

# Linux指令

[百度 Vmware centos7安装参考](#)

[百度 centos7镜像下载地址](#)

选everything

## 查看已经安装的CentOS版本信息

```
cat /etc/issue
```

## vi/vim文本编辑器

vi 正常模式/编辑模式/命令行模式(底行模式)

```
vi 输入行号 shift+g 跳转行号    gg跳转第一行    G跳转伟行
yy复制,
p粘贴
u撤销
dd删行(centos6 dd centos7 d删行 dd删单字符)
```

## vim 字符串替换：底行模式下：

%s/源字符串/目的字符串/g

```
%s/https/http/g
```

## 切换用户

su - 切换用户名

例如： su - root

## 切换到指定运行级别的指令

基本语法

init[012356]

(centos6)

vim /etc/inittab

(centos7)

systemctl get-default

# To view current default target, run:

# systemctl get-default

#

# To set a default target, run:

# systemctl set-default TARGET.target

Default runlevel. The runlevels used are:

- 0 - halt (Do NOT set initdefault to this)
- 1 - Single user mode
- 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
- 3 - Full multiuser mode
- 4 - unused
- 5 - X11
- 6 - reboot (Do NOT set initdefault to this)

centos 7 inittab文件已经不再使用了，而是使用systemd了。

multi-user.target多用户界面，类似以前的level3

graphical.target图形界面，类似以前的level5

## linux如何修改root密码

(前提:不能用远程连接，相当于你接触这台虚拟机所在的电脑，所以安全性不会降低)

- > 开机
- > 在引导时输入 回车键
- > 看到一个界面输入
- > 看到一个新的界面，选择第二行(editor kernel编辑内核) 再输入e
- > 在这行输入1 然后再输入回车键
- > 再次输入b，然后将进入单用户模式。

此时，我们可以进入单用户模式，使用passwd命令修改root密码。 1

## 帮助指令 man/help

```
main ls
help cd
```

**pwd指令: Print the name of thhe current working directory. 显示当前目录的绝对路径**

**ls指令: ls - list directory contents**

```
ls -l
```

**cd指令: Change the shell working directory. 切换目录**

绝对路径: 例如 cd /home

相对路径: 从当前工作目录开始定位到需要的目录去 例如现在 pwd 显示在 /bin 下。 然后

cd ../home

使用相对路需要知道当前在哪个路径下才能写出这个指令

**mkdir指令: mkdir - make directories 创建目录**

一次创建多级目录，带上 -p参数 mkdir -p /home/animal/tiger

-p意思为: no error if existing, make parent directories as needed

如果存在(错误)的话，不报错，根据需要创建父目录

## rmmdir指令：rmmdir - remove empty directories 删除目录

如果需要删除非空目录，需要使用 `rm -rf` 指定目录

## touch指令：创建空文件

```
touch 文件名称
touch hello.txt hello2.txt
```

## cp指令：cp - copy files and directories 拷贝指令

```
cp [选项] source dest
cp [选项] 源文件 拷贝的目录
常用选项 -r: 递归复制整个文件夹
-R, -r, --recursive 递归 adj.
copy directories recursively 递归地 adv.
# 准确定位源目录和目标目录
```

## rm指令：rm - remove files or directories 移除文件或目录

```
-r 递归删除整个文件夹
-f 强制删除，不提示
-f, --force
ignore nonexistent files, never prompt
```

## mv指令 mv - move (rename) files 移动文件与目录或重命名

移动(如果有则重命名)

## cat指令：cat - concatenate files and print on the standard output 以只读方式查看文件内容

```
-n 显示行号
|more 管道符
cat -n /etc/profile |more 使用空格 翻页
```

## more指令：一个基于VI编辑器的文本过滤器，它以全屏幕的方式按页显示文本文件的内容

```
more /etc/profile 空格/ctrl + f 翻页 enter 下一行 q退出more模式 看上一页 ctrl+b
```

####vim cat more less 翻页/上一页

空格/ctrl + f 翻页 enter 下一行 q退出more模式 看上一页 ctrl+b

## less指令：less - opposite of more 用来分屏查看文件内容

**less**比**more**指令更强大支持各种显示终端，并不是一性将整个文件加载之后才显示，而是根据显示需要加载内容。对于大型文件的显示具有较高效率

空格/**PgUp** 翻页/下一页

**enter** 下一行

**q**退出**more**模式

上一页 **PgDn**

/字符串 向下搜寻字符串

**n**: 向下查找

**N**: 向上查找

?字符串 向上搜寻

字符串 **n**: 向下查找 **N**: 向上查找

## echo指令

**echo**输出内容到控制台

输出环境变量

```
[root@pengtao ~]# echo $PATH
```

```
/usr/lib64/qt-
```

```
3.3/bin:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bin
```

## head指令

显示文件的开头部分内容，默认文件的前**10**行

**head -n 5** 文件路径文件名 制定显示文件的多少行

例如

```
[root@pengtao /]# head -n 5 /etc/profile
```

```
# /etc/profile
```

## tail指令

**tail**用于输出文件尾部的内容，默认后**10**行

**tail -n 5** 文件 查看文件后**5**行的文件内容，**5**可以是任何行数

**tail -f** 文件 实时追踪该文档的所有更新

查看文件最后**5**行并且实时追踪该文档的更新

```
[root@pengtao /]# tail -5f /etc/profile
```

```
fi
```

```
done
```

```
unset i
```

```
unset -f pathmunge
```

## ln指令

软链接也叫符号链接，类似于windows的快捷方式，主要存放了链接其他文件的路径

例如：

```
ln -s [源文件或目录] [软链接名]
```

## history指令

显示所有执行过的指令 `history`  
显示最近执行过的10个指令 `history 10`  
执行历史编号为5的指令 `history`查看编号      !编号

## 时间日期类:

```
date "+%Y-%m-%d %H:%M:%S"
显示日期时间
[root@pengtao /]# date "+%Y-%m-%d %H:%M:%S"
2022-01-25 22:49:25
[root@pengtao /]#
```

## date指令: 显示当前日期

```
date
date "+%Y-%m-%d%H:%M:%S"

[root@localhost /]# date
Tue Jan 25 11:07:12 PST 2022
[root@localhost /]# date "+%Y-%m-%d%H:%M:%S"
2022-01-2511:07:15
设置系统时间
date -s "2022-02-10 14:34:00"
```

## cal指令 calendar日历指令

```
cal
cal 2022
```

## find指令

find [搜索范围] [选项]

案例:

- 按文件名-name 根据名称hello.txt 查找/home目录下的 文件hello.txt  
`find /home -name hello.txt`
- 按文件所有者 根据文件名 查找  
`find /opt -user nobody`
- 查找整个linux系统下 按大小 查找 + 大于 -小于 等于直接写 20M  
`find /usr -size +20M`  
`find /home -size -20k`  
`find / -size 20M`
- 查询 / 目录下所有txt文件  
`find / name *.txt`  
1M = 1024k

## locate指令：快速定位文件路径

locate指令基于数据库进行查询，第一次运行该指令前，必须使用 updatedb指令创建locate数据库

```
[root@pengtao mysql]# clear
[root@pengtao mysql]# updatedb
[root@pengtao mysql]# locate hello.txt
/home/hello.txt
[root@pengtao mysql]#
```

## grep指令和管道符号 |

grep过滤查找，管道符“|”表示将前一个命令的处理结果输出传递给后面的命令进行处理

-n 显示匹配行及行号

-i 忽略字母大小写

案例：

```
[root@pengtao mysql]# cat /home/hello.txt
public class OrderFeignMain80 {

    public static void main(String[] args) {
        System.out.println("hello world");
        SpringApplication.run(OrderFeignMain80.class, args);
    }
}
[root@pengtao mysql]# cat /home/hello.txt | grep world
System.out.println("hello world");
[root@pengtao mysql]# cat /home/hello.txt | grep -n world
4:         System.out.println("hello world");
[root@pengtao mysql]#
```

###压缩和解压缩指令

## gzip/gunzip

gzip 压缩文件 (\*.gz文件 压缩后原文件不保留)

gunzip 解压缩文件

```
[root@pengtao home]# ls
animal a.txt c c.txt hello.txt
[root@pengtao home]# gzip hello.txt c.txt
[root@pengtao home]# ls
animal a.txt c c.txt.gz hello.txt.gz
[root@pengtao home]# gunzip hello.txt.gz
[root@pengtao home]# ls
animal a.txt c c.txt.gz hello.txt
[root@pengtao home]#
```

## zip/unzip

zip 压缩文件/ unzip解压缩文件

zip [选项]xxx.zip 将要压缩的内容

unzip [选项]

```
将home目录下所有文件压缩成home.zip
zip -r home.zip /home/

zip -r home.zip /home/

[root@pengtao home]# unzip -d /opt/tmp/ home.zip
[root@pengtao home]# ll /opt/tmp/ /opt/tmp/home
/opt/tmp/:
总用量 4
drwxr-xr-x. 7 root root 4096 1月 26 20:19 home

/opt/tmp/home:
总用量 36
drwxr-xr-x. 3 root root 4096 1月 25 04:24 animal
-rw-r--r--. 1 root root 874 1月 25 06:28 a.txt
-rw-r--r--. 1 root root 12 1月 25 06:31 c
-rw-r--r--. 1 root root 436 1月 25 06:34 c.txt.gz
-rw-r--r--. 1 root root 190 1月 26 20:07 hello.txt
drwx-----. 25 root root 4096 1月 25 03:06 pengtao
drwx-----. 4 root root 4096 1月 23 23:35 pt
drwx-----. 4 root root 4096 1月 23 23:44 yangxiao
drwx-----. 4 root root 4096 1月 23 23:43 zhangwuji
```

**tar指令 tar指令是打包指令，最后打包后的文件是 .tar.gz的文件**

**压缩 tar -zcvf**

**解压 tar -zxvf**

```
tar [选项] XXX.tar.gz 打包的内容
-c 产生.tar打包文件      create a new archive
-v 产生详细信息          verbosely list files processed【详细的列出已处理的文件】
-z (支持gzip压缩/解压文件) 通过gzip指令处理备份文件      filter the archive through
gzip 【通过gzip过滤存档文件】
-x 解压.tar文件          extract files from an archive【解压---从归档文件中解出文件】
-f (指定文件)使用存档文件 use archive file or device ARCHIVE【使用存档文件或设备
ARCHIVE】
```

案例：

```

tar -zcvf a.tar.gz hello.txt a.txt
tar -zcvf all.tar.gz /home/          打包整个home下所有文件
解压到当前目录：
tar -zxvf a.tar.gz
解压到指定目录 -C
tar -zvf a.tar.gz -C /opt/tmp/
[root@pengtao home]# tar -zcvf a.tar.gz hello.txt a.txt
hello.txt
a.txt
[root@pengtao home]# ls
animal a.tar.gz a.txt c c.txt.gz hello.txt home.zip pengtao pt yangxiao
zhangwuji
[root@pengtao home]#

```

## xz -z 压缩文件、 xz -d 解压文件 指令

**xz -z** 要压缩的文件

如果要保留被压缩的文件加上参数 **-k**，如果要设置压缩率加入参数 **-0** 到 **-9** 调节压缩率。如果不设置，默认压缩等级是6.

**xz** 解压文件方法或命令

**xz -d** 要解压的文件

同样使用 **-k** 参数来保留被解压缩的文件。

创建**tar.xz**文件：只要先 **tar cvf xxx.tar xxx/** 这样创建**xxx.tar**文件先，然后使用 **xz -z xxx.tar** 来将 **xxx.tar**压缩成为 **xxx.tar.xz**

## xz -d 解压tar.xz文件：

```

先 xz -d xxx.tar.xz      #将 xxx.tar.xz解压成 xxx.tar
然后，再用
tar xvf xxx.tar          #来解包。

```

###权限管理(文件和目录的权限)

[本段需要参考](#)

## chmod指令

```

r:read w:write x:execute
u:root g:group o:other a:all

[root@pengtao home]# chmod u=rwx,g=rw,o=r hello.txt
[root@pengtao home]# ls -lh hello.txt
-rwxrw-r--. 1 root root 190 1月 26 20:07 hello.txt

```

## chown 指令 修改文件所有者



```
chown newowner file
```

使用root目录进行操作:

将 /usr/local/mysql 目录 所有的文件和目录的 所有者 都改成 mysql

```
chown -R mysql /usr/local/mysql
```

-R 如果是目录, 则使其下所有子文件或目录递归生效

## chgrp修改文件所在组

```
chown group file
```

```
chgrp -R /home/pengtao pengtao
```

```
[root@pengtao pt]# chown -R pengtao /home/pt
```

```
[root@pengtao pt]# ll -s /home/pt
```

总用量 0

```
0 -rw-r--r--. 1 pengtao root 0 1月 26 21:31 hello.txt
```

```
[root@pengtao pt]# chgrp -R wudang /home/pt
```

chgrp: 无效的组: "wudang"

```
[root@pengtao pt]# chgrp -R mojiao /home/pt
```

```
[root@pengtao pt]# ll -s /home/pt
```

总用量 0

```
0 -rw-r--r--. 1 pengtao mojiao 0 1月 26 21:31 hello.txt
```

```
[root@pengtao pt]#
```

## crond 任务调度

crontab 进行任务调度基本语法:

```
crontab [选项]
```

-e 编辑crontab定时任务

-l 查看crontab任务

-r 删除当前用户所有的crontab任务

快速入门案例:

设置任务调度文件: /etc/crontab

设置个人任务调度

执行crontab -e 命令。

然后输入任务到调度文件,

例如:

```
*/1 * * * * ls -l /etc >> /tmp/to.txt
```

每小时的每分钟执行 ls -l /etc/ > /tmp/to.txt 命令

5个占位符的说明:

1. 如果只是简单的任务, 可以不用写脚本, 直接在crontab中加入任务即可

2. 对于比较复杂的任务, 需要写脚本 (shell 编程)

3. 以下百度即可:

第1列 分钟1~59

第2列 小时1~23 (0表示子夜)

第3列 日1~31

第4列 月1~12

第5列 星期0~6 (0表示星期天)

## 第6列 要运行的命令

\* 表示任何时间，比如第一个\*代表一个小时中每分钟都执行一次

，代表不连续的时间，如"0 8,12,16 \* \* \*"代表每天8点0分、12点0分、16点0分执行一次命令

- 代表连续的时间范围，如"0 5 \* \* 1-6"代表每周一至周六的凌晨5点0分执行命令

\*/n 代表每隔多久执行一次。如上述示例中代表每隔一分钟执行一次命令

## 案例参考 96页

案例：

1、编写shell脚本，如

```
vim /home/mytask.sh
```

```
date >> /tmp/mydate
```

2、给mytask.sh一个可执行的权限

```
chmod 744 /home/mytask.sh
```

3、crontab -e

crond相关指令：

```
crontab -e 编辑任务
```

```
crontab -r 终止任务调度
```

```
crontab -l 列出当前有哪些任务调度
```

```
service crond restart 重启任务调度
```

```
*/1 * * * * /home/mytask.sh
```

```
// TODO
```

## Linux磁盘分区、挂载

分区基本知识：了解 mbr分区、windows下磁盘分区、Linux磁盘分区

mount挂载

umount卸载

## 老师不离开指令lsblk：查看系统的分区和挂载的情况

```
[root@pengtao home]# lsblk -f
NAME      FSTYPE LABEL                UUID                                  MOUNTPOINT
sr0       iso9660 CentOS_6.8_Final
/media/CentOS_6.8_Final
sda
├─sda1 ext4                16783e93-0db1-4f38-8861-0eeb058c3fab /boot
├─sda2 ext4                81715f87-a9dd-4ba7-9af7-5a7058ea8f76 /
└─sda3 swap              150bc620-7328-4952-9543-5a0ba183bc13 [SWAP]
```

```
[root@pengtao home]#
```

分区情况 分区类型

唯一标识分区的40位不重复的字符串

挂载点

## df指令：报告文件系统磁盘空间使用情况 report file system disk space usage

```
df -lh
```

## du指令：查询指定目录的磁盘占用情况（查看内存大小） estimate file space usage

du -h /目录

- s 指定目录占用大小汇总
- h 带计量单位
- a all 包含文件
- max-depth=1 子目录深度
- c 列出明细的同时，增加汇总量

**du -sh /usr/local/nginx**

**du -ch /usr/local/nginx**

案例：

```
[root@localhost tar-gz-bak]# du -sh /usr/local/nginx
3.9M    /usr/local/nginx
[root@localhost tar-gz-bak]# du -ch /usr/local/nginx
3.8M    /usr/local/nginx/sbin
72K    /usr/local/nginx/conf
12K    /usr/local/nginx/logs
8.0K    /usr/local/nginx/html
0      /usr/local/nginx/client_body_temp
0      /usr/local/nginx/proxy_temp
0      /usr/local/nginx/fastcgi_temp
0      /usr/local/nginx/uwsgi_temp
0      /usr/local/nginx/scgi_temp
3.9M    /usr/local/nginx
3.9M    总用量
[root@localhost tar-gz-bak]#
```

du -ach --max-depth=1 /opt

```
[root@pengtao ~]# du -ach --max-depth=1 /opt
163M    /opt/vmware-tools-distrib
4.0K    /opt/rh
54M    /opt/vmwareTools-10.3.10-13959562.tar.gz
1008K    /opt/tmp
217M    /opt
217M    总用量
[root@pengtao ~]#
```

## 指令

1)统计/home文件夹下文件的个数               wc统计个数

```
ls -l /home |grep "^-" | wc -l
```

2)统计/home文件夹下目录的个数

```
ls -l /home |grep "^d" | wc -l
```

3)统计/home文件夹下文件的个数， 包括子文件夹里的

```
ls -lR /home |grep "^-" | wc -l
```

4)统计文件夹下目录的个数， 包括子文件夹里的

```
ls -lR /home |grep "^d" | wc -l
```

5)以树状显示目录结构

```
tree
```

```
yum install tree
```

```
tree
```

## centos6 不支持yum

1.虚拟机能否连接外网

2.DNS配置是否有问题！ 检验DNS配置是否正常可以这样做：nslookup www.baidu.com

3.centos6 不支持yum解决：

### [参考博客](#)

centos6 不支持yum 一键复制解决：

```
sed -i "s|enabled=1|enabled=0|g" /etc/yum/pluginconf.d/fastestmirror.conf
```

```
mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup
```

```
curl -o /etc/yum.repos.d/CentOS-Base.repo https://www.xmpan.com/Centos-6-Vault-Aliyun.repo
```

```
yum clean all
```

```
yum makecache
```

## Linux网络配置固定ip

目前我们的网络配置采用的是NAT模式

一。自动ip

linux-系统-首选项-网络连接-编辑-自动连接-应用

缺点：每次自动获取的IP地址可能不一样。如果是个网站，每次IP地址不一样是不行的，不适用于做服务器。

服务器的IP是需要固定的

二 指定固定的ip

```
vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

追加：

```
IPADDR=192.168.159.131# 和vmware8的ip同一网段(ipv4前三位一样)的ip
```

```
GATEWAY=192.168.159.2
```

```
DNS1=192.168.159.2
```

```
PREFIX=24
```

找到BOOTPROTO修改为：

```
BOOTPROTO=static
```

确认

```
ONBOOT=yes
```

### ###Linux 进程管理 ps -aux 或 ps -ef

ps 查看进程使用的指令[report a snapshot of the current processes], 一般来说使用的参数是

ps -aux

-a all显示当前终端的所有进程信息

-u userList 以用户的格式显示进程信息

-x 显示后台进程运行的参数 Register format

-e 在命令之后显示环境。(Show the environment after the command.)

-f 【ASCII-art过程层次结构(森林)】ASCII-art process hierarchy (forest)

To see every process on the system using standard syntax: 【使用标准语法查看系统中的每个进程: 】

ps -e

ps -ef

ps -eF

ps -ely

To see every process on the system using BSD syntax: 【使用BSD语法查看系统中的每个进程: 】

ps ax

ps axu

ps -aux|grep sshd

ps -ef|grep java

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.1	0.0	19364	1552	?	Ss	17:18	0:05	/sbin/init
root	2	0.0	0.0	0	0	?	S	17:18	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	17:18	0:00	[migration/0]
root	4	0.0	0.0	0	0	?	S	17:18	0:00	[ksoftirqd/0]
root	5	0.0	0.0	0	0	?	S	17:18	0:00	[stopper/0]
root	6	0.0	0.0	0	0	?	S	17:18	0:00	[watchdog/0]
root	7	0.0	0.0	0	0	?	S	17:18	0:01	[migration/1]
root	8	0.0	0.0	0	0	?	S	17:18	0:00	[stopper/1]
root	9	0.0	0.0	0	0	?	S	17:18	0:00	[ksoftirqd/1]
root	10	0.0	0.0	0	0	?	S	17:18	0:00	[watchdog/1]
root	11	0.0	0.0	0	0	?	S	17:18	0:00	[migration/2]

[root@pengtao ~]# ps -ax|grep sshd:pengtao

Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ

4151 pts/3 S+ 0:00 grep sshd:pengtao

[root@pengtao ~]# kill -9 4151

-bash: kill: (4151) - 没有那个进程

## kill指令

根据 进程号结束进程

kill -9 pid

根据名字结束进程

killall progressName

### #### 【/bin/bash 说明是一个终端】

ps -ef|grep bash

pstree -u 显示进程树

## 服务(service)管理

一般一个服务都会监听一个端口

默认:

sshd 22

mysql 3306

####CentOS 6服务指令

centos6 上的服务管理工具为chkconfig, Linux系统所有的预设服务都可以通过查看/etc/init.d/目录得到。但里边只有屈指可数的几个文件,

因为CentOS 7已经不再延续CentOS6版本的服务管理方案了。但是我们依然可以继续使用chkconfig这个命令。系统的服务都可以通过这样的命令实现:

```
service 服务名 start|stop|restart|status
```

```
service iptables start
```

## CentOS 7 服务指令

centos7不使用SysV而改为systemd了, 这是因为systemd支持多个服务并发启动, 而SysV只能一个一个地启动,

这样最终导致的结果是systemd方式启动会快很多。

**列出系统所有的服务**

```
systemctl list-units -all --type=service
```

这些服务对应的启动脚本在/usr/lib/systemd/system/

常用命令

#让某个服务开机启动

systemctl enable crond.service (.service可以省略)

#不让开机启动

systemctl disable crond.service

#查看服务状态

systemctl status crond.service

#启动某个服务

systemctl start crond.service

#停止某个服务

systemctl stop crond.service

#重启某个服务

systemctl restart crond.service

#查看某个服务是否开机启动

systemctl is-enabled crond #查看某个服务是否开机启动

###centos 6 防火墙操作:

[centos6防火墙操作](#)

```
vim /etc/sysconfig/iptables    # yy复制    p粘贴    u撤回    d删行    ESC    shift + 冒号
```

```
service iptables restart
```

###centos 7 防火墙操作:

#查看防火墙规则

```
firewall-cmd --list-all
```

# 查询端口是否开放

```
firewall-cmd --query-port=8080/tcp
```

# 开放80端口

```
firewall-cmd --permanent --add-port=80/tcp
```

```
firewall-cmd --permanent --add-port=6379/tcp
```

```
firewall-cmd --permanent --add-port=5672/tcp
```

```
firewall-cmd --permanent --add-port=21/tcp
```

```
firewall-cmd --permanent --add-port=20/tcp
```

```
firewall-cmd --permanent --add-port=39000/tcp
```

```
firewall-cmd --permanent --add-port=40000/tcp
```

```
firewall-cmd --permanent --add-port=15672/tcp
```

```
firewall-cmd --permanent --add-port=5672/tcp
```

```
firewall-cmd --permanent --add-port=4369/tcp
```

```
firewall-cmd --permanent --add-port=25672/tcp
```

# 移除端口

```
firewall-cmd --permanent --remove-port=8080/tcp
```

#重启防火墙(修改配置后要重启防火墙)

```
firewall-cmd --reload
```

## telnet指令

通过telnet指令来检查linux的某个端口是否在监听，并且可以访问

```
telnet ip 端口
```

windows cmd窗口

```
telnet 192.168.159.131 22
```

【windows telnet不是内部或外部命令 解决: 】

开始->"控制面板"->"程序和功能"-> 左侧"启动或关闭windows功能"->

→ 在"windows功能"界面勾选Telnet client →点击"确定"等待安装。

## 如果希望设置某个服务自启动或关闭永久生效，要使用chkconfig指令:

使用setup指令，选择系统服务 \* 是自启动

或

```
ls -l /etc/init.d/
```

可以查看服务

查看或者修改默认级别: vi /etc/inittab

Linux系统有7种运行级别(runlevel): 常用的是级别3和5

- 运行级别0: 系统停机状态, 系统默认运行级别不能设为0, 否则不能正常启动
  - 运行级别1: 单用户工作状态, **root**权限, 用于系统维护, 禁止远程登陆
  - 运行级别2: 多用户状态(没有**NFS**), 不支持网络
  - 运行级别3: 完全的多用户状态(有**NFS**), 登陆后进入控制台命令行模式
  - 运行级别4: 系统未使用, 保留
  - 运行级别5: **X11**控制台, 登陆后进入图形**GUI**模式
  - 运行级别6: 系统正常关闭并重启, 默认运行级别不能设为6, 否则不能正常启动
- 查看或修改运行级别
- ```
vi /etc/inittab
```

备忘:

linux 50节课学到 20分钟了

49已学完

```
chkconfig --list 指令
chkconfig --list | grep 服务名
或
chkconfig 服务名 --list

[root@localhost ~]# chkconfig --list|grep sshd
sshd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@localhost ~]# chkconfig sshd --list
sshd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@localhost ~]#

chkconfig iptables --list
设置某个服务在某个级别下是否自启动
chkconfig --level 服务级别[0-6] 服务名 off/on

chkconfig --level 5 sshd off

[root@localhost ~]# chkconfig --level 5 sshd off
[root@localhost ~]# chkconfig --list|grep sshd
sshd          0:off  1:off  2:on   3:on   4:on   5:off  6:off
[root@localhost ~]#

1) linux 重新设置服务后自启动或关闭, 需要重启机器reboot才能生效.
```

## 进程监控指令

动态进程监控指令top

网络服务监控指令netstat

**top指令: top- display Linux tasks 【显示 linux 任务: 实时显示进程】**

**netstat指令: netstat - Print network connections, routing tables, interface statistics,masquerade connections, and mul-ticast memberships**

netstat -打印网络连接, 路由表, 接口统计信息, 伪装连接和多播成员关系

```
netstat -anp | more
```



# RPM 和 YUM

介绍:

RPM软件包管理器

RPM 是Red-Hat Package Manager（红帽软件包管理器）的缩写，这一文件格式名称虽然打上了RedHat的标志，

但是其原始设计理念是开放式的，包括OpenLinux、S.u.S.E.以及Turbo Linux等Linux的分发版本都有采用，

可以算是公认的行业标准了。

英文原义：RPM Package Manager（原Red Hat Package Manager，是一个递归缩写）

注解：一种用于互联网下载包的打包及安装工具，它包含在某些Linux分发版中。它生成具有.RPM扩展名的文件。与Dpkg类似。

查看已安装的 rpm 列表

rpm -qa|grep glibc

```
rpm -qa
rpm -qa|more
rpm -qa xx  查询 xx 的 rpm是否安装
rpm -qi xx  查询安装的 rpm包软件信息
rpm -ql xx  查询软件包名软件包中的文件(软件包的安装位置)
rpm -qf 文件  查询某个文件属于哪个rpm包
rpm -e rpm包名
```

```
[root@localhost ~]# rpm -qa|grep firefox
firefox-17.0.10-1.el6.centos.x86_64
[root@localhost ~]# rpm -qf /etc/passwd
setup-2.8.14-20.el6_4.1.noarch
[root@localhost ~]#
```

## rpm包的管理

安装rpm包:

基本语法:

rpm -ivh RPM包全路径名称

参数说名字:

i install 安装

v verbose 提示

h hash 进度条

实例:

## yun Shell前端软件包管理器

Yum（全称为 Yellow dog Updater, Modified）是一个在Fedora和RedHat以及CentOS中的Shell前端软件包管理器。

基于RPM包管理，能够从指定的服务器自动下载RPM包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包，无须繁琐地一次次下载、安装。

1.yum查询功能

```
yum [options] [command] [package ...]
```

-y: 默认yum需要是交互模式, -y表示自动提供yes响应

search: 搜索某个软件名或关键字

list: 列出所有yum所管理的软件包和名称

info: 同上, 也类似rpm -qai

provides: 查找该命令是由软件安装生成的, 类似rpm -df的功能

repolist: 列出所有可用的repo

yum list|grep firefox 查询 yum镜像服务器上firefox的版本

yum install firefox具体的linux版本

TODO linux 安装 tomcat idea mysql

## mysql命令

mysql查看版本号:

```
mysql -V
```

```
[root@localhost mysql]# mysql -V
```

```
mysql Ver 14.14 Distrib 5.7.24, for linux-glibc2.12 (x86_64) using EditLine wrapper
```

mysql基本命令:

```
mysql -uroot -p
```

```
show databases;
```

```
use nacos;
```

```
show tables;
```

## [oracle官网下载jdk](#)

1.wget 复制的tar.gz链接

2.解压:

3.配置环境变量jdk:

```
vim /etc/profile
```

末尾追加:

```
export JAVA_HOME=/usr/local/jdk-17.0.2
```

```
export CLASSPATH=$JAVA_HOME/lib
```

```
export PATH=$JAVA_HOME/bin:$PATH
```

4.使环境变量重新生效、刷新、重新加载 source /etc/profile

5.验证 linux jdk安装是否成功

```
java
```

```
javac
```

```
java -version
```

####重启网络 /etc/init.d/network restart

## 两台linux远程传输指令：

scp 文件 用户名@ip:路径

```
scp nginx-1.20.2.tar.gz root@192.168.159.132:/usr/local
```

将本地文件拷贝到远程

scp 文件名 -用户名@计算机IP或者计算机名称:远程路径

从远程将文件拷回本地

scp -用户名@计算机IP或者计算机名称:文件名 本地路径

[测试第二台linux安装好nginx](#) --第二台机器的ip

## Linux 安装telnet

```
rpm -qa telnet-server
yum -y install telnet-server
rpm -qa telnet
yum -y install telnet
service xinetd restart
yum -y install xinetd
```

## cmd窗口 连接 linux

SSH端口

ssh root@ip -p 22

FTP linux搭建命令记录：

cd /etc/vsftpd

vim vsftpd.conf

## 启动

systemctl start vsftpd

ftp://ftpuser:ftpuser@192.168.159.132/

[安装博客](#)

本机笔记本按照上面博客安装成功。

但是只能用root账号密码登陆并且 可以各种跳转上下目录

2022年3月3日16:15:39

配置好了root用户不限上传下载删除的vsftpd

需要优化为 匿名用户可以登陆可以上传下载

可以使用某一个文件夹以及子文件夹。不能使用上级文件夹