[1.基础 3](#_Toc498275278)

[常用命令 3](#_Toc498275279)

[A.2. 文件所有权的显示 3](#_Toc498275280)

[A.3. 删除非常见字符命名的文件 3](#_Toc498275281)

[A.4 常识介绍 3](#_Toc498275282)

[A.5 文件访问方式 4](#_Toc498275283)

[A.6 配置是否支持ping 4](#_Toc498275284)

[A.7 查看磁盘信息／使用情况 4](#_Toc498275285)

[A.8 环境变量 4](#_Toc498275286)

[1.1 联网nat 4](#_Toc498275287)

[1.2. U盘挂载 6](#_Toc498275288)

[1.3 基本命令 6](#_Toc498275289)

[1.3.1 mkdir 6](#_Toc498275290)

[1.3.2 cp-copy 6](#_Toc498275291)

[1.3.3 cat 6](#_Toc498275292)

[1.3.4 ls-list 6](#_Toc498275293)

[1.3.5 fuser 6](#_Toc498275294)

[1.3.5 ps 6](#_Toc498275295)

[1.3.5 top 6](#_Toc498275296)

[1.3.6 grep 6](#_Toc498275297)

[1.3.6 pgrep 7](#_Toc498275298)

[1.3.7 kill 7](#_Toc498275299)

[1.3.8 | 管道 7](#_Toc498275300)

[1.3.9 grep 查找指定字符串 7](#_Toc498275301)

[1.3.10 tar [主选项+辅选项] 文件或者目录 7](#_Toc498275302)

[1.3.11 xz、sz、rz 7](#_Toc498275303)

[1.3.12 sudo—superuser do 8](#_Toc498275304)

[1.3.13 df—disk free 9](#_Toc498275305)

[1.3.14 make 9](#_Toc498275306)

[1.3.15 yum 9](#_Toc498275307)

[1.3.16 rm 9](#_Toc498275308)

[1.3.17 mv 9](#_Toc498275309)

[1.3.18 gcc 9](#_Toc498275310)

[1.3.19 chmod 10](#_Toc498275311)

[1.3.20 alias 10](#_Toc498275312)

[1.3.21 function 函数 10](#_Toc498275313)

[1.3.22 type 10](#_Toc498275314)

[1.3.23 vim 10](#_Toc498275315)

[1.3.23 tail 10](#_Toc498275316)

[1.3.24 find 10](#_Toc498275317)

[1.3.25 man 11](#_Toc498275318)

[1.3.26 wget 11](#_Toc498275319)

[1.3.26 nohup 11](#_Toc498275320)

[1.3.27 touch 11](#_Toc498275321)

[1.3.28 extglob 11](#_Toc498275322)

[1.3.29 netstat 11](#_Toc498275323)

[1.3.30 crontab 11](#_Toc498275324)

[1.3.30 top 12](#_Toc498275325)

[1.3.31 du 12](#_Toc498275326)

[1.3.32 yum 12](#_Toc498275327)

[1.3.33 curl 12](#_Toc498275328)

[1.3.34 tcl/tool command language 12](#_Toc498275329)

[1.3.35 tee 12](#_Toc498275330)

[1.3.36 nano 12](#_Toc498275331)

[1.3.37 sar (system activity reporter) 12](#_Toc498275332)

[1.3.38 lsof (list open files) 12](#_Toc498275333)

[1.3.39 nmap(network mapper) 12](#_Toc498275334)

[1.3.40 ifconfig 12](#_Toc498275335)

[1.3.41 nslookup 13](#_Toc498275336)

[1.3.42 lsb\_release 13](#_Toc498275337)

[1.3.42 su 13](#_Toc498275338)

[1.3.43 cut 13](#_Toc498275339)

[1.4 文件名颜色 13](#_Toc498275340)

[1.5 vmware简介 13](#_Toc498275341)

[1.6 script 13](#_Toc498275342)

[1.6.1 日期 14](#_Toc498275343)

[1.7 添加用户 14](#_Toc498275344)

[1.8 端口配置 14](#_Toc498275345)

[2. 机制 14](#_Toc498275346)

[2.1 系统软件及其存储位置 14](#_Toc498275347)

[2.1.1 iptable/firewalld/ufw 14](#_Toc498275348)

[2.1.2 /etc 18](#_Toc498275349)

[2.1.3 服务器配置 18](#_Toc498275350)

[2.2 网络 18](#_Toc498275351)

[route vs gateway 19](#_Toc498275352)

[2.2.1 IPv4 vs IPv6 19](#_Toc498275353)

[2.2.2 简介 19](#_Toc498275354)

[2.2.3 搭建服务器 20](#_Toc498275355)

[2.3 linux服务器内存已满解决方案 20](#_Toc498275356)

[2.3.1 扩容 20](#_Toc498275357)

[2.3.2 设置swap配置 20](#_Toc498275358)

[2.4 进程间通信 20](#_Toc498275359)

[3. 软件 21](#_Toc498275360)

[3.1 使用 21](#_Toc498275361)

[3.1.1 redis 21](#_Toc498275362)

[3.1.2 mongo 21](#_Toc498275363)

[3.1.3 ssh服务器配置 21](#_Toc498275364)

[3.1.4 内网vs外网 (公网) 21](#_Toc498275365)

[3.1.5 shadowsocks 21](#_Toc498275366)

[3.1.6 PostgresSql 21](#_Toc498275367)

[4. shell 脚本 22](#_Toc498275368)

[4.1 简介 22](#_Toc498275369)

## Frontier

Zsh

<http://zsh.sourceforge.net/Doc/Release/Shell-Grammar.html>

# 1.基础

## 常用命令

Cat /etc/redhat-release：用于查看当前redhat的版本

Uname –r：表示kernel的版本

Hostname 显示主机

Hostname -f

&: linux中的&符号置于程序尾让程序自动运行。如/usr/local/mysql/bin/mysqld\_safe –user=mysql &

查看linux中的运行的服务

---service --status-all

查看环境变量

echo $PATH

<http://blog.csdn.net/qjwcn/article/details/18267431>

使用命令

Source /etc/profile即可将改变即刻生效

查看linux内核信息：

rpm -qa |grep -i kernel

### A.2. 文件所有权的显示

- --- --- ---

第一个字符表示文件类型，如d(目录)、-(普通文件)、系统文件等

第一组表示文件所有者的权限

第二组为组内用户的权限

第三组为others的权限

### A.3. 删除非常见字符命名的文件

如何：!、#、-命名的文件

rm -rf -- !

对于编码错误的文件进行删除，利用文件的节点号

Find . –inum 276831 -exec rm -rf {} \;

查看文件节点号

ls –i

### A.4 常识介绍

Linux（unix）中:

---文件描述符 0 表示standard input

---文件描述符 1 表示standard output

---文件描述符2 表示standard error

2>&1 表示将标准错误导向标准输出， 但是为了区分1是文件描述符，而不是filename， 故添加了&;

tty---(telebytes 、teletypewriter电传打字机，电报交换机)

### A.5 文件访问方式

### A.6 配置是否支持ping

/proc/sys/net/ipv4/icmp\_echo\_ignore\_all

### A.7 查看磁盘信息／使用情况

Windows分区按照c disk、d disk、等。

Linux的分区使用/dev/sda、／dev／sdb，以此类推。硬盘中分primary partition、extended partition、logical partition。扩展分区和逻辑分区在创建时需要显示说明。一张硬盘中只能有一个扩展分区；逻辑分区，理论上无分区数量的限制。

注意：显示free的磁盘空间多指不被系统控制的空间，为不可用空间。但是在扩展分区中的free的磁盘空间，则为可被利用的磁盘空间。

分区表：记录了硬盘分区情况。存在两种分区表标准。

---MBR( Master Boot Record)：

缺点： 不支持超过4个主分区的配置、disk 分区被限制在2tb

---GPT(GUID partition table)

command line:

fdisk –l：查看linux系统使用的分区表方式。

Disklabel显示为dos，其是mbr的一种。

### A.8 环境变量

环境变量，可以通过$PATH进行获取。

该$PATH的内容存储在/etc/environment文件中。

当安装某软件后，其命令仅限与在安装目录使用，需要使用时，需要将其加入到$PATH所指定的目录中去，使用systemlink的方式。

## 1.1 联网nat

公网：能够被dns感知，具有公有地址，由inter nic负责。

内网：局域网，无法被公网上的网络访问。

内网中的多台计算机可以通过公网ip进行网络访问。公网路由器通过

时经过路由器，通过宽带，最终去到运营商那边，数据是从运营商出去，最终数据是回到运营商那边，运营商再把数据发送到用户的电脑。

查看public interface和private interface

Ip –4 addr scope global

使用curl查看某个网址是否可以联通：

Curl –connect-timeout seconds url，示例：

Curl –connect-timeout 5 10.144.60.90

内网中的计算机区分方式：

端口映射：

端口转发：port forwarding，通过修改/etc/sysctl.conf进行永久性的修改。

## 1.2. U盘挂载

使用fdisk查看磁盤情況，移動硬盤為sdb1

mount /dev/sdb1 /mnt/usb

## 1.3 基本命令

### 1.3.1 mkdir

参数：

-p: parent级创建路径

-v：每次创建显示信息

### 1.3.2 cp-copy

-i：interactive覆盖询问

-f：force强制复制

-l：link建立硬链接，而非赋值文件

-p：preserve保留文件或目录属性，所有者、所属组、权限与时间

-P： parent保留源文件或目录的路径

-b：备份，备份文件字尾加上备份字符串，~

-v：verbose显示执行过程

### 1.3.3 cat

### 1.3.4 ls-list

ll 将显示文件的所有信息

ls –i 显示每位文件独有的节点号

### 1.3.5 fuser

显示正在使用指定file、file system、sockets的进程信息

### 1.3.5 ps

显示process status。

参考：<http://www.cnblogs.com/peida/archive/2012/12/19/2824418.html>

-ps aux：显示所有包含其他使用者的行程

-ps –e|grep mysql：显示mysql程序的对应pid号

### 1.3.5 top

动态查看系统资源分配

### 1.3.6 grep

Global Regular Expression Print是表示全局表达式输出。使用权限是所有用户。

-c：count()，只输出匹配行

-I：忽略大小写

-h：查询多文件时不显示文件名。

-n：显示匹配行及行号

-s：不显示不存在或无匹配文本的错误信息。

-v：显示不含匹配文本的所有行。

-r: recursive 要求搜索其子目录。默认情况是跳过目录。

-w：匹配整个单词，而非字符串的一部分。

使用grep –para pattern filename

其中pattern可以多个，有|进行连接，如grep pattern1 | pattern2 file

Grep pattern1 files | grep pattern2

### 1.3.6 pgrep

用来查看进程

-n：表示查找最新的进程，并返回进程号

-o：表示查找最老的进程，并返回进程号

其他参数与grep相同，如pgrep –c node

pgrep –f node

### 1.3.7 kill

-9：强制

### 1.3.8 | 管道

常用more，less结合显示内容

如：ps –ef | more

### 1.3.9 grep 查找指定字符串

-ps aux|grep mysql

查找指定字符串：cat text.txt | grep ‘hello world’

查找指定字符串所在行及其上述多行: cat text.txt | grep –B 10 ‘hello world’ (-B --> Before)

查找指定字符串所在行及其下述多行：cat text.txt | grep –A 10 ‘hello world’ (-A ---> After)

### 1.3.10 tar [主选项+辅选项] 文件或者目录

參考<http://www.linfo.org/tar.html>

c：创建新的档案文件，用于备份、打包

x：拆包，档案文件中释放

t：列出档案文件

辅助选项

-z：是否具备gzip的属性

-j：是否同时具备bzip属性

-v：verbose压缩过程中显示文件

-f：使用档名，后必须直接接档名

tar –xvf filename 用于解压具备gz的文件

tar –xzvf filename 用于解压不具备gz的文件

tar –cvf qmtrip.tar ./qmtrip 用于压缩项目qmtrip

注：

压缩率：Bzip2>Gzip>zip

Bzip耗费更多cpu，gzip更快

### 1.3.11 xz、sz、rz

mongodb-linux-x86\_64-rhel62-3.2.10

Sz命令用于将文件send到某处，即下载。rz用于接受文件，即上传。

同时注意使用 软件secureCRT终端，该终端会设置上传、下载的文件的目录，故在使用sz时，直接无需提供存放目录

sz downloadFile.txt

### 1.3.12 sudo—superuser do

#### 1.3.12.1 创建用户

可以使用visudo来查看、编辑文档，此时，该命令会调用vim进行编辑，但是会提供自动查错的功能。

超级用户创建普通用户：

Adduser new\_username

为新用户创建密码：

Passwd new\_username

给予普通用户的超级权利

设置时，处于栏目/etc/sbin/visudo

语法：User-name Machine——name=（Effective-user） command

#### 1.3.12.2 创建组

--- 创建组：Groupadd test

---修改组：groupmod

---删除组：groupdel test2

### 1.3.13 df—disk free

-df /home –h：查看某区域的存储使用情况

注：du ---disk usage

注意：free操作会显示mem的相关信息，包块已用、未用，swap，buffer等。

<http://serverfault.com/questions/85470/meaning-of-the-buffers-cache-line-in-the-output-of-free>

### 1.3.14 make

使用命令make test进行测试，直接在软件目录使用make，即可

Make install

### 1.3.15 yum

Yellow dog Updater, modified, shell前段软件包管理器(为centos指定的包管理器，相当于ubuntu的apt-get包管理器，从指定服务器下载RPM包并安装，自动处理依赖性关系，一次安装所有依赖的软件包。

Yum install “Development Tools”

与自身source源码编译安装。

Make configure //生成configure文件

./configure –prefix=/path //进行编译，并且指明安装目录

<https://www.digitalocean.com/community/tutorials/how-to-install-git-on-centos-7>

参考： 如何安装

对于使用源码安装的软件，其删除方法是：

<http://serverfault.com/questions/444289/how-to-uninstall-git-installed-from-the-source>

1.3.16 rm

-f：--force，强制删除

-i：--interactive

-r：--recursive递归删除，参数中列出的全部目录和文件

-v：详细显示过程

注：

Rm -rvf /home/Jackie ../将文件目录jackie下的所有文件移植上层目录

### 1.3.17 mv

mv /path/oldFileName /path/newFileName

同目录中进行文件的重命名：mv oldName newName

### 1.3.18 gcc

编译器

### 1.3.19 chmod

改变文件的权限

-chmod 755 my\_script

权限有三：读、写、执行，共10个字节表示，第一个字节表示目录或者文件，其后分别表示读、写、执行

9个字节代表三组：（owner、group、other）的权限

#### 1.3.19.1 chown

改变某文件的归属文

chown user file\_name

### 1.3.20 alias

取别名，使用方法： alias l=’ls -l’ ，该别名功能只在当前的session中

该功能是shell builtin

### 1.3.21 function 函数

该功能是shell builtin

### 1.3.22 type

返回命令的类型，如alias、shell function、executable file

### 1.3.23 vim

编辑软件

<http://www.radford.edu/~mhtay/CPSC120/VIM_Editor_Commands.htm>

---按键i进入编辑模式

---按键esc，后冒号：，输入wq，表示保存退出；输入q！表示不保存退出；q表示强制退出

---按键esc，后冒号：，输入/keyword，系统将查找字段为keyword的值，n表示下一个，N表示回溯

---按键:n，表示跳转到第n行

---按键:/，表示向后查询；:?表示向前查询

使用美元符 SHIFT $，跳轉到vim編輯的一行末尾

SHIFT O,跳轉到vim編輯的上一行開頭

SHIFT ^ 跳轉到vim編輯的一行開頭，在普通模式下

SHIFT 0往下跳過多行(此處為零)， SHIFT 9 往上跳过多行

SHIFT A 跳到该行末尾并进入编辑模式

### 1.3.23 tail

查看文件尾部信息，对于日志文件，由于不断更新，可以看到最新的更新内容。

-f 循环读取

-q 不显示处理信息

-v 显示详细的处理信息

-c<数目> 显示的字节数

-n<行数> 显示行数，默认显示10行

--pid=PID 与-f合用,表示在进程ID,PID死掉之后结束.

-q, --quiet, --silent 从不输出给出文件名的首部

-s, --sleep-interval=S 与-f合用,表示在每次反复的间隔休眠S秒

tail –n [+|-] num 显示第num行开始，到文件末尾

### 1.3.24 find

<http://www.cnblogs.com/wanqieddy/archive/2011/06/09/2076785.html>

Find path –options

-print：将查找到的文件输出到标准输出

-exec：将匹配的文件执行该参数所给出的shell命令。

-ok: 与exec相同，但是每次会主动询问。

-name：查询名为name的

例如：

查找部分文件，并执行删除操作：find ./dist/ -name \*js.map -exec rm -f {} \;

#### 1.3.24.1 locate

Linux中查找文件或者目录。

### 1.3.25 man

查看帮助文档的命令，manual为其全称。命令路径/usr/bin/man

如想要查看命令ls的详细介绍，如参数等，可以使用man ls

### 1.3.26 wget

Linux中用于download网页内容。支持http、https、ftp、http proxies

简单命令：wget <http://www.minjieren.com/wordpress-3.1-zh_CN.zip>

-O：表示下载并以不同文件名进行保存。wget -O new\_name old\_name （O表示output）

#### Curl

Curl支持多种协议进行文件的上传下载。

常用命令：

Curl –O path url // path是下载后文件的存储路径，同时设置其下载后的命名

### 1.3.26 nohup

表示不挂断地运行命令。Nohup /root/start.sh &

### 1.3.27 touch

创建新文件，或者改变文件的timestamp

<http://www.linfo.org/touch.html>

### 1.3.28 extglob

Extended pattern matching

Bash提供的比较实用的工具之一，允许使用强于正则的表达式，但是默认关闭

---shopt -s extglob #：开启这项服务

---shopt –u extglob #：关闭此项服务

### 1.3.29 netstat

显示与网络相关的的所有信息，如网络连接、路由表、借口状态、masquerade连接，multicast membership

是在内核中访问网络链接状态及其相关信息的程序。提供tcp链接，tcp和udp监听、进程内存管理

#### 1.3.19.2 显示内容简介

显示端口链接状态时，state字段可能存在的情况：LAST\_ACK， CLOSE\_WAIT

### 1.3.30 crontab

该服务的执行依赖于crond服务

启动服务

/etc/init.d/cron stop（start|restart）----------ubuntu中

Redhat中位于/etc/下

### 1.3.30 top

查看系统资源

### 1.3.31 du

Disk usage ，用于查看文件夹下的文件用的存储大小。

Du -para folder\_path

参数：

-h：全称—human-readable，将可读性较好的方式显示。

### 1.3.32 yum

Yellow dog update, modified， 为前端软件包管理器。

Yum install gcc

Yum remove –skip-broken gcc

### 1.3.33 curl

Commandline uniform resource locator

### 1.3.34 tcl/tool command language

### 1.3.35 tee

tee指令会从标准输入设备读取数据，将其内容输出到标准输出设备，同时保存成文件。

### 1.3.36 nano

文本编辑器，比vi简单。

### 1.3.37 sar (system activity reporter)

### 1.3.38 lsof (list open files)

显示已打开的文件

### 1.3.39 nmap(network mapper)

网络扫描和嗅探工具包

### 1.3.40 ifconfig

查看电脑使用的ip等信息。在单个网卡中可以配置多个ip地址。

### 1.3.41 nslookup

查询dns信息命令。

交互式模式： nslookup – destination-server-name (注意中间的“-”)

非交互模式。Nslookup destination-server-name

外网环境下，ping l.jingli365.com，此时显示的ip地址为115.159.75.86， 内网环境下，ping l.jingli365.com表示192.168.1.2

### 1.3.42 lsb\_release

全称： linux standard base release

查看ubuntu的型号

### 1.3.42 su

linux系统切换用户。

Su root

Su – root

切换用户时，添加“-”符号，表示切换用户和该用户的shell环境。

### 1.3.43 cut

截取文件的部分内容进行显示。是对每一行进行字节、字段、字符进行处理

显示linux系统中的所有用户： cut –d : -f 1 /etc/passwd

-c：按照字符为单位

-b：按照字节为单位， 可以与-n连用，-n表示取消分割多字节字符。

-d：按照自定义分隔符进行截取。一般与-f连用， 表示截取的那段区域

who | cut –c 1： 表示截取每行的第一个字符。

who | cut –c 1-n： 表示截取每行的第一个字符。

### 1.3.43 ln

创建某个文件的链接

hard link: 指该链接直接指向了该文件，ls –l, 将链接视为文件处理

soft link：指该链接简洁指向了某文件, 如systemlink。区别，当使用ls –l时，soft link有—>指向

当显示使用-s时，表明创建soft link

### 1.3.44 curl

computer software project to provide an tool and command-line to transfer data using various protocols, like: ftp, http, ssh.

Originally means see URL.

### 1.3.45 awk

### 1.3.45 scp

scp ./times.sql [test@j.jingli365.com:/home/test](mailto:test@j.jingli365.com:/home/test) 本地文件复制到远程

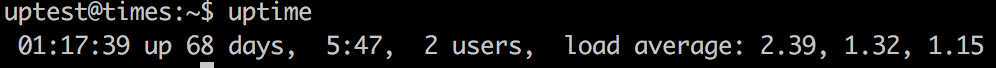
scp [test@j.jingli365.com:/home/test/times.sql](mailto:test@j.jingli365.com:/home/test/times.sql) ./ 远程文件复制到本地

### 1.3.46 size

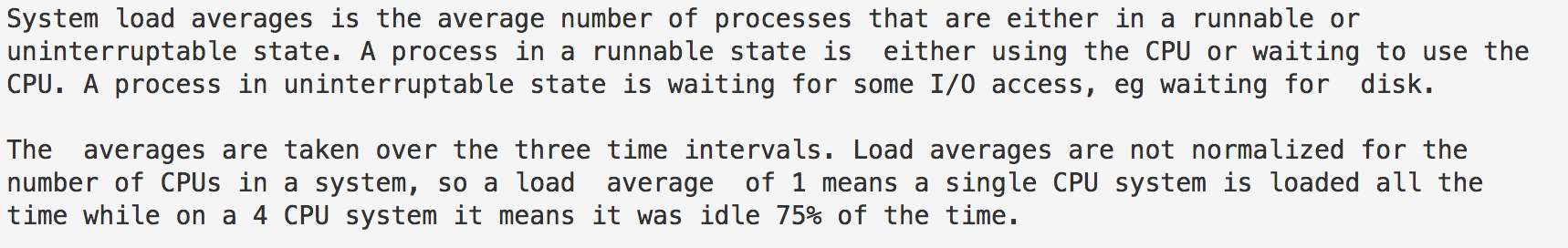
print the size of segments of an object file.

### 1.3.47 uptime

系统检测命令：



当前时间；该系统持续运行时长； 登陆的用户数；系统平均负载



## 1.4 文件名颜色

---绿色：可执行

---白色：普通文件

---红色：压缩文件

---蓝色：目录

--先蓝色：连接

## 1.5 vmware简介

关闭vmware中linux的beep声，参考<http://www.programgo.com/article/54791231276/>

在没有X11环境下，linux控制台使用setterm –blength 0

在/root/.bash\_profile文件中添加setterm –blength 0；则系统每次启动时，加载文件bash\_profile时，设置beep stopped

## 1.6 script

<http://linuxcommand.org/writing_shell_scripts.php>

脚本的执行，需要注意其执行的环境，系统不会搜索整个文件系统进行查找

但是系统对于可执行的文件，写有一个目录，即我们所说的path，

添加directory到path中，使用export PATH=$PATH:your directory

### 1.6.1 日期

日期显示：date (该结果为星期、月份、日期、小时、时间标准、年份)

格式化：date %A, %B %-d, %Y

## 1.7 添加用户

Useradd –d /usr/jack –m jack

为用户jack创建目录和用户名；

超级用户下：

Passwd jack可以设置用户的密码

Vim /etc/sudoers 修改用户权限。

修改权限

/etc/sudoers 显示了用户/组的权限问题

Usermod 命令：

Usermod –g root tommy： -g: group， root：组，tommy: 账号

对于/etc/sudoers中的内容，包含了多个用户/用户组的权限设置。 对于普通用户，若使其具有root权限，需要添加jack ALL=(ALL) ALL类似语句。当普通用户执行某操作(sudo vim test.json)时，若无权限，此时根据该表，进行密码提示，此时的密码是用户自己的密码。

Jack ALL =( ALL) ALL ： 表示登录者：登陆的主机： （可以变换的身份）可以下达的命令。

## 1.8 端口配置

/etc/ssh/sshd\_config --开放端口号

## 1.9 utilities/实用工具类

### 1.9.1 hash

report and remember the location of most recently used command; when no arguments is provided, then list all the location of utilities.

### 1.9.2 ulimit

限制shell进程及其子进程的系统资源大小，包括

-m: 最大内存大小

-p: 管道缓冲区大小

-s: 线程栈大小

-f: 创建文件大小

-u: 设置最大进程数

ulimit -s 512

设置方式：

终端设置 : ulimit –s 512

程序中调用: ulimit –s 512

bash脚本类型： 在文件.bashrc中添加ulimit –u 64， 此时对所有的shell进程有效

修改/etc/security/limits 配置文件，实现对于单个用户的资源限制。

# 2. 机制

2.1 系统文件

.bash\_profile: shell 脚本，每次login时会执行

## 2.1 系统软件及其存储位置

### 2.1.1 iptables

参考：<http://blog.csdn.net/ai_net/article/details/7693788>

Linux(Ubuntu) 虚拟机开启http服务，host主机进行访问，需要开启虚拟机防火墙服务，

/sbin/iptables -I INPUT -p tcp --dport 80 -j ACCEPT

#### 2.1.1.1 位置

reference： <https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-using-iptables-on-ubuntu-14-04>

##### 2.1.1.2 iptables

linux command：

iptables –L : 该命令显示所有的iptables 规则

### 2.1.2 iptable/firewalld/ufw

#### 2.1.2.1 简介

<http://www.cnblogs.com/ggjucheng/archive/2012/08/19/2646466.html>

Iptable的核心为netfilter，netfilter是真正实现防火墙功能，是linux内核中实现包过滤的内部结构。

防火墙配置：/etc/sysconfig/iptables或者为/etc/init.d/iptables（版本不一）

三种功能：

---包过滤、

---NAT（网络地址转换）:分为源地址NAT和目的地址NAT

---通用的pre-route packet mangling：

Iptable：存在filter表(实现包过滤)，nat表(网络地址转换)，mangle表(包重构（修改), raw表（数据跟踪处理）

Filter表系统chain：input、forward、output

Nat表系统chain：input、forward、output

Mangle表系统chain：prerouting,output

Input规则链：表示目的地址为本地的包

Output规则链：表示目的地址为本机之外的包

① 当一个数据包进入网卡时，它首先进入PREROUTING链，内核根据数据包目的IP判断是否需要转送出去。

② 如果数据包就是进入本机的，它就会沿着图向下移动，到达INPUT链。数据包到了INPUT链后，任何进程都会收到它。本机上运行的程序可以发送数据包，这些数据包会经过OUTPUT链，然后到达POSTROUTING链输出。

③ 如果数据包是要转发出去的，且内核允许转发，数据包就会如图所示向右移动，经过FORWARD链，然后到达POSTROUTING链输出

Linux中的防火墙关闭：

1. service iptables stop

systemctl status iptables.service

查看iptables的状态为。

Service iptables status可以查看iptables是否启动

关闭firewalld服务

systemctl mask firewalld

systemctl stop firewalld

使用iptable

yum install iptables-services

systemctl enable iptables

systemctl start iptables

Failed to start IPv4 firewall with iptables

方法：

service iptables save

systemctl stop firewalld

systemctl disable firewalld

systemctl start iptables

systemctl status iptables

systemctl enable iptables

ufw： uncomplicated firewall 配置简单，功能强大的firewall

启动： ufw disable/ enable

注意：

iptables 仅能维护IPv4的地址

#### 2.1.2.2 查看命令

查看ufw的状态信息：ufw status

#### 2.1.2.3 过滤规则的编写

dump防火墙iptables的内容到文件: iptables-save > path (如： iptables-save > /etc/iptables/rules.v4)

restore规则：

iptables-restore < path (如： iptables-store < /etc/iptables/rules.v4)

##### iptables 的过滤规则：

---规则1: 允许大部分的链接请求，通过在各个表中添加配置项，针对某些请求，拒绝链接

---规则2: 拒绝大部分的链接请求，通过在各个表中添加配置项，针对某些请求，允许链接。

Iptables –P chain\_name [DROP|ACCEPT]:

上述命令用于设定使用规则1还是规则2

添加规则，设置iptables去追踪链接状态。

Iptables –A INPUT –m conntrack –ctstate ESTABLISHED, RELATED –j ACCEPT

##### 命令行的添加

---iptables –A [INPUT|OUTPUT|FORWARD] –p [udp|tcp] --match multiport –dports 4070 –j [DROP|ACCEPT]

其中：

-A: 表示append，在iptables的某张表中进行追加。

-I： 表示insert，其后跟policy\_name number: 表示将该条规则插入tables，并指定某行。将其他规则往下压。

-p: 表示协议，udp或者tcp

-j：表示操作

--dport：表示destination port，可以是数字(逗号分隔)，可以是命令，如ssh. 对于想开放多个端口，可以使用 port1:port2，表示从port1到port2的之间的所有port

-m： 表示iptables提供的一些extensions或者modules，其后添加模块名，表示该rule将要使用该rule。如conntrack

-ctstate：是module中conntrack提供的命令。-m:

##### 命令行的删除

---iptables –F：F🡪flush, 表示所有iptables的表的删除。

---iptables –D chain\_name: 删除某张表，如INPUT、OUTPUT、FORWARD表。

---iptables –D chain\_name rulenum: 指定某张表的第n条规则被删除。

##### 常用的规则设定

iptables –A INPUT –p tcp –dport 22 –j ACCEPT; (或者iptables –A INPUT –p tcp –dport ssh –j ACCEPT;)

iptables –A INPUT –p tcp –dport 22 –j ACCEPT;

##### icmp-ping的过滤

iptables控制是否使用ping功能。0表示echo-reply， 8表示echo-request

iptables –A INPUT –p icmp –icmp-type 8 –m state –state NEW,ESTABLSHED,RELATED –j ACCEPT;

其中—icmp-type可以使用echo-request表示，也可以使用数字8表示。

/etc/sysconfig目录下文件介绍

/etc/sysconfig/network-scripts/ifcfg-eth0 描述了网关和掩码

 Tunneling 技术？

Iptables的三张表：filter，nat，mangle，iptables中默认的就是filter表，故curl时可以省略

Iptables的三个阶段：prerouting、forwarding、postrouting

实现linux的forwarding功能：

/proc/sys/net/ipv4/ip\_forward中的sysctl net.ipv4.ip\_forward=1，设置为1，则代表需要将其转发功能打开。即，当发送的包packet的目的地址不是本地时，将进行转发，否则将这个包丢弃。

Packet通过iptables的层层路由后，若不是本地包，则实现转发，借助SNAT进行正确转发。

Iptables每个阶段都有不同的规则，策略。Iptables –P 可以设置Chain的策略。

对于链的操作：

--- -l：表示插入，

--- -A：表示追加

--- -R：表示替换

--- -D：表示删除

--- -L：表示列表显示。

一台linux服务器，使用ifconfig查看，发现存在

Eth0: 内网地址inet：xxx，子网掩码：xxx，广播地址：xxx

Lo：内网地址inet：127.0.0.1 子网掩码：xxx

如何搭建网络服务器：

<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-node-js-application-for-production-on-centos-7>

对于nodejs网络服务器来说：

首先启动nodejs web服务器，指定监听的服务器地址和端口：http.createServer().listen(port,[hostname,callback])

通过使用curl <http://local.ip.address/port>来确定该nodejs服务成功。

其次，安装nginx，yum install nginx)。修改配置文件/etc/nginx/nginx.conf。

通过命令凯斯nginx服务

Systemctl restart nginx

Systemctl enable nginx

### 2.1.2 /etc

#### 2.1.2.1 /etc/profile

全局配置脚本文件，应用于所有用户

/etc/

对于登录启动系统，会自动装入configuration文件，对于login的系统，存在/etc/profile、~/.bash\_profile（个人启动文件）、~/.bash\_login（当.bash\_profile不存在时）、./profile（默认的配置文件）

对于无需登录启动系统，存在/etc/bash.bashrc和文件~/.bashrc

#### 2.1.2.2 /etc/sysctl.conf

设置系统初始化配置。如设置共享cache的大小，设置ip转发。此ip转发是指，当存在多个nic时，各个内网之间相互链接，需要设置ip允许转发。

Net.ipv4.ip\_forward=1

### 2.1.3 服务器配置

#### 2.1.3.1 /proc/

查看服务器内存：cat /proc/meminfo

查看服务器cpu信息：cat /proc/cpuinfo

查看内存；

memfree：指的是实际的物理ram内存的大小，系统未曾使用的。

Memavailable： 系统预测的启动新程序可用的内存。不包含swapping。

#### 2.1.3.2

## 2.2 网络

参考：http://www.tldp.org/HOWTO/Networking-Overview-HOWTO.html#toc6

流程：

通过增加电脑的硬件配置（有线、无线网卡）后，电脑可以接受有线、无线信号，完成可以上网的可能。

当网络链接，用户访问某个网址，首先会在本地查找常用dns列表，将域名转成ip地址。当本地不存在对应的域名时，此时会根据配置的dns服务器地址，请求该域名对应的ip地址。（google提供免费的dns的域名解析服务器，8.8.8.8和8.8.4.4）。完成域名解析后，会将请求发送到对应的router中(或者ISP服务商)。继而完成请求的转发。

对于部署了内网服务器的网络，一般该局域网网络中的主机通过DHCP动态获取ip地址，同时会设置dns为该内网服务器的网关路由器，

### route vs gateway

相同点： 两种设备都是用来在多个网络间进行network traffic的。

不同点： gateway用来在两种不相似的网络中，调节网络流量，更多的是在多个不相同网络中进行通信；router用来在相似的网络中，调节网络流量。

### 2.2.1 IPv4 vs IPv6

IPv4: 0.0.0.0在服务器层面上指所有机器

IPv6: [::] 双 : 表示所有机器

IPv4: 127.0.0.1表示回环地址，即本地测试地址。

IPv6: ::1表示本地地址。

### 2.2.2 简介

#### 2.2.2.1 ifconfig

通过ifconfig查看ip地址，此时显示的

lo: 表示本地地址（回环地址），仅用于本地测试，外网和局域网内的其他用户不可见

#### 2.2.2.2 网络接口配置

添加网络接口，修改目录/etc/network/interfaces(Ubuntu)

dns服务器地址设置： /etc/resolv.conf

#### 2.2.2.3 route路由相关查询

每台服务器存在默认的网关地址。可以是公网的网关地址，亦可以是内网网关地址。

#### 2.2.2.4 拨号上网 vs 路由器上网(宽带上网)

拨号上网： 通过电话线路， 拨打某个电话号码，建立与ISP的网络链接。无须架设宽带设备。

宽带上网：通过使用cable、digital subscriber line、 satellite and FTTx进行网络信号传输。

problems:

RTNETLINK answers: File exists

RTNETLINK answers: File exists Failed to bring up ens7.

Failed to bring up ens7.

sudo systemctl disable systemd-resolved

$ sudo reboot

### 2.2.3 搭建服务器

reference： <http://www.tk4479.net/u013177446/article/details/53996037>

<https://askubuntu.com/questions/304898/trying-to-configure-eth0-and-eth1-but-failed-to-bring-up-eth1>

## 2.3 linux服务器内存已满解决方案

nodejs: webpack-dev-server 启动时，提示173 exit。查看内存使用：只剩下几十兆，cpu使用%100

### 2.3.1 扩容

### 2.3.2 设置swap配置

https://www.digitalocean.com/community/tutorials/how-to-add-swap-on-ubuntu-12-04

https://www.google.co.jp/search?source=hp&q=kswapd0&oq=kswapd0&gs\_l=psy-ab.3..0i13k1l10.593.3621.0.4021.4.3.0.0.0.0.599.1107.5-2.2.0....0...1.1j2.64.psy-ab..2.2.1105.0..0j35i39k1.0.7kB98CPm6mc

## 2.4 进程间通信

reference： http://tldp.org/LDP/lpg/node7.html

# 3. 软件

## 3.1 使用

### 3.1.1 redis

参考：<http://blog.csdn.net/jiangguilong2000/article/details/8114740>

在其文件目录下执行

Redis-server /etc/redis.conf

3.1.1.1 安装

apt-get install redis-server

### 3.1.2 mongo

### 3.1.3 ssh服务器配置

#### 3.1.3.1 ssh免秘登陆

使用ssh进行远程服务器（vps）等登陆，配置眠密登陆。

远程服务器端：

1. 用户目录下，创建.ssh目录，设置权限700（chmod +700 .ssh/），.ssh目录下创建文件authorized\_keys， 将由客户端生成的.pub文件中的key值拷贝到authorized\_keys文件中。
2. 设置ssh的系统服务配置项(/etc/ssh/目录下)：文件sshd\_config 中修改

---RSAAuthentication yes

---PubkeyAuthentication yes

---AuthorizedKeysFile .ssh/authorized\_keys .ssh/authorized\_keys2

1. 重启ssh服务 /etc/init.d/ssh restart

### 3.1.3.2 introduction

full-name is secure shell, used to login remote machine and execute commands on a remote machine.

ssh [options] [command]

configurations:

---command-line options

---user’s configuration file(~/.ssh/config)

---client的system-wide配置文件/etc/ssh/ssh\_config

### 3.1.3.3 ssh-add

### 3.1.3.4 ssh-agent

### 3.1.4 内网vs外网 (公网)

内网（私有网）的ip地址固定在： 10/8， 172.16/ 12，192.168/16

私有网络中的客户端需要访问internet时，需要私有域的服务器（亦可以理解为路由器）进行nat, proxy或者隧道技术进行地址转换。

如公司l.jingli365.com的外网地址时115.159.75.86， 该服务器上（路由器上存在内网地址192.168.1.1），当外网访问l.jingli365.com的服务时，此时该服务器会做端口映射（即是nat）

### 3.1.5 shadowsocks

服务器搭建vpn，reference：<https://wiki.archlinux.org/index.php/Shadowsocks>

通过apt-get进行安装的shadowsocks，根据shadowsock的配置，启动shadowsocks

nohup /usr/bin/python /usr/local/bin/ssserver -c ./lib/python2.7/dist-packages/shadowsocks/config.json &

### 3.1.6 PostgresSql

#### 3.1.6.1 安装postgres

注意：

---当linux中安装postgres，会默认为linux系统添加postgres用户

配置文件： /etc/postgresql-common/createcluster.conf

安装脚本： apt-get install postgresql-9.6

安装log:

postgresql-client-common

-----创建集群

Creating new cluster 9.6/main ...

config /etc/postgresql/9.6/main

data /var/lib/postgresql/9.6/main

locale en\_US.UTF-8

socket /var/run/postgresql

port 5432

postgresql-common

#### 3.1.6.2 使用系统命令进入系统

psql

直接输入时，会根据当前的linux的用户身份进行postgres数据库的登陆。一般登陆时需要su – postgres, 此时宜系统用户postgres的身份登陆。

# 4. shell 脚本

references: <http://www.gnu.org/software/bash/manual/html_node/index.html#SEC_Contents>

## 4.1 简介

### 4.1.1 Keywords

#### 4.1.1.1 echo

simple command: 如echo

pipeline：带有 | 或者 |&的命令行，可以并行的执行多个command。|& 表示，2>&1 |, 即奖前一个命令的标准输出和错误流—>下一个command的标准输入

#### 4.1.1.2 special sign

##### 4.1.1.2.1 $? Get Exit Status

Reference: http://www.gnu.org/software/bash/manual/html\_node/Special-Parameters.html#Special-Parameters

Expands to the exit status of the most recently executed foreground pipeline.

##### 4.1.1.2.2 $$

Expands to the process id of the invoking shell.

##### 4.1.1.2.3 $!

Expands to the process id of the most recently placed into the background.

##### 4.1.1.2.4 $#

the number of positional parameters

##### 4.1.1.2.5 $@

equivalent to $1, $2

#### 4.1.1.5 $[number]

$0: the name of shell or shell script.

$1: the first parameter in command-line

#### 4.1.1.6 pipeline (|& vs |)

|: to represent pipeline in linux command-line. The first command output will be the input of the next command.

|&: equivalent to 2>$1, standard out and error info will be put into the pipeline, and the next command receives.

### 4.1.2 Escape Character

#### 4.1.2.1 double-quote

“”: will convert content inside into literal meaning, with exception of character backquote, dollar-sign, backslash

#### 4.1.2.2 single-quote

## 4.2 Clause

----IF从句

If []; then

fi

为if条件句

在/etc/profile中存在多种shell脚本，如profile等，出现设置路径的问题，PATH=$PATH:$HOME/bin

The export command tells the shell to make the contents of PATH available to child processes of this shell.

----引号

Linux中的引号可以包含回车键，但是引号中存在正常的引号可以使用backslash进行转义

## 4.3 operand

### 4.3.1 bitwise operation

>>， >>>， <<

### 4.3.2 AND vs OR vs XOR

&， |，^

### 4.3.3 shortcut

\*=， +=， -=（a = a –b），&= (a = a & b)，|= (a = a|b)， ^=(a = a^b)， >>=， <<=， /=， %=

## 4.4 脚本文件

## 4.4.1 执行某种命令

shell脚本文件中/home/test/bin/update-nginx.sh的效果等同于sh /home/test/bin/update-nginx.sh

## exec sh –c “/home/test/bin/update-nginx.sh”

# 5. 通信协议

## 5.1层结构(OSI) vs TCP/IP结构

#### 5.1.1.1 Open system interconnection，开放式网络互联

物理层： 负责控制传输介质、使用的物理物质标准

数据链路层：

网络层：数据的路由选择传输方式、定义了逻辑地址，物理地址。最大传输单位， 协议： ip，icmp

传输层： 传输过层中的差错处理。存在协议有udp、tcp

会话层： 开始、控制和结束会话

表示层：定义数据的展现方式，加密、编码等。

应用层：定义了一些应用协议，如SMTP，FTP，Http, Telnet, NFS

#### 5.1.1.2 tcp/ip 结构

reference： <https://www.cnblogs.com/BlueTzar/articles/811160.html>

主机到网络层（osi的物理层、数据链路层）

网络互联层 （网络层）

传输层（osi的传输层）

应用层（osi的会话层、表示层、应用层）

## 5.2 实用协议介绍

### 5.2.1 icmp

Internet control message Protocol。控制报文协议。为网络层协议。实用该协议的的命令（工具）有： ping， tracert。

该协议为错误侦测和回报机制。目的是检测网络的链接情况。

Ping 命令为应用层命令，实用网络层的ICMP协议。调用ping命令，

### 5.2.3 安全性

ping of death:　当icmp的数据包的尺寸超过64kb上限时，主机出现内存分配错误，导致tcp/ip堆栈崩溃，导致死机。

## 5.3 各层详细信息

### 5.3.1 运输层

控制信息的接收，错误的处理。使用端口。

### 5.3.2 网络层

无端口信息，存在字段: type、code、和id，id时发送方发送报文时区别而设，接收方利用此信息进行区别回复。Type表示时echo-reques、echo-reply或者其他

## 5.4 实例

### 5.4.1 iptables

## 5. 5 Socket-套接字

### 5.5.1 connection associated

#### 5.5.1.1 tcp

generally， the maximum number of connect request is 128 by default.

#### 5.5.1.2 POSIX Defined Interface

##### 5.5.1.2.1 Service Establishment

##### 5.5.1.2.2 Data Transfer

Three method for sending data:

---send(int sockfd, const void \*buf, size\_t nbytes, int flags): connection established.

---sendto(int sockfd, const void \*buf, size\_t nbytes, int flags,const struct sockaddr \*destaddr, socklen\_t destlen)

: connectionless

---sendmsg(int sockfd, const struct msghdr \*msg, int flags) : designate multiple buffers from which to transmit data.

Three method for receiving data:

---recv: will receive all data once and for all.

---recvfrom: get the address of sender.

---recvmsg(int sockfd, struct msghdr \*msg, int flags): get data into multiple buffers

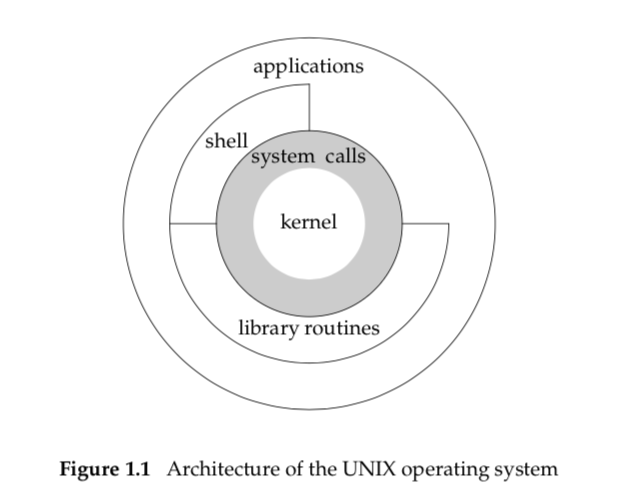
### 5.5.2 connectionless

#### 5.5.2.1 SOCK-DGRAM 数据报

### 5.5.3 out-bound data

# 6. 内核（kernel ）

6.1 简介



## 6.2 kernel

## 6.3 系统调用

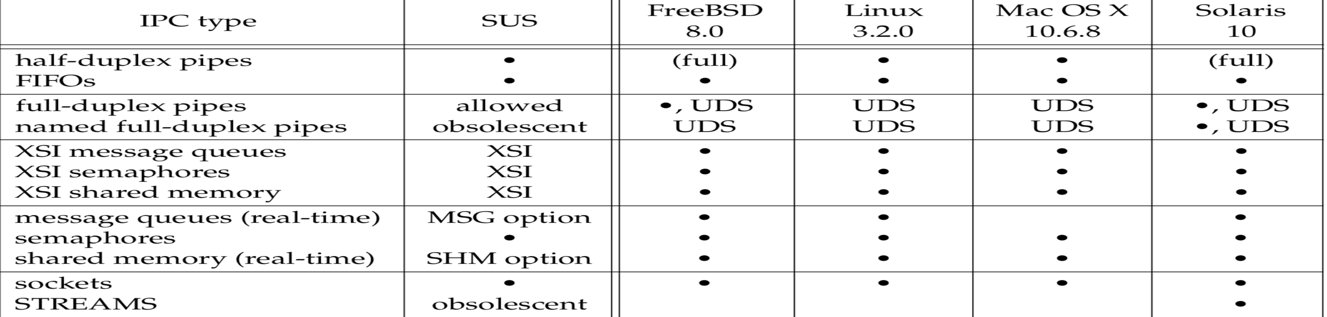
int brk(void \*addr)

void \*sbrk(intptr\_t increment)

# 7. IPC—Inter-Process-Communication

## 7.1 Introduction

12 forms of IPC:



The first ten forms of IPC, are usually restricted to IPC between processes on the same host. The final two rows—sockets and STREAMS—are the only two forms that are generally supported for IPC between processes on different hosts.

To solve the communication problems between processes, three approaches are adopted.

---Pipe

---FIFOs

---message queue

---semaphore

---shared memory (shared registry)

## 7.2 Category

### 7.2.1 socket

#### 7.2.1.1 Unix Domain Socket

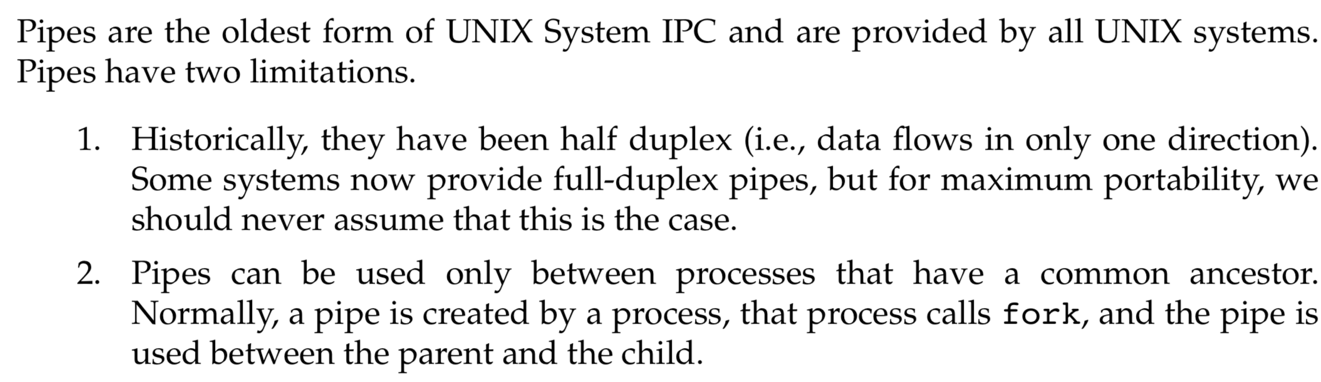
like a combination of pipe and socket

##### 7.2.1.1.1 stream

##### 7.2.1.1.1 datagram

#### 7.2.1.2 Internet Domain Socket

### 7.2.2 pipe



# 8. Mechanism

## 8.1 introduction

## 8.2 Message Queue

### 8.2.1 Micro-level usage

#### 8.2.1.1 IPC

### 8.2.2 Macro-level usage

#### 8.2.2.1 rabbit mq

## 8.3 Semaphore

## 8.4 Shared registry