

# 嵌入式系统设计

Vxwork

## Linux

快捷键：

ctrl + l 清空屏幕

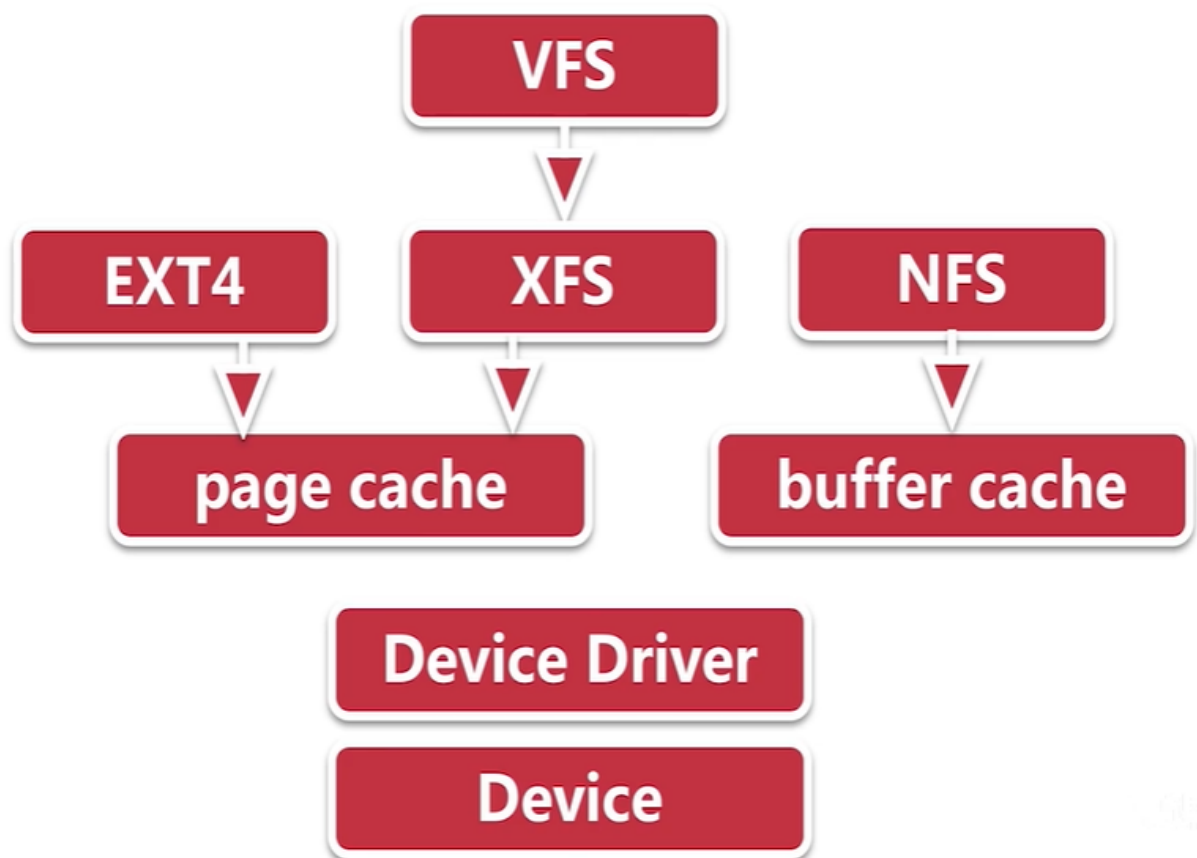
## Linux文件系统

FHS3.0 (File system Hierarchy Standard)

	可分享	不可分享
不可变	/usr	/etc
	/opt	/boot
可变	/var/mail	/var/run
	/var/spool/news	/var/lock

- /
  - etc 配置文件
  - bin 必要命令
  - usr 二级目录
  - home 家目录
  - var 动态数据

## VFS虚拟文件系统



- 内核层抽象出通用的文件系统接口
- 支持文件、网络、特殊文件系统

抽象对象：

- 超级块：文件系统
- 目录项：文件路径
- 索引节点：具体文件
- 文件：进程打开的文件

属性分层结构

一切皆文件

## 数据盘挂载

```
1 fdisk -l
```

inode：存储数据的元数据

Linux没有盘符的概念，只有一个根目录/，所有文件都在其下

/

- 根目录
- 层级关系

# 命令

通用格式：

```
1 command [-options] [parameter]
```

- command：命令本身
- -options：[可选，非必填] 命令的一些选项，可以通过选项控制命令的行为细节
- parameter：[可选，非必填]命令的参数，多数用于命令的指向目标等

语法中[]表示可选

## ls

```
1 ls [-a -l -h] [Linux路径]
```

- -a all
  - 前面带.的文件使隐藏文件/文件夹，只有通过-a选项才能看到
- -l 以列表（竖向排列）
- -h 表示以易于阅读的形式，列出文件大小，如K，M，G

/home/用户名

组合使用

## cd

change directory

```
1 cd [Linux路径]
```

不写参数回到用户的HOME目录下

- 绝对路径
- 相对路径

. 表示当前目录

.. 表示上一级目录

~ 表示HOME目录

# mkdir

创建目录

```
1 mkdir [-p] 路径
```

-p可选，创建多级不存在的目录时使用

# touch

创建文件

touch 路径

# cat

查看内容

cat 路径

# more

查看内容，支持翻页，空格翻页，q退出

more 路径

# cp

可以用于复制文件\文件夹

```
1 cp [-r] 参数1 参数2
```

- -r选项，可选，用于**复制文件夹**使用，表示递归
- 参数1，Linux路径，表示被复制的文件或文件夹
- 参数2，Linux路径，表示要复制去的地方

# mv

```
1 mv 参数1 参数2
```

可以用于改名

# rm

删除文件，文件夹

```
1 rm [-r -f] 参数1 参数2 参数n
```

- -r, 删除文件夹
- -f, force, 强制删除（不会弹出提示信息）
  - 普通用户删除内容不会弹出提示，只有**root管理员用户删除内容会有提示**
  - **所以一般普通用户用不到-f**
- 参数1, 参数2, ..., 参数n 表示要删除的文件或文件夹路径，按照空格隔开。

支持用通配符来模糊匹配

# pwd

Print Work Directory

# tree

树状目录

# which

Linux命令本体就是一个个的二进制可执行文件

```
1 which 命令
```

# find

```
1 find 起始路径 -name "被查找文件名"
```

```
1 find 起始路径 -size + | -n[kMG]
```

- +, - 表示大于和小于
- n表示大小数字
- kMG表示大小单位，k表示kb,M表示MB，G表示GB

# grep

通过grep命令，从文件中通过关键字过滤文件行

```
1 grep [-n] 关键字 文件路径
```

- 选项 `-n` 可选，表示在结果中显示匹配的行的行号
- 参数，**关键字**，必填，表示过滤的关键字，带有空格或其它特殊符号，建议使用“”将关键字包围起来
- 参数，**文件路径**，必填，表示要过滤内容的文件路径，可作为**内容输入端口**

## WC

```
1 wc [-c -m -l -w] 文件路径
```

- 选项，`-c`，统计bytes数量
- 选项，`-m`，统计字符数量
- 选项，`-l`，统计行数
- 选项，`-w`，统计单词数量
- 参数，文件路径，被统计的文件，可作为内容输入端口

## 管道符

左 | 右。将左边的结果作为右边的输入

## echo

命令行内输出指定内容

## 反引号`

在echo中，用``括起来表示命令信息

```
1 echo `pwd`
```

## 重定向符号

```
1 > 将左侧命令的结果，覆盖写入到符号右侧指定的文件中
```

```
1 >> 将左侧命令的结果，追加写入到符号右侧指定的文件中
```

## tail

```
1 tail [-f -num] 参数
```

- 参数，linux命令，表示被跟踪的文件路径
- 选项，`-f`，表示持续跟踪
- 选项，`-num`，表示尾部多少行，不填默认10行

# Vim

## 命令模式

i: 在当前光标位置进入输入模式

a: 在当前光标位置 之后 进入输入模式

I: 在当前行的开头, 进入输入模式

A: 在当前行的结尾, 进入输入模式

o: 在当前行的下一行进入输入模式

O: 在当前行的上一行进入输入模式

0: 移动光标至开头

\$: 移动给光标至行结尾

pageup: 向上翻页

pagedown: 向下翻页

/: 进入搜索模式

n: 向下继续搜索

N: 向上继续搜索

dd: 删除光标所在的行

ndd: n是数字, 表示删除当前光标向下n行

yy: 复制当前行

nyy: 复制当前行和下面的n行

p: 粘贴复制的内容

u: 撤销修改

ctrl + r: 反向撤销修改

gg: 跳到首行

G: 跳到尾行

dG: 从当前行开始, 向下全部删除

dgg: 从当前行开始, 向上全部删除

dS : 从当前光标开始，删除到本行的结尾

d0 : 从当前光标开始，删除到本行的开头

## 底线命令模式

:wq 保存并退出

:q 仅退出

:q! 强制退出

:w 仅保存

**:set nu 显示行号**

:set paste 设置粘贴模式

# Linux权限和用户

## su

```
1 su [-] [用户名]
```

## sudo

普通用户使用sudo使用root权限

用户与用户组

## 用户和用户组

```
1 groupadd 创建用户组
2
3 groupdel 用户组名
```

```
1 useradd [-g -d] 用户名
```

- 选项：-g指定用户的组，不指定-g，会创建同名组加入，指定-g需要组已经存在，如已存在同名组，必须使用-g。
- 选项：-d指定用户HOME路径，不指定，HOME目录默认在：/home/用户名

```
1 userdel [-r] 用户名
```

- 选项：-r，删除用户的HOME目录，不使用-r，删除目录时，HOME目录保留



## 1 id [用户名]

- 参数：用户名，被查看的用户，如果不提供则查看自身。

- 1 usermod -aG 用户组 用户名
- 2 将指定用户加入指定用户组

- 1 getent passwd
- 2 getent group

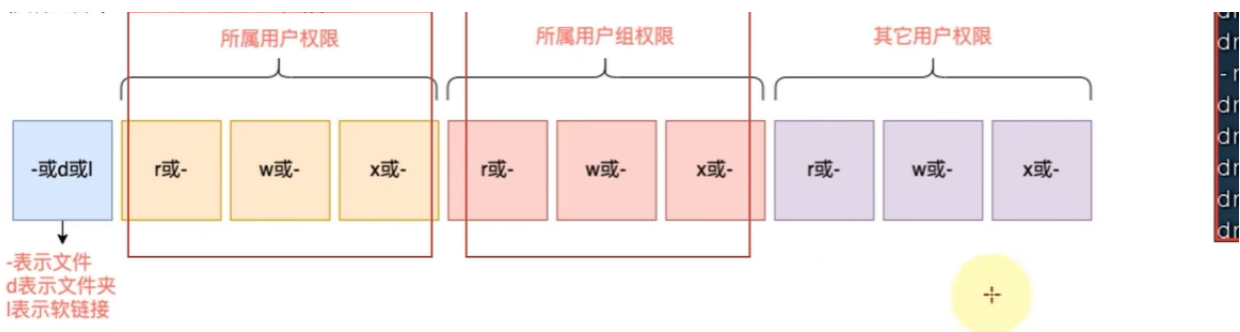
使用getent命令，可以查看当前系统内有那些命令

七份信息：

用户名： 密码（X）： 用户ID： 组ID： 描述信息（无用）： HOME目录： 执行终端（默认bash）

## 查看权限管控信息

- 文件或文件夹的控制信息
- 文件或文件夹所属用户
- 文件或文件夹所属用户组



举例：drwxr-xr-x, 表示：

- 这是一个文件夹，首字母d表示
- 所属用户(右上角图序号2)的权限是：有r有w有x, rwx
- 所属用户组(右上角图序号3)的权限是：有r无w有x, r-x（-表示无此权限）
- 其它用户的权限是：有r无w有x, r-x

## 什么意思

- 第一个d表示文件夹
- 所属用户
- 所有用户组
- 所属其他用户
  - o **r代表读**
    - o 文件夹表示可以查看文件夹内容
  - o **w代表写**
    - o 文件夹表示可以在文件夹内：创建，删除，改名等操作

- **x代表可执行权限**，针对文件表示可以将文件作为程序执行
- 针对文件夹，表示可以更改工作目录到此文件夹，即cd进入

## chmod

修改文件或目录的权限。

```
1 chmod u = rwx, g = rx, o = x hello.txt
```

**-R选项**可以将文件夹以及文件夹内**全部内容**权限设置为：rwxrwxrwx

```
1 chmod -R u=rwx, g=rwx, o=rwx hello.tx
```

使用数字序号

```
1 chmod 751 hello.txt
```

## chown

更改文件或目录的所有者为指定的用户或用户组。

```
1 chown [-R] [用户] [:] [用户组] 文件或文件夹
```

- **选项，-R，同chmod，对文件夹内全部内容应用相同规则**
- 选项，用户，修改所属用户
- 选项，用户组，修改所属用户组
- **:** 用于分隔用户和用户组

**普通用户无法使用，只能用root用户**

# Linux使用操作

**快捷键：**

- 1 `ctrl + c` 强制停止,退出当前命令输入
- 2 `ctrl + d` 退出账户的登录
- 3 `history` 查看历史输入的命令
- 4 `!`搜索历史命令, 自动匹配, 例如! `py`
- 5 `ctrl + a`, 跳到命令开头
- 6 `ctrl + e`, 跳到命令结尾
- 7 `ctrl +` 键盘左键, 向左跳一个单词
- 8 `ctrl +` 键盘右键, 向右跳一个单词
- 9 `ctrl + l` 清空终端内容
- 10 `clear` 清空终端内容

## 软件安装

yum: RPM软件管理器,用于自动化安装配置Linux软件,并可以自动解决依赖问题。

- 1 `yum [-y] [install | remove | search]` 软件名称

- `-y` 自动确认,无需手动确认安装或卸载过程

yum命令需要root权限,可以su切换到root,或使用sudo权限,yum命令需要联网

## Ubuntu

- 1 `apt [-y] [install | remove | search]` `wget`

## systemctl

- 1 `systemctl start | stop | status | enable | disable` 服务名

- NetworkManager, 主网络服务
- networkd, 副网络服务
- firewalld, 防火墙服务
- sshd, ssh服务 (FinalShell远程登录Linux使用的就是此服务)

除了内置的服务以外,部分第三方软件安装后也可以用systemctl进行控制

## 软链接

在系统中创建软链接,可以将文件、文件夹链接到其他位置。类似快捷方式

- 1 `ln -s` 参数1 参数2

- `-s`, 创建软链接
- 参数1: 被链接的文件或文件夹

- 参数2：要链接去的目的地

## 日期，时区

```
1 date [-d] [+格式化字符串]
```

- -d按照给定的字符串显示日期，一般用于日期计算
- 格式化字符串：通过特定的字符串标记，来控制显示的日期格式
  - %Y， 年
  - %y, 年份后两位数字（00，99）
  - %M 月份（01，12）
  - %d 日(01,31)
  - %H 小时（00，23）
  - %M 分钟（00，59）
  - %S 秒（00，59）
  - %s 自1970-01-01 00:00:00到现在的秒数
- 使用-d支持的时间标记：(同样支持格式化字符串)
  - year 年
  - month 月
  - day 天
  - hour 小时
  - minute 分钟
  - second 秒

## ntp

可以自动联网同步时间，也可以通过ntp -u ntp.aliyun.com 手动校准时间

## IP地址

DHCP：动态获取IP地址，即每次重启设备后都会获取一次，可能导致IP地址频繁变更

每一台联网的电脑都会有一个地址，用于和其他计算机进行通信，IP地址主要有两个版本，V4和V6版本

IPv4的地址格式为a.b.c.d，其中abcd表示0~255的数字，如192.168.88.101

通过 `ipconfig` 查看本机的IP地址。

127.0.0.1表示本机

0.0.0.0

- 可以用于指代本机
- 可以在端口绑定中用来确定绑定关系
- 在一些IP地址中，表示所有IP的意思，如放行规则设置为0.0.0.0，表示允许任意IP访问

# 主机名

```
1 hostname
```

修改主机名

```
1 hostnamectl set-hostname name
```

## 域名解析

### ping

可以通过ping命令来检查指定的网络服务器是否是可联通的。

```
1 ping [-c num] ip或主机名
```

- 选项，-c，检查的次数，不适用-c选项，将无限次数持续检查
- 参数：ip或主机名，被检查的服务器的ip地址或主机名地址

### wget

非交互式的文件下载器，可以在命令行内下载网络文件

```
1 wget [-b] url
```

- 选项：-b，后台下载，会将日志写入到当前工作目录的wget-log文件
- 参数：url，下载链接

### curl

发送http网络请求，可用于下载文件，获取信息等

```
1 curl [-O] url
```

- 选项：-O，用于下载文件，当url是下载链接时，可以使用此选项保存文件
- 参数：url，要发起请求的网络地址

## 端口

端口，是设备与外界通讯交流的出入口，端口可以分为物理端口和虚拟端口

- 物理端口：又可称之为接口，是可见的端口，如USB接口，RJ45网口，HDMI端口等
- 虚拟端口：是指计算机内部的端口，是不可见的，是用来操作系统和外部进行交互使用的

Linux支持65535个端口，分为3类进行使用：

- 公认端口：1~1023，通常用于一些系统内置或知名程序的预留使用，如SSH服务的22端口，HTTPS服务的443端口，非特殊需要，不要占用这个范围的端口
- 注册端口：1024~49151，通常可以随意使用，用于松散的绑定一些程序/服务
- 动态端口：49152~65535，通常不会固定绑定程序，而是当程序对外进行网络链接时，用于临时使用。

## 进程管理

```
1 ps [-e -f]
```

- -e 显示出全部的进程
- -f 以完全格式化的形式展示信息（展示全部信息）
- 一般来说，固定用法就是 ps -ef 列出全部进程的全部信息

从左到右分别是：

- UID：进程所属的用户ID
- PID：进程的进程号ID
- PPID：进程的父ID（启动此进程的其它进程）
- C：此进程的CPU占用率（百分比）
- STIME：进程的启动时间
- TTY：启动此进程的终端序号，如显示？，表示非终端启动
- TIME：进程占用CPU的时间
- CMD：进程对应的名称或启动路径或启动命令

- UID：进程所属的用户ID
- PID：进程的进程号ID
- PPID：进程的父ID（启动此进程的其他进程）
- C：此进程的CPU占用率（百分比）
- STIME：进程的启动时间
- TTY：启动此进程的终端序号，如果显示？，表示非终端启动
- TIME：进程启用CPU的时间
- CMD：进程对应的名称或启动路径和启动命令

## 关闭进程

```
1 kill -9 进程ID
```

- -9, 表示**强制关闭进程**，不适用此选项会向进程发送信号要求其关闭，但是否关闭看进程自身的处理机制

## 主机状态

查看CPU，内存使用情况

```
1 top
```

top - 14:08:23 up 1:00, 2 users, load average: 0.00, 0.00, 0.00										
任务: 333 total, 1 running, 332 sleeping, 0 stopped, 0 zombie										
%Cpu(s): 0.2 us, 0.2 sy, 0.0 ni, 99.6 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st										
MiB Mem : 7914.4 total, 4666.2 free, 1573.7 used, 1674.4 buff/cache										
MiB Swap: 9216.0 total, 9216.0 free, 0.0 used. 6067.1 avail Mem										
进程号	USER	PR	NI	VIRT	RES	SHR	%CPU	%MEM	TIME+	COMMAND
877	root	20	0	266152	20392	17372 S	0.7	0.3	0:00.74	NetworkManager
10	root	20	0	0	0	0 I	0.3	0.0	0:00.35	rcu_sched
797	root	20	0	169280	7616	6596 S	0.3	0.1	0:08.44	vmtoolsd
1775	topeet	20	0	22036	12632	8052 S	0.3	0.2	0:00.87	systemd
1	root	20	0	168564	12016	8588 S	0.0	0.1	0:04.92	systemd
2	root	20	0	0	0	0 S	0.0	0.0	0:00.03	kthreadd
3	root	0	-20	0	0	0 I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0 I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0 I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
8	root	0	-20	0	0	0 I	0.0	0.0	0:00.00	mm_percpu_wq
9	root	20	0	0	0	0 S	0.0	0.0	0:00.03	ksoftirqd/0
11	root	rt	0	0	0	0 S	0.0	0.0	0:00.02	migration/0
12	root	-51	0	0	0	0 S	0.0	0.0	0:00.00	idle_inject/0
13	root	20	0	0	0	0 I	0.0	0.0	0:00.84	kworker/0:1-mpt_poll_0
14	root	20	0	0	0	0 S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0 S	0.0	0.0	0:00.00	cpuhp/1
16	root	-51	0	0	0	0 S	0.0	0.0	0:00.00	idle_inject/1
17	root	rt	0	0	0	0 S	0.0	0.0	0:00.95	migration/1
18	root	20	0	0	0	0 S	0.0	0.0	0:00.04	ksoftirqd/1
20	root	0	-20	0	0	0 I	0.0	0.0	0:00.00	kworker/1:0H-kblockd
21	root	20	0	0	0	0 S	0.0	0.0	0:00.00	cpuhp/2

## 第一行：

top：命令名称

14: 08 : 23 当前系统时间，up 6min：启动了六分钟

2 users: 2个用户登录，load：15分钟负载

## 第二行：

Tasks: 175个进程 1 running: 1个子进程在运行

174 sleeping: 174个进程睡眠，0个停止进程，0个僵尸进程

## 第三行：

%Cpu(s): CPU使用率，us：用户CPU使用率，sy：系统CPU使用率，ni：高优先级进程占用CPU时间百分比，id：空闲CPU率，wa：IO等待CPU占用率，hi：CPU硬件中断率，si：CPU软件中断率，st：强制等待占用CPU率

## 第四、五行

Kib Mem：物理内存，total：总量，free：空闲，used：使用，buff/cache：buff和cache占用

KibSwap：虚拟内存（交换空间），total：总量，free：空闲，used：使用，buff/cache：buff和cache占用

- PID：进程ID
- USER：进程所属用户

- PR：进程优先级，越小越好
- NI：负值表示高优先级，正表示低优先级
- VIRT：进程使用虚拟内存，单位KB
- RES：进程使用物理内存，单位KB
- SHR：进程使用共享内存，单位KB
- S：进程状态(S休眠，R运行，Z僵死状态，N负数优先级，I空闲状态)
- %CPU：进程占用CPU率
- %MEM：进程占用内存率
- TIME+：进程使用CPU时间总计，单位10毫秒
- COMMAND：进程的命令或名称或程序文件路径

- 1 `-p`：只显示某个进程信息
- 2 `-d`：设置刷新时间，默认为5s
- 3 `-c`：显示产生进程的完整命令，默认是进程名
- 4 `-n`：制定刷新次数，比如 `top -n 3` 是新输出三次后退出
- 5 `-b`：以非交互非全屏模式运行，以批次的方式执行top，一般配合-n制定输出几次统计信息，将输出重定向到制定文件，比如 `top -b -n 3 > /tmp/top.tmp`
- 6 `-i`：不显示任何闲置（idle）或无用的进程
- 7 `-u`：查找特定用户启动的进程

### top以交互式运行：

- 1 `h`：按下h键，会显示帮助画面
- 2 `c`：按下c键，会显示产生进程的完整命令，等同于-c参数
- 3 `f`：可以选择需要展示的项目
- 4 `M`：根据驻留内存大小（RES）排序
- 5 `T`：根据CPU使用百分比大小进行排序
- 6 `T`：根据时间/累计时间进行排序
- 7 `E`：切换顶部内存显示单位
- 8 `e`：切换进程内存显示单位
- 9 `l`：切换显示平均负载和启动时间信息
- 10 `i`：不显示闲置或无用的进程，等同于-i参数
- 11 `t`：切换显示CPU状态信息
- 12 `m`：切换显示内存信息

### 硬盘使用情况：

- 1 `df -h`

- -h，以更加人性化的单位显示

### 磁盘信息监控：

- 1 `iostat [-x] [num1] [num2]`



## 网络状态监控：

- 可以使用sar命令查看网络的相关统计

```
1 sar -n DEV num1 num2
```

- -n 查看网络，DEV表示查看网络接口
- num1: 刷新闻隔（不填就查看一次阶数） num2: 查看次数（不填无限次数）

## 环境变量

环境变量是一组信息记录，类型是Key Value类型（名称=值），用于操作系统运行的时候记录关键信息

```
1 env
```

查看环境变量

环境变量：PATH，通过\$取出环境变量的值

环境变量PATH会记录一组目录，目录之间用：隔开。记录的是命令的搜索路径。当执行命令会从记录中记录的目录中挨个搜索要执行的命令并执行

可以通过这个项目的值，加入自定义的命令搜索路径

如

```
1 export PATH = $PATH
```

修改环境变量

- 临时生效： export 名称= 值
- 永久生效
  - 针对用户： ~/bashrc文件中配置
  - 针对全部用户

## 文件上传和下载

通过finalShell或者xshell 上传或下载，拖动

# 压缩和解压

- .tar 称之为tarball，对党文件，即简单的将文件组装到一个.tar的文件中，并没有太多文件体积的减少，仅仅是简单的封装
- .gz，也常见为.tar.gz，gzip格式压缩文件，即使用gzip压缩算法将文件压缩到一个文件内，可以极大的减少压缩后的体积

```
1 tar [-c -v -x -f -z -C] 参数1 参数2 ... 参数N
```

- -c 创建压缩文件，用于压缩模式
- -v 显示压缩，解压过程，用于查看进度
- -x 解压模式
- -f 要创建的文件，或者要解压的文件，-f选项必须在所有选项中位置处于最后一个
- -z gzip模式，不适用-z就是普通的tarball模式
- -C 选择解压的目的地，用于解压模式

```
1 zip [-r] 参数
```

- -r 压缩文件夹使用

```
1 unzip
```

- unzip [-d] 参数
  - -d 制定解压的目录

## Mysql