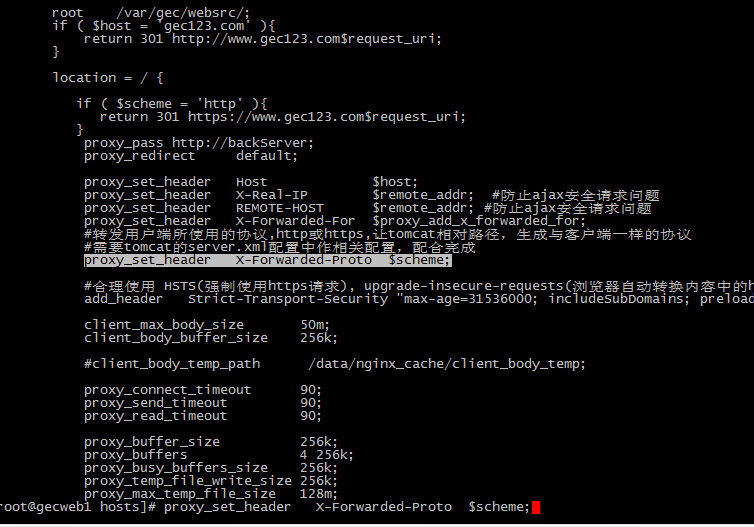
# 复制与粘贴屏幕内容

使用鼠标左键在屏幕上拉选要复制的内容，然后右键可以快速将内容复制到，当前录入位置，如下：



也可以，选择后按“Shift+home”组合键复制系统“粘贴板”，在需要粘贴的地址使用对应的粘贴功能完成粘贴，在操作控制台是使用“Shift+insert”组合键。

# 中止程序执行

组合键“Ctr+c”

# 文件和命令自动补全

（1）只需输入文件或目录名的前几个字符，然后按TAB键，如无相重的，完整的文件名立即自动在命令行出现；如有相重的，再按一下TAB键，系统会列出当前目录下所有以这几个字符开头的名字。

（2）在命令行下，只需输入例如m，再连续按两次TAB键，系统将列出所有以m开头的命令，(包括自定义的Bshell命令函数)，对查找某些记不清楚的命令特有用。例如输入ftp，将查到ftp，ftpcount，ftpwho，ftpshut等本不熟悉的命令。

# 返回前一次路径

[root@localhost ~]# cd -

# 搜索历史命令

（1）↑↓

前、后查找近期使用命令

（2）Ctrl + R

Linux下的神器Ctrl + R(reverse-i-search ) 的使用方法：  
(press Ctrl + R ) 输入任意字符，例如： "vim"  就会出现 (reverse-i-search)`vim': vim /etc/network/interfaces    
如果我想找另一个vim命令呢？  就多按几次 Ctrl + R ，就可以继续向前搜索 “vim" 的内容了 :(reverse-i-search)`vim': vim .ssh/known\_hosts  
如果找到了，按 -> 或者直接回车

# 设置环境变量

linux系统是多用户的任务系统，对于每个用户都可以指定特定的用户变量。针对不同程序在不同的环境下运行就需要修改环境变量进行定制。

1. **显示所有变量**

**[root@mail ~]#env**

1. **显示环境变量**

**[root@mail ~]#echo $PATH** 路径变量path

1. **变量的周期划分**
2. **临时性变量：使用命令如export等命令设置，设置之后马上生效。当关闭shell的时候失效（这种主要用于测试比较多）**

**常见的设置方法是通过命令export指定**linux最常见的就是lamp配置，指定软件运行路径为例。

比如指定php 或mysql执行的路径变量

[root@mail ~]#export PATH=$PATH:/usr/local/php/bin  为php环境

[root@mail ~]#export PATH=$PATH:/usr/local/mysql/bin 为mysql环境

如果设置多个可以采用格式**[root@mail ~]#export PATH=$PATH:/usr/local/mysql/bin**;

1. **永久变量：**

**通过修改配置文件，配置之后变量永久生效**

1. **按照影响范围分**
2. **用户变量（局部变量）**：修改的设置只对某个用户的路径或执行起作用

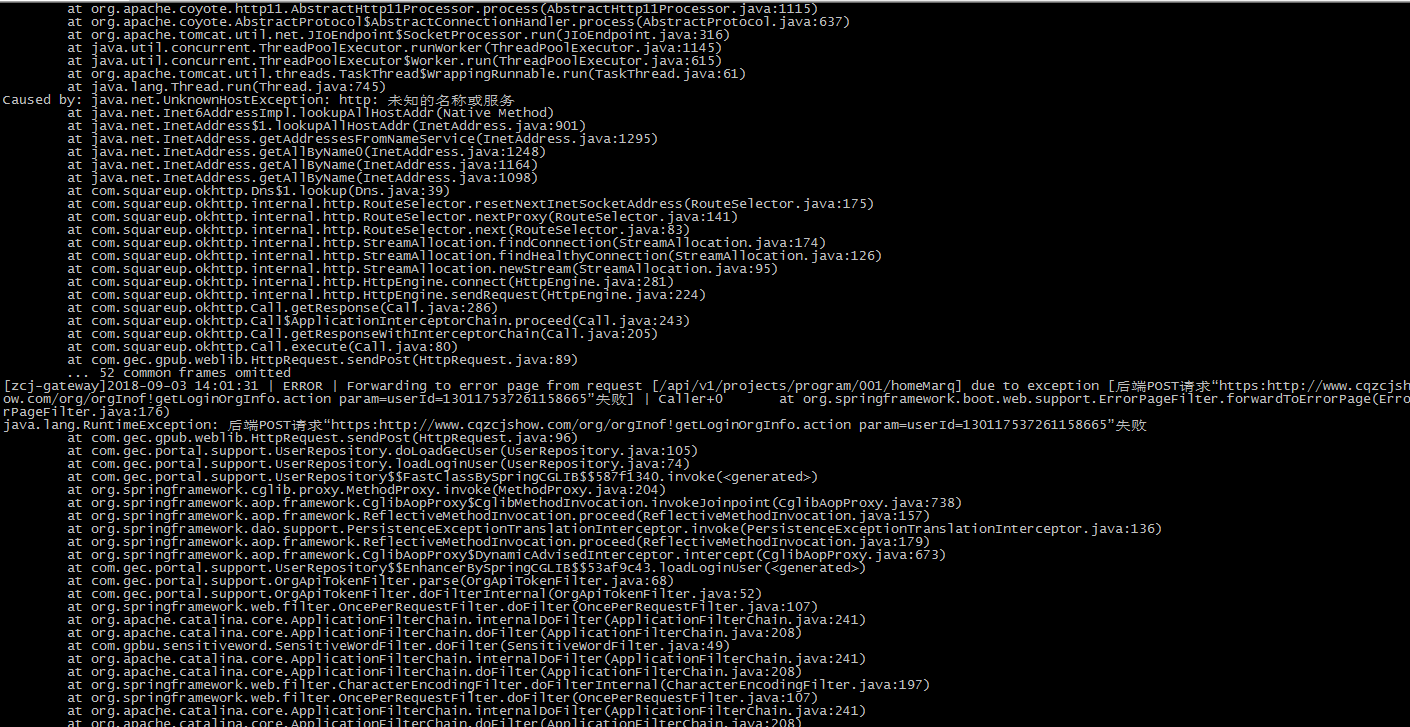
 前面介绍的变量设置关闭会话失效，对于需要写入系统变量永久生效的需要修改配置文件。一般有用户级别的变量设置，和全局系统级别设置。**修改用户变量**    linux下每个用户宿主目录下都有一个**~/.bash\_profile**文件，里面记录直接加到文件里面，最后面export path=$path:/path1:/path2

1. **系统变量（全局变量**）：影响范围是整个系统

**全局变量，系统环境变量**linux支持多用户多任务，如果要让所有用户都采用相同的变量设置。**修改/etc/profile文件，通过vi或vim增加变量**

修改变量之后，需要马上生效变量设置。可执行**[root@mail ~]source /etc/profile**

# 设置本地域名解析



在linux的文件系统里面,有这样一个文件:

# Vi /etc/hosts

此文件属于本地静态域名映射表，即在linux系统中 访问某域名时，会优先采用此文件的解析方式，然后再参考更下游的DNS服务器节点。

[复制代码](javascript:void(0);)

(pyvenv) ➜ ~ cat /etc/hosts

##

# Host Database

#

# localhost is used to configure the loopback interface

# when the system is booting. Do not change this entry.

##

127.0.0.1 localhost

255.255.255.255 broadcasthost

::1 localhost

119.14.121.48 web\_server

119.14.121.42 database\_server

119.14.121.40 nginx\_server

[复制代码](javascript:void(0);)

对于一个经常要和各种服务器打交道的开发人员来说，可以不必花脑力去记住各种服务器的IP，而是通过容易记忆的名称来给IP标记成别名，即DNS的功能。

比如，如果要远程登录到Web服务器，在设置hosts之前 ，需要执行如下命令：

ssh root@119.14.121.48

而经过以上设置之后，只需要执行下面的操作：

ssh root@web\_server

可能一台服务器的登录并没有显示出此方式的 **奇技淫巧**,但是只要服务器数量增加后，比如到10台，记忆10台机器的名称应该比记忆10个数字IP要容易得多吧，因为你标记的别名会引起你的联想，但是IP则不会。

# 使用stat命令查看一个文件的属性

1. 访问时间（**Access**）、修改时间（**modify**）、状态改变时间（**Change**）
2. **# stat** **index**.php
3. **Access**: 2018**-05-10** 02:37:44.169014602 **-0500**
4. **Modify**: 2018**-05-09** 10:53:14.395999032 **-0400**
5. **Change**: 2018**-05-09** 10:53:38.855999002 **-0400**

# 文本内容查看、查找

1. 直接将文件内容全录出

cat 文件名

1. 查看文件开始指定行

head –100 (查看前100行 )

1. 日志文件不断查看其输出

需要用tail命令，也是我们使用得很多的

tail -100（查看最后100行）

当需要一直输出查看日志时

tail –f

-f方式，会让操作控制台持续挂起，有新的内容时，持续输出。要退出，请按Ctr+z。

1. 查看文本中20至30行内容（总共100行）

方法1：

**#** sed -n '20,30p' test.txt

方法2：

**#** head -30 test.txt |tail

1. 如何查看指定内容的输出，用于跟踪问题

tail –f |grep ‘gec’(查看含gec串的输出行，多个条件可以再加｜grep,另外他的查找是支持正规表达式的，可以更灵活的查找)。

1. **近期修改文件查找：**

***find命令的路径是作为第一个参数的， 基本命令格式为 find [path] [option] [action]***   
与时间相关的参数：   
-atime 最后访问时间   
-ctime 创建时间   
-mtime 最后修改时间

列出home目录中，当天（24小时之内）有改动的文件：

# findt / -mtime 0

1. 当日志文件很大，需要将后面的部分复制下来，如何处理

使用tail,当结果输出到另一文件

tail 1000 >new.log

<https://www.cnblogs.com/cangqiongbingchen/p/5023886.html>

# 如何修改配置文件

命令:vi或vim(本人一般用vi就够了)

1. 打开如新建文件

vi 文件名

1. 切换编辑与查看状态

按键盘字母“i”,进行入插入编辑状态，再按“Esc”键退出编辑状态。

1. **删除行**

在非编辑状态下，双击键盘字母“d”

1. **定位到行号**

在非编辑状态下”:n”回车,n为行号

1. 保存

在非编辑状态下”:w”回车

1. 保存退出

在非编辑状态下”:wq”回车

1. 退出

在非编辑状态下”:q”回车

1. 放弃修改直接退出

在非编辑状态下”:q!”回车

1. **查找**

在非编辑状态下”/字符串”回车，查找下一个，继续按键盘字母“n”。

1. **查找替换**

在非编辑状态下” :%s/查找串/替换串/g”回车(特殊字符需要用“\”转义)

# 使用ssh-key认证身份

说到ssh，对于linux环境下的开发人员来说，并不陌生。一般情况下，使用频率最高的来自于如下两个场景：

* 远程服务器登录操作身份认证
* git代码服务器身份认证

这两种使用场景都需要 **身份认证** ，传统的方式都是直接输入密码，此方式的缺点是：

* 输入过程繁琐
* 需要用户记忆密码
* 密码可能会被截获

而使用ssh-key则使用一组 公钥/私钥 来的加密串来进行身份认证。

生成方法如下：

# ssh-keygen -t rsa

Generating public/private rsa key pair.

Enter file in which to save the key (/home/harmo/.ssh/id\_rsa):

根据向导完成后续的设置，会在本机生成一组文件：

-rw------- 1 harmo staff 1.6K 1 25 16:26 id\_rsa

-rw------- 1 harmo staff 401B 1 25 16:26 id\_rsa.pub

其中：

* id\_rsa 私钥
* id\_rsa.pub 公钥

这组文件直接存放在本机，只要将公钥信息复制到相应的服务器(用户目录.ssh/authorized\_keys)，设置为信任公钥即可。

后续的过程中，客户机和服务器之间的身份认证则自动通过ssh-key进行认证，不再需要用户输入甚至记忆密码了。

虽然对于免密码输入还有一些别的技术解决方案，例如：

* sshpass
* .netrc

但是它们的缺点是：仍然需要把账号和密码明文保存在本机。这样会存在这样的危险：只要本机被hack，这些明文直接会丢失，造成的影响也远比ssh-key丢失要严重。

# 生成登录证书

1. 生成SSH密钥对 # ssh-keygen -t rsa

用默认安装路径

1. 2次输入密钥文件密码 
2. 进入~/.ssh 目录，拷贝证书文件至本地：id\_rsa、 id\_rsa.pub
3. 在 /root/.ssh/... 下创建 <strong> authorized\_keys </strong> 文件，执行：

cat id\_rsa.pub >> authorized\_keys

如果已有authorized\_keys文件，则：

将id\_rsa.pub文件里面的密钥串加入authorized\_keys文件中

1. 编辑sshd\_config文件： # vi /etc/ssh/sshd\_config

编辑内容如下：

#禁用密码验证

PasswordAuthentication no

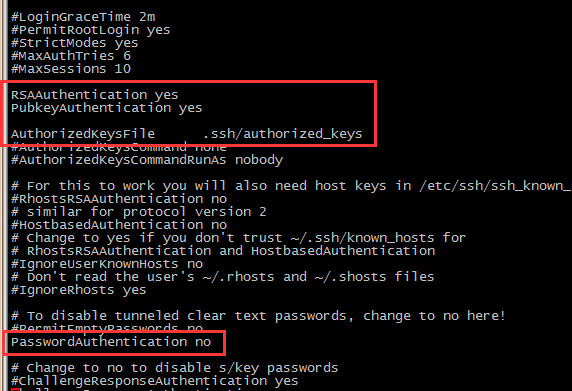
#启用密钥验证

RSAAuthentication yes

PubkeyAuthentication yes

#指定公钥数据库文件

AuthorsizedKeysFile .ssh/authorized\_keys



PasswordAuthentication no禁用密码验证这个参数,一定要在证书安装好，验证能用证书登录后，这个参数才能关闭，否则可能远程无法连接

1. 重启SSH服务前建议多保留一个会话以防不测

service sshd restart 或 systemctl restart sshd.service

tail -100  /var/log/secure

有日志打印：Authentication refused: bad ownership or modes for file /home/xxxxxx/.ssh/authorized\_keys

解决方法：chmod 600  /home/xxxx/.ssh/authorized\_keys

# 查看网络状态命令(端口)

1. *netstat -antp #查看所有网络连接*
2. *netstat  -anp  |grep   端口号 查看某个端口是否被占用*
3. *netstat -lntp #只查看监听的端口信息*

# IP调整

1. 查看本机网络配置：

ip a

1. 找到对应网卡配置调整，修改：

/etc/sysconfig/network-scripts

vi enp2s0f0;

1. 重启网卡服务

service network restart

# 如何查看磁盘分区及使用情况(清理磁盘空间)

1. df -lh --查看磁盘分区及使用情况
2. 查找大文件
3. 方案一：du命令

案例：查看当前目录下所有文件及目录点用空间大小

du -h --max-depth=2 | sort -n

（说明：--max-depth为深度，1则为当前目录下，如何想递归2级，则设置为2，sort按大小排序）

1. 方案二：find命令

find / -type f -size +100M

1. 删除大文件：
2. 方式一：cat nul>/var/log/nginx/szwhmaster.log【具体的日志文件名】

或者： echo "">catalina.out

1. 方式二：进入大文件目录 🡪 删除文件：rm –f xxxx 【rm命令一定要慎用】
2. 无大文件，但是空间就是降不下来：

lsof -n |grep delete > /root/io.txt

查看io.txt，将占用IO磁盘很大的进程给kill掉

<https://www.cnblogs.com/xxdfly/p/6564294.html>

# Vi 保存一个没有权限的已编辑文件

用Vim对vim /etc/ssh/sshd\_config文件做了更改，保存时才发现对该文件仅有只读权限，不能进行保存操作。  
告警内容如下：

解决方法1：:把文件关闭，获取权限以后再重新打开。但是如果你已经做了大量更改，这样做会很浪费时间，因为在这个过程中所有的更改都会丢失。  
解决方法2：:在保存文件前更改文件权限。  
命令是：  
**:w !sudo tee %**  
这个命令将会要你输入密码，就像在命令行中使用 sudo vim一样。

# Bash快捷键

在控制台窗口中，输入一长串命令时，发现有个错误，可以用下面几个快捷键

Ctrl+a:光标移到行首  
Ctrl + u ：从光标处删除至命令行首  
Ctrl + k ：从光标处删除至命令行尾  
Ctrl + w ：从光标处删除至字首  
Alt + d ：从光标处删除至字尾

# 文本中两列位置替换

***#*** *cat a.txt*

60.35.1.15 www.baidu.com

45.46.26.85 www.sina.com.cn

***#*** *awk '{print $2"\t"$1}' a.txt*

<https://blog.csdn.net/aywb1314/article/details/52239281>

# 压缩文件操作

1. **zip打包压缩命令**

# zip -r myfile.zip ./\*  
将当前目录下的所有文件和文件夹全部压缩成myfile.zip文件,－r表示递归压缩子目录下所有文件.

你可以使用 zip 命令同时处理多个文件和目录，方法是将它们逐一列出，并用空格间隔：

# zip -r filename.zip file1 file2 file3 /usr/work/school

1. **Unzip解压命令**

#unzip -o -d /home/sunny myfile.zip  
把myfile.zip文件解压到 /home/sunny/  
-o:不提示的情况下覆盖文件；  
-d:-d /home/sunny 指明将文件解压缩到/home/sunny目录下；

1. **删除压缩文件中某个文件**#zip -d myfile.zip filename
2. **向压缩文件中添加某个文件**#zip -m myfile.zip ./rpm\_info.txt
3. **tar命令压缩排除某个目录**

**#** tar zcvf data.tar.gz /data --exclude=tmp *#--exclude参数为不包含某个目录或文件，后面也可以跟多个*

1. **查看tar包存档文件，不解压**

# tar tf data.tar.gz #t是列出存档文件目录，f是指定存档文件

1. **解压tar包存档文件**

# tar zxvf data.tar.gz

<https://www.cnblogs.com/zdz8207/p/3765604.html>

# 找出IP地址

# tail -50000 /var/gpwshow/tomcat/logs/catalina.out |grep -o '[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}' #-o只显示匹配字符

参考原文：

<https://blog.csdn.net/zhenliang8/article/details/80800330>

<https://blog.csdn.net/u010102951/article/details/44116087>