截取用户名及对应的家目录
32 cut -d ":" -f 1,6 /etc/passwd
33
34 从/etc/passwd文件中, 截取linux对应的用户名, 密码, 及家目录
35 head -41 /etc/passwd | tail -1 | cut -d ":" -f 1-2,6
36 grep -n "^linux" /etc/passwd | cut -d ":" -f 2-3,7
37 grep "linux" /etc/passwd | cut -d ":" -f 1-2,6
38

```

```

1 练习题:
2 1. 将/etc/passwd文件中的linux用户的家目录对应的字符串截取出来。
3   grep "linux" /etc/passwd | cut -d ":" -f 6
4 2. 将/etc/passwd文件中的linux用户的用户名和密码对应的字符串截取出来。
5   grep -n "linux" /etc/passwd | cut -d ":" -f 2-3
6 3. 将/etc/passwd文件中的linux用户的用户名,家目录,
7   使用的bash解析器对应的字符串截取出来。
8   grep "linux" /etc/passwd | cut -d ":" -f 1,6,7
9 4. 将file hello.c 文件中的C字符截取出来。
10  file hello.c | cut -d " " -f 2

```

8.9 重定向的命令

```

1 > : 将命令的执行结果重定向到某个文件中, 如果文件不存在, 则创建文件,
2   并将结果写到文件中。如果文件存在, 在将文件中的内容清空之后,
3   再写入对应的内容。
4 >> : 将命令的执行结果重定向到某个文件中, 如果文件不存在, 则创建文件,
5   并将结果写到文件中。如果文件存在, 则将写入的内容从文件的末尾开始追加。
6
7 cat hello.c world.c > helloworld.c ----> 实现两个文件的拼接合并
8
9 使用场合:
10  可以实现多个文件的拼接,
11  将程序执行的结果重定向到对应的日志文件中。

```

9、通配符的使用

```
1 在终端中执行以下命令，创建以下文件：
2 touch a1.c a2.c a3.c a4.c aa.c ab.c ac.c ad.c aaa.c abb.c acc.c add.c
3
4 1. * ---> 匹配多个字符
5     ls a*.c ---> 查找文件名第一字符为a,后边多个任意字符
6     a1.c a2.c a3.c a4.c aaa.c aa.c abb.c ab.c ac.c acc.c ad.c
7     add.c
8
9 2. ? ---> 匹配一个字符
10    ls a?.c ---> 文件名有2个字符，第一个字符为a,第二个字符任意。
11    a1.c a2.c a3.c a4.c aa.c ab.c ac.c ad.c
12
13 3. [] ----> 匹配[]中的任意一个字符
14    ls a[abcd].c ---> 第二个字符为abcd中的任意一个
15    aa.c ab.c ac.c ad.c
16    ls a[a-d].c ---> 第二个字符为a-d中的任意一个
17    aa.c ab.c ac.c ad.c
18    ls a[^a].c ---> 第二个字符为除了a字符的任意一个
19    a1.c a2.c a3.c a4.c ab.c ac.c ad.c
20    ls a[1-4].c ---> 第二个字符为1-4中的任意一个
21    a1.c a2.c a3.c a4.c
22    ls a[1234].c ---> 第二个字符为1234中的任意一个
23    a1.c a2.c a3.c a4.c
24
25    rm hello.[^c] ---> 删除文件名为hello,结尾不是.c的全部删除。
26
```

10、明日授课内容

1. 文件权限操作相关的命令
2. 软链接文件和硬链接文件的创建
3. 磁盘操作相关的命令
4. 网络相关的概念及命令
5. shell编程(概念介绍及shell中变量的定义及使用)

11、作业

- 1 熟悉今天的shell命令。
- 2 C试卷。

