



作者 wujunze 2016.02.25 22:48:00

写了41篇文章,回复382人,

# Linux运维常用命令

评论: 7 · 阅读: 5809 · 喜欢: 1

自己买了一台ECS,经常折腾,安装一些好玩儿的软件,工具.配置LNMP环境,跑网站程序,挺有意思的.经常用Linux,所以就搜集整理了一些Linux服务器运维常用命令



## 1.删除0字节文件

```
1 | find -type f -size 0 -exec rm -rf {}
```

## 2.查看进程

按内存从大到小排列

```
1 | ps -e -o "%C : %p : %z : %a"|sort -k5 -nr
```

## 3.按cpu利用率从大到小排列

```
1 | ps -e -o "%C : %p : %z : %a"|sort -nr
```

## 4.打印说cache里的URL

```
1 | grep -r -a jpg /data/cache/* | strings | grep "http:" | awk -F'http:' '{}
```

## 5.查看http的并发请求数及其TCP连接状态:

```
1 | netstat -n | awk '/^tcp/ {++S[$NF]} END {for(a in S) print a, S[a}]'
```

6. sed -i '/Root/s/no/yes/' /etc/ssh/sshd\_config sed在这个文里Root的一行, 匹配Root一行, 将no替换成yes.

## 7.1.如何杀掉mysql进程:

```
1 | ps aux |grep mysql |grep -v grep |awk '{print $2}' |xargs kill -9 (从中了解  
2  
3 | killall -TERM mysqld  
4  
5 | kill -9 cat /usr/local/apache2/logs/httpd.pid 试试查杀进程PID
```

## 8.显示运行3级别开启的服务:

```
1 | ls /etc/rc3.d/S* |cut -c 15- (从中了解到cut的用途, 查看数据)
```

输入关键字搜索

```

1 | cat << EOF
2 | +-----+
3 | |           === Welcome to Tunoff services === |
4 | +-----+
5 | EOF

```

## 10. for 的巧用(如给mysql建软链接)

```
cd /usr/local/mysql/bin for i in * do ln /usr/local/mysql/bin/$i /usr/bin/$i done
```

## 11. 取IP地址:

```

ifconfig eth0 |grep "inet addr:" |awk '{print $2}'|cut -c 6- 或者
ifconfig | grep 'inet addr:' | grep -v '127.0.0.1' | cut -d: -f2 | awk '{ print $1}'

```

## 12.内存的大小:

```
1 | free -m |grep "Mem" | awk '{print $2}'
```

## 13. netstat -an -t | grep ":80" | grep ESTABLISHED | awk '{printf "%s %sn", \$5, \$6}' | sort

## 14.查看Apache的并发请求数及其TCP连接状态:

```
1 | netstat -n | awk '/^tcp/ {++S[$NF]} END {for(a in S) print a, S[a}]'
```

## 15.因为同事要统计一下服务器下面所有的jpg的文件的大小,写了个shell给他来统计.原来用xargs实现,但他一次处理一部分,搞的有多个总和...,下面的命令就能解决啦.

```
1 | find / -name *.jpg -exec wc -c {} ;|awk '{print $1}'|awk '{a+= $1}END{print
```

CPU的数量 (多核算多个CPU, `cat /proc/cpuinfo |grep -c processor`) 越多, 系统负载越低, 每秒能处理的请求数也越多。

## 16 CPU负载 # `cat /proc/loadavg`

检查前三个输出值是否超过了系统逻辑CPU的4倍。

## 18 CPU负载 #`mpstat 1 1`

检查%idle是否过低(比如小于5%)

## 19 内存空间 # `free`

检查free值是否过低 也可以用 # `cat /proc/meminfo`

## 20 swap空间 # `free`

检查swap used值是否过高 如果swap used值过高, 进一步检查swap动作是否频繁:

# vmstat 1 5

观察si和so值是否较大

## 21 磁盘空间 # `df -h`

检查是否有分区使用率(Use%)过高(比如超过90%) 如发现某个分区空间接近用

```
1 | du -cks ^ | sort -rn | head -n 10
```

## 22 磁盘I/O负载 # `iostat -x 1 2`

检查I/O使用率(%util)是否超过100%

## 23 网络负载 # `sar -n DEV`

检查网络流量(rxbyt/s, txbyt/s)是否过高

## 24 网络错误 # `netstat -i`

检查是否有网络错误(drop fifo colls carrier) 也可以用命令: # `cat /proc/net/dev`

## 25 网络连接数目 # `netstat -an | grep -E "(tcp)" | cut -c 68- | sort | uniq -c | sort -n`

## 26 进程总数 # `ps aux | wc -l`

检查进程个数是否正常 (比如超过250)

## 27 可运行进程数目 # `vmstat 1 5`

列给出的是可运行进程的数目, 检查其是否超过系统逻辑CPU的4倍

## 28 进程 # `top -id 1`

观察是否有异常进程出现

## 29 网络状态 检查DNS, 网关等是否可以正常连通

## 30 用户 # `who | wc -l`

检查登录用户是否过多 (比如超过50个) 也可以用命令: # `uptime`

## 31 系统日志 # `cat /var/log/rflogview/*errors`

检查是否有异常错误记录 也可以搜寻一些异常关键字, 例如:

```
1 | # grep -i error /var/log/messages
2 | # grep -i fail /var/log/messages
```

## 32 核心日志 # `dmesg`

检查是否有异常错误记录

## 33 系统时间 # `date`

检查系统时间是否正确

## 34 打开文件数目 # `lsof | wc -l`

检查打开文件总数是否过多

35 日志 # `logwatch -print` 配置/etc/log.d/logwatch.conf, 将 `Mailto` 设置为自己的email 地址, 启动mail服务 (sendmail或者postfix), 这样就可以每天收到日志报告了。

缺省logwatch只报告昨天的日志, 可以用# `logwatch -print -range all` 获得所有的[日志分析](#)结果。

可以用# `logwatch -print -detail high` 获得更具体的日志分析结果(而不仅仅是出错日志)。

## 36.杀掉80端口相关的进程

```
1 | lsof -i :80|grep -v "PID"|awk '{print "kill -9",$2}'|sh
```

## 37.清除僵死进程。

```
1 | # tcpdump -c 10000 -i eth0 -n dst port 80 > /root/pkts
```

39.然后检查IP的重复数 并从小到大**排序** 注意“-t +0”中间是两个空格

```
1 | # less pkts | awk {'printf $3"n"'} | cut -d. -f 1-4 | sort | uniq -c | awk
```

40.查看有多少个活动的**php-cgi**进程

```
1 | netstat -anp | grep php-cgi | grep ^tcp | wc -l
2 |
3 | chkconfig -list | awk '{if ($5=="3:on") print $1}'
```

41.kudzu查看网卡型号

```
1 | kudzu -probe -class=network
```

匹配中文字符的**正则表达式**: `[u4e00-u9fa5]`

评注: 匹配中文还真是个头疼的事, 有了这个表达式就好办了

匹配双字节字符(包括汉字在内): `[\x00-\xff]`

评注: 可以用来计算字符串的长度 (一个双字节字符长度计2, ASCII字符计1)

匹配空白行的正则表达式: `ns*r`

评注: 可以用来删除空白行

匹配HTML标记的正则表达式: `<(S*?)[^>]*>.*?</1>|<.*? />`

评注: 网上流传的版本太糟糕, 上面这个也仅仅能匹配部分, 对于复杂的嵌套标记依旧无能为力

匹配首尾空白字符的正则表达式: `^s*|s*$`

评注: 可以用来删除行首行尾的空白字符(包括空格、制表符、换页符等等), 非常有用的表达式

匹配Email地址的正则表达式: `w+([-+.]w+)*@w+([-.]w+)*.w+([-.]w+)*`

评注: 表单验证时很实用

匹配网址URL的正则表达式: `[a-zA-Z]+://[^\s]*`

评注: 网上流传的版本功能很有限, 上面这个基本可以满足需求

匹配帐号是否合法(字母开头, 允许5-16字节, 允许字母数字下划线): `^[a-zA-Z][a-zA-Z0-9_]{4,15}$`

评注: 表单验证时很实用

匹配国内电话号码: `d{3}-d{8}|d{4}-d{7}`

评注: 匹配形式如 0511-4405222 或 021-87888822

匹配腾讯QQ号: `[1-9][0-9]{4,}`

评注: 腾讯QQ号从10000开始

匹配中国邮政编码: `[1-9]d{5}(?!d)`

评注: 中国邮政编码为6位数字

匹配身份证: `d{15}|d{18}`

评注: 中国的身份证为15位或18位

评注：提取ip地址时有用

匹配特定数字：

```
1  ^[1-9]d*$           //匹配正整数
2  ^-[1-9]d*$          //匹配负整数
3  ^-[1-9]d*$          //匹配整数
4  ^[1-9]d*|0$         //匹配非负整数（正整数 + 0）
5  ^-[1-9]d*|0$        //匹配非正整数（负整数 + 0）
6  ^[1-9]d*.d*|0.d*[1-9]d*$ //匹配正浮点数
7  ^-([1-9]d*.d*|0.d*[1-9]d*)$ //匹配负浮点数
8  ^-?([1-9]d*.d*|0.d*[1-9]d*|0?.0+|0)$ //匹配浮点数
9  ^[1-9]d*.d*|0.d*[1-9]d*|0?.0+|0$ //匹配非负浮点数（正浮点数 + 0）
10 ^(-([1-9]d*.d*|0.d*[1-9]d*))|0?.0+|0$ //匹配非正浮点数（负浮点数 + 0）
```

评注：处理大量数据时有用，具体应用时注意修正

匹配特定字符串：

```
1  ^[A-Za-z]+$        //匹配由26个英文字母组成的字符串
2  ^[A-Z]+$           //匹配由26个英文字母的大写组成的字符串
3  ^[a-z]+$           //匹配由26个英文字母的小写组成的字符串
4  ^[A-Za-z0-9]+$     //匹配由数字和26个英文字母组成的字符串
5  ^w+$              //匹配由数字、26个英文字母或者下划线组成的字符串
```

评注：最基本也是最常用的一些表达式

赞 | 1

赏

标签： linux运维

7 条评论

添加新评论

- 
-  刘瑞斌  March 11th, 2016 at 06:57 pm 回复
- 好
- 
-  卧龙大熊猫 博主  March 11th, 2016 at 09:36 pm 回复
- [给力]
- 
-  Ghost  March 16th, 2016 at 04:37 pm 回复
- 收藏
- 
-  wujunze 博主  April 9th, 2017 at 01:29 pm 回复
- 😊
- 
-  李力  March 31st, 2016 at 01:11 pm 回复
- 学习学习
- 
-  卧龙大熊猫 博主  September 18th, 2016 at 05:03 pm
- 输入关键字搜索