
回到首页

芋道源码 —— 知识星球

我是一段不羁的公告！
记得给芳芳这 3 个项目加油，添加一个 STAR 噢。
https://github.com/YunaiV/SpringBoot-Labs
https://github.com/YunaiV/onemall
https://github.com/YunaiV/ruoyi-vue-pro
2020-01-10

精尽 Linux 面试题「最新更新时间：2023-07」

以下面试题，基于网络整理，和自己编辑。具体参考的文章，会在文末给出所有的链接。

如果胖友有自己的疑问，欢迎在星球提问，我们一起整理吊吊的【网络】面试题的大保健。

而题目的难度，芳芳尽量按照从容易到困难的顺序，逐步下去。

另外，本面试题面向的是开发，而不是运维。所以会写的相对简单，以开发常见的 Linux 面试题为准。并且，本文会整理一些常用的 Linux 的命令，对应标题为 `命令` 结果为的小节，例如说 `cp` 命令，对应到 “cp 命令” 小节。

常用命令

> 芳芳：这一小节会非常非常非常长，当做温故知新吧。
>
> 另外，面试官也可能会问，你熟悉 Linux 么？你平时使用哪些 Linux 命令。酱紫的连环炮~
>
> 当然，建议重点看下 「性能相关」 小节。

目录相关

find 命令

《Linux 命令大全 —— find 命令》

> 查找指定文件名的文件(不区分大小写)：`find -iname "MyProgram.c"` 。
> 对找到的文件执行某个命令：`find -iname "MyProgram.c" -exec md5sum {} \;` 。
> 查找 home 目录下的所有空文件：`find ~ -empty` 。

【常用】如何在 `/usr` 目录下找出大小超过 10MB 的文件?

> 输入命令 `find /usr -type f -size +10240k` 命令。
>
> 输出结果如下：

```
/usr/lib/locale/locale-archive
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.151-1.b12.el7_4.x86_64/jre/lib/amd64/server/classes.jsa
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.151-1.b12.el7_4.x86_64/jre/lib/amd64/server/libjvm.so
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.151-1.b12.el7_4.x86_64/jre/lib/rt.jar

// ... 省略
```

另外，du 命令也可以做类似的事情，可以看看 《查找 Linux 系统中的占用磁盘空间最大的前 10 个文件或文件夹》 文章。

如何在 /var 目录下找出 90 天之内未被访问过的文件？

输入命令：find /var \! -atime -90 。

如何在 /home 目录下找出 120 天之前被修改过的文件？

输入命令：find /home -mtime +120 。

在整个目录树下查找文件 “core” ，如发现则无需提示直接删除它们？

输入命令：find / -name core -exec rm {} \; 。

ls 命令

《Linux 命令大全 —— ls 命令》

以易读的方式显示文件大小(显示为 MB,GB...)：ls -lh 。
以最后修改时间升序列出文件：ls -ltr 。
在文件名后面显示文件类型：ls -F 。

pwd 命令

《Linux 命令大全 —— pwd 命令》

输出当前工作目录：pwd 。

cd 命令

《Linux 命令大全 —— cd 命令》

cd ： 可以在最近工作的两个目录间切换。

mkdir 命令

《Linux 命令大全 —— mkdir 命令》

在 home 目录下创建一个名为 temp 的目录：mkdir ~/temp 。
使用 -p 选项可以创建一个路径上所有不存在的目录：mkdir -p dir1/dir2/dir3/dir4/ 。

df 命令

《Linux 命令大全 —— df 命令》

显示文件系统的磁盘使用情况，默认情况下 `df -k` 将以字节为单位输出磁盘的使用量。
使用 `df -h` 选项可以以更符合阅读习惯的方式显示磁盘使用量。
使用 `df -T` 选项显示文件系统类型。

rm 命令

《Linux 命令大全 —— rm 命令》

删除文件前先确认：`rm -i filename.txt` 。
在文件名中使用 shell 的元字符会非常有用。删除文件前先打印文件名并进行确认：`rm -i file*` 。
递归删除文件夹下所有文件，并删除该文件夹：`rm -r example` 。

mv 命令

《Linux 命令大全 —— mv 命令》

将 file1 重命名为 file2 ，如果 file2 存在则提示是否覆盖：`mv -i file1 file2` 。
`-v` 会输出重命名的过程，当文件名中包含通配符时，这个选项会非常方便：`mv -v file1 file2` 。

cp 命令

《Linux 命令大全 —— cp 命令》

拷贝 file1 到 file2 ，并保持文件的权限、属主和时间戳：`cp -p file1 file2` 。
拷贝 file1 到 file2 ，如果 file2 存在会提示是否覆盖：`cp -i file1 file2` 。

有一普通用户想在每周日凌晨零点零分定期备份 `/user/backup`到 `/tmp` 目录下，该用户应如何做?

配置如下：

```
crontab -e
0 0 * * 7 /bin/cp /user/backup /tmp
```

每周一下午三点将 `/tmp/logs` 目录下面的后缀为 `*.log` 的所有文件 rsync 同步到备份服务器 192.168.1.100 中同样的目录下面

配置如下：

```
crontab -e
00 15 * * 1 rsync -avzP /tmp/logs/*.log root@192.168.1.100:/tmp/logs
```

相比来说，`rsync` 比 `scp` 的性能更好。具体可以看看 《scp 与 rsync 性能实测》 文章。

mount 命令

《Linux 命令大全 —— mount 命令》

如果要挂载一个文件系统，需要先创建一个目录，然后将这个文件系统挂载到这个目录上：

```
# mkdir /u01

# mount /dev/sdb1 /u01
```

也可以把它添加到 fstab 中进行自动挂载，这样任何时候系统重启的时候，文件系统都会被加载：

```
/dev/sdb1 /u01 ext2 defaults 0 2
```

cat 命令

[《Linux 命令大全 —— cat 命令》]

你可以一次查看多个文件的内容，下面的命令会先打印 file1 的内容，然后打印 file2 的内容：`cat file1 file2` 。
`-n` 命令可以在每行的前面加上行号：`cat -n /etc/logrotate.conf` 。

如何看当前 Linux 系统有几颗物理 CPU 和每颗 CPU 的核数？

```
[root@centos6 ~ 10:55 #35]# cat /proc/cpuinfo|grep -c 'physical id'
4

[root@centos6 ~ 10:56 #36]# cat /proc/cpuinfo|grep -c 'processor'
4
```

tail 命令

[《Linux 命令大全 —— tail 命令》]

tail 命令默认显示文件最后的 10 行文本：`tail filename.txt` 。
你可以使用 `-n` 选项指定要显示的行数：`tail -n N filename.txt` 。
你也可以使用 `-f` 选项进行实时查看，这个命令执行后会等待，如果有新行添加到文件尾部，它会继续输出新的行，在查看日志时这个选项会非常有用。你可以通过 `CTRL-C` 终止命令的执行：`tail -f log-file` 。

less 命令

[《Linux 命令大全 —— less 命令》]

这个命名可以在不加载整个文件的前提下显示文件内容，在查看大型日志文件的时候这个命令会非常有用：`less huge-log-file.log` 。

当你用 less 命令打开某个文件时，下面两个按键会给你带来很多帮助，他们用于向前和向后滚屏：

```
CTRL+F - forward one window
CTRL+B - backward one window
```

通用命令

grep 命令

《Linux 命令大全 —— grep 命令》

在文件中查找字符串(不区分大小写)：grep -i "the" demo_file 。
输出成功匹配的行，以及该行之后的三行：grep -A 3 -i "example" demo_text 。
在一个文件夹中递归查询包含指定字符串的文件：grep -r "ramesh" * 。

sed 命令

《Linux 命令大全 —— sed 命令》

当你将 Dos 系统中的文件复制到 Unix/Linux 后，这个文件每行都会以 \r\n 结尾，sed 可以轻易将其转换为 Unix 格式的文件，使用\n 结尾的文件：sed 's/.$//' filename 。
反转文件内容并输出：sed -n '1!G; h; p' filename 。
为非空行添加行号：sed '/./=' thegeekstuff.txt | sed 'N; s/\n/ /' 。

用 sed 命令将指定的路径 /usr/local/http 替换成为 /usr/src/local/http ？

```
[root@centos7 ~]# echo "/usr/local/http/" | sed 's#/usr/local/#/usr/src/local/#'
```

打印 /etc/ssh/sshd_config 的第一百行？

```
sed -n '100p' /etc/ssh/sshd_config
```

用 sed 命令永久关闭防火墙？

```
[root@centos7 ~]# sed -i.bak 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config
[root@centos7 ~]# cat /etc/selinux/config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

awk 命令

《Linux 命令大全 —— awk 命令》

删除重复行：$ awk '!($0 in array) { array[$0]; print}' temp 。
打印 /etc/passwd 中所有包含同样的 uid 和 gid 的行：awk -F ':' '$3=$4' /etc/passwd 。

打印文件中的指定部分的字段：`awk '{print $2,$5;}' employee.txt` 。

可能会有胖友刚开始会懵逼，awk 和 sed 命令不是类似的么，那么就可以看看 《【总结】awk 与 sed 的区别》 。

打印 /etc/passwd 的 1 到 3 行？

使用 sed 命令：

```
[root@centos7 ~]# sed -n '1,3p' /etc/passwd
root:x:0:0:root:/root:/bin/bash
system:x:0:0::/home/system:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
```

使用 awk 命令：

```
[root@centos7 ~]# awk 'NR>=1&&NR<=3{print $0}' /etc/passwd
root:x:0:0:root:/root:/bin/bash
system:x:0:0::/home/system:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
```

vim 命令

《Linux vi/vim》

打开文件并跳到第 10 行：`vim +10 filename.txt` 。
打开文件跳到第一个匹配的行：`vim +/search-term filename.txt` 。
以只读模式打开文件：`vim -R /etc/passwd` 。

diff 命令

貌似不太常用，当学习下。

《Linux 命令大全 —— diff 命令》

比较的时候忽略空白符：`diff -w name_list.txt name_list_new.txt` 。

sort 命令

貌似不太常用，当学习下。

《Linux 命令大全 —— sort 命令》

以升序对文件内容排序：`sort names.txt` 。
以降序对文件内容排序：`sort -r names.txt` 。
以第三个字段对 `/etc/passwd` 的内容排序：`sort -t: -k 3n /etc/passwd | more` 。

****

xargs 命令

《Linux 命令大全 —— xargs 命令》

将所有图片文件拷贝到外部驱动器：`ls *.jpg | xargs -n1 -i cp {} /external-hard-drive/directory` 。
将系统中所有 jpg 文件压缩打包：`find / -name *.jpg -type f -print | xargs tar -cvzf images.tar.gz` 。
下载文件中列出的所有 url 对应的页面：`cat url-list.txt | xargs wget -c` 。

把当前目录下所有后缀名为 .txt 的文件的权限修改为 777 ？

方式一，使用 xargs 命令：`find ./ -type f -name "*.txt" |xargs chmod 777` 。
方式二，使用 exec 命令：`find ./ -type f -name "*.txt" -exec chmod 777 {}` 。

压缩相关

tar 命令

《tar 压缩解压缩命令详解》

创建一个新的 tar 文件：`tar cvf archive_name.tar dirname/` 。
解压 tar 文件：`tar xvf archive_name.tar` 。
查看 tar 文件：`tar tvf archive_name.tar` 。

gzip 命令

《Linux 命令大全 —— gzip 命令》

创建一个 *.gz 的压缩文件：`gzip test.txt` 。
解压 *.gz 文件：`gzip -d test.txt.gz` 。
显示压缩的比率：`gzip -l *.gz` 。

bzip2 命令

《Linux 命令大全 —— bzip2 命令》

创建 *.bz2 压缩文件：`bzip2 test.txt` 。
解压 *.bz2 文件：`bzip2 -d test.txt.bz2` 。

unzip 命令

《Linux 命令大全 —— unzip 命令》

解压 *.zip 文件：`unzip test.zip` 。
查看 *.zip 文件的内容：`unzip -l jasper.zip` 。

系统命令

export 命令

《Linux 命令大全 —— export 命令》

输出跟字符串 oracle 匹配的环境变量：`export | grep ORCALE` 。
设置全局环境变量：`export ORACLE_HOME=/u01/app/oracle/product/10.2.0` 。

kill 命令

《Linux 命令大全 —— kill 命令》

kill 用于终止一个进程。一般我们会先用 ps -ef 查找某个进程得到它的进程号，然后再使用 kill -9 进程号终止该进程。你还可以使用killall、pkill、xkill 来终止进程

> 芳芳：注意，-9 表示强制终止指定进程。实际场景下，不会这么做。
>
> 但一般情况下，只需要 kill 进程编号 就可结束。

```
$ ps -ef | grep vim
ramesh    7243  7222  9 22:43 pts/2    00:00:00 vim

$ kill -9 7243
```

passwd 命令

《Linux 命令大全 —— passwd 命令》

> passwd 用于在命令行修改密码，使用这个命令会要求你先输入旧密码，然后输入新密码：passwd 。
> 超级用户可以用这个命令修改其他用户的密码，这个时候不需要输入用户的密码：passwd USERNAME 。
> passwd 还可以删除某个用户的密码，这个命令只有 root 用户才能操作，删除密码后，这个用户不需要输入密码就可以登录到系统：passwd -d USERNAME 。

su 命令

《Linux 命令大全 —— su 命令》

> su 命令用于切换用户账号，超级用户使用这个命令可以切换到任何其他用户而不用输入密码：su - USERNAME 。
>
> 用另外一个用户名执行一个命令下面的示例中用户 john 使用 raj 用户名执行 ls 命令，执行完后返回 john 的账号：

```
[john@dev-server]$ su - raj -c 'ls'

[john@dev-server]$
```

> 用指定用户登录，并且使用指定的 shell 程序，而不用默认的：su -s 'SHELLNAME' USERNAME 。

yum 命令

《Linux 命令大全 —— yum 命令》

> 使用 yum 安装 apache ： yum install httpd 。
> 更新 apache ： yum update httpd 。
> 卸载/删除 apache ： yum remove httpd 。

rpm 命令

《Linux 命令大全 —— rpm 命令》

使用 rpm 安装 apache ： rpm -ivh httpd-2.2.3-22.0.1.el5.i386.rpm 。
更新 apache ： rpm -uvh httpd-2.2.3-22.0.1.el5.i386.rpm 。
卸载/删除 apache ： rpm -ev httpd 。

shutdown 命令

《Linux 命令大全 —— shutdown 命令》

关闭系统并立即关机： shutdown -h now 。
10 分钟后关机： shutdown -h +10 。
重启： shutdown -r now 。
重启期间强制进行系统检查： shutdown -Fr now 。

crontab 命令

《Linux 命令大全 —— crontab 命令》

查看某个用户的 crontab 配置： crontab -u user -l 。
设置一个每十分钟执行一次的计划任务： */10 * * * * /home/ramesh/check-disk-space 。【前提是，在 crontab 下】

service 命令

《Linux 命令大全 —— service 命令》

service 命令用于运行 System V init 脚本，这些脚本一般位于 /etc/init.d 文件下，这个命令可以直接运行这个文件夹里面的脚本，而不用加上路径。
查看服务状态： service ssh status 。
查看所有服务状态： service --status-all 。
重启服务： service ssh restart 。

另外，使用 chkconfig 命令，可以设置服务在系统启动时，是否自动启动。详细的，见 《Linux 命令大全 —— chkconfig 命令》 文章。

chmod 命令

《Linux 命令大全 —— chmod 命令》

chmod 用于改变文件和目录的权限。
给指定文件的属主和属组所有权限(包括读、写、执行)： chmod ug+rwx file.txt 。
删除指定文件的属组的所有权限： chmod g-rwx file.txt 。
修改目录的权限，以及递归修改目录下面所有文件和子目录的权限： chmod -R ug+rwx file.txt 。

chown 命令

《Linux 命令大全 —— chown 命令》

chown 用于改变文件属主和属组。
同时将某个文件的属主改为 oracle ，属组改为 db ： chown oracle:dba dbora.sh 。

使用 -R 选项对目录和目录下的文件进行递归修改：chown -R oracle:dba /home/oracle 。

uname 命令

《Linux 命令大全 —— uname 命令》

uname 可以显示一些重要的系统信息，例如内核名称、主机名、内核版本号、处理器类型之类的信息：uname -a 。

whereis 命令

《Linux 命令大全 —— whereis 命令》

当你不知道某个命令的位置时可以使用 whereis 命令，下面使用 whereis 查找 ls 的位置：whereis ls 。
当你想查找某个可执行程序的位置，但这个程序又不在 whereis 的默认目录下，你可以使用 -B 选项，并指定目录作为这个选项的参数。下面的命令在 /tmp 目录下查找 lsmk 命令：whereis -u -B /tmp -f lsmk 。

locate 命令

《Linux 命令大全 —— locate 命令》

locate 命名可以显示某个指定文件（或一组文件）的路径，它会使用由 updatedb 创建的数据库。
下面的命令会显示系统中所有包含 crontab 字符串的文件：locate crontab 。

另外，胖友如果使用 CentOS 找不到 locate 命令，可以看看 《CentOS 系统找到 locate 命令及 locate 搜索不到存在的文件》 文章。

man 命令

《Linux 命令大全 —— man 命令》

显示某个命令的 man 页面：man crontab 。
些命令可能会有多个 ma n页面，每个 man 页面对应一种命令类型：man SECTION-NUMBER commandname 。
命令类型：

- man 页面一般可以分为 8 种命令类型。
  1. 用户命令
  2. 系统调用
  3. c 库函数
  4. 设备与网络接口
  5. 文件格式
  6. 游戏与屏保
  7. 环境、表、宏
  8. 系统管理员命令和后台运行命令
- 例如，我们执行 whatis crontab ，你可以看到 crontab 有两个命令类型 1 和 5 ，所以我们可以通过下面的命令查看命令类型 5 的 man 页面：

```
$ whatis crontab
crontab (1)          - maintain crontab files for individual users (V3)
crontab (5)          - tables for driving cron

$ man 5 crontab
```

网络相关

ifconfig 命令

[《Linux 命令大全 —— ifconfig 命令》]

ifconfig 用于查看和配置 Linux 系统的网络接口。
查看所有网络接口及其状态：`ifconfig -a` 。
使用 up 和 down 命令启动或停止某个接口：`ifconfig eth0 up` 和 `ifconfig eth0 down` 。

用一条命令显示本机 eth0 网卡的 IP 地址，不显示其它字符？

输入命令任一一个命令即可：

```
# 方法一：
ifconfig eth0|grep inet|awk -F ':' '{print $2}'|awk '{print $1}'
# 方法二
ifconfig eth0|grep "inet addr"|awk -F '[ :]+' '{print $4}'
# 方法三：
ifconfig eth0|awk -F '[ :]+' 'NR==2 {print $4}'
# 方法四：
ifconfig eth0|sed -n '2p'|sed 's#^.*addr:##g'|sed 's# Bc.*$##g'
# 方法五：
ifconfig eth0|sed -n '2p'|sed -r 's#^.*addr:(.*)  Bc.*$#\1#g'
# 方法六(CENTOS7 也适用)：
ip addr|grep eth0|grep inet|awk '{print $2}'|awk -F '/' '{print $1}'
```

ping 命令

[《Linux 命令大全 —— ping 命令》]

ping 一个远程主机，只发 5 个数据包：`ping -c 5 gmail.com` 。

如何禁止服务器被 ping ？

```
[root@node0 ~]# echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_all  // 这个时候，别人是可以 ping 通自己的
[root@node1 ~]# ping 192.168.6.6
PING 192.168.6.6 (192.168.6.6) 56(84) bytes of data.
64 bytes from 192.168.6.6: icmp_seq=1 ttl=64 time=1.79 ms
64 bytes from 192.168.6.6: icmp_seq=2 ttl=64 time=0.597 ms

[root@node0 ~]# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all
[root@node1 ~]# ping 192.168.6.6    // ping 不能了
PING 192.168.6.6 (192.168.6.6) 56(84) bytes of data.
--- 192.168.6.6 ping statistics ---
93 packets transmitted, 0 received, 100% packet loss, time 92168ms
```

curl 命令

《Linux 命令大全 —— curl 命令》

如果我们使用 ping 测试某个地址是否能连接，那么 curl 测试用个 URL 是否可以访问。

写出一个 curl 命令，访问指定服务器 61.135.169.121 上的如下 URL ： http://www.baidu.com/s?wd=test ，访问的超时时间是 20 秒

输入命令 curl --connect-timeout 20 http://61.135.169.121/s?wd=test 。

wget 命令

《Linux 命令大全 —— wget 命令》

使用 wget 从网上下载软件、音乐、视频：wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-3.2.1.tar.gz 。
下载文件并以指定的文件名保存文件：wget -O taglist.zip http://www.vim.org/scripts/download_script.php?src_id=7701 。

ftp 命令

《Linux 命令大全 —— ftp 命令》

ftp 命令和 sftp 命令的用法基本相似。

连接 ftp 服务器并下载多个文件：

```
$ ftp IP/hostname
ftp> mget *.html
```

显示远程主机上文件列表：

```
ftp> mls *.html -
/ftptest/features.html
/ftptest/index.html
/ftptest/othertools.html
/ftptest/samplereport.html
/ftptest/usage.html
```

ssh 命令

《Linux 命令大全 —— ssh 命令》

登录到远程主机：ssh username@remotehost.example.com 。
显示 ssh 客户端版本：ssh -V 。

【重要】服务器状态相关

通过如下命令，我们可以知道 Linux 服务器运行情况，从而可以排查性能的情况。

因为我们是每小节一个命令，胖友后面可以看看 《Linux 性能分析工具介绍（CPU，内存，磁盘

I/O，网络）》 文章，它将本小节的命令，又做了一次归类，和介绍。所以，可以结合着一起读读。

ps 命令

《Linux 命令大全 —— ps 命令》

> ps 命令用于显示正在运行中的进程的信息。
> 查看当前正在运行的所有进程：`ps -ef | more` 。
> 以树状结构显示当前正在运行的进程，H 选项表示显示进程的层次结构：`ps -efH | more` 。

查看后台所有 java 进程？

> 方式一：`ps -ef |grep java` 。
> 方式二：`jps -m` 。

uptime 命令

《Linux 命令大全 —— uptime 命令》

```
$ uptime
23:51:26 up 21:31,  1 user,  load average: 30.02, 26.43, 19.02
```

这个命令可以快速查看机器的负载情况。在 Linux 系统中，这些数据表示等待 CPU 资源的进程和阻塞在不可中断 IO 进程（进程状态为 D）的数量。这些数据可以让我们对系统资源使用有一个宏观的了解。

命令的输出分别表示 1 分钟、5 分钟、15 分钟的平均负载情况。通过这三个数据，可以了解服务器负载是在趋于紧张还是趋于缓解。

> 如果 1 分钟平均负载很高，而 15 分钟平均负载很低，说明服务器正在命令高负载情况，需要进一步排查 CPU 资源都消耗在了哪里。
> 反之，如果 15 分钟平均负载很高，1 分钟平均负载较低，则有可能是 CPU 资源紧张时刻已经过去。
> 上面例子中的输出，可以看见最近 1 分钟的平均负载非常高，且远高于最近 15 分钟负载，因此我们需要继续排查当前系统中有什么进程消耗了大量的资源。可以通过下文将会介绍的 vmstat、mpstat 等命令进一步排查。

另外，还有一个 《Linux 命令大全 —— w 命令》 ，也是使用比较方便的，快速查看系统负载情况的命令。

dmesg 命令

《Linux 命令大全 —— dmesg 命令》

```
$ dmesg | tail
[1880957.563150] perl invoked oom-killer: gfp_mask=0x280da, order=0, oom_score_adj=0
[...]
[1880957.563400] Out of memory: Kill process 18694 (perl) score 246 or sacrifice child
[1880957.563408] Killed process 18694 (perl) total-vm:1972392kB, anon-rss:1953348kB, file-rss:0kB
[2320864.954447] TCP: Possible SYN flooding on port 7001. Dropping request.  Check SNMP counters.
```

该命令会输出系统日志的最后 10 行。示例中的输出，可以看见一次内核的 oom kill 和一次 TCP 丢包。这些日志可以帮助排查性能问题。千万不要忘了这一步。

vmstat 命令

[《Linux 命令大全 —— vmstat 命令》]

```
$ vmstat 1
procs ----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
34  0    0 200889792  73708 591828    0    0     0     5    6   10 96  1  3  0  0
32  0    0 200889920  73708 591860    0    0     0   592 13284 4282 98  1  1  0  0
32  0    0 200890112  73708 591860    0    0     0     0 9501 2154 99  1  0  0  0
32  0    0 200889568  73712 591856    0    0     0    48 11900 2459 99  0  0  0  0
32  0    0 200890208  73712 591860    0    0     0     0 15898 4840 98  1  1  0  0
^C
```

vmstat 命令，每行会输出一些系统核心指标，这些指标可以让我们更详细的了解系统状态。后面跟的参数 1 ，表示每秒输出一次统计信息，表头提示了每一列的含义，这几介绍一些和性能调优相关的列：

r：等待在 CPU 资源的进程数。这个数据比平均负载更加能够体现 CPU 负载情况，数据中不包含等待 IO 的进程。如果这个数值大于机器 CPU 核数，那么机器的 CPU 资源已经饱和。
free：系统可用内存数（以千字节为单位），如果剩余内存不足，也会导致系统性能问题。下文介绍到的 free 命令，可以更详细的了解系统内存的使用情况。
si，so：交换区写入和读取的数量。如果这个数据不为 0 ，说明系统已经在使用交换区（swap），机器物理内存已经不足。
us，sy，id，wa，st：这些都代表了 CPU 时间的消耗，它们分别表示用户时间(user)、系统（内核）时间(sys)、空闲时间(idle)、IO等待时间(wait)和被偷走的时间(stolen，一般被其他虚拟机消耗)。

上述这些 CPU 时间，可以让我们很快了解 CPU 是否处于繁忙状态。一般情况下，如果用户时间和系统时间相加非常大，CPU 出于忙于执行指令。如果IO等待时间很长，那么系统的瓶颈可能在磁盘 IO 。

示例命令的输出可以看见，大量 CPU 时间消耗在用户态，也就是用户应用程序消耗了 CPU 时间。这不一定是性能问题，需要结合 r 队列，一起分析。

mpstat 命令

```
$ mpstat -P ALL 1
Linux 3.13.0-49-generic (titanclusters-xxxxx)  07/14/2015  _x86_64_ (32 CPU)
07:38:49 PM  CPU   %usr  %nice   %sys %iowait   %irq  %soft  %steal  %guest  %gnice  %idle
07:38:50 PM  all  98.47   0.00   0.75    0.00   0.00   0.00    0.00    0.00    0.00   0.78
07:38:50 PM    0  96.04   0.00   2.97    0.00   0.00   0.00    0.00    0.00    0.00   0.99
07:38:50 PM    1  97.00   0.00   1.00    0.00   0.00   0.00    0.00    0.00    0.00   2.00
07:38:50 PM    2  98.00   0.00   1.00    0.00   0.00   0.00    0.00    0.00    0.00   1.00
07:38:50 PM    3  96.97   0.00   0.00    0.00   0.00   0.00    0.00    0.00    0.00   3.03
[...]
```

该命令可以显示每个 CPU 的占用情况，如果有一个 CPU 占用率特别高，那么有可能是一个单线程应用程序引起的。

pidstat 命令

```
$ pidstat 1
Linux 3.13.0-49-generic (titanclusters-xxxxx)  07/14/2015    _x86_64_    (32 CPU)
07:41:02 PM   UID       PID    %usr %system  %guest    %CPU   CPU  Command
07:41:03 PM     0         9    0.00    0.94    0.00    0.94     1  rcuos/0
07:41:03 PM     0      4214    5.66    5.66    0.00   11.32    15  mesos-slave
07:41:03 PM     0      4354    0.94    0.94    0.00    1.89     8  java
07:41:03 PM     0      6521 1596.23    1.89    0.00 1598.11    27  java
07:41:03 PM     0      6564 1571.70    7.55    0.00 1579.25    28  java
07:41:03 PM 60004     60154    0.94    4.72    0.00    5.66     9  pidstat
07:41:03 PM   UID       PID    %usr %system  %guest    %CPU   CPU  Command
07:41:04 PM     0      4214    6.00    2.00    0.00    8.00    15  mesos-slave
07:41:04 PM     0      6521 1590.00    1.00    0.00 1591.00    27  java
07:41:04 PM     0      6564 1573.00   10.00    0.00 1583.00    28  java
07:41:04 PM   108      6718    1.00    0.00    0.00    1.00     0  snmp-pass
07:41:04 PM 60004     60154    1.00    4.00    0.00    5.00     9  pidstat
^
```

pidstat 命令输出进程的 CPU 占用率，该命令会持续输出，并且不会覆盖之前的数据，可以方便观察系统动态。如上的输出，可以看见两个 JAVA 进程占用了将近 1600% 的CPU时间，既消耗了大约 16 个 CPU 核心的运算资源。

iostat 命令

```
$ iostat -xz 1
Linux 3.13.0-49-generic (titanclusters-xxxxx)  07/14/2015  _x86_64_ (32 CPU)
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          73.96    0.00    3.73    0.03    0.06   22.21
Device:   rrqm/s   wrqm/s     r/s     w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await r_await w_await  svctm
xvda        0.00     0.23    0.21    0.18     4.52     2.08    34.37     0.00    9.98   13.80    5.42   2.44
xvdb        0.01     0.00    1.02    8.94   127.97   598.53   145.79     0.00    0.43    1.78    0.28   0.25
xvdc        0.01     0.00    1.02    8.86   127.79   595.94   146.50     0.00    0.45    1.82    0.30   0.27
dm-0        0.00     0.00    0.69    2.32    10.47    31.69    28.01     0.01    3.23    0.71    3.98   0.13
dm-1        0.00     0.00    0.00    0.94     0.01     3.78     8.00     0.33  345.84    0.04  346.81   0.01
dm-2        0.00     0.00    0.09    0.07     1.35     0.36    22.50     0.00    2.55    0.23    5.62   1.78
[...]
```

r/s, w/s, rkB/s, wkB/s：分别表示每秒读写次数和每秒读写数据量（千字节）。读写量过大，可能会引起性能问题。

await：IO 操作的平均等待时间，单位是毫秒。这是应用程序在和磁盘交互时，需要消耗的时间，包括 IO 等待和实际操作的耗时。如果这个数值过大，可能是硬件设备遇到了瓶颈或者出现故障。

avgqu-sz：向设备发出的请求平均数量。如果这个数值大于 1 ，可能是硬件设备已经饱和（部分前端硬件设备支持并行写入）。

%util：设备利用率。这个数值表示设备的繁忙程度，经验值是如果超过 60 ，可能会影响 IO 性能（可以参照 IO 操作平均等待时间）。如果到达 100% ，说明硬件设备已经饱和。

如果显示的是逻辑设备的数据，那么设备利用率不代表后端实际的硬件设备已经饱和。值得注意的是，即使 IO 性能不理想，也不一定意味这应用程序性能会不好，可以利用诸如预读取、写缓存等策略提升应用性能。

free 命令

《Linux 命令大全 —— free 命令》

```
$ free -m
             total       used       free     shared    buffers     cached
Mem:        245998      24545     221453         83         59        541
-/+ buffers/cache:      23944     222053
Swap:            0          0          0
```

free 命令可以查看系统内存的使用情况，-m 参数表示按照兆字节展示。最后两列分别表示用于IO缓存的内存数，和用于文件系统页缓存的内存数。需要注意的是，第二行 -/+ buffers/cache ，看上去缓存占用了大量内存空间。

这是 Linux 系统的内存使用策略，尽可能的利用内存，如果应用程序需要内存，这部分内存会立即被回收并分配给应用程序。因此，这部分内存一般也被当成是可用内存。

如果可用内存非常少，系统可能会动用交换区(如果配置了的话)，这样会增加 IO 开销(可以在 iostat 命令中体现)，降低系统性能。

【重要】Linux系统里，您知道 buffer 和 cache 如何区分吗？

Buffer 和 Cache 都是内存中的一块区域。

> 当 CPU 需要写数据到磁盘时，由于磁盘速度比较慢，所以 CPU 先把数据存进 Buffer ，然后 CPU 去执行其他任务，Buffer中的数据会定期写入磁。
> 当 CPU 需要从磁盘读入数据时，由于磁盘速度比较慢，可以把即将用到的数据提前存入 Cache ，CPU 直接从 Cache中 拿数据要快的多。

详细的，可以看看 《Linux中 buffer/cache、swap、虚拟内存和Page》 。

sar 命令

《Linux 命令大全 —— sar 命令》

```
$ sar -n DEV 1
Linux 3.13.0-49-generic (titanclusters-xxxxx)  07/14/2015     _x86_64_    (32 CPU)
12:16:48 AM     IFACE   rxpck/s   txpck/s    rxkB/s    txkB/s   rxcmp/s   txcmp/s  rxmcst/s   %ifutil
12:16:49 AM      eth0  18763.00   5032.00  20686.42    478.30      0.00      0.00      0.00      0.00
12:16:49 AM        lo     14.00     14.00      1.36      1.36      0.00      0.00      0.00      0.00
12:16:49 AM   docker0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:16:49 AM     IFACE   rxpck/s   txpck/s    rxkB/s    txkB/s   rxcmp/s   txcmp/s  rxmcst/s   %ifutil
12:16:50 AM      eth0  19763.00   5101.00  21999.10    482.56      0.00      0.00      0.00      0.00
12:16:50 AM        lo     20.00     20.00      3.25      3.25      0.00      0.00      0.00      0.00
12:16:50 AM   docker0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
```

> sar 命令在这里可以查看网络设备的吞吐率。在排查性能问题时，可以通过网络设备的吞吐量，判断网络设备是否已经饱和。如示例输出中，eth0 网卡设备，吞吐率大概在 22 Mbytes/s

，既 176 Mbits/sec ，没有达到 1Gbit/sec 的硬件上限。

sar -n TCP,ETCP 1

```
$ sar -n TCP,ETCP 1
Linux 3.13.0-49-generic (titanclusters-xxxxx)  07/14/2015    _x86_64_    (32 CPU)
12:17:19 AM  active/s passive/s    iseg/s    oseg/s
12:17:20 AM      1.00      0.00  10233.00  18846.00
12:17:19 AM  atmptf/s  estres/s retrans/s isegerr/s   orsts/s
12:17:20 AM      0.00      0.00      0.00      0.00      0.00
12:17:20 AM  active/s passive/s    iseg/s    oseg/s
12:17:21 AM      1.00      0.00   8359.00   6039.00
12:17:20 AM  atmptf/s  estres/s retrans/s isegerr/s   orsts/s
12:17:21 AM      0.00      0.00      0.00      0.00      0.00
```

sar命令在这里用于查看 TCP 连接状态，其中包括：

- active/s：每秒本地发起的TCP连接数，既通过connect调用创建的TCP连接；
- passive/s：每秒远程发起的TCP连接数，即通过accept调用创建的TCP连接；
- retrans/s：每秒TCP重传数量；

TCP 连接数可以用来判断性能问题是否由于建立了过多的连接，进一步可以判断是主动发起的连接，还是被动接受的连接。TCP 重传可能是因为网络环境恶劣，或者服务器压力。

我们可以使用哪个命令查看系统的历史负载（比如说两天前的）？

```
sar -q -f /var/log/sa/sa22 # 查看 22 号的系统负载
```

top 命令

《Linux 命令大全 —— top 命令》

```
$ top
top - 00:15:40 up 21:56,  1 user,  load average: 31.09, 29.87, 29.92
Tasks: 871 total,   1 running, 868 sleeping,   0 stopped,   2 zombie
%Cpu(s): 96.8 us,  0.4 sy,  0.0 ni,  2.7 id,  0.1 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem:  25190241+total, 24921688 used, 22698073+free,    60448 buffers
KiB Swap:        0 total,        0 used,        0 free.   554208 cached Mem
   PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 20248 root      20   0  0.227t 0.012t  18748 S  3090  5.2  29812:58 java
  4213 root      20   0 2722544  64640  44232 S  23.5  0.0 233:35.37 mesos-slave
 66128 titancl+  20   0   24344   2332   1172 R   1.0  0.0   0:00.07 top
  5235 root      20   0 38.227g 547004  49996 S   0.7  0.2   2:02.74 java
  4299 root      20   0 20.015g 2.682g  16836 S   0.3  1.1  33:14.42 java
     1 root      20   0   33620   2920   1496 S   0.0  0.0   0:03.82 init
     2 root      20   0       0      0      0 S   0.0  0.0   0:00.02 kthreadd
     3 root      20   0       0      0      0 S   0.0  0.0   0:05.35 ksoftirqd/0
     5 root       0 -20       0      0      0 S   0.0  0.0   0:00.00 kworker/0:0H
     6 root      20   0       0      0      0 S   0.0  0.0   0:06.94 kworker/u256:0
     8 root      20   0       0      0      0 S   0.0  0.0   2:38.05 rcu_sched
```

top 命令包含了前面好几个命令的检查的内容。比如系统负载情况（uptime）、系统内存使用情况（free）、系统 CPU 使用情况（vmstat）等。因此通过这个命令，可以相对全面的查看系统负载的来源。同时，top 命令支持排序，可以按照不同的列排序，方便查找出诸如内存占用最多的进程、CPU占用率最高的进程等。

但是，top 命令相对于前面一些命令，输出是一个瞬间值，如果不持续盯着，可能会错过一些线索。这时可能需要暂停 top 命令刷新，来记录和比对数据。

netstat 命令

[《Linux 命令大全 —— netstat 命令》]

如何查看系统都开启了哪些端口？

```
[root@centos6 ~ 13:20 #55]# netstat -lnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State       PID/Program name
tcp        0      0 0.0.0.0:22                  0.0.0.0:*                   LISTEN      1035/sshd
tcp        0      0 :::22                       :::*                        LISTEN      1035/sshd
udp        0      0 0.0.0.0:68                  0.0.0.0:*                               931/dhclient
Active UNIX domain sockets (only servers)
Proto RefCnt Flags       Type       State         I-Node PID/Program name    Path
unix  2      [ ACC ]     STREAM     LISTENING     6825   1/init              @/com/ubuntu/upstart
unix  2      [ ACC ]     STREAM     LISTENING     8429   1003/dbus-daemon    /var/run/dbus/system_bus_socket
```

如何查看网络连接状况？

```
[root@centos6 ~ 13:22 #58]# netstat -an
Active Internet connections (servers and established)
```

```
Proto Recv-Q Send-Q Local Address               Foreign Address             State
tcp        0      0 0.0.0.0:22                  0.0.0.0:*                   LISTEN
tcp        0      0 192.168.147.130:22          192.168.147.1:23893         ESTABLISHED
tcp        0      0 :::22                       :::*                        LISTEN
udp        0      0 0.0.0.0:68                  0.0.0.0:*

// ... 省略其它
```

如何统计系统当前进程连接数？

输入命令 `netstat -an | grep ESTABLISHED | wc -l` 。
输出结果 `177` 。一共有 177 连接数。

用 netstat 命令配合其他命令，按照源 IP 统计所有到 80 端口的 ESTABLISHED 状态链接的个数？

严格来说，这个题目考验的是对 awk 的使用。

首先，使用 `netstat -an|grep ESTABLISHED` 命令。结果如下：

```
tcp        0      0 120.27.146.122:80       113.65.18.33:62721      ESTABLISHED
tcp        0      0 120.27.146.122:80       27.43.83.115:47148      ESTABLISHED
tcp        0      0 120.27.146.122:58838    106.39.162.96:443       ESTABLISHED
tcp        0      0 120.27.146.122:52304    203.208.40.121:443      ESTABLISHED
tcp        0      0 120.27.146.122:33194    203.208.40.122:443      ESTABLISHED
tcp        0      0 120.27.146.122:53758    101.37.183.144:443      ESTABLISHED
tcp        0      0 120.27.146.122:27017    23.105.193.30:50556     ESTABLISHED

// ... 省略其它
```

然后，进一步修改命令，使用 `netstat -an|grep ESTABLISHED|grep ":80"|awk 'BEGIN{FS="[[:space:]]+"}{print $4}'` 命令。结果如下：

```
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
120.27.146.122
10.47.111.216

// ... 省略其它
```

- 说明：FS 是字段分隔符，简单的可以用多个 awk 过滤。

最后，再进一步修改命令，使用 `netstat -an|grep ESTABLISHED|grep ":80"|awk 'BEGIN{FS="[[:space:]:]+"}{print $4}'|sort|uniq -c|sort -nr` 命令。结果如下：

```
47 120.27.146.122
 1 10.47.111.216
```

- 第一列为连接数，第二列为 IP 。

虽然我们这里罗列了很多的命令，下面，还是会有其它命令。

Linux 概述

Linux 的体系结构

从大的方面讲，Linux 体系结构可以分为两块：Linux 体系结构

用户空间(User Space) ：用户空间又包括用户的应用程序(User Applications)、C 库(C Library) 。
内核空间(Kernel Space) ：内核空间又包括系统调用接口(System Call Interface)、内核(Kernel)、平台架构相关的代码(Architecture-Dependent Kernel Code) 。

为什么 Linux 体系结构要分为用户空间和内核空间的原因？

1、现代 CPU 实现了不同的工作模式，不同模式下 CPU 可以执行的指令和访问的寄存器不同。
2、Linux 从 CPU 的角度出发，为了保护内核的安全，把系统分成了两部分。

用户空间和内核空间是程序执行的两种不同的状态，我们可以通过两种方式完成用户空间到内核空间的转移：1）系统调用；2）硬件中断。

什么是 Linux 内核？

了解即可。

Linux 系统的核心是内核。内核控制着计算机系统上的所有硬件和软件，在必要时分配硬件，并根据需要执行软件。

1. 系统内存管理
2. 应用程序管理
3. 硬件设备管理
4. 文件系统管理

详细的，可以看看 《是时候深入了解 Linux 的系统结构了》 。

Linux 开机启动过程？

了解即可。

1、主机加电自检，加载 BIOS 硬件信息。

2、读取 MBR 的引导文件(GRUB、LILO)。

> 芳芳：比较“闲”的胖友，可以感兴趣看看 《引导加载程序之争：了解 LILO 和 GRUB》 文章。

3、引导 Linux 内核。

4、运行第一个进程 init （进程号永远为 1 ）。

5、进入相应的运行级别。

6、运行终端，输入用户名和密码。

Linux系 统缺省的运行级别？

0. 关机。
1. 单机用户模式。
2. 字符界面的多用户模式(不支持网络)。
3. 字符界面的多用户模式。
4. 未分配使用。
5. 图形界面的多用户模式。
6. 重启。

Linux 使用的进程间通信方式？

> 了解即可，不需要太深入。

1、管道(pipe)、流管道(s_pipe)、有名管道(FIFO)。
2、信号(signal) 。
3、消息队列。
4、共享内存。
5、信号量。
6、套接字(socket) 。

详细的，可以看看：

> 《Linux 进程间通信的几种方式总结 —— linux 内核剖析（七）》
> 《目前 Linux 进程间通信的常用方法是什么(pipe？信号量？消息队列？)?》

Linux 有哪些系统日志文件？

参见 《Linux 系统日志及日志分析》 文章，比较重要的是 /var/log/messages 日志文件。

> 该日志文件是许多进程日志文件的汇总，从该文件可以看出任何入侵企图或成功的入侵。
>
> 另外，如果胖友的系统里有 ELK 日志集中收集，它也会被收集进去。

Linux 虚拟内存是什么？

> 芳芳：选读，面试不问。作为一个知识点，比较底层~

直接看 《虚拟内存的那点事儿》 文章。

磁盘、目录、文件

简单 Linux 文件系统？

在 Linux 操作系统中，所有被操作系统管理的资源，例如网络接口卡、磁盘驱动器、打印机、输入输出设备、普通文件或是目录都被看作是一个文件。

也就是说在 Linux 系统中有一个重要的概念：一切都是文件。其实这是 Unix 哲学的一个体现，而 Linux 是重写 Unix 而来，所以这个概念也就传承了下来。在 Unix 系统中，把一切资源都看作是文件，包括硬件设备。UNIX系统把每个硬件都看成是一个文件，通常称为设备文件，这样用户就可以用读写文件的方式实现对硬件的访问。

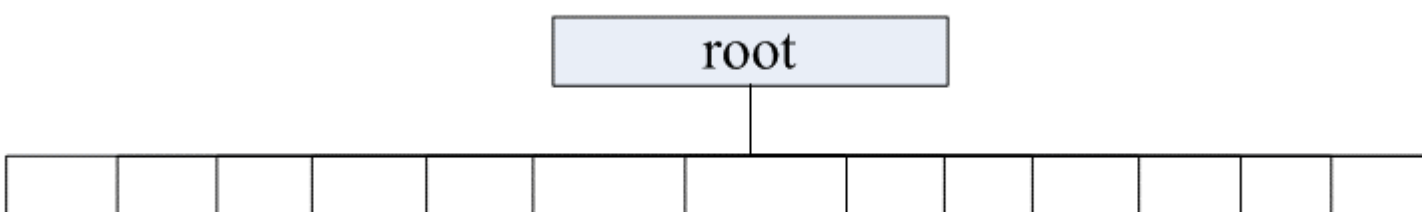
Linux 支持 5 种文件类型，如下图所示：

文件类型	描述	示例
普通文件	用来在辅助存储设备（如磁盘）上存储信息和数据	包含程序源代码（用Java等语言所编写）序、图片、声音、图
目录文件	用于表示和管理系统中的文件，目录文件中包含一些文件名和子目录名	/root、/home
链接文件	用于不同目录下文件的共享	当创建一个已存在文接时，系统就创建一这个链接文件指向已
设备文件	用来访问硬件设备	包括键盘、硬盘、光等
命名管道（FIFO）	是一种特殊类型的文件，Linux系统下，进程之间通信可以通过该文件完成	

Linux 的目录结构是怎样的？

芳芳：这个问题，一般不会问。更多是实际使用时，需要知道。

Linux 文件系统的结构层次鲜明，就像一棵倒立的树，最顶层是其根目录：



常见目录说明：

/bin： 存放二进制可执行文件(ls,cat,mkdir等)，常用命令一般都在这里；
/etc： 存放系统管理和配置文件；
/home： 存放所有用户文件的根目录，是用户主目录的基点，比如用户user的主目录就是/home/user，可以用~user表示；
/usr ： 用于存放系统应用程序；
/opt： 额外安装的可选应用程序包所放置的位置。一般情况下，我们可以把tomcat等都安装到这里；
/proc： 虚拟文件系统目录，是系统内存的映射。可直接访问这个目录来获取系统信息；
/root： 超级用户（系统管理员）的主目录（特权阶级^o^）；
/sbin: 存放二进制可执行文件，只有root才能访问。这里存放的是系统管理员使用的系统级别的管理命令和程序。如ifconfig等；
/dev： 用于存放设备文件；
/mnt： 系统管理员安装临时文件系统的安装点，系统提供这个目录是让用户临时挂载其他的文件系统；
/boot： 存放用于系统引导时使用的各种文件；
/lib ： 存放着和系统运行相关的库文件 ；
/tmp： 用于存放各种临时文件，是公用的临时文件存储点；
/var： 用于存放运行时需要改变数据的文件，也是某些大文件的溢出区，比方说各种服务的日志文件（系统启动日志等。）等；
/lost+found： 这个目录平时是空的，系统非正常关机而留下“无家可归”的文件（windows下叫什么.chk）就在这里。

什么是 inode ？

芳芳：一般来说，面试不会问 inode 。但是 inode 是一个重要概念，是理解 Unix/Linux 文件系统和硬盘储存的基础。

查看 《理解 inode》 文章。

《缺页中断 —— FIFO、LRU、OPT 这三种置换算法》 。

简述 Linux 文件系统通过 i 节点把文件的逻辑结构和物理结构转换的工作过程？

如果看的一脸懵逼，也没关系。一般来说，面试官不太会问这个题目。

Linux 通过 inode 节点表将文件的逻辑结构和物理结构进行转换。

inode 节点是一个 64 字节长的表，表中包含了文件的相关信息，其中有文件的大小、文件所有者、文件的存取许可方式以及文件的类型等重要信息。在 inode 节点表中最重要的内容是磁盘地址表。在磁盘地址表中有 13 个块号，文件将以块号在磁盘地址表中出现的顺序依次读取相应的块。
Linux 文件系统通过把 inode 节点和文件名进行连接，当需要读取该文件时，文件系统在当前目录表中查找该文件名对应的项，由此得到该文件相对应的 inode 节点号，通过该 inode 节点的磁盘地址表把分散存放的文件物理块连接成文件的逻辑结构。

什么是硬链接和软链接？

1）硬链接

由于 Linux 下的文件是通过索引节点(inode)来识别文件，硬链接可以认为是一个指针，指向文件索引节点的指针，系统并不为它重新分配 inode 。每添加一个一个硬链接，文件的链接数就加 1

。

不足：1）不可以在不同文件系统的文件间建立链接；2）只有超级用户才可以为目录创建硬链接。

2）软链接

软链接克服了硬链接的不足，没有任何文件系统的限制，任何用户可以创建指向目录的符号链接。因而现在更为广泛使用，它具有更大的灵活性，甚至可以跨越不同机器、不同网络对文件进行链接。

不足：因为链接文件包含有原文件的路径信息，所以当原文件从一个目录下移到其他目录中，再访问链接文件，系统就找不到了，而硬链接就没有这个缺陷，你想怎么移就怎么移；还有它要系统分配额外的空间用于建立新的索引节点和保存原文件的路径。

实际场景下，基本是使用软链接。详细的，胖友可以看看 《关于硬链接和软连接（符号链接）的区别》 。总结区别如下：

硬链接不可以跨分区，软件链可以跨分区。
硬链接指向一个 inode 节点，而软链接则是创建一个新的 inode 节点。
删除硬链接文件，不会删除原文件，删除软链接文件，会把原文件删除。

RAID 是什么?

RAID 全称为独立磁盘冗余阵列(Redundant Array of Independent Disks)，基本思想就是把多个相对便宜的硬盘组合起来，成为一个硬盘阵列组，使性能达到甚至超过一个价格昂贵、 容量巨大的硬盘。RAID 通常被用在服务器电脑上，使用完全相同的硬盘组成一个逻辑扇区，因此操作系统只会把它当做一个硬盘。

RAID 分为不同的等级，各个不同的等级均在数据可靠性及读写性能上做了不同的权衡。在实际应用中，可以依据自己的实际需求选择不同的 RAID 方案。

详细的，胖友可以看看 《RAID 技术介绍和总结》 。当然，因为很多公司都使用云服务，大家很难接触到 RAID 这个概念，更多的可能是普通云盘、SSD 云盘酱紫的概念。

网络

iptables 命令

iptables ，是一个配置 Linux 内核防火墙的命令行工具。功能非常强大，对于我们开发来说，主要掌握如何开放端口即可。例如：

把来源 IP 为 192.168.1.101 访问本机 80 端口的包直接拒绝：`iptables -I INPUT -s 192.168.1.101 -p tcp --dport 80 -j REJECT` 。

开启 80 端口，因为web对外都是这个端口

```
iptables -A INPUT -p tcp --dport 80 -j ACCEP
```

另外，要注意使用 `iptables save` 命令，进行保存。否则，服务器重启后，配置的规则将丢失。

详细的，可以看看：

《Linux 命令大全 —— iptables 命令》
《Linux 面试经历：iptables 面试题》
《百度 Linux 运维防火墙 iptables 的面试题》
《奇虎 360 Linux 运维工程师 iptables 防火墙面试题》

route 命令

《Linux 命令大全 —— route 命令》

添加一条到 192.168.3.0/24 的路由，网关为 192.168.1.254 ？

输入命令 route add -net 192.168.3.0/24 netmask 255.255.255.0 gw 192.168.1.254 。

查看本机路由的三种方式？

```
[root@centos6 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.220.0   0.0.0.0         255.255.255.0   U     1      0        0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U     1002   0        0 eth0
172.16.0.0      0.0.0.0         255.255.0.0     U     0      0        0 eth0
0.0.0.0         172.16.0.1      0.0.0.0         UG    0      0        0 eth0

[root@centos6 ~]# netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
192.168.220.0   0.0.0.0         255.255.255.0   U         0 0          0 eth1
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 eth0
172.16.0.0      0.0.0.0         255.255.0.0     U         0 0          0 eth0
0.0.0.0         172.16.0.1      0.0.0.0         UG        0 0          0 eth0

[root@centos6 ~]#  ip route
192.168.220.0/24 dev eth1  proto kernel  scope link  src 192.168.220.157  metric 1
169.254.0.0/16 dev eth0  scope link  metric 1002
172.16.0.0/16 dev eth0  proto kernel  scope link  src 172.16.251.6
default via 172.16.0.1 dev eth0  proto static
```

tcpdump 命令

《Linux 命令大全 —— tcpdump 命令》

在 Linux 系统下如何按照下面要求抓包：只过滤出访问 HTTP 服务的，目标 IP 为 192.168.0.111 ，一共抓 1000 个包，并且保存到 1.cap 文件中？？

```
tcpdump -nn -s0 host 192.168.0.111 and port 80 -c 1000 -w 1.cap
```

如何配置静态 IP ？

参见文章 《Linux 设置静态IP》 。这是一个必备技能。

是否可以给一个网卡配置多个 IP？

可以，参见文章 《Linux 下一个网卡配置多个 IP【虚拟ip】》 。不过，一般比较少这么做。更多的是，一台服务器有两个网卡，配置了两个不同的 IP 。

如何查看某个网卡是否连接着交换机？

芳芳：原来还有酱紫的命令！学习到新技能了。

不过阿里云不支持酱紫的操作。

mii-tool eth0 或者 mii-tool eth1 。

设置 DNS 需要修改哪个配置文件？

全局的配置，可以在 /etc/resolv.conf 文件中配置。
指定网卡的配置，可以在 /etc/sysconfig/network-scripts/ifcfg-eth0 文件中配置。
一般来说，肯定是全局配置即可。

/etc/hosts 文件什么做用？

在 /etc/hosts 文中，我们可以配置指定域名和 IP 的映射关系。详细的，可以看看 《Linux环境下 /etc/hosts 文件详解》 文章。

在 Linux 下如何指定dns服务器，来解析某个域名？

使用 dig 命令：dig @DNSip http://domain.com 。例如：

```
dig @8.8.8.8 www.baidu.com # 使用谷歌 DNS 解析百度
```

安全

一台 Linux 系统初始化环境后需要做一些什么安全工作？

1、添加普通用户登陆，禁止 root 用户登陆，更改 SSH 端口号。

修改 SSH 端口不一定绝对哈。当然，如果要暴露在外网，建议改下。

2、服务器使用密钥登陆，禁止密码登陆。

3、开启防火墙，关闭 SElinux ，根据业务需求设置相应的防火墙规则。

4、装 fail2ban 这种防止 SSH 暴力破击的软件。

5、设置只允许公司办公网出口 IP 能登陆服务器(看公司实际需要)

也可以安装 VPN 等软件，只允许连接 VPN 到服务器上。

6、修改历史命令记录的条数为 10 条。

7、只允许有需要的服务器可以访问外网，其它全部禁止。

8、做好软件层面的防护。

- 8.1 设置 nginx_waf 模块防止 SQL 注入。
- 8.2 把 Web 服务使用 www 用户启动，更改网站目录的所有者和所属组为 www 。

什么叫 CC 攻击？什么叫 DDOS 攻击？

CC 攻击，主要是用来攻击页面的，模拟多个用户不停的对你的页面进行访问，从而使你的系统资源消耗殆尽。

DDOS 攻击，中文名叫分布式拒绝服务攻击，指借助服务器技术将多个计算机联合起来作为攻击平台，来对一个或多个目标发动 DDOS 攻击。

> 攻击，即是通过大量合法的请求占用大量网络资源，以达到瘫痪网络的目的。

怎么预防 CC 攻击和 DDOS 攻击？

防 CC、DDOS 攻击，这些只能是用硬件防火墙做流量清洗，将攻击流量引入黑洞。

流量清洗这一块，主要是买 ISP 服务商的防攻击的服务就可以，机房一般有空余流量，我们一般是买服务，毕竟攻击不会是持续长时间。

例如说，《阿里云 —— DDoS 高防IP》 。

什么是网站数据库注入？

由于程序员的水平及经验参差不齐，大部分程序员在编写代码的时候，没有对用户输入数据的合法性进行判断。
应用程序存在安全隐患。用户可以提交一段数据库查询代码，根据程序返回的结果，获得某些他想得知的数据，这就是所谓的 SQL 注入。
SQL注入，是从正常的 WWW 端口访问，而且表面看起来跟一般的 Web 页面访问没什么区别，如果管理员没查看日志的习惯，可能被入侵很长时间都不会发觉。

如何过滤与预防？

数据库网页端注入这种，可以考虑使用 nginx_waf 做过滤与预防。

Shell

芳芳：本小节为选读。我也不太会写 Shell 脚本，都是写的时候，在网络上拼拼凑凑。。。

Shell 脚本是什么？

一个 Shell 脚本是一个文本文件，包含一个或多个命令。作为系统管理员，我们经常需要使用多个命令来完成一项任务，我们可以添加这些所有命令在一个文本文件(Shell 脚本)来完成这些日常工作任务。

什么是默认登录 Shell ？

在 Linux 操作系统，"/bin/bash" 是默认登录 Shell，是在创建用户时分配的。

使用 chsh 命令可以改变默认的 Shell 。示例如下所示：

```
# chsh <用户名> -s <新shell>
# chsh linuxtechi -s /bin/sh
```

在 Shell 脚本中，如何写入注释？

注释可以用来描述一个脚本可以做什么和它是如何工作的。每一行注释以 # 开头。例子如下：

```
#!/bin/bash
# This is a command
echo “I am logged in as $USER”
```

语法级

可以在 Shell 脚本中使用哪些类型的变量？

在 Shell 脚本，我们可以使用两种类型的变量：

系统定义变量

系统变量是由系统系统自己创建的。这些变量通常由大写字母组成，可以通过 set 命令查看。

用户定义变量

用户变量由系统用户来生成和定义，变量的值可以通过命令 "echo $<变量名>" 查看。

Shell脚本中 $? 标记的用途是什么？

在写一个 Shell 脚本时，如果你想要检查前一命令是否执行成功，在 if 条件中使用 $? 可以来检查前一命令的结束状态。

如果结束状态是 0 ，说明前一个命令执行成功。例如：

```
root@localhost:~# ls /usr/bin/shar
/usr/bin/shar
root@localhost:~# echo $?
0
```

如果结束状态不是0，说明命令执行失败。例如：

```
root@localhost:~# ls /usr/bin/share
ls: cannot access /usr/bin/share: No such file or directory
root@localhost:~# echo $?
2
```

Bourne Shell(bash) 中有哪些特殊的变量？

下面的表列出了 Bourne Shell 为命令行设置的特殊变量。

```
内建变量     解释
$0     命令行中的脚本名字
$1     第一个命令行参数
$2     第二个命令行参数
…..     …….
$9     第九个命令行参数
$#     命令行参数的数量
$*     所有命令行参数，以空格隔开
```

如何取消变量或取消变量赋值？

unset 命令用于取消变量或取消变量赋值。语法如下所示：

```
# unset <变量名>
```

Shell 脚本中 if 语法如何嵌套?

```
if [ 条件 ]
then
命令1
命令2
…..
else
if [ 条件 ]
then
命令1
命令2
….
else
命令1
命令2
…..
fi
fi
```

在 Shell 脚本中如何比较两个数字？

在 if-then 中使用测试命令（ -gt 等）来比较两个数字。例如：

```
#!/bin/bash
x=10
y=20
if [ $x -gt $y ]
then
echo “x is greater than y”
else
echo “y is greater than x”
fi
```

Shell 脚本中 case 语句的语法?

基础语法如下：

```
case 变量 in
值1)
命令1
命令2
…..
最后命令
!!
值2)
命令1
命令2
……
最后命令
;;
esac
```

Shell 脚本中 for 循环语法？

基础语法如下：

```
for 变量 in 循环列表
do
命令1
命令2
….
最后命令
done
```

Shell 脚本中 while 循环语法？

如同 for 循环，while 循环只要条件成立就重复它的命令块。
不同于 for循环，while 循环会不断迭代，直到它的条件不为真。

基础语法：

```
while [ 条件 ]
do
命令…
done
```

do-while 语句的基本格式？

do-while 语句类似于 while 语句，但检查条件语句之前先执行命令（LCTT 译注：意即至少执行一次。）。下面是用 do-while 语句的语法：

```
do
{
命令
} while (条件)
```

Shell 脚本中 break 命令的作用？

break 命令一个简单的用途是退出执行中的循环。我们可以在 while 和 until 循环中使用 break 命令

跳出循环。

Shell 脚本中 continue 命令的作用？

continue 命令不同于 break 命令，它只跳出当前循环的迭代，而不是整个循环。continue 命令很多时候是很有用的，例如错误发生，但我们依然希望继续执行大循环的时候。

如何使脚本可执行?

使用 chmod 命令来使脚本可执行。例子如下：chmod a+x myscript.sh 。

#!/bin/bash 的作用？

#!/bin/bash 是 Shell 脚本的第一行，称为释伴（shebang）行。

这里 # 符号叫做 hash ，而 ! 叫做 bang。
它的意思是命令通过 /bin/bash 来执行。

如何调试 Shell脚本？

使用 -x' 数（sh -x myscript.sh）可以调试 Shell脚本。
另一个种方法是使用 -nv 参数(sh -nv myscript.sh)。

如何将标准输出和错误输出同时重定向到同一位置?

方法一：2>&1（如# ls /usr/share/doc > out.txt 2>&1 ） 。
方法二：&>（如# ls /usr/share/doc &> out.txt ） 。

在 Shell 脚本中，如何测试文件？

test 命令可以用来测试文件。基础用法如下表格：

Test	用法
-d 文件名	如果文件存在并且是目录，返回true
-e 文件名	如果文件存在，返回true
-f 文件名	如果文件存在并且是普通文件，返回true
-r 文件名	如果文件存在并可读，返回true
-s 文件名	如果文件存在并且不为空，返回true
-w 文件名	如果文件存在并可写，返回true
-x 文件名	如果文件存在并可执行，返回true

在 Shell 脚本如何定义函数呢？

函数是拥有名字的代码块。当我们定义代码块，我们就可以在我们的脚本调用函数名字，该块就会被执行。示例如下所示：

```
$ diskusage () { df -h ; }
译注：下面是我给的shell函数语法，原文没有
[ function ] 函数名 [()]
{
命令;
[return int;]
}
```

如何让 Shell 就脚本得到来自终端的输入?

read 命令可以读取来自终端（使用键盘）的数据。read 命令得到用户的输入并置于你给出的变量中。例子如下：

```
# vi /tmp/test.sh
#!/bin/bash
echo ‘Please enter your name’
read name
echo “My Name is $name”
# ./test.sh
Please enter your name
LinuxTechi
My Name is LinuxTechi
```

如何执行算术运算?

有两种方法来执行算术运算：

1、使用 expr 命令：# expr 5 + 2 。
2、用一个美元符号和方括号（$[表达式]）：test=$[16 + 4] ; test=$[16 + 4] 。

编程题

判断一文件是不是字符设备文件，如果是将其拷贝到 /dev 目录下?

```
#!/bin/bash
read -p "Input file name: " FILENAME
if [ -c "$FILENAME" ];then
    cp $FILENAME /dev
fi
```

添加一个新组为 class1 ，然后添加属于这个组的 30 个用户，用户名的形式为 stdxx ，其中 xx 从 01 到 30 ?

```
#!/bin/bash
groupadd class1
for((i=1;i<31;i++))
do
        if [ $i -le 10 ];then
                useradd -g class1 std0$i
        else
                useradd -g class1 std$i
        fi
done
```

编写 Shell 程序，实现自动删除 50 个账号的功能，账号名为stud1 至 stud50 ?

```
#!/bin/bash
for((i=1;i<51;i++))
```

```
do
                userdel -r stud$i
done
```

写一个 sed 命令，修改 /tmp/input.txt 文件的内容？

要求：

删除所有空行。
一行中，如果包含 "11111"，则在 "11111" 前面插入 "AAA"，在 "11111" 后面插入 "BBB" 。比如：将内容为 0000111112222 的一行改为 0000AAA11111BBB2222 。

```
[root@~]# cat -n /tmp/input.txt
     1  000011111222
     2
     3  000011111222222
     4  11111000000222
     5
     6
     7  111111111111122222222222
     8  2211111111
     9  112222222
    10  1122
    11

# 删除所有空行命令
[root@~]# sed '/^$/d' /tmp/input.txt
000011111222
000011111222222
11111000000222
111111111111122222222222
2211111111
112222222
1122

# 插入指定的字符
[root@~]# sed 's#\(11111\)#AAA\1BBB#g' /tmp/input.txt
0000AAA11111BBB222
0000AAA11111BBB222222
AAA11111BBB000000222
AAA11111BBBAAA11111BBB11122222222222
22AAA11111BBB111
112222222
1122
```

实战

如何选择 Linux 操作系统版本?

一般来讲，桌面用户首选 Ubuntu ；服务器首选 RHEL 或 CentOS ，两者中首选 CentOS 。

根据具体要求：

安全性要求较高，则选择 Debian 或者 FreeBSD 。

需要使用数据库高级服务和电子邮件网络应用的用户可以选择 SUSE 。

想要新技术新功能可以选择 Feddora ，Feddora 是 RHEL 和 CentOS 的一个测试版和预发布版本。

【重点】根据现有状况，绝大多数互联网公司选择 CentOS 。现在比较常用的是 6 系列，现在市场占有大概一半左右。另外的原因是 CentOS 更侧重服务器领域，并且无版权约束。

CentOS 7 系列，也慢慢使用的会比较多了。

如何规划一台 Linux 主机，步骤是怎样？

1、确定机器是做什么用的，比如是做 WEB 、DB、还是游戏服务器。

不同的用途，机器的配置会有所不同。

2、确定好之后，就要定系统需要怎么安装，默认安装哪些系统、分区怎么做。
3、需要优化系统的哪些参数，需要创建哪些用户等等的。

请问当用户反馈网站访问慢，你会如何处理？

有哪些方面的因素会导致网站网站访问慢？

1、服务器出口带宽不够用

- 本身服务器购买的出口带宽比较小。一旦并发量大的话，就会造成分给每个用户的出口带宽就小，访问速度自然就会慢。
- 跨运营商网络导致带宽缩减。例如，公司网站放在电信的网络上，那么客户这边对接是长城宽带或联通，这也可能导致带宽的缩减。

2、服务器负载过大，导致响应不过来

可以从两个方面入手分析：

- 分析系统负载，使用 w 命令或者 uptime 命令查看系统负载。如果负载很高，则使用 top 命令查看 CPU ，MEM 等占用情况，要么是 CPU 繁忙，要么是内存不够。
- 如果这二者都正常，再去使用 sar 命令分析网卡流量，分析是不是遭到了攻击。一旦分析出问题的原因，采取对应的措施解决，如决定要不要杀死一些进程，或者禁止一些访问等。

3、数据库瓶颈

- 如果慢查询比较多。那么就要开发人员或 DBA 协助进行 SQL 语句的优化。
- 如果数据库响应慢，考虑可以加一个数据库缓存，如 Redis 等。然后，也可以搭建 MySQL 主从，一台 MySQL 服务器负责写，其他几台从数据库负责读。

4、网站开发代码没有优化好

- 例如 SQL 语句没有优化，导致数据库读写相当耗时。

针对网站访问慢，怎么去排查？

1、首先要确定是用户端还是服务端的问题。当接到用户反馈访问慢，那边自己立即访问网站看看，如果自己这边访问快，基本断定是用户端问题，就需要耐心跟客户解释，协助客户解决问题。

芳芳：不要上来就看服务端的问题。一定要从源头开始，逐步逐步往下。

2、如果访问也慢，那么可以利用浏览器的调试功能，看看加载那一项数据消耗时间过多，是图片加载慢，还是某些数据加载慢。

3、针对服务器负载情况。查看服务器硬件(网络、CPU、内存)的消耗情况。如果是购买的云主机，比如阿里云，可以登录阿里云平台提供各方面的监控，比如 CPU、内存、带宽的使用情况。

4、如果发现硬件资源消耗都不高，那么就需要通过查日志，比如看看 MySQL慢查询的日志，看看是不是某条 SQL 语句查询慢，导致网站访问慢。

怎么去解决？

1、如果是出口带宽问题，那么久申请加大出口带宽。
2、如果慢查询比较多，那么就要开发人员或 DBA 协助进行 SQL 语句的优化。
3、如果数据库响应慢，考虑可以加一个数据库缓存，如 Redis 等等。然后也可以搭建MySQL主从，一台 MySQL 服务器负责写，其他几台从数据库负责读。
4、申请购买 CDN 服务，加载用户的访问。
5、如果访问还比较慢，那就需要从整体架构上进行优化咯。做到专角色专用，多台服务器提供同一个服务。

如何排查 CPU load 过高问题？

《cpu load 过高问题排查》

Linux 性能调优都有哪几种方法？

1、Disabling daemons (关闭 daemons)。
2、Shutting down the GUI (关闭 GUI)。
3、Changing kernel parameters (改变内核参数)。
4、Kernel parameters (内核参数)。
5、Tuning the processor subsystem (处理器子系统调优)。
6、Tuning the memory subsystem (内存子系统调优)。
7、Tuning the file system (文件系统子系统调优)。
8、Tuning the network subsystem (网络子系统调优)。

666. 彩蛋

Linux 的面试题，主要是以整理为主。但是，整理的过程，比较痛苦。因为，对于开发的 Linux 面试题，真的问的灰常简单。

参考与推荐如下文章：

《Linux 笔试面试常见题目（整理）》

题目比较琐碎，偏运维~

《总结：Linux 体系结构和内核结构区别》

《39 条常见的 Linux 系统简单面试题》

《Linux 面试题哪里有》

《Linux 面试题》

《如何用九条命令在一分钟内检查 Linux 服务器性能？》

《后端程序员必备的 Linux 基础知识》

《中国 100 强互联网企业 Linux 运维面试题合集曝光！》 <!-- 以后在细整理 -->

《Linux 运维面试题》 <!-- 以后在细整理 -->

文章目录

回到首页