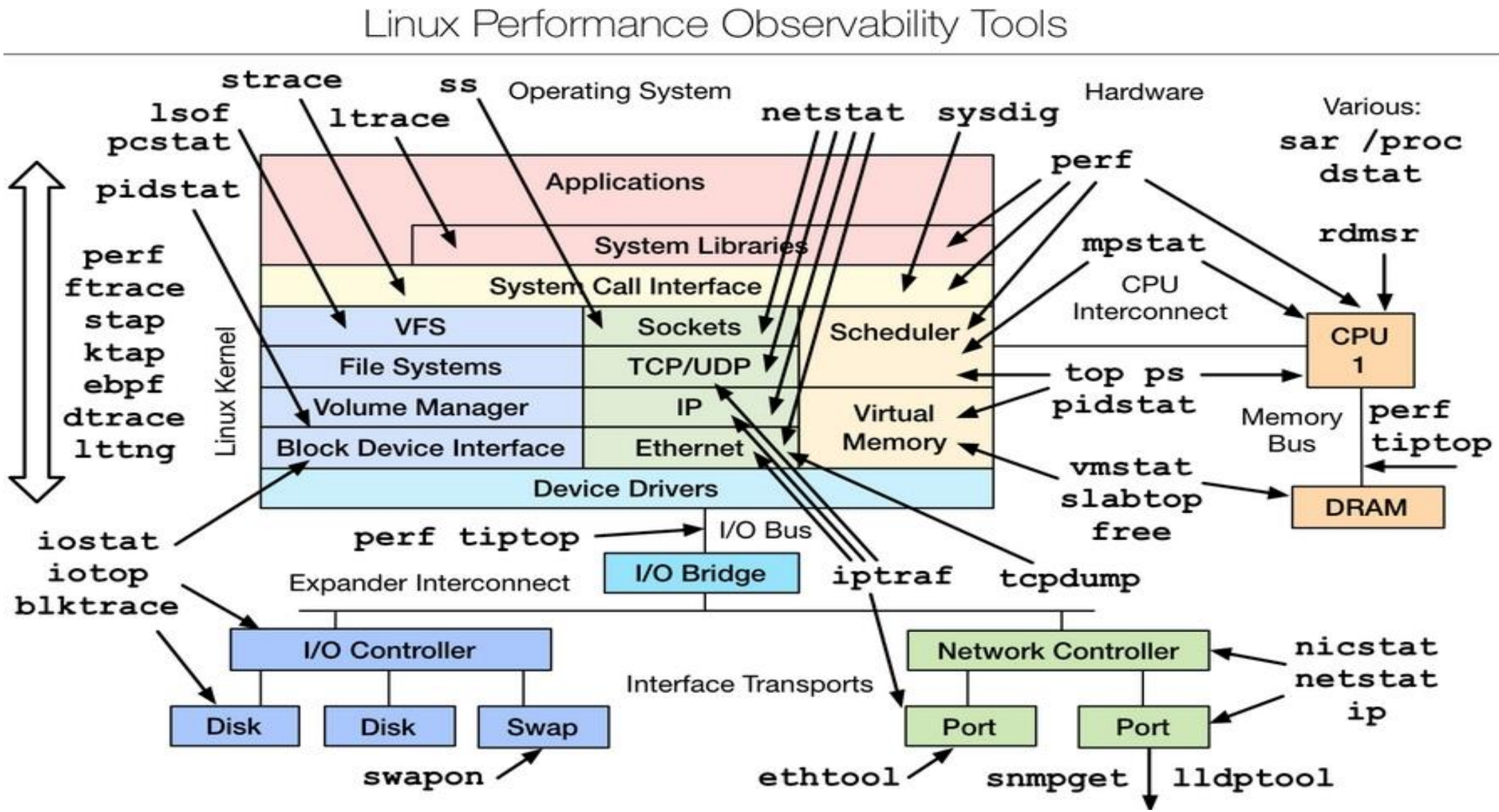
Linux系统问题排查&上云交流

文昌
阿里云售后
2015-6

Linux系统排查工具



Linux系统稳定性&性能常见问题

CPU问题

Cpu的使用率为什么这么高？
究竟谁占用了我的CPU？
怎么下手去优化？

内存问题

内存的使用率为什么这么高？
究竟谁占用了我的内存
怎么下手去优化？

网络问题

网络延时大？网络丢包？网络不通？
如何定位哪里出了问题？
是否有优化空间？

磁盘问题

磁盘IO情况是多少？
谁在占用我们的磁盘IO资源？

Linux常见问题之CPU篇

常见用户问题

CPU使用率	为什么Linux的负载很高？
CPU去哪啦	机器的CPU都用到哪里啦？
CPU使用优化	我们的CPU使用率很高，有没有什么优化建议？

Linux CPU常见问题排查思路

CPU问题排查思路

CPU使用率	Cpu的使用分布：user/sys/nice/iowait/hirq/sirq/steal time?
CPU去哪啦	用户态进出那个的内存使用以及top 5， sys 内核态哪些程序占用了CPU？
CPU优化	用户态top 5的使用cpu情况？中断、切换、进程占用？

CPU型号查看-dmidecode

```
[root@iZ28lzm2ehvZ ~]# dmidecode -t processor
# dmidecode 2.12
SMBIOS 2.4 present.

Handle 0x0401, DMI type 4, 26 bytes
Processor Information
        Socket Designation: CPU 1
        Type: Central Processor
        Family: Other
        Manufacturer: Intel
        ID: D7 06 02 00 FF FB 89 17
        Version: Not Specified
        Voltage: Unknown
        External Clock: Unknown
        Max Speed: 2300 MHz
        Current Speed: 2300 MHz
        Status: Populated, Enabled
        Upgrade: Other
```

Linux CPU负载情况——uptime/w

```
[root@iZ94cjwg86gZ ~]# uptime
 20:24:30 up 20 days,  5:45,  1 user,  load average: 0.00, 0.01, 0.05
[root@iZ94cjwg86gZ ~]# w
 20:24:32 up 20 days,  5:45,  1 user,  load average: 0.00, 0.01, 0.05
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
root     pts/0    42.120.74.89     20:16    0.00s  0.00s  0.00s w
```

Linux CPU负载情况——uptime/w

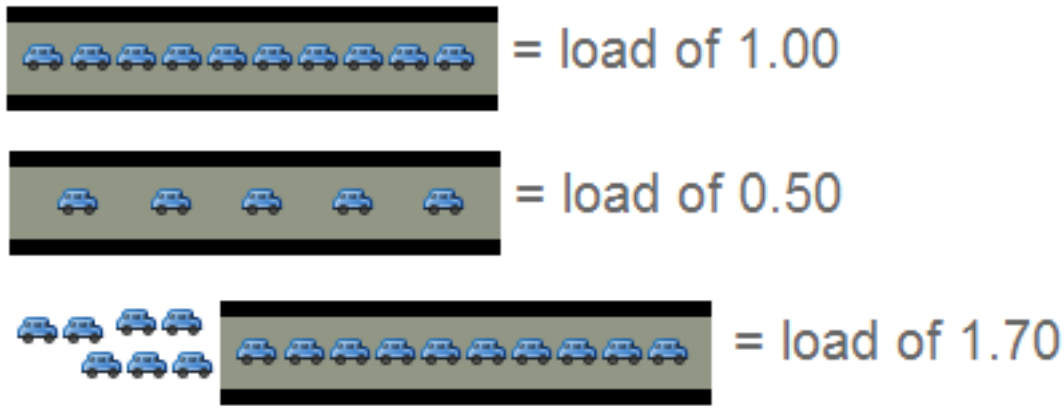
```
[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ uptime
 11:46:34 up 189 days, 27 min, 15 users,  load average: 3.67, 4.10, 4.76

[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ w
 11:47:06 up 189 days, 27 min, 15 users,  load average: 2.72, 3.82, 4.64
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
admin    pts/6    10.143.34.156    Tue19   37:19m  0.08s  0.08s -bash
admin    pts/14   10.143.34.156    Wed22   11:20m  0.13s  0.09s /bin/bash /usr/
admin    pts/16   10.143.34.156    Tue17   22:44m  3.54s  0.12s -bash
admin    pts/17   10.143.34.156    Tue17   18:34m  0.09s  0.09s -bash
admin    pts/19   10.143.34.156    09:55    1:50m  0.12s  0.10s /bin/bash /usr/
admin    pts/28   10.143.34.156    09:56   22:52   0.02s  0.02s -bash
admin    pts/30   10.143.34.156    00:28   10:57m  0.13s  0.00s sshd: admin [pr
admin    pts/18   10.143.34.156    11:37    9:16   0.04s  0.04s -bash
admin    pts/33   10.143.34.156    22Dec14 30days  0.15s  0.09s /bin/bas
zhiyan.c pts/34   10.143.0.37      10:08    8:15   1.27s  0.00s sshd: zh
admin    pts/35   10.143.34.156    10:25    1:21m  0.02s  0.02s -bash
admin    pts/37   10.143.34.156    07Jan15  3days  0.09s  0.09s -bash
admin    pts/43   10.143.34.156    10:32    1:10m  0.19s  0.19s -bash
jinli.zj pts/45   10.143.0.37      11:45    1:27   0.02s  0.02s -bash
admin    pts/46   10.143.34.156    11:46    0.00s  0.02s  0.00s w
```

CPU负载建议:Load/cores<=0.7

需要进行调查法则”：如果长期你的系统负载在 0.70 上下，那么你需要在事情变得更糟糕之前，花些时间了解其原因。

“现在就要修复法则”：1.00。 如果你的服务器系统负载长期徘徊于 1.00，那么就应该马上解决这个问题。否则，你将半夜接到你上司的电话，这可不是件令人愉快的事情。

“凌晨三点半锻炼身体法则”：5.00。 如果你的服务器负载超过了 5.00 这个数字，那么你将失去你的睡眠，还得在会议中说明这情况发生的原因，总之千万不要让它发生。



CPU去哪儿啦——TOP查看进程使用CPU大户

```
top - 14:41:35 up 189 days,  3:22, 12 users,  load average: 2.88, 4.48, 4.85
Tasks: 440 total,   1 running, 438 sleeping,   1 stopped,   0 zombie
Cpu(s):  0.7%us,  0.5%sy,  0.0%ni, 95.2%id,  3.5%wa,  0.0%hi,  0.1%si,  0.0%st
Mem:  49449340k total, 14460340k used, 34989000k free,   609852k buffers
Swap:  1052216k total,      400k used,  1051816k free, 11076872k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
19090 root      15   0  540m  16m 2644 S 12.3  0.0 462:42.02 python
 8604 root      15   0  796m  34m 8992 S  2.3  0.1   1612:12 shennongAgentSe
 4444 root      34  19     0    0    0 S  0.7  0.0   4761:39 kipmi0
18173
 5128
 9761
20302
29088
    1
    2
```

```
top - 14:40:32 up 189 days,  3:21, 12 users,  load average: 5.22, 5.18, 5.08
Tasks: 441 total,   2 running, 438 sleeping,   1 stopped,   0 zombie
Cpu0  : 17.6%us,  2.7%sy,  0.0%ni, 79.4%id,  0.0%wa,  0.0%hi,  0.3%si,  0.0%st
Cpu1  :  3.0%us,  1.0%sy,  0.0%ni, 96.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  0.7%us,  1.0%sy,  0.0%ni, 98.3%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  1.0%us,  0.3%sy,  0.0%ni, 98.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu4  : 13.6%us,  1.7%sy,  0.0%ni, 84.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu5  :  3.3%us,  2.0%sy,  0.0%ni, 94.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu6  :  5.0%us,  1.3%sy,  0.0%ni, 93.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu7  :  2.0%us,  1.3%sy,  0.0%ni, 96.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu8  :  6.3%us,  0.7%sy,  0.0%ni, 93.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu9  : 10.6%us,  5.0%sy,  0.0%ni, 84.4%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu10 :  6.6%us,  1.7%sy,  0.0%ni, 91.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu11 :  0.3%us,  0.7%sy,  0.0%ni, 99.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu12 :  7.6%us,  0.7%sy,  0.0%ni, 91.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu13 :  8.6%us,  3.0%sy,  0.0%ni, 88.4%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu14 :  3.6%us,  1.3%sy,  0.0%ni, 94.7%id,  0.3%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu15 :  2.7%us,  1.3%sy,  0.0%ni, 96.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:  49449340k total, 14473212k used, 34976128k free,   609852k buffers
Swap:  1052216k total,      400k used,  1051816k free, 11075540k cached
```

内存指标说明：

Shift+p按照cpu使用率排序

1：则显示所有CPU的使用率信息

进程状态。 D=不可中断的睡眠状态 R=运行 S=睡眠 T=跟踪/停止 Z=僵尸进程

阿里云 aliyun.com

CPU去哪儿啦—PS工具查看CPU大户

```
[root@iZ28lzm2ehvZ ~]# ps -eo rss,pmem,pcpu,vsize,args |  sort -k 1 -r -n
25480  5.0  0.0 181372 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --user=
lib/mysql/mysql.sock
10884  2.1  0.0 275736 /usr/local/aegis/aegis_client/aegis_00_65/AliYunDun
 9352  1.8  0.0 286532 /usr/local/aegis/alihids/AliHids
 5120  1.0  0.0  19428 ntpd -u ntp:ntp -p /var/run/ntpd.pid -g
 3324  0.6  0.0  88080 sshd: root@pts/0
 2164  0.4  0.0  81468 /usr/local/aegis/aegis_update/AliYunDunUpdate
 1828  0.3  0.0  77464 SCREEN -S anders.zhangw
 1736  0.3  0.0  77208 SCREEN -S anders.zhangw
 1604  0.3  0.0  66084 /bin/bash
 1604  0.3  0.0  66084 -bash
 1592  0.3  0.0  66084 /bin/bash
 1584  0.3  0.0  66084 /bin/bash
 1576  0.3  0.0  66084 /bin/bash
 1572  0.3  0.0  66084 /bin/bash
 1088  0.2  0.0  74828 crond
  952  0.1  0.0 129572 /usr/sbin/nscd
  844  0.1  0.0  63516 ps -eo rss,pmem,pcpu,vsize,args
  768  0.1  0.0  62648 /usr/sbin/sshd
  644  0.1  0.0  66084 -bash
  576  0.1  0.0  10128 syslogd -m 0
  520  0.1  0.0   3808 /sbin/mingetty tty2
  520  0.1  0.0   3808 /sbin/mingetty tty1
  388  0.0  0.0  12636 /sbin/udevd -d
  288  0.0  0.0   3824 klogd -x
  204  0.0  0.0  63856 /bin/sh /usr/bin/mysqld_safe --datadir=/var/lib/mysql --socket=/var
  148  0.0  0.0  10368 init [3]
   68  0.0  0.0  31628 /usr/sbin/gshelld
  RSS %MEM %CPU    VSZ COMMAND
```

CPU查看明细——VMStat工具使用

```
$ vmstat -n 3
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 2  0    400 34948972 609948 11106840    0    0    16   266    0    0 14  5 73  8  0
 2  0    400 34952036 609948 11106852    0    0     0   716 4087 10302  6  2 92  0  0
 0  0    400 34956912 609948 11106856    0    0     0    41 3476 11248  7  2 91  0  0
 1  0    400 34957224 609948 11106860    0    0     0   708 1139 8720  1  1 99  0  0
 1  0    400 34959396 609948 11106864    0    0     0    72 1092 8956  0  1 99  0  0
 2  0    400 34958676 609948 11106904    0    0     0   411 2040 47304  2  1 97  0  0
 0  0    400 34953336 609948 11107352    0    0     0   371 2505 73654  3  2 95  0  0
 0  0    400 34952944 609948 11107372    0    0     0    45 2873 10295  3  1 96  0  0
 2  0    400 34953340 609948 11107384    0    0     0   652 4694 11909  2  3 95  0  0
 1  0    400 34954192 609948 11107448    0    0     0     0 3219 13841  5  6 89  0  0
 2  0    400 34953208 609948 11107472    0    0     0   503 1242 13963  4  4 93  0  0
```

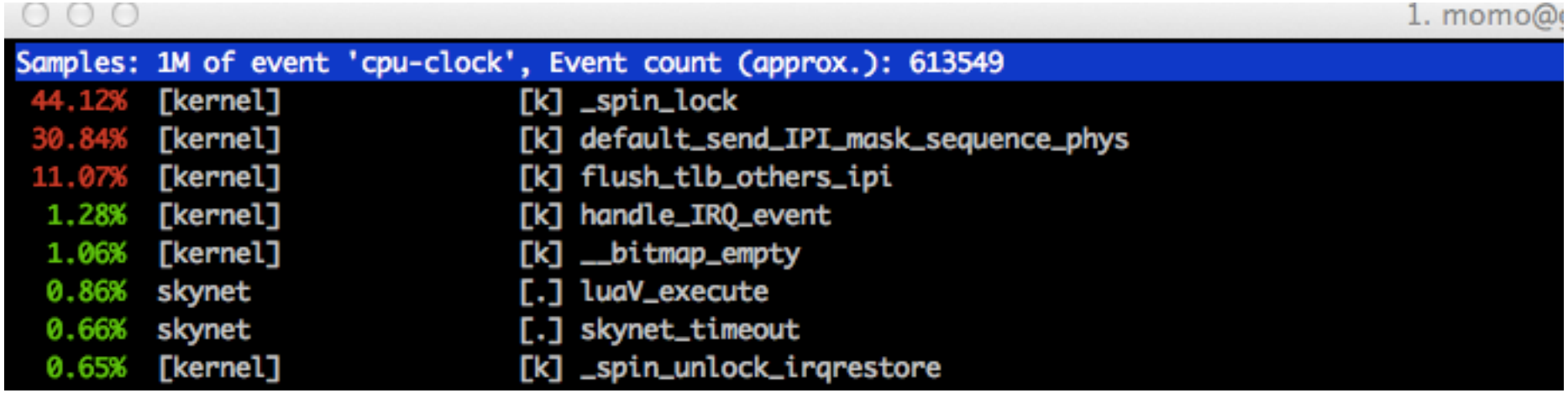
sar工具使用——查看历史明细

```
$ sar -u ALL -f /var/log/sa/sa14|head -10
Linux
$ sar  -P ALL -u ALL -f /var/log/sa/sa14|head -40
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/14/2015      _x86_64_        (16 CPU)
12:00
12:01
12:02
12:03
12:04
12:05
12:06
12:07

12:00:01 AM     CPU      %usr     %nice      %sys   %iowait    %steal      %irq     %soft    %guest     %idle
12:01:01 AM     all     34.30      0.00      7.15      0.05      0.00      0.01      0.47      0.00     58.00
12:01:01 AM       0     39.9
12:01:01 AM       1     32.5
12:01:01 AM       2     32.6
12:01:01 AM       3     31.9
12:01:01 AM       4     39.7
12:01:01 AM       5     32.3
12:01:01 AM       6     31.9
12:01:01 AM       7     31.8
12:01:01 AM       8     34.1
12:01:01 AM       9     37.3
12:01:01 AM      10     34.3
12:01:01 AM      11     32.6
12:01:01 AM      12     34.3
12:01:01 AM      13     35.6
12:01:01 AM      14     34.5
12:01:01 AM      15     33.6
12:02:01 AM     all      6.2
$ sar -q  -f /var/log/sa/sa14 |head -20
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/14/2015      _x86_64_        (16 CPU)

12:00:01 AM   runq-sz  plist-sz   ldavg-1   ldavg-5  ldavg-15   blocked
12:01:01 AM        14      1355     11.49      7.21      6.52         0
12:02:01 AM        14       897      5.36      6.28      6.25         0
12:03:01 AM        27       974      2.67      5.37      5.94         0
12:04:01 AM        13       903      5.73      6.15      6.19         0
12:05:01 AM        32      1388      2.95      5.28      5.89         0
12:06:01 AM        16       938      4.74      5.70      6.01         1
12:07:01 AM         8       893      7.68      6.93      6.45         0
12:08:01 AM         8       892      4.28      6.11      6.20         0
12:09:01 AM        11       867      3.55      5.58      6.01         0
12:10:01 AM        26       967      6.85      6.77      6.43         0
12:11:01 AM        17       972      7.75      7.49      6.73         0
12:12:01 AM         8       882      3.66      6.40      6.40         0
12:13:01 AM        12       913      5.18      6.67      6.51         0
12:14:01 AM         7      1259      2.96      5.78      6.22         0
12:15:01 AM        18       978      1.74      4.93      5.90         0
12:16:01 AM        12       923     13.53      8.89      7.27         0
12:17:01 AM         8       927      5.46      7.43      6.87         0
```

CPU详情查看-mpstat

```
$ mpstat -P ALL 2 10
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/22/2015      _x86_64_        (16 CPU)

03:28:04 PM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
03:28:06 PM  all   11.96    0.00    7.59    0.00    0.00    0.37    0.00    0.00   80.07
03:28:06 PM    0    4.52    0.00    5.03    0.00    0.00    5.03    0.00    0.00   85.43
03:28:06 PM    1   15.42    0.00    7.46    0.00    0.00    0.00    0.00    0.00   77.11
03:28:06 PM    2   13.43    0.00    7.46    0.00    0.00    0.00    0.00    0.00   79.10
03:28:06 PM    3    6.00    0.00    9.00    0.00    0.00    0.00    0.00    0.00   85.00
03:28:06 PM    4   12.44    0.00    7.46    0.00    0.00    1.00    0.00    0.00   79.10
03:28:06 PM    5   14.50    0.00    3.50    0.00    0.00    0.00    0.00    0.00   82.00
03:28:06 PM    6   13.00    0.00    2.50    0.00    0.00    0.00    0.00    0.00   84.50
03:28:06 PM    7    2.01    0.00    2.01    0.00    0.00    0.00    0.00    0.00   95.98
03:28:06 PM    8   10.05    0.00   11.56    0.00    0.00    0.00    0.00    0.00   78.39
03:28:06 PM    9   17.41    0.00   10.95    0.00    0.00    0.00    0.00    0.00   71.64
03:28:06 PM   10   12.00    0.00   12.50    0.00    0.00    0.00    0.00    0.00   75.50
03:28:06 PM   11   14.00    0.00   14.00    0.00    0.00    0.00    0.00    0.00   72.00
03:28:06 PM   12   18.50    0.00    3.50    0.00    0.00    0.00    0.00    0.00   78.00
03:28:06 PM   13   14.50    0.00    9.50    0.00    0.00    0.00    0.00    0.00   76.00
03:28:06 PM   14   20.10    0.00    8.54    0.00    0.00    0.00    0.00    0.00   71.36
03:28:06 PM   15    4.00    0.00    6.00    0.00    0.00    0.00    0.00    0.00   90.00
```

CPU详情查看-perf



CPU负载高调优常见方法

小技巧

1. 如果确实是用户进程使用率高，则弹性扩容；
2. 如果是IO Wait高，则IO 性能进行调整；
3. 如果是sys高，则检查in/cs的比例；
4. 如果是in高， 则看看是否可以使用cpu in在哪里？网络、IO？
5. 如果是某个CPU高，则可以使用打散技术；
6. 如果莫个整体CPU高导致某个进程慢，则可以绑定CPU；
7. 如果上下文切换多，则可以绑定CPU；
8. 清理僵尸进程；

Linux常见问题之内存篇

常见用户问题

内存使用率	为什么机器使用的内存很高？
内存去哪啦	机器的内存都用到哪里啦？
内存优化	我们的内存使用率很高，有没有什么优化建议？

Linux 内存常见问题排查思路

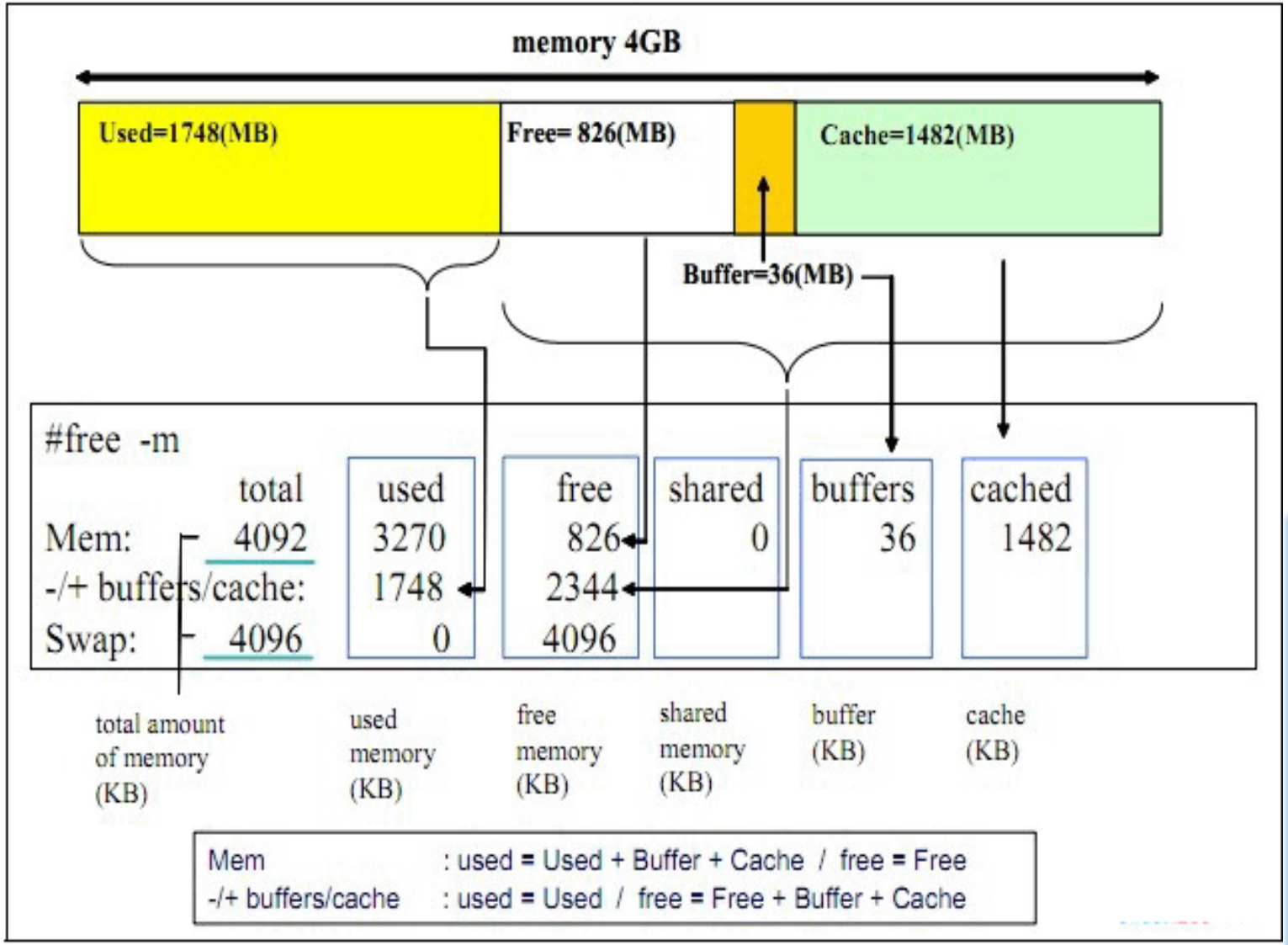
内存问题排查思路

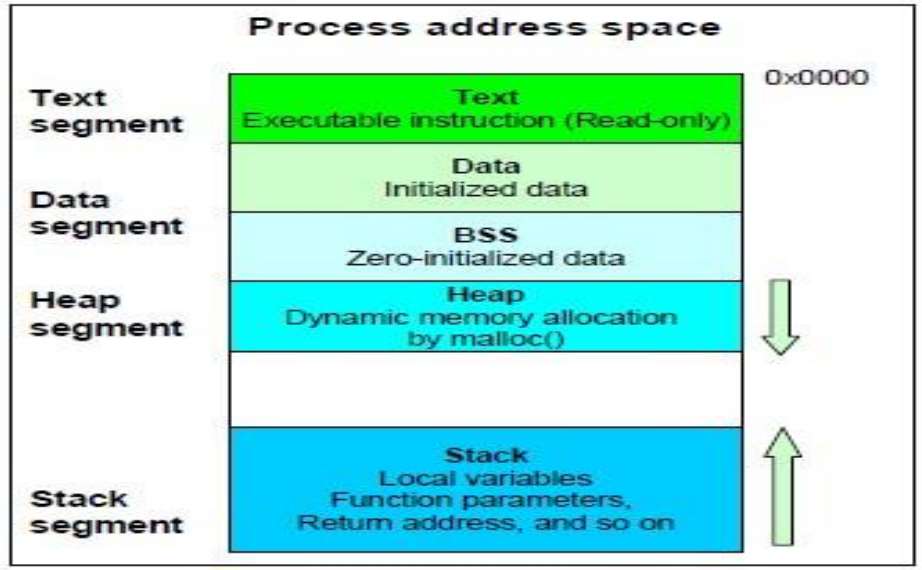
内存使用率	物理内存、虚拟内存、常驻内存、cache/buffer、swap、slab分布
内存去哪啦	用户态进出那个的内存使用以及top 5， 内核态的slab哪些对象占用了内存？
内存优化	用户态top 5的使用内存情况？代码段、数据段、堆、栈？内核态slab的哪些对象可以优化？

内存型号查看-dmidecode

```
[root@iZ28lzm2ehvZ ~]# dmidecode -t memory
# dmidecode 2.12
SMBIOS 2.4 present.

Handle 0x1000, DMI type 16, 15 bytes
Physical Memory Array
        Location: Other
        Use: System Memory
        Error Correction Type: Multi-bit ECC
        Maximum Capacity: 512 MB
        Error Information Handle: Not Provided
        Number Of Devices: 1

Handle 0x1100, DMI type 17, 21 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: 0x0000
        Total Width: 64 bits
        Data Width: 64 bits
        Size: 512 MB
        Form Factor: DIMM
        Set: None
        Locator: DIMM 0
        Bank Locator: Not Specified
        Type: RAM
        Type Detail: None
```

Linux内存使用情况——free





Buffer && Cache && Used

1. 物理内存分为=进程内存空间、Buffer、Cache、Slab、Pagetable
2. buffer : 作为buffer cache的内存，是块设备的读写缓冲区
3. cache: 作为page cache的内存, 文件系统的cache
4. Used = 进程非共享内存（Res-Shared)+共享内存+Slab使用+pagetable

Linux Swap—mkswap/swapon/swapoff

```
#free -m
             total       used       free     shared
buffers      cached
Mem:           493        484          8          0
61         321
-/+ buffers/cache:        101        391
Swap:            0          0          0
```

上述数据说明：VM没有swap分区

```
dd if=/dev/zero of=/swapfile1 bs=1024 count=524288
536870912 bytes (537 MB) copied, 9.77603 seconds, 54.9 MB/s
# chown root:root /swapfile1
# chmod 0600 /swapfile1
# mkswap /swapfile1
Setting up swapspace version 1, size = 536866 kB
# swapon /swapfile1
```

以上方法设置swap分区

```
# free -m
             total       used       free     shared
buffers      cached
Mem:           493        484          9          0
1          418
-/+ buffers/cache:         64        429
Swap:          511          0        511
```

Swap分区说明

1. 内存分为=物理内存+Swap分区（虚拟内存）
2. Swap是硬盘空间，不是真正的物理内存
3. Swap分区的使用场景：物理内存不够用时，内存交换至swap分区；
4. 当可用内存不足时，系统有两个选择：一个是通过SWAP来释放内存，另一个是删除Cache中的Page来释放内存。一个很常见的例子是：当拷贝大文件的时候，时常会发生SWAP现象。这是因为拷贝文件的时候，系统会把文件内容在Cache中按Page来缓存，此时一旦可用内存不足，系统便会倾向于通过SWAP来释放内存
5. Swap分区：OOM与程序性能的平衡

内存去哪儿啦——TOP查看进程使用内存大户

```
top - 16:19:32 up 49 days, 23:18,  1 user,  load average: 0.00, 0.03, 0.02
Tasks:  53 total,   2 running,  51 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.0%us,  0.0%sy,  0.0%ni,100.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:    505284k total,   496532k used,     8752k free,     1388k buffers
Swap:   524280k total,        0k used,   524280k free,   429268k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
31349 mysql     15   0  177m  24m 3904 S  0.0  5.0   0:05.10 mysqld
 2887 root      15   0  269m  10m 7828 S  0.0  2.2   0:58.87 AliYunDun
 2903 root      15   0  279m 9352 7020 S  0.0  1.9   9:13.63 AliHids
 1307 ntp       15   0 19428 5120 4012 S  0.0  1.0   0:00.85 ntpd
10485 root      15   0 88080 3324 2592 R  0.0  0.7   0:00.04 sshd
 4316 root      15   0 81468 2164 1784 S  0.0  0.4   0:14.43 AliYunDunUpdate
 3480 root      15   0 77464 1828  596 S  0.0  0.4   0:00.09 screen
 7527 root      15   0 77208 1736  748 S  0.0  0.3   0:00.15 screen
 3493 root      15   0 66084 1604 1200 S  0.0  0.3   0:00.03 bash
10489 root      16   0 66084 1600 1204 S  0.0  0.3   0:00.05 bash
 3481 root      16   0 66084 1592 1200 S  0.0  0.3   0:00.10 bash
 3505 root      16   0 66084 1584 1196 S  0.0  0.3   0:00.03 bash
 7528 root      15   0 66084 1576 1192 S  0.0  0.3   0:00.02 bash
 7540 root      15   0 66084 1572 1188 S  0.0  0.3   0:00.02 bash
 1331 root      15   0 74828 1088  508 S  0.0  0.2   0:00.46 crond
10539 root      15   0 12628 1060  836 R  0.0  0.2   0:00.06 top
 1274 nscd      15   0  126m  952  556 S  0.0  0.2   0:09.24 nscd
 1294 root      15   0 62648  768  204 S  0.0  0.2   0:00.31 sshd
 1262 root      15   0 10128  576  412 S  0.0  0.1   0:01.30 syslogd
 1345 root      17   0  3808  520  436 S  0.0  0.1   0:00.01 mingetty
 1346 root      18   0  3808  520  436 S  0.0  0.1   0:00.00 mingetty
  365 root      16  -4 12636  388    0 S  0.0  0.1   0:00.18 udevd
 1265 root      15   0  3824  288  200 S  0.0  0.1   0:00.00 klogd
31267 root      25   0 63856  204    4 S  0.0  0.0   0:00.00 mysqld_safe
    1 root      15   0 10368  148   52 S  0.0  0.0   0:02.04 init
 1336 root      25   0 31628   68    0 S  0.0  0.0   0:00.00 gshelld
```

内存指标说明

1. Virt： 虚拟内存
2. Res： 物理内存（常驻内存）
3. Shr： 共享内存
4. 实际使用内存是：RES。
Free中看到的used，不是virt的和，而是res的和。
5. shift+m, 按照内存排序

内存去哪儿啦—PS工具查看内存大户

```
[root@iZ28lzm2ehvZ ~]# ps -eo rss,pmem,pcpu,vsize,args |  sort -k 1 -r -n
25480  5.0  0.0 181372 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --user=
lib/mysql/mysql.sock
10884  2.1  0.0 275736 /usr/local/aegis/aegis_client/aegis_00_65/AliYunDun
 9352  1.8  0.0 286532 /usr/local/aegis/alihids/AliHids
 5120  1.0  0.0  19428 ntpd -u ntp:ntp -p /var/run/ntpd.pid -g
 3324  0.6  0.0  88080 sshd: root@pts/0
 2164  0.4  0.0  81468 /usr/local/aegis/aegis_update/AliYunDunUpdate
 1828  0.3  0.0  77464 SCREEN -S anders.zhangw
 1736  0.3  0.0  77208 SCREEN -S anders.zhangw
 1604  0.3  0.0  66084 /bin/bash
 1604  0.3  0.0  66084 -bash
 1592  0.3  0.0  66084 /bin/bash
 1584  0.3  0.0  66084 /bin/bash
 1576  0.3  0.0  66084 /bin/bash
 1572  0.3  0.0  66084 /bin/bash
 1088  0.2  0.0  74828 crond
  952  0.1  0.0 129572 /usr/sbin/nscd
  844  0.1  0.0  63516 ps -eo rss,pmem,pcpu,vsize,args
  768  0.1  0.0  62648 /usr/sbin/sshd
  644  0.1  0.0  66084 -bash
  576  0.1  0.0  10128 syslogd -m 0
  520  0.1  0.0   3808 /sbin/mingetty tty2
  520  0.1  0.0   3808 /sbin/mingetty tty1
  388  0.0  0.0  12636 /sbin/udevd -d
  288  0.0  0.0   3824 klogd -x
  204  0.0  0.0  63856 /bin/sh /usr/bin/mysqld_safe --datadir=/var/lib/mysql --socket=/var
  148  0.0  0.0  10368 init [3]
   68  0.0  0.0  31628 /usr/sbin/gshelld
  RSS %MEM %CPU    VSZ COMMAND
```

内存指标说明

1. Virt： 虚拟内存
2. Res： 物理内存（常驻内存）
3. Shr： 共享内存
4. 实际使用内存是：RES。

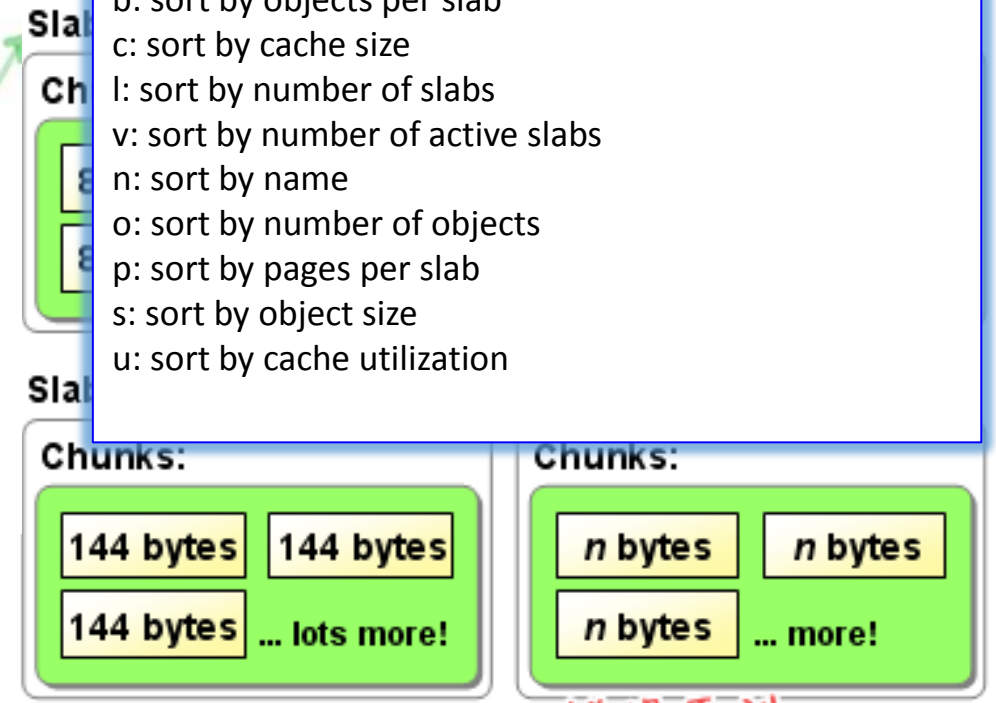
Free中看到的used，不是virt的和，而是res的和。

内存去哪儿啦—Slabtop工具查看内存大户

```
Active / Total Objects (% used)    : 30644 / 59248 (51.7%)
Active / Total Slabs (% used)      : 2937 / 2938 (100.0%)
Active / Total Caches (% used)     : 86 / 133 (64.7%)
Active / Total Size (% used)       : 7393.91K / 11274.10K (65.6%)
Minimum / Average / Maximum Object : 0.02K / 0.19K / 128.00K

  OBJS ACTIVE  USE OBJ SIZE  SLABS OBJ/SLAB CACHE SIZE NAME
  2485   1467  59%    0.52K    355        7      1420K radix_tree_node
  5274   1698  32%    0.21K    293       18      1172K dentry_cache
 10840   3848  35%    0.09K    271       40      1084K buffer_head
  1055    521  49%    0.74K    211        5       844K ext3_inode_cache
  9840   1703  17%    0.08K    205       48       820K selinux_inode_security
   132    132 100%    2.62K    132        1       528K kmem_cache
   124    124 100%    4.00K    124        1       496K size-4096
   488    448  91%    1.00K    122        4       488K size-1024
   228    226  99%    2.00K    114        2       456K size-2048
  6077   1855  30%    0.06K    103       59       412K size-64
```

options:
--delay=n, -d n　每隔n秒刷新信息
--once, -o　只显示一次
--sort=S, -s S　按照S排序，其中S为排序标准
--version, -V　显示版本信息
--help　显示帮助信息

排序标准
a: sort by number of active objects
b: sort by objects per slab
c: sort by cache size
l: sort by number of slabs
v: sort by number of active slabs
n: sort by name
o: sort by number of objects
p: sort by pages per slab
s: sort by object size
u: sort by cache utilization



内存查看明细——VMStat工具使用

```
procs -----------memory---------- ---swap-- -----io----
 r  b   swpd   free   buff  cache   si   so    bi
 0  0 254176  14456   1136 421524    0    0    12
 0  0 254176  14456   1136 421512    0    0     0
 0  0 254176  14456   1136 421512    0    0     0
 0  0 254176  14456   1144 421504    0    0     0    14 1245  205  0  1 99  1  0
 0  0 254176  14456   1144 421512    0    0     0     0 1
 1  2 254176 203044   1880 232408    0    0   312  6384 1
 0  4 254176 103916   2048 326180    0    0   312 44334 2
 1  4 254176   6268   1616 421500    0    0   174 51072 2
 0  4 254176   5780   1720 419336    0    0     2 51102 2
 0  5 254176   5992   1852 419704    0    0    12 51416 2
 0  5 254176   5976   1928 419940    0    0     0 49282 2
 1  6 254176   6384   2000 420340    0    0     4 52028 2410  981  0 10  0 90  0
 0  6 254176   6552   1984 420308    0    0     2 49962 2239 1837  0 10  0 90  0
 0  7 254176   6280   1964 419260    0    0     2 51870 2283 1538  0 10  0 90  0
 0  7 254176   6448   1948 420508    0    0     2 48592 2188 1026  0 10  0 90  0
 0  7 254176   6372   1928 420156    0    0     0 47272 2273 1711  0  9  0 91  0
 0  8 254176   6944    944 420076    0    0     2 48194 2210  941  0 10  0 91  0
 1  8 254176   5580    924 422312    0    0     2 52318 2267 1810  0 10  0 90  0
 0  8 254176   6108    924 420512    0    0     2 50934 2259 1792  0 11  0 89  0
 0  8 254176   5868    940 422132    0    0     2 50572 2275 1204  2 11  0 88  0
 0  8 254176   5528    948 422052    0    0     2 49982 2219 1683  0 10  0 90  0
 0  8 254176   6252   1004 420608    0    0     2 51808 2514 1052  0 11  0 89  0
```

```
[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ vmstat  -a
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu------
 r  b   swpd   free  inact active   si   so    bi    bo   in   cs us sy id wa st
30  0    412 42637896 477228 5788424    0    0    16   269    0    0 14  5 72  8  0
```

```
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+  COMMAND
31349 mysql     15   0  177m   24m 3904 S   0.0   5.0   0:05.16 mysqld
 2887 root      15   0  269m   10m 7832 S   0.0   2.2   1:06.24 AliYunDun
 2903 root      15   0  279m 9352 7020 S   0.0   1.9  10:11.65 AliHids
12848 root      18   0 71384 8800 8708 D   5.3   1.7   0:05.87 dd
 1307 ntp       15   0 19428 5120 4012 S   0.0   1.0   0:01.02 ntpd
12804 root      16   0 88872 3320 2592 S   0.0   0.7   0:00.03 sshd
```

内存情况查看-dstat工具使用

```
[root@iZ28lzm2ehvZ ~]# dstat -gms
Color support is disabled, python-curses is not installed.
---paging-- ------memory-usage----- ----swap---
  in   out | used  buff  cach  free| used  free
  57B   57B|  61M 1972k  276M  154M|   0   512M
   0     0 |  61M 1972k  276M  154M|   0   512M
   0     0 |  61M 1972k  276M  154M|   0   512M
   0     0 |  61M 1972k  276M  154M|   0   512M
   0     0 |  61M 1972k  276M  154M|   0   512M
   0     0 |  61M 1980k  276M  154M|   0   512M
   0     0 |  61M 1980k  276M  154M|   0   512M
   0     0 |  61M 1980k  276M  154M|   0   512M
   0     0 |  61M 1980k  276M  154M|   0   512M
```

内存指标说明

1. Paging in : Paging swap in
2. Paging out: Paging swap out

内存情况查看-pmap工具使用

```
[root@iZ28lzm2ehvZ ~]# pmap -d 1
1:   init [3]
Address           Kbytes Mode  Offset           Device    Mapping
0000000000400000      36 r-x-- 0000000000000000 003:00001 init
0000000000609000       4 rw--- 0000000000009000 003:00001 init
000000000c634000     132 rw--- 000000000c634000 000:00000   [ anon ]
00000033cba00000     112 r-x-- 0000000000000000 003:00001 ld-2.5.so
00000033cbc1c000       4 r---- 000000000001c000 003:00001 ld-2.5.so
00000033cbc1d000       4 rw--- 000000000001d000 003:00001 ld-2.5.so
00000033cbe00000     236 r-x-- 0000000000000000 003:00001 libsepol.so.1
00000033cbe3b000    2048 ----- 000000000003b000 003:00001 libsepol.so.1
00000033cc03b000       4 rw--- 000000000003b000 003:00001 libsepol.so.1
00000033cc03c000      40 rw--- 00000033cc03c000 000:00000   [ anon ]
00000033cc200000    1336 r-x-- 0000000000000000 003:00001 libc-2.5.so
00000033cc34e000    2048 ----- 000000000014e000 003:00001 libc-2.5.so
00000033cc54e000      16 r---- 000000000014e000 003:00001 libc-2.5.so
00000033cc552000       4 rw--- 0000000000152000 003:00001 libc-2.5.so
00000033cc553000      20 rw--- 00000033cc553000 000:00000   [ anon ]
00000033cc600000       8 r-x-- 0000000000000000 003:00001 libdl-2.5.so
00000033cc602000    2048 ----- 0000000000002000 003:00001 libdl-2.5.so
00000033cc802000       4 r---- 0000000000002000 003:00001 libdl-2.5.so
00000033cc803000       4 rw--- 0000000000003000 003:00001 libdl-2.5.so
00000033cca00000      84 r-x-- 0000000000000000 003:00001 libselinux.so.1
00000033cca15000    2048 ----- 0000000000015000 003:00001 libselinux.so.1
00000033ccc15000       8 rw--- 0000000000015000 003:00001 libselinux.so.1
00000033ccc17000       4 rw--- 00000033ccc17000 000:00000   [ anon ]
00002b2d6b1aa000       8 rw--- 00002b2d6b1aa000 000:00000   [ anon ]
00002b2d6b1b0000       8 rw--- 00002b2d6b1b0000 000:00000   [ anon ]
00007fff3c652000      84 rw--- 00007ffffffe9000 000:00000   [ stack ]
00007fff3c679000      12 r-x-- 00007fff3c679000 000:00000   [ anon ]
ffffffffff600000    8192 ----- 0000000000000000 000:00000   [ anon ]
mapped: 18556K    writeable/private: 324K    shared: 0K
```

内存指标说明

1. mapped : VRT
2. Waiteable/private: RSS
3. Shared: 共享内存

参数 解释 **Address:00378000-0038d000** 进程所占的地址空间 **Kbytes** 该虚拟段的大小 **RSS** 设备号（主设备：次设备） **Anon** 设备的节点号，**0**表示没有节点与内存相对应 **Locked** 是否允许**swapped** **Mode** 权限：**r=read, w=write, x=execute, s=shared, p=private(copy on write)** **Mapping: bash** 对应的映像文件名

内存历史情况—sar工具使用

```
[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ sar -S -f /var/log/sa/sa14|head -10
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/14/2015      _x86_64_        (16 CPU)

12:00:01 AM kbswpfree kbswpused  %swpused  kbswpcad   %swpcad
12:01:01 AM   1051804       412      0.04         0      0.00
12:02:01 AM   1051804       412      0.04         0      0.00
12:03:01 AM   1051804       412      0.04         0      0.00
12:04:01 AM   1051804       412      0.04         0      0.00
12:05:01 AM   1051804       412      0.04         0      0.00
12:06:01 AM   1051804       412      0.04         0      0.00
12:07:01 AM   1051804       412      0.04         0      0.00

[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ sar -W -f /var/log/sa/sa14|head -10
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/14/2015      _x86_64_        (16 CPU)

12:00:01 AM  pswpin/s pswpout/s
12:01:01 AM      0.00      0.00
12:02:01 AM      0.00      0.00
12:03:01 AM      0.00      0.00
12:04:01 AM      0.00      0.00
12:05:01 AM      0.00      0.00
12:06:01 AM      0.00      0.00
12:07:01 AM      0.00      0.00

[admin@AY41A_AG:/home/admin] [cn-hangzhou-1:cn-hangzhou-dg-a01:raf:classic]
$ sar -B -f /var/log/sa/sa14 |head -10
Linux 2.6.18-164.11.1.el5 (r14a02001.dg.aliyun.com)     01/14/2015      _x86_64_        (16 CPU)

12:00:01 AM  pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
12:01:01 AM      0.00   9284.93 475513.86      0.00 183078.03      0.00      0.00      0.00      0.00
12:02:01 AM      0.00  16993.91  51698.74      0.00  21753.19      0.00      0.00      0.00      0.00
12:03:01 AM      0.00  19322.48  61195.07      0.00  24810.51      0.00      0.00      0.00      0.00
12:04:01 AM      0.07  13610.58 271824.91      0.00 105425.37      0.00      0.00      0.00      0.00
12:05:01 AM      0.00  22524.17  58754.46      0.00  23585.62      0.00      0.00      0.00      0.00
12:06:01 AM      0.00  16222.59 242841.63      0.00  93804.77      0.00      0.00      0.00      0.00
12:07:01 AM      0.00  11663.75 264723.91      0.00 102034.70      0.00      0.00      0.00      0.00

[admin@AY41A_AG:/home/ad
$ sar -r -f /var/log/sa/
Linux 2.6.18-164.11.1.el

12:00:01 AM kbmemfree kb
12:01:01 AM  22382428  2
12:02:01 AM  21466456  27982884
12:03:01 AM  20474580  28974760
12:04:01 AM  19370572  30078768
12:05:01 AM  18269044  31180296
12:06:01 AM  17525088  31924252
12:07:01 AM  16460548  32988792

12:02:01 AM  -3788.77      0.30   3853.67
12:03:01 AM  -4171.75      0.12   4009.22
12:04:01 AM  -4549.23      0.40   4569.47
12:05:01 AM  -4638.40      0.39   4421.58
12:06:01 AM  -3071.15      0.41   3152.25
12:07:01 AM  -4493.25      0.42   4445.15
```

内存的清理方法——dropcache/kill $pid

```
[root@iZ28lzm2ehvZ ~]# free -m
             total       used       free     shared    buffers     cached
Mem:           493        487          6          0         58        262
-/+ buffers/cache:        165        328
Swap:          511         93        418
[root@iZ28lzm2ehvZ ~]# sync
[root@iZ28lzm2ehvZ ~]# free -m
             total       used       free     shared    buffers     cached
Mem:           493        487          6          0         58        262
-/+ buffers/cache:        165        328
Swap:          511         93        418
[root@iZ28lzm2ehvZ ~]# echo 3 > /proc/sys/vm/drop_caches
[root@iZ28lzm2ehvZ ~]# free -m
             total       used       free     shared    buffers     cached
Mem:           493        330        163          0          0        178
-/+ buffers/cache:        151        341
Swap:          511         93        418
```

cache释放

To free pagecache:
echo 1 > /proc/sys/vm/drop_caches

To free dentries and inodes:
echo 2 > /proc/sys/vm/drop_caches

To free pagecache, dentries and inodes:
echo 3 > /proc/sys/vm/drop_caches

安全起见，先sync

内存调优常见方法

小技巧

1. 降低swap的使用率：
sysctl -a | grep swappiness
vm.swappiness = 60

2. 关闭Swap:swapoff

3. 限制其他用户的内存使用
vim /etc/security/limits.conf
user1 hard as 1000（用户user1所有累加起来，内存不超过1000kiB）
user1 soft as 800（用户user1一次运行，内存不超过800kiB）

4. 大量连续内存数据：
vim /etc/sysctl.conf
vm.nr_hugepage=20

5. 调节page cache（大量一样的请求 调大page cache）
vm.lowmem_reserve_ratio = 256 256 32（保留多少内存作为pagecache 当前 最大 最小）
vm.vfs_cache_pressure=100（大于100，回收pagecache）
vm.page.cluster=3（一次性从swap写入内存的量为2的3次方页）
vm.zone_reclaim_mode=0/1（当内存危机时，是否尽量回收内存 0:尽量回收 1:尽量不回收）
min_free_kbytes：该文件表示强制Linux VM最低保留多少空闲内存（Kbytes）。

6 脏页
vm.dirty_background_radio=10（当脏页占内存10%，pdflush工作）
vm.dirty_radio=40（当进程自身脏页占内存40%，进程自己处理脏页，将其写入磁盘）
vm.dirty_expire_centisecs=3000（脏页老化时间为30秒 3000/100=30秒）
vm.dirty_writeback_centisecs=500（每隔5秒，pdflush监控一次内存数量 500/100=5秒）

Linux系统问题之网络篇

常见用户问题

网络不可达	机器不可达，出去不可达
网络丢包	进包丢包或者出包丢包
网速慢	传输速度慢
网络调优	我们的网络使用率很高，有没有什么优化建议？

Linux 内存常见问题排查思路

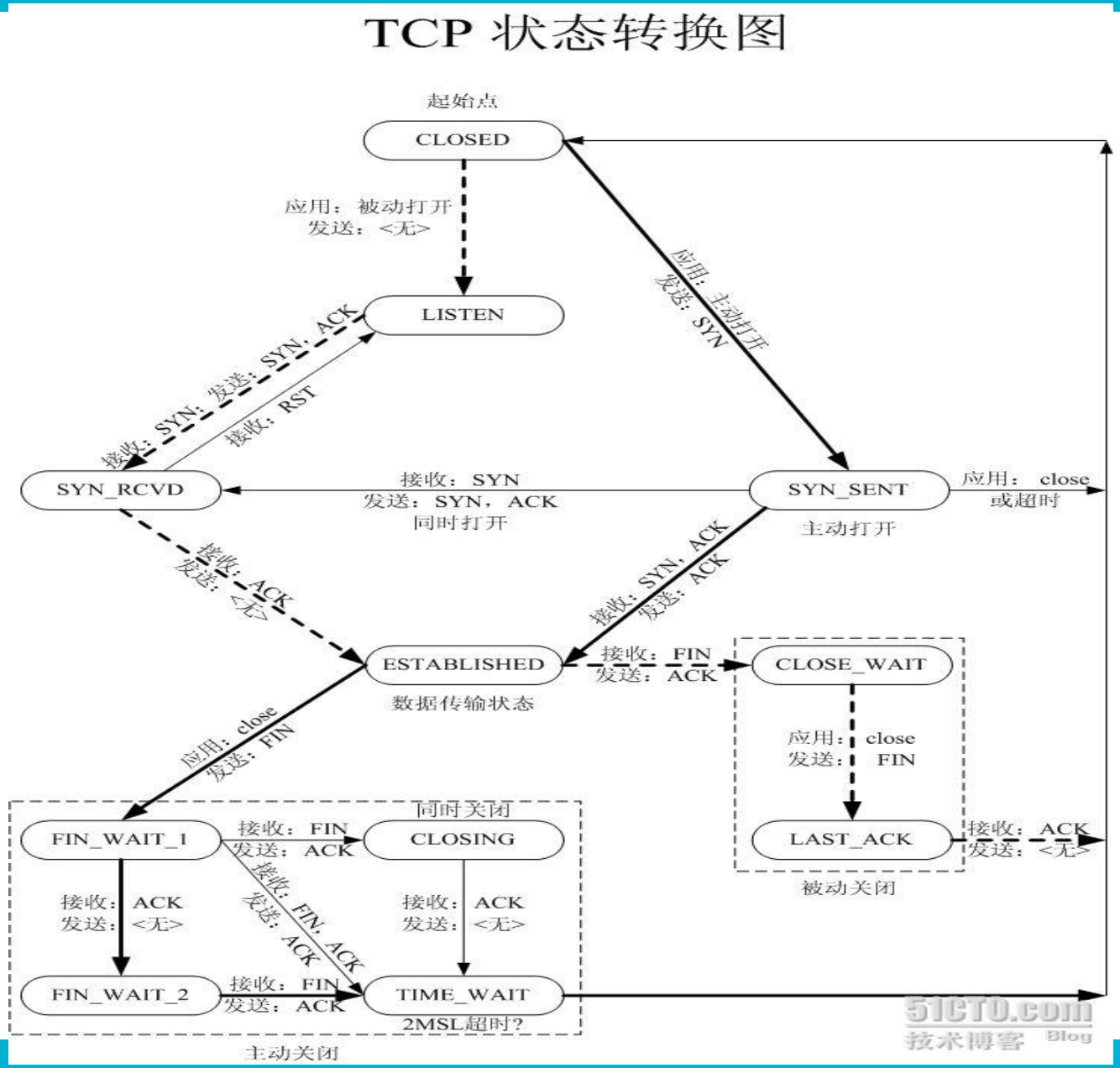
网络问题排查思路

定位问题段	在哪一段出现问题，丢包或者不通
网络包去哪啦	如果丢包或者不通，网络包去哪里啦
网络优化	网络协议栈是否有优化空间

Linux 网卡信息查看——ifconfig

```
[root@iZ233h7oezdZ ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:16:3E:02:36:07
          inet addr:10.165.6.202  Bcast:10.165.7.255  Mask:255.255.248.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:402713 errors:0 dropped:0 overruns:0 frame:0
          TX packets:181604 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:27841384 (26.5 MiB)  TX bytes:12157898 (11.5 MiB)
          Interrupt:19

eth1      Link encap:Ethernet  HWaddr 00:16:3E:02:22:A7
          inet addr:114.215.175.140  Bcast:114.215.175.255  Mask:255.255.252.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7369052 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6801 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:338848139 (323.1 MiB)  TX bytes:812259 (793.2 KiB)
          Interrupt:20

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

Linux连接状态查看——netstat & SS



路由跟踪——tracert/mtr

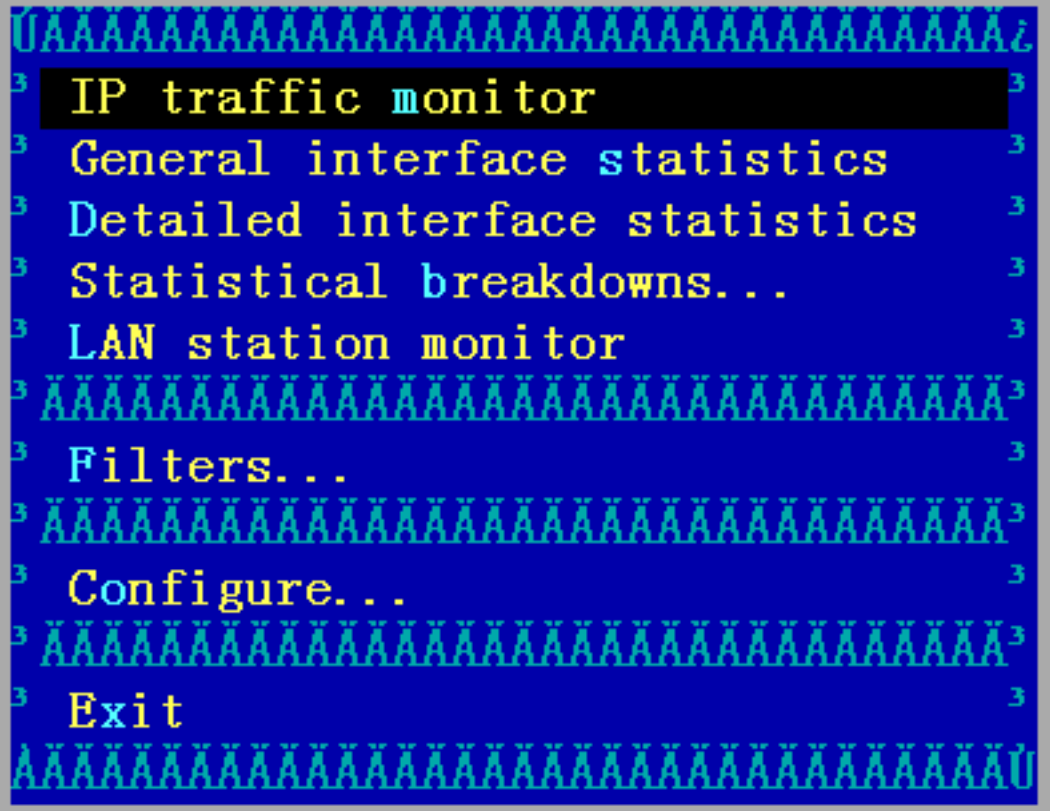
```
[root@iZ233h7oezdZ ~]# traceroute www.baidu.com
traceroute to www.baidu.com (220.181.112.244), 30 hops max, 60 byte packets
 1  218.244.139.248 (218.244.139.248)  0.593 ms  0.734 ms  0.850 ms
 2  10.106.14.26 (10.106.14.26)  0.435 ms 10.106.14.18 (10.106.14.18)  0.405 ms 10.106.14.26 (10.106.14.26)  0.418 ms
 3  42.120.244.201 (42.120.244.201)  0.857 ms 42.120.244.209 (42.120.244.209)  0.995 ms 42.120.244.205 (42.120.244.205)  0.989 ms
 4  42.120.244.173 (42.120.244.173)  2.355 ms 42.120.244.169 (42.120.244.169)  1.801 ms 42.120.244.173 (42.120.244.173)  2.393 ms
 5  122.224.187.42 (122.224.187.42)  1.564 ms 115.238.21.118 (115.238.21.118)  1.389 ms 122.224.187.34 (122.224.187.34)  24.805 ms
 6  61.164.31.165 (61.164.31.165)  5.856 ms 61.164.13.145 (61.164.13.145)  1.289 ms 61.164.31.237 (61.164.31.237)  1.244 ms
 7  202.97.68.161 (202.97.68.161)  26.297 ms  26.091 ms  26.313 ms
 8  * 220.181.0.38 (220.181.0.38)  295.444 ms *
 9  * * *
10  220.181.182.38 (220.181.182.38)  29.474 ms 220.181.17.90 (220.181.17.90)  31.541 ms 220.181.17.22 (220.181.17.22)  43.304 ms
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```

1. AY66H_AG:~ (mtr)

```
                                My traceroute  [v0.85]
anderszhangwdeMacBook-Air.local (0.0.0.0)                Tue May  5 17:13:45 2015
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                                 Packets               Pings
 Host                                          Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. 10.1.32.1                                   0.0%    16    1.8   2.1   1.4   6.0   1.0
 2. 10.64.200.33                                0.0%    16    1.9   5.2   1.6  17.1   4.3
 3. 10.64.1.1                                   0.0%    16    2.0   2.6   1.5   6.8   1.5
 4. 42.120.74.4                                 0.0%    16    1.7   3.2   1.7  13.6   2.8
 5. 42.120.253.233                              0.0%    16    2.9   4.1   2.7  11.7   2.2
 6. 115.238.21.114                              0.0%    16    2.6  16.3   2.5 101.4  28.8
 7. 115.233.23.198                              0.0%    16    5.5   6.0   3.3  15.9   3.5
 8. 115.239.209.34                             12.5%    16    3.8   4.5   3.6   8.6   1.3
 9. ???
10. 115.239.210.27                              6.7%    15    3.7   4.4   3.6  10.0   1.6
```

Ifstat & iftop & iptraf

```
[root@iZ233h7oezdZ ~]# iftop --help
iftop: unknown option --
iftop: display bandwidth usage on an interface by host

Synopsis: iftop -h | [-npblNBP] [-i interface] [-f filter code]
                     [-F net/mask] [-G net6/mask6]

   -h                  display this message
   -n                  don't do hostname lookups
   -N                  don't convert port numbers to services
   -p                  run in promiscuous mode (show traffic between other
                       hosts on the same network segment)
   -b                  don't display a bar graph of traffic
   -B                  Display bandwidth in bytes
   -i interface        listen on named interface
   -f filter code      use filter code to select packets to count
                       (default: none, but only IP packets are counted)
   -F net/mask         show traffic flows in/out of IPv4 network
   -G net6/mask6       show traffic flows in/out of IPv6 network
   -l                  display and count link-local IPv6 traffic (default: off)
   -P                  show ports as well as hosts
   -m limit            sets the upper limit for the bandwidth scale
   -c config file      specifies an alternative configuration file
   -t                  use text interface without ncurses

   Sorting orders:
   -o 2s                Sort by first column (2s traffic average)
   -o 10s               Sort by second column (10s traffic average) [default]
   -o 40s               Sort by third column (40s traffic average)
   -o source            Sort by source address
   -o destination       Sort by destination address

   The following options are only available in combination with -t
   -s num              print one single text output afer num seconds, then quit
   -L num              number of lines to print

iftop, version 1.0pre4
copyright (c) 2002 Paul Warren <pdw@ex-parrot.com> and contributors
```

Nfcontrack——查看连接数

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

返回结果如下：

```
LAST_ACK 14
SYN_RECV 348
ESTABLISHED 70
FIN_WAIT1 229
FIN_WAIT2 30
CLOSING 33
TIME_WAIT 18122
```

链接数统计：
查看每秒的链接数：cat /proc/sys/net/netfilter/nf_conntrack_count

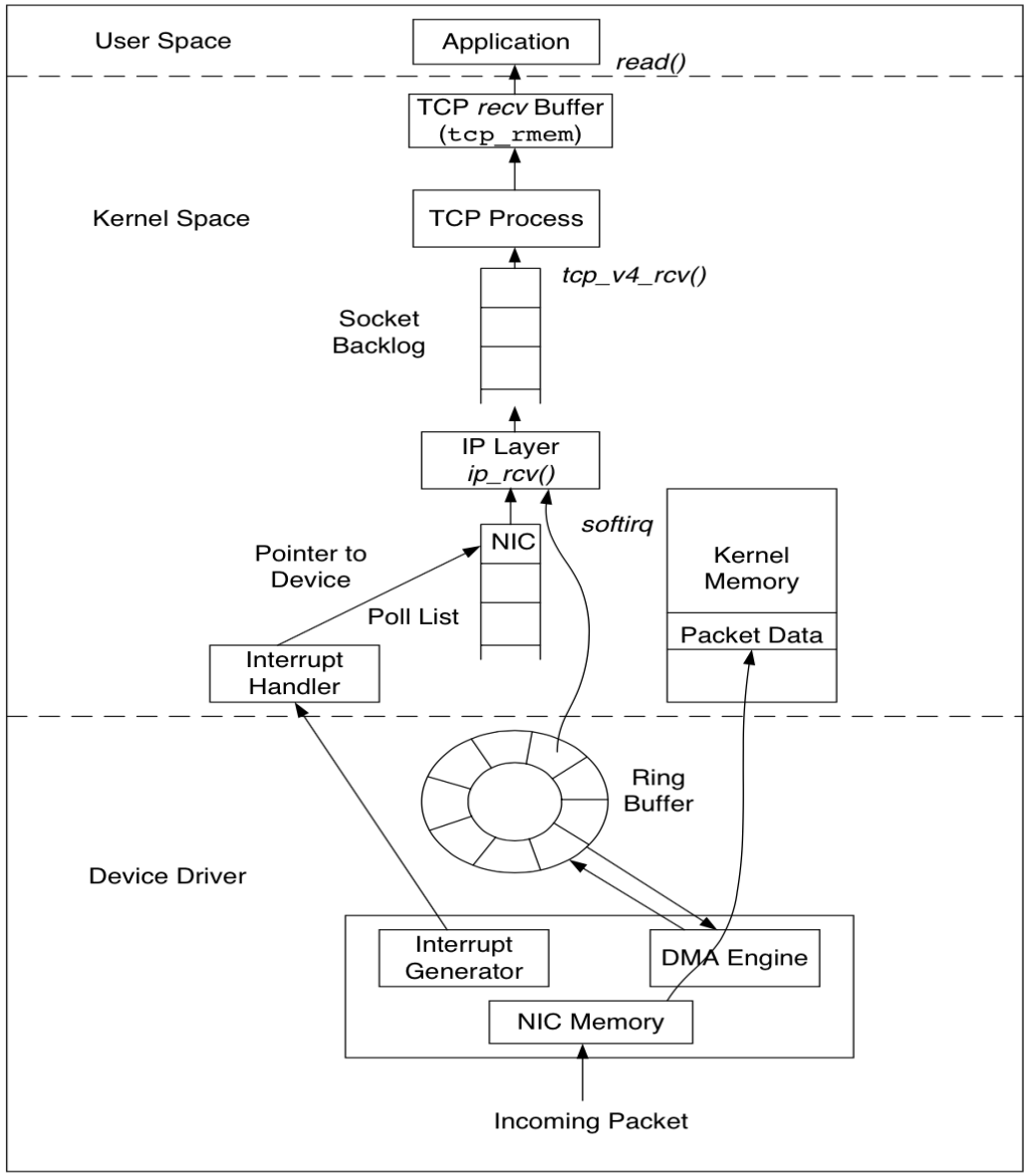
tcpdump

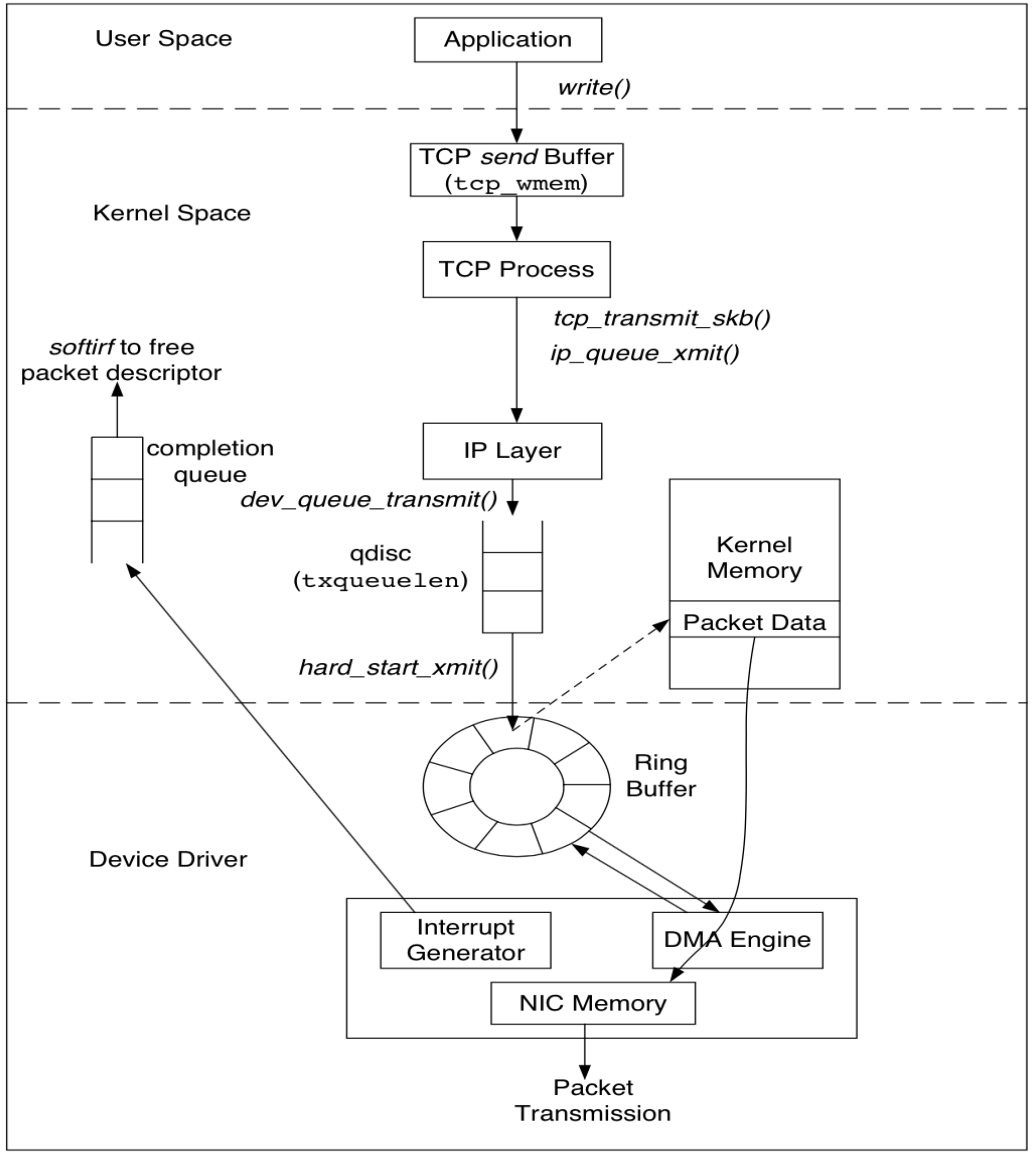
Sudo tcpdump –i eth0 host 10.192.168.1 and port 80 and tcp

压测

典型工具：netperf & iperf

Linux网络接发包原理-调优





Linux网络队列&调优

Syn 队列：保存SYN_RECV状态的连接。队列长度由net.ipv4.tcp_max_syn_backlog设置
接收队列：net.core.netdev_max_backlog
Ringbuffer: ethtool -g eth0

Qdisc: ifconfig 可以看到txqueuelen, ifconfig eth0 txqueuelen 2000

MSL(maximum segment lifetime): timewait状态的持续时间与该参数有关联

参考：http://blog.sina.com.cn/s/blog_e59371cc0102vg4n.html

Nagie算法&delay ACK:
参考：http://www.cnblogs.com/polymorphism/archive/2012/12/10/high_latency_for_small_size_entities_in_table_service.html

典型案例

1、用户案例：拷贝文件速度不高，应用协议不一样导致的发包速度不一样
Tcpdump抓包，看到包的大小和包的间隔不一样；

2、用户案例：系统消耗大，丢包，Timewait状态的连接很大
Netstat –anlp|grep TIME_WAIT|wc –l

3、用户案例：Nagie算法&delay ACK，导致TPS上不去
Tcpdump发现ack与回包间隔20ms

4、用户案例：运营商链路质量问题-MTR/tracert
Mtr发现链路丢包

5、网络配置导致故障：ping网关不通
Gateway无法ping通（持续）

6、访问域名慢：DNS配置错误——dig+ping
Dig看dns是否慢

7、网络吞吐率压测：netperf

Linux常见问题之磁盘篇

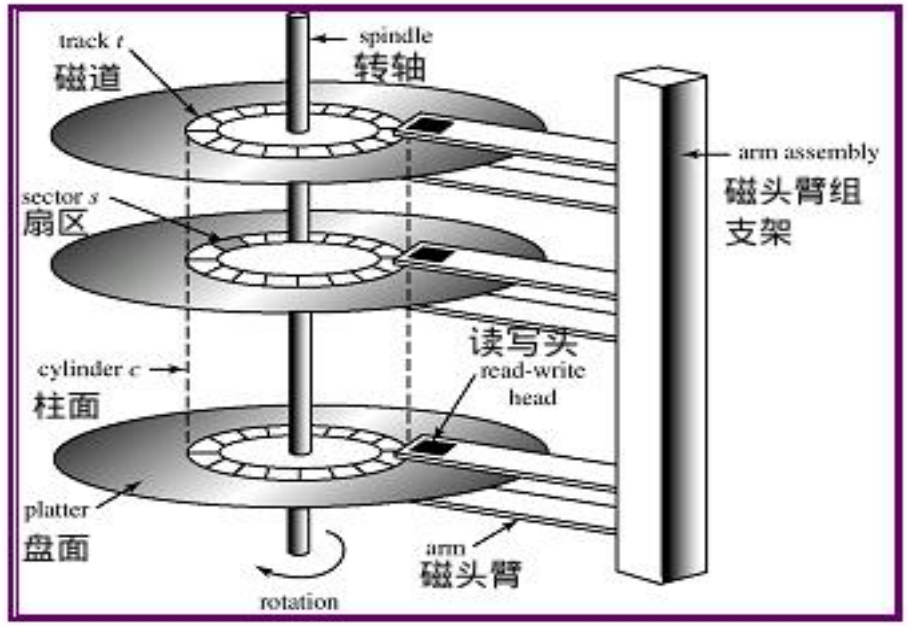
常见用户问题

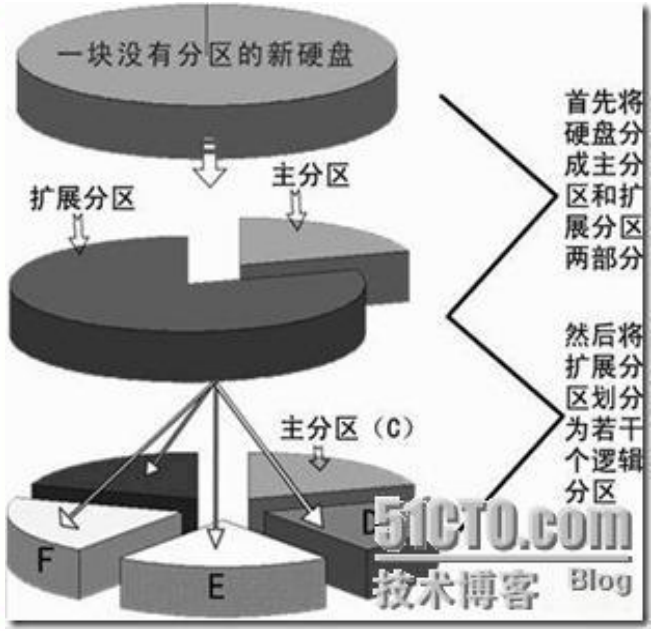
磁盘使用率	为什么机器使用的IO很高？
IO去哪啦	IO都用到哪里啦？

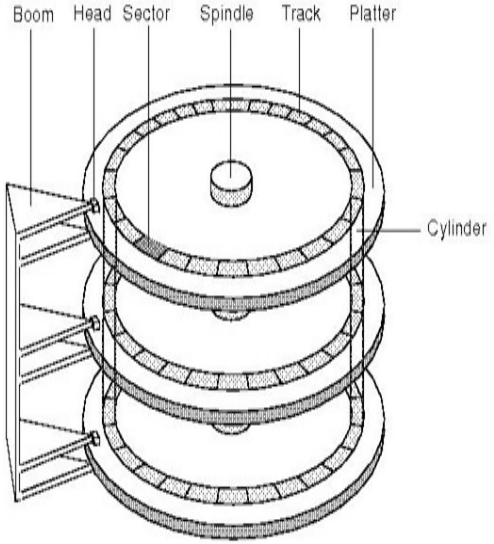
fdisk工具使用

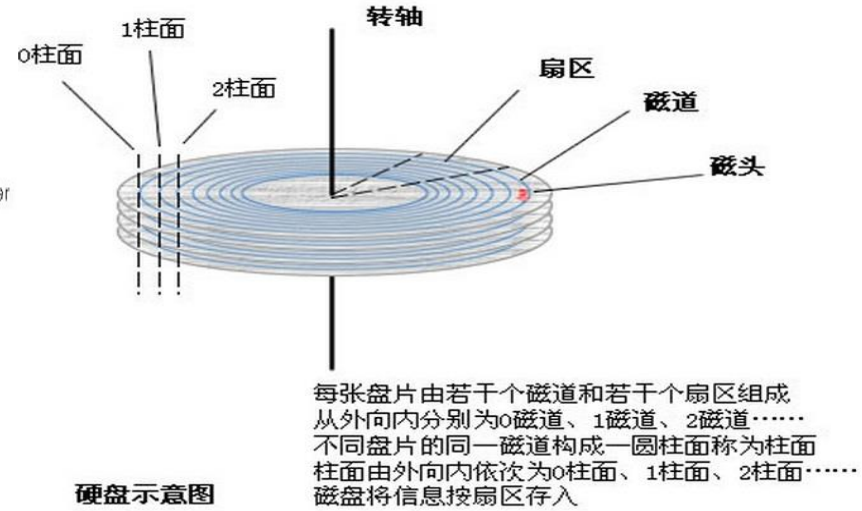
```
[root@www.linuxidc.com ~]# fdisk -l
Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 b
Sector size (logical/physical): 512 bytes /
I/O size (minimum/optimal): 512 bytes / 512
Disk identifier: 0x00044938
   Device Boot      Start         End      Blo
System
/dev/sda1   *           1         638     512000
Linux
Partition 1 does not end on cylinder boundar
/dev/sda2             638         893    2048000   83
Linux
Partition 2 does not end on cylinder boundary.
/dev/sda3             893        1020    1024000   82
Linux swap / Solaris
Partition 3 does not end on cylinder boundary.
/dev/sda4            1020        1306    2292736    5
Extended
/dev/sda5            1021        1306    2291712   83
Linux
```

head<磁头>， sectors<扇区>， track<磁道>，
cylinder <柱面>，Unit<柱面大小>









FIO工具使用——测试磁盘性能（读写能力&IOPS）

顺序读：

fio -filename=/dev/hda1 -direct=1 -iodepth 1 -thread -rw=read -ioengine=psync -bs=16k -size=2G -numjobs=30 -runtime=1000 -group_reporting -name=mytest

随机写：

fio -filename=/dev/hda1 -direct=1 -iodepth 1 -thread -rw=randwrite -ioengine=psync -bs=16k -size=2G -numjobs=30 -runtime=1000 -group_reporting -name=mytest

顺序写：

fio -filename=/dev/hda1 -direct=1 -iodepth 1 -thread –rw=write -ioengine=psync -bs=16k -size=2G -numjobs=30 -runtime=1000 -group_reporting -name=mytest

混合随机读写：

fio -filename=/dev/hda1 -direct=1 -iodepth 1 -thread -rw=randrw -rwmixread=70 -ioengine=psync -bs=16k -size=2G -numjobs=30 -runtime=100 -group_reporting -name=mytest -ioscheduler=noop

```
[root@iZ28lzm2ehvZ ~]# fio -filename=/dev/hda1 -direct=1 -iodepth 1 -thread -rw=read -ioengine=psync -bs=16k -size
mytest: (g=0): rw=read, bs=16K-16K/16K-16K/16K-16K, ioengine=psync, iodepth=1
...
mytest: (g=0): rw=read, bs=16K-16K/16K-16K/16K-16K, ioengine=psync, iodepth=1
fio-2.0.14
Starting 30 threads
Jobs: 28 (f=28): [RRRRRRRRRRRR__RRRRRRRRRRRRRRR] [74.6% done] [17238K/0K/0K /s] [1077 /0 /0  iops] [eta 05m:40s]
mytest: (groupid=0, jobs=30): err= 0: pid=4688: Wed Jan  7 01:02:54 2015
  read : io=51272MB, bw=52501KB/s, iops=3281 , runt=1000019msec
    clat (usec): min=71 , max=3878.3K, avg=9034.54, stdev=75409.23
```

dd工具使用

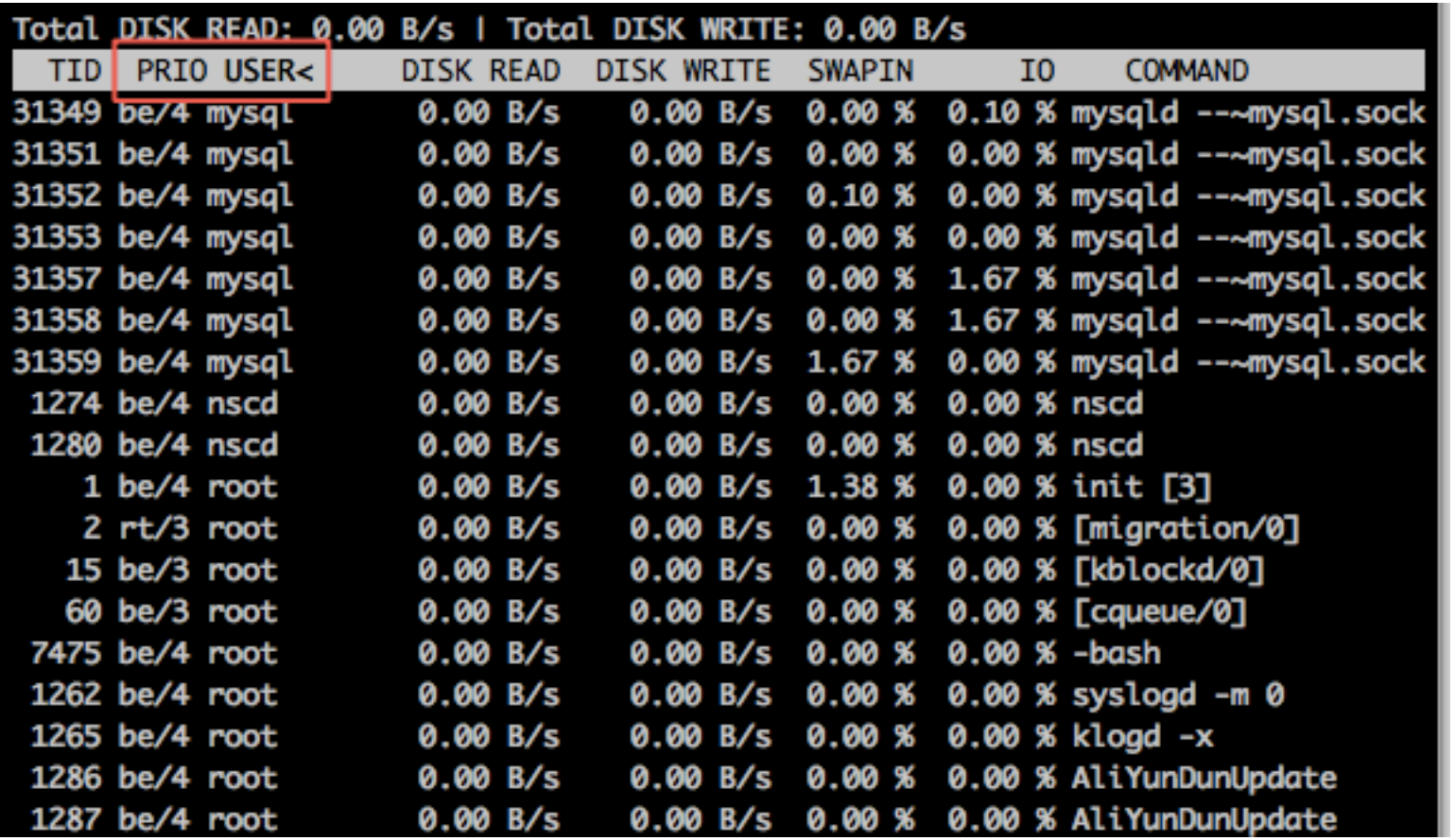
12.测试硬盘的读写速度
dd if=/dev/zero bs=1024 count=1000000 of=/root/1Gb.file
dd if=/root/1Gb.file bs=64k | dd of=/dev/null
通过以上两个命令输出的命令执行时间，可以计算出硬盘的读、写速度。
13.确定硬盘的最佳块大小：
dd if=/dev/zero bs=1024 count=1000000 of=/root/1Gb.file
dd if=/dev/zero bs=2048 count=500000 of=/root/1Gb.file
dd if=/dev/zero bs=4096 count=250000 of=/root/1Gb.file
dd if=/dev/zero bs=8192 count=125000 of=/root/1Gb.file
通过比较以上命令输出中所显示的命令执行时间，即可确定系统最佳的块大小。

blktrace工具使用——查看机器当前IO情况

```
[root@iZ28lzm2ehvZ ~]#blktrace -d /dev/hda1 -a issue -a complete -w 120  -o - |  blkiomon -I
2 -h -
```

```
sizes read (bytes): num 0, min -1, max 0, sum 0, squ 0, avg nan, var nan
sizes write (bytes): num 2, min 4096, max 16384, sum 20480, squ 285212672, avg 1
0240.0, var 37748736.0
d2c read (usec): num 0, min -1, max 0, sum 0, squ 0, avg nan, var nan
d2c write (usec): num 2, min 1356, max 1900, sum 3256, squ 5448736, avg 1628.0,
var 73984.0
throughput read (bytes/msec): num 0, min -1, max 0, sum 0, squ 0, avg nan, var n
an
throughput write (bytes/msec): num 2, min 3020, max 8623, sum 11643, squ 8347652
9, avg 5821.5, var 7848402.2
sizes histogram (bytes):
             0:      0          1024:      0          2048:      0          4096:      1
          8192:      0         16384:      1         32768:      0         65536:      0
        131072:      0        262144:      0        524288:      0       1048576:      0
       2097152:      0       4194304:      0       8388608:      0     > 8388608:      0
d2c histogram (usec):
             0:      0             8:      0            16:      0            32:      0
            64:      0           128:      0           256:      0           512:      0
          1024:      0          2048:      2          4096:      0          8192:      0
         16384:      0         32768:      0         65536:      0        131072:      0
        262144:      0        524288:      0       1048576:      0       2097152:      0
       4194304:      0       8388608:      0      16777216:      0      33554432:      0
    >33554432:      0
bidirectional requests: 0
```

iotop工具使用——发现问题进程&线程



```
Total DISK READ: 0.00 B/s | Total DISK WRITE: 0.00 B/s
  TID  PRIO  USER<     DISK READ  DISK WRITE  SWAPIN      IO    COMMAND
31349 be/4 mysql        0.00 B/s    0.00 B/s  0.00 %  0.10 % mysqld --~mysql.sock
31351 be/4 mysql        0.00 B/s    0.00 B/s  0.00 %  0.00 % mysqld --~mysql.sock
31352 be/4 mysql        0.00 B/s    0.00 B/s  0.10 %  0.00 % mysqld --~mysql.sock
31353 be/4 mysql        0.00 B/s    0.00 B/s  0.00 %  0.00 % mysqld --~mysql.sock
31357 be/4 mysql        0.00 B/s    0.00 B/s  0.00 %  1.67 % mysqld --~mysql.sock
31358 be/4 mysql        0.00 B/s    0.00 B/s  0.00 %  1.67 % mysqld --~mysql.sock
31359 be/4 mysql        0.00 B/s    0.00 B/s  1.67 %  0.00 % mysqld --~mysql.sock
 1274 be/4 nscd         0.00 B/s    0.00 B/s  0.00 %  0.00 % nscd
 1280 be/4 nscd         0.00 B/s    0.00 B/s  0.00 %  0.00 % nscd
    1 be/4 root         0.00 B/s    0.00 B/s  1.38 %  0.00 % init [3]
    2 rt/3 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % [migration/0]
   15 be/3 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % [kblockd/0]
   60 be/3 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % [cqueue/0]
 7475 be/4 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % -bash
 1262 be/4 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % syslogd -m 0
 1265 be/4 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % klogd -x
 1286 be/4 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % AliYunDunUpdate
 1287 be/4 root         0.00 B/s    0.00 B/s  0.00 %  0.00 % AliYunDunUpdate
```

iotop命令的键盘快捷键：

1、左右箭头改变排序方式，默认是按IO排序

2、r键是反向排序

3、o键是只显示有IO输出的进程

4、同样q是退出

5、iotop –P只显示进程，而非线程

通过IO TOP工具，能够定位到哪个进程或者线程的IO量很大

iotop按列显示每个进程/线程的I/O读写带宽，同时也显示进程/线程做swap交换和等待I/O所占用的百分比

iostat工具使用——发现问题设备

```
[root@iZ28lzm2ehvZ ~]# iostat -d -x 1
Linux 2.6.18-274.el5 (iZ28lzm2ehvZ)     2015年01月07日

Device:         rrqm/s   wrqm/s   r/s   w/s    rsec/s    wsec/s avgrq-sz avgqu-sz   await  svctm  %util
hda               0.09     4.23  0.84  0.75     29.13     39.86    43.57     0.06   35.68   0.84   0.13
hda1              0.09     4.23  0.84  0.75     29.13     39.86    43.57     0.06   35.68   0.84   0.13
```

rrqm/s：每秒这个设备相关的读取请求有多少被Merge了;wrqm/s：每秒这个设备相关的写入请求有多少被Merge了。
rsec/s：每秒读取的扇区数；wsec/：每秒写入的扇区数。
rKB/s：The number of read requests that were issued to the device per second；wKB/s;
avgrq-sz 平均请求扇区的大小
avgqu-sz 是平均请求队列的长度。毫无疑问，队列长度越短越好。
await：每一个IO请求的处理的平均时间（单位是微秒毫秒）。
这里可以理解为IO的响应时间，一般地系统IO响应时间应该低于5ms，如果大于10ms就比较大了。
这个时间包括了队列时间和服务时间，也就是说，
一般情况下，await大于svctm，它们的差值越小，则说明队列时间越短，反之差值越大，队列时间越长，说明系统出了问题。
svctm 表示平均每次设备I/O操作的服务时间（以毫秒为单位）。
如果svctm的值与await很接近，表示几乎没有I/O等待，磁盘性能很好，
如果await的值远高于svctm的值，则表示I/O队列等待太长，
系统上运行的应用程序将变慢。
%util：在统计时间内所有处理IO时间，除以总共统计时间。:例如，如果统计间隔1秒，该设备有0.8秒在处理IO，而0.2秒闲置，
那么该设备的%util = 0.8/1 = 80%，所以该参数暗示了设备的繁忙程度一般地，如果该参数是100%表示设备已经接近满负荷运行了

lsof工具使用——发现问题文件&进程

```
[root@iZ28lzm2ehvZ ~]# lsof  /tmp/test.1
COMMAND  PID USER   FD   TYPE DEVICE  SIZE/OFF  NODE NAME
dd      4797 root    1w   REG    3,1 386105344 81935 /tmp/test.1
[root@iZ28lzm2ehvZ ~]#
```

```
[root@iZ28lzm2ehvZ ~]# ps axu|grep dd
root      4869  1.1  0.1  63180   616 pts/0    D+   01:50   0:00 dd oflag direct nonblock if /dev/zero of /tmp/test.1 bs 8k count 80000
root      4871  0.0  0.1  61184   748 pts/6    R+   01:50   0:00 grep dd
[root@iZ28lzm2ehvZ ~]# sudo yum install kernel-debuginfo
[root@iZ28lzm2ehvZ ~]# lsof -p 4869
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
dd      4869 root  cwd    DIR    3,1     4096 1007617 /root
dd      4869 root  rtd    DIR    3,1     4096       2 /
dd      4869 root  txt    REG    3,1    45848  811078 /bin/dd
dd      4869 root  mem    REG    3,1   143600  409855 /lib64/ld-2.5.so
dd      4869 root  mem    REG    3,1  1722304  409856 /lib64/libc-2.5.so
dd      4869 root  mem    REG    3,1   145824  409863 /lib64/libpthread-2.5.so
dd      4869 root  mem    REG    3,1    53448  409864 /lib64/librt-2.5.so
dd      4869 root  mem    REG    3,1 56419408  788468 /usr/lib/locale/locale-archive
dd      4869 root    0r   CHR    1,5      0t0    1084 /dev/zero
dd      4869 root    1w   REG    3,1 75497472   81935 /tmp/test.1
dd      4869 root    2u   CHR  136,0      0t0       2 /dev/pts/0
[root@iZ28lzm2ehvZ ~]#
```

上云交流之上云前的准备

云产品的功能&选型

用户系统对应的功能是否有云产品的替代方案
用户系统中现有的功能是否有替代的云产品

云上架构和迁云方案

性价比、容灾方案、稳定性
数据库如何迁移
网络架构如何迁移过程中兼容

性能测试

对比用户系统架构中的性能要求，测试云产品是否满足
包括：服务器、数据库、缓存

运维平台的兼容

监控、部署、扩容

欢迎讨论

Thank you

共建中国云计算生态圈