

CentOS 丛书目录 — 系统管理 — 网络服务 — 应用部署

sysstat 工具

内容提要

1. 了解 **sysstat** 工具包的组成
2. 理解 **sar** 累计工作方式的配置
3. 掌握 **sar**、**iostat**、**mpstat** 命令的使用

sysstat 工具简介

sysstat 是 Linux 系统中的常用工具包。它的主要用途是观察服务负载，比如CPU和内存的占用率、网络的使用率以及磁盘写入和读取速度等。

sysstat 工具包中包含两类工具：

- 即时查看工具：**iostat**、**mpstat**、**sar**
- 累计统计工具：**sar**

也就是说，**sar** 具有这两种功能。因此，**sar** 是 **sysstat** 中的核心工具。

为了实现 **sar** 的累计统计，系统必须周期地记录当时的信息，这是通过调用 **/usr/lib/sa/** 中的三个工具实现的：

- **sa1**：收集并存储每天系统动态信息到一个二进制的文件中，用作 **sadc** 的前端程序
- **sa2**：收集每天的系统活跃信息写入总结性的报告，用作 **sar** 的前端程序
- **sadc**：系统动态数据收集工具，收集的数据被写入一个二进制的文件中，它被用作 **sar** 工具的后端

在 CentOS 系统的默认设置中，以如下的方式使用这三个工具：

1. 在守护进程 **/etc/rc.d/init.d/sysstat** 中使用 **/usr/lib/sa/sadc -F -L -** 命令创建当日记录文件，文件为 **/var/log/sa/saDD**，其中 **DD** 为当天的日期。当系统重新启动后，会向文件 **/var/log/sa/saDD** 输出类似 **11:37:16 AM LINUX RESTART** 这样的行信息。
2. 在 **cron** 任务 **/etc/cron.d/sysstat** 中每隔10分钟执行一次 **/usr/lib/sa/sa1 1 1** 命令，将信息写入文件 **/var/log/sa/saDD**
3. 在 **cron** 任务 **/etc/cron.d/sysstat** 中每天 **23:53** 执行一次 **/usr/lib/sa/sa2 -A** 命令，将当天的汇总信息写入文件 **/var/log/sa/saDD**

您可以修改 **/etc/cron.d/sysstat** 以适合您的需要。

另外，文件 **/var/log/sa/saDD** 为二进制文件，不能使用 **more**、**less** 等文本工具查看，必须用 **sar** 或 **sadf** 命令查看。

sar

在使用 Linux 系统时，常常会遇到各种各样的问题，比如系统容易死机或者运行速度突然变慢，这时我们常常猜测：是否硬盘空间不足，是否内存不足，是否 I/O 出现瓶颈，还是系统的核心参数出了问题？这时，我们应该考虑使用 **sar** 工具对系统做一个全面了解，分析系统的负载状况。

sar (System Activity Reporter) 是系统活动情况报告的缩写。**sar** 工具将对系统当前的状态进行取样，然后通过计算数据和比例来表达系统的当前运行状态。它的特点是可以连续对系统取样，获得大量的取样数据；取样数据和分析的结果都可以存入文件，所需的负载很小。**sar** 是目前 Linux 上最为全面的系统性能分析工具之一，可以从多方面对系统的活动进行报告，包括：文件的读写情况、系统调用的使用情况、磁盘I/O、CPU效率、内存使用状况、进程活动及IPC有关的活动等。为了提供不同的信息，**sar** 提供了丰富的选项、因此使用较为复杂。

sar 的命令格式

sar 的命令格式为：

```
sar [-A] [-b] [-B] [-c] [-d] [-i interval] [-p] [-q]
    [-r] [-R] [-t] [-u] [-v] [-V] [-w] [-W] [-y]
    [-n {DEV|EDEV|NFS|NFSD|SOCK|ALL}]
    [-x {pid|SELF|ALL}] [-X {pid|SELF|ALL}]
    [-I {irq|SUM|ALL|XALL}] [-P {cpu|ALL}]
    [-o filename] [-f filename]
    [-s hh:mm:ss] [-e hh:mm:ss]
    [interval [count]]
```

其中：

- **interval**：为取样时间间隔
- **count**：为输出次数，若省略此项，默认值为 1

常用选项：

选项	说明
-A	等价于 -bBcdqrRuvWwy -I SUM -I XALL -n ALL -P ALL
-b	显示I/O和传送速率的统计信息
-B	输出内存页面的统计信息
-c	输出进程统计信息，每秒创建的进程数
-d	输出每一个块设备的活动信息
-i interval	指定间隔时长，单位为秒
-p	显示友好设备名字，以方便查看，也可以和-d 和-n 参数结合使用，比如 -dp 或-np
-q	输出进程队列长度和平均负载状态统计信息
-r	输出内存和交换空间的统计信息
-R	输出内存页面的统计信息
-t	读取 /var/log/sa/saDD 的数据时显示其中记录的原始时间，如果没有这个参数使用用户的本地时间
-u	输出CPU使用情况的统计信息
-v	输出inode、文件和其他内核表的统计信息
-V	输出版本号信息
-w	输出系统交换活动信息
-W	输出系统交换的统计信息
-y	输出TTY设备的活动信息
-n {DEV EDEV NFS NFSD SOCK ALL}	分析输出网络设备状态统计信息。
DEV	报告网络设备的统计信息
EDEV	报告网络设备的错误统计信息
NFS	报告 NFS 客户端的活动统计信息
NFSD	报告 NFS 服务器的活动统计信息
SOCK	报告网络套接字（sockets）的使用统计信息
ALL	报告所有类型的网络活动统计信息
-x {pid SELF ALL}	输出指定进程的统计信息。
pid	用 pid 指定特定的进程
SELF	表示 sar 自身
ALL	表示所有进程
-X {pid SELF ALL}	输出指定进程的子进程的统计信息
-I {irq SUM ALL XALL}	输出指定中断的统计信息。
irq	指定中断号
SUM	指定输出每秒接收到的中断总数
ALL	指定输出前16个中断
XALL	指定输出全部的中断信息
-P {cpu ALL}	输出指定 CPU 的统计信息
-o filename	将输出信息保存到文件 filename
-f filename	从文件 filename 读取数据信息。filename 是使用-o 选项时生成的文件。
-s hh:mm:ss	指定输出统计数据的起始时间
-e hh:mm:ss	指定输出统计数据的截至时间，默认为18:00:00

sar 使用举例

从 `/var/log/sa/saDD` 中读取累计统计信息

1、输出CPU使用情况的统计信息

```
[root@cnetos5 ~]# sar
[root@cnetos5 ~]# sar -u
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM      CPU      %user      %nice    %system    %iowait    %steal     %idle
12:10:01 AM      all       0.02       0.00       0.14       0.01       0.00     99.84
12:20:01 AM      all       0.02       0.00       0.12       0.01       0.00     99.86
12:30:01 AM      all       0.01       0.00       0.12       0.01       0.00     99.86
Average:         all       0.03       0.00       0.13       0.01       0.00     99.84
```

输出项说明:

CPU	all 表示统计信息为所有 CPU 的平均值。
%user	显示在用户级别(application)运行使用 CPU 总时间的百分比。
%nice	显示在用户级别,用于nice操作,所占用 CPU 总时间的百分比。
%system	在核心级别(kernel)运行所使用 CPU 总时间的百分比。
%iowait	显示用于等待I/O操作占用 CPU 总时间的百分比。
%steal	管理程序(hypervisor)为另一个虚拟进程提供服务而等待虚拟 CPU 的百分比。
%idle	显示 CPU 空闲时间占用 CPU 总时间的百分比。

1. 若 **%iowait** 的值过高,表示硬盘存在I/O瓶颈
2. 若 **%idle** 的值高但系统响应慢时,有可能是 CPU 等待分配内存,此时应加大内存容量
3. 若 **%idle** 的值持续低于 10,则系统的 CPU 处理能力相对较低,表明系统中最需要解决的资源是 CPU。

2、显示I/O和传送速率的统计信息

```
[root@cnetos5 ~]# sar -b
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM      tps      rtps      wtps    bread/s    bwrtn/s
12:10:01 AM      1.58      0.00      1.58      0.00     16.71
12:20:01 AM      1.09      0.00      1.09      0.00     10.85
12:30:01 AM      1.08      0.00      1.08      0.00     10.74
Average:         1.24      0.00      1.24      0.00     12.70
```

输出项说明:

tps	每秒钟物理设备的 I/O 传输总量
rtps	每秒钟从物理设备读入的数据总量
wtps	每秒钟向物理设备写入的数据总量
bread/s	每秒钟从物理设备读入的数据量,单位为 块/s
bwrtn/s	每秒钟向物理设备写入的数据量,单位为 块/s

3、输出内存页面的统计信息

```
[root@cnetos5 ~]# sar -B
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM    pgpgin/s    pgpgout/s    fault/s    majflt/s
12:10:01 AM      0.00      4.17      9.74      0.00
12:20:01 AM      0.00      2.71      2.24      0.00
12:30:01 AM      0.00      2.69      2.25      0.00
Average:         0.00      3.17      4.07      0.00
```

输出项说明:

pgpgin/s	每秒钟从磁盘读入的系统页面的 KB 总数
pgpgout/s	每秒钟向磁盘写出的系统页面的 KB 总数
fault/s	系统每秒产生的页面失效(major + minor)数量

majflt/s	系统每秒产生的页面失效(major)数量
----------	----------------------

4、输出每秒创建的进程数的进程统计信息

```
[root@cnetos5 ~]# sar -c
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM      proc/s
12:10:01 AM          0.05
12:20:01 AM          0.03
12:30:01 AM          0.03
Average:          0.03
```

输出项说明:

proc/s	每秒钟创建的进程数
--------	-----------

5、输出网络设备状态的统计信息

```
[root@cnetos5 ~]# sar -n DEV |grep eth0
12:00:01 AM      IFACE      rxpck/s      txpck/s      rxbyt/s      txbyt/s      rxcmp/s      txcmp/s      rxmcsst/s
12:10:01 AM      eth0         0.59         0.92        41.57        893.98          0.00          0.00          0.00
12:20:01 AM      eth0         0.55         0.88        37.50        859.56          0.00          0.00          0.00
12:30:01 AM      eth0         0.55         0.86        38.17        871.98          0.00          0.00          0.00
Average:         eth0         0.29         0.42        21.05        379.29          0.00          0.00          0.00
```

输出项说明:

IFACE	网络设备名
rxpck/s	每秒接收的包总数
txpck/s	每秒传输的包总数
rxbyt/s	每秒接收的字节 (byte) 总数
txbyt/s	每秒传输的字节 (byte) 总数
rxcmp/s	每秒接收压缩包的总数
txcmp/s	每秒传输压缩包的总数
rxmcsst/s	每秒接收的多播 (multicast) 包的总数

6、输出网络设备状态的统计信息（查看网络设备故障）

```
[root@cnetos5 ~]# sar -n EDEV |egrep 'eth0|IFACE'
12:00:01 AM      IFACE      rxerr/s      txerr/s      coll/s      rxdrop/s      txdrop/s      txcarr/s      rxfram/s      rxfifo/s      txfifo/s
12:10:01 AM      eth0         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
12:20:01 AM      eth0         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
12:30:01 AM      eth0         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
Average:         eth0         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
```

输出项说明:

IFACE	网络设备名
rxerr/s	每秒接收的坏包总数
txerr/s	传输包时每秒发生错误的总数
coll/s	传输包时每秒发生冲突 (collision) 的总数
rxdrop/s	接收包时, 由于缺乏缓存, 每秒丢弃 (drop) 包的数量
txdrop/s	传输包时, 由于缺乏缓存, 每秒丢弃 (drop) 包的数量
txcarr/s	传输包时, 每秒发生的传输错误 (carrier-error) 的数量
rxfram/s	接收包时, 每秒发生帧校验错误 (frame alignment error) 的数量
rxfifo/s	接收包时, 每秒发生队列 (FIFO) 一错误的数量
txfifo/s	传输包时, 每秒发生队列 (FIFO) 一错误的数量

7、输出进程队列长度和平均负载状态统计信息

```
[root@cnetos5 ~]# sar -q
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM      runq-sz      plist-sz      ldavg-1      ldavg-5      ldavg-15
```

```
12:10:01 AM      0      85      0.02      0.01      0.00
12:20:01 AM      0      85      0.01      0.00      0.00
12:30:01 AM      0      85      0.03      0.01      0.00
Average:         0      85      0.01      0.00      0.00
```

输出项说明:

runq-sz	运行队列的长度（等待运行的进程数）
plist-sz	进程列表中进程（processes）和线程（threads）的数量
ldavg-1	最后1分钟的系统平均负载（System load average）
ldavg-5	过去5分钟的系统平均负载
ldavg-15	过去15分钟的系统平均负载

8、输出内存和交换空间的统计信息

```
[root@cnetos5 ~]# sar -r
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM kbmemfree kbmemused %memused kbbuffers kbcached kbswpfree kswpused %swpused kbswpcad
12:10:01 AM 262068 253408 49.16 43884 156456 1048568 0 0.00 0
12:20:01 AM 261572 253904 49.26 44580 156448 1048568 0 0.00 0
12:30:01 AM 260704 254772 49.42 45124 156472 1048568 0 0.00 0
Average: 259551 255925 49.65 46453 156470 1048568 0 0.00 0
```

输出项说明:

kbmemfree	可用的空闲内存数量，单位为 KB
kbmemused	已使用的内存数量（不包含内核使用的内存），单位为 KB
%memused	已使用内存的百分数
kbbuffers	内核缓冲区（buffer）使用的内存数量，单位为 KB
kbcached	内核高速缓存（cache）数据使用的内存数量，单位为 KB
kbswpfree	可用的空闲交换空间数量，单位为 KB
kswpused	已使用的交换空间数量，单位为 KB
%swpused	已使用交换空间的百分数
kbswpcad	交换空间的高速缓存使用的内存数量

9、输出内存页面的统计信息

```
[root@cnetos5 ~]# sar -R
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM frmpg/s bufpg/s campg/s
12:10:01 AM -0.10 0.23 0.01
12:20:01 AM -0.21 0.29 -0.00
12:30:01 AM -0.36 0.23 0.01
Average: -0.21 0.22 0.00
```

输出项说明:

frmpg/s	每秒系统中空闲的内存页面（memory page freed）数量
bufpg/s	每秒系统中用作缓冲区（buffer）的附加内存页面（additional memory page）数量
campg/s	每秒系统中高速缓存的附加内存页面（additional memory pages cached）数量

10、输出inode、文件和其他内核表的信息

```
[root@cnetos5 ~]# sar -v
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM dentunusd file-sz inode-sz super-sz %super-sz dquot-sz %dquot-sz rtsig-sz %rtsig-sz
12:10:01 AM 7253 576 5126 0 0.00 0 0.00 0 0.00
12:20:01 AM 7253 576 5126 0 0.00 0 0.00 0 0.00
12:30:01 AM 7253 576 5126 0 0.00 0 0.00 0 0.00
Average: 7253 589 5125 0 0.00 0 0.00 0 0.00
```

输出项说明:

dentunusd	目录高速缓存中未被使用的条目数量
file-sz	文件句柄（file handle）的使用数量
inode-sz	i节点句柄（inode handle）的使用数量
super-sz	由内核分配的超级块句柄（super block handle）数量
%super-sz	已分配的超级块句柄占总超级块句柄的百分比
dquot-sz	已经分配的磁盘限额条目数量
%dquot-sz	分配的磁盘限额条目数量占总磁盘限额条目的百分比
rtsig-sz	已排队的 RT 信号的数量
%rtsig-sz	已排队的 RT 信号占总 RT 信号的百分比

11、输出系统交换活动信息

```
[root@cnetos5 ~]# sar -w
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM cswch/s
12:10:01 AM 44.74
12:20:01 AM 44.41
12:30:01 AM 44.41
Average: 44.50
```

输出项说明：

cswch/s	每秒的系统上下文切换数量
---------	--------------

12、输出系统交换的统计信息

```
[root@cnetos5 ~]# sar -W
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM pswpin/s pswpout/s
12:10:01 AM 0.00 0.00
12:20:01 AM 0.00 0.00
12:30:01 AM 0.00 0.00
Average: 0.00 0.00
```

输出项说明：

pswpin/s	每秒系统换入的交换页面（swap page）数量
pswpout/s	每秒系统换出的交换页面（swap page）数量

13、输出TTY设备的活动信息

```
[root@cnetos5 ~]# sar -y
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

12:00:01 AM TTY rcvin/s xmtin/s framerr/s prtyerr/s brk/s overrun/s
12:10:01 AM 0 0.00 0.00 0.00 0.00 0.00 0.00
12:10:01 AM 1 0.00 0.00 0.00 0.00 0.00 0.00
12:20:01 AM 0 0.00 0.00 0.00 0.00 0.00 0.00
12:20:01 AM 1 0.00 0.00 0.00 0.00 0.00 0.00
12:30:01 AM 0 0.00 0.00 0.00 0.00 0.00 0.00
12:30:01 AM 1 0.00 0.00 0.00 0.00 0.00 0.00
.....
Average: 0 0.00 0.00 0.00 0.00 0.00 0.00
Average: 1 0.00 0.00 0.00 0.00 0.00 0.00
```

输出项说明：

TTY	TTY 串行设备号
rcvin/s	每秒接收的中断数量
xmtin/s	每秒传送的中断数量
framerr/s	每秒发生的帧错误数（frame error）量
prtyerr/s	每秒发生的奇偶校验错误（parity error）数量
brk/s	每秒发生的暂停（break）数量
ovrun/s	每秒发生的溢出错误（overrun error）数量

14、显示全面的累计统计信息

```
# sar -A
```

15、默认配置不提供的累计统计信息

```
[root@cnetos5 ~]# sar -d
Requested activities not available in file
[root@cnetos5 ~]# sar -x ALL
Requested activities not available in file
[root@cnetos5 ~]# sar -X ALL
Requested activities not available in file
```

1. 默认情况下，为了防止统计数据文件 `/var/log/sa/saDD` 迅速增大，`/usr/lib/sa/sadc` 没有记录每个块设备的统计信息。
2. 可以在 `-d -x -X` 参数后添加取样参数获得即时统计信息。
3. 带有 `-x -X` 选项的 `sar` 命令从来不能记录到二进制统计数据文件。

查看即时统计信息

1、使用取样选项查看即时统计信息

例如：每30秒取样一次，连续取样5次

```
# sar -n DEV 30 5
# sar -u 30 5
```

2、输出和读取统计信息文件

例如：

```
# sar -u 30 5 -o sar-dump-001
# sar -u -f sar-dump-001
```

3、输出每一个块设备的活动信息

```
# sar -dp 5 2
Linux 2.6.18-53.el5 (cnetos5) 01/22/2008

07:12:11 AM      DEV          tps   rd_sec/s   wr_sec/s  avgrq-sz  avgqu-sz   await    svctm     %util
07:12:16 AM      sda             0.40         0.00        17.56     44.00         0.00         1.00         1.00         0.04
07:12:16 AM      sdb             0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00

07:12:16 AM      DEV          tps   rd_sec/s   wr_sec/s  avgrq-sz  avgqu-sz   await    svctm     %util
07:12:21 AM      sda             0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
07:12:21 AM      sdb             0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00

Average:          DEV          tps   rd_sec/s   wr_sec/s  avgrq-sz  avgqu-sz   await    svctm     %util
Average:          sda             0.20         0.00         8.78     44.00         0.00         1.00         1.00         0.02
Average:          sdb             0.00         0.00         0.00         0.00         0.00         0.00         0.00         0.00
```

输出项说明：

DEV	正在监视的块设备
tps	每秒钟物理设备的 I/O 传输总量
rd_sec/s	每秒从设备读取的扇区（sector）数量
wr_sec/s	每秒向设备写入的扇区（sector）数量
avgrq-sz	发给设备请求的平均扇区数
avgqu-sz	发给设备请求的平均队列长度
await	设备 I/O 请求的平均等待时间（单位为毫秒）
svctm	设备 I/O 请求的平均服务时间（单位为毫秒）
%util	在 I/O 请求发送到设备期间，占用 CPU 时间的百分比。用于体现设备的带宽利用率。

1. `avgqu-sz` 的值较低时，设备的利用率较高。
2. 当 `%util` 的值接近 100% 时，表示设备带宽已经占满。

要判断一个系统瓶颈问题，有时需要几个 `sar` 命令选项结合起来使用，例如：

- 怀疑CPU存在瓶颈，可用 `sar -u` 和 `sar -q deng` 等来查看

- 怀疑内存存在瓶颈，可用 `sar -B`、`sar -r` 和 `sar -W` 等来查看
- 怀疑I/O存在瓶颈，可用 `sar -b`、`sar -u` 和 `sar -d` 等来查看

iostat

iostat 的命令格式

`iostat` 用于输出CPU和磁盘I/O相关的统计信息。命令格式为：

```
iostat [ -c | -d ] [ -k | -m ] [ -t ] [ -V ] [ -x ] [ device [ ... ] | ALL ]
[ -p [ device | ALL ] ] [ interval [ count ] ]
```

其中：

- **interval**：为取样时间间隔
- **count**：为输出次数，若指定了取样时间间隔且省略此项，将不断产生统计信息

常用选项：

选项	说明
-c	仅显示CPU统计信息。与-d选项互斥。
-d	仅显示磁盘统计信息。与-c选项互斥。
-k	以KB为单位显示每秒的磁盘请求数。默认单位块。
-m	以MB为单位显示每秒的磁盘请求数。默认单位块。
-p {device ALL}	用于显示块设备及系统分区的统计信息。与-x选项互斥。
-t	在输出数据时，打印搜集数据的时间。
-V	打印版本号信息。
-x	输出扩展信息。

iostat 使用举例

下面给出几个例子：

```
# 显示一条包括所有的CPU和设备吞吐率的统计信息
# iostat
Linux 2.6.18-53.el5 (cnetos5) 01/21/2008

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.10    0.04   0.37    0.07    0.00   99.42

Device:            tps    Blk_read/s    Blk_wrtn/s    Blk_read    Blk_wrtn
sda                  1.44         16.79         10.58      800430      504340
sdb                   0.01          0.07          0.00        3314         8
sdc                   0.86          8.56          0.00      407892        24

# 每隔5秒显示一次设备吞吐率的统计信息（单位为 块/s）
# iostat -d 5

# 每隔5秒显示一次设备吞吐率的统计信息（单位为 KB/s），共输出3次
# iostat -dk 5 3

# 每隔2秒显示一次 sda 及上面所有分区的统计信息，共输出5次
# iostat -p sda 2 5

# 每隔2秒显示一次 sda 和 sdb 两个设备的扩展统计信息，共输出6次
# iostat -x sda sdb 2 6
Linux 2.6.18-53.el5 (cnetos5) 01/21/2008

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.10    0.04   0.37    0.07    0.00   99.42

Device:    rrqm/s   wrqm/s   r/s    w/s   rsec/s   wsec/s  avgrq-sz  avgqu-sz   await   svctm   %util
sda         0.17     0.84   0.96   0.47   16.67    10.56    19.01     0.01     7.11    1.25    0.18
sdb         0.00     0.00   0.01   0.00    0.07     0.00     5.16     0.00     0.22    0.19    0.00
.....
```


iostat 的输出项说明

avg-cpu 部分输出项说明:

%user	在用户级别运行所使用的 CPU 的百分比。
%nice	nice 操作所使用的 CPU 的百分比。
%system	在核心级别 (kernel) 运行所使用 CPU 的百分比。
%iowait	CPU 等待硬件 I/O 所占用 CPU 的百分比。
%steal	当管理程序 (hypervisor) 为另一个虚拟进程提供服务而等待虚拟 CPU 的百分比。
%idle	CPU 空闲时间的百分比。

Device 部分基本输出项说明:

tps	每秒钟物理设备的 I/O 传输总量。
Blk_read	读入的数据总量, 单位为块。
Blk_wrtn	写入的数据总量, 单位为块。
kB_read	读入的数据总量, 单位为KB。
kB_wrtn	写入的数据总量, 单位为KB。
MB_read	读入的数据总量, 单位为MB。
MB_wrtn	写入的数据总量, 单位为MB。
Blk_read/s	每秒从驱动器读入的数据量, 单位为 块/s。
Blk_wrtn/s	每秒向驱动器写入的数据量, 单位为 块/s。
kB_read/s	每秒从驱动器读入的数据量, 单位为KB/s。
kB_wrtn/s	每秒向驱动器写入的数据量, 单位为KB/s。
MB_read/s	每秒从驱动器读入的数据量, 单位为MB/s。
MB_wrtn/s	每秒向驱动器写入的数据量, 单位为MB/s。

Device 部分扩展输出项说明:

rrqm/s	将读入请求合并后, 每秒发送到设备的读入请求数。
wrqm/s	将写入请求合并后, 每秒发送到设备的写入请求数。
r/s	每秒发送到设备的读入请求数。
w/s	每秒发送到设备的写入请求数。
rsec/s	每秒从设备读入的扇区数。
wsec/s	每秒向设备写入的扇区数。
rkB/s	每秒从设备读入的数据量, 单位为 KB/s。
wkB/s	每秒向设备写入的数据量, 单位为 KB/s。
rMB/s	每秒从设备读入的数据量, 单位为 MB/s。
wMB/s	每秒向设备写入的数据量, 单位为 MB/s。
avgrq-sz	发送到设备的请求的平均大小, 单位为扇区。
avgqu-sz	发送到设备的请求的平均队列长度。
await	I/O请求平均执行时间。包括发送请求和执行的时间。单位为毫秒。
svctm	发送到设备的I/O请求的平均执行时间。单位为毫秒。
%util	在I/O请求发送到设备期间, 占用CPU时间的百分比。用于显示设备的带宽利用率。当这个值接近100%时, 表示设备带宽已经占满。

mpstat

mpstat 的命令格式

mpstat 输出每一个 CPU 的运行状况, 为多处理器系统中的 CPU 利用率提供统计信息。命令格式为:

```
mpstat [-P { cpu | ALL } ] [-V ] [ interval [ count ] ]
```

其中:

- **interval**：为取样时间间隔。指定0则输出自系统启动后的一个统计信息。
- **count**：为输出次数。若指定了取样时间间隔且省略此项，将不断产生统计信息。

常用选项：

选项	说明
-P {cpu ALL}	指定 CPU。用 CPU-ID 指定，CPU-ID 是从0开始的，即第一个CPU为0。ALL 表示所有CPU。
-V	输出版本号信息。

mpstat 使用举例

下面给出几个例子：

```
# 输出所有 CPU 使用情况的统计信息。
# mpstat
Linux 2.6.18-53.el5 (cnetos5) 01/21/2008

10:39:06 AM CPU %user %nice %sys %iowait %irq %soft %steal %idle intr/s
10:39:06 AM all 0.10 0.04 0.31 0.06 0.04 0.01 0.00 99.45 1012.99

# 输出第一个 CPU 使用情况的统计信息。
# mpstat -P 0
Linux 2.6.18-53.el5 (cnetos5) 01/21/2008

10:41:03 AM CPU %user %nice %sys %iowait %irq %soft %steal %idle intr/s
10:41:03 AM 0 0.09 0.02 0.40 0.09 0.08 0.01 0.00 99.32 1012.79

# 每隔2秒输出所有CPU的统计信息，共输出5次。
# mpstat 2 5

# 每隔2秒输出一共所有CPU的统计信息，共输出5次。
# mpstat -P ALL 2 5

# 每隔2秒输出一共第二个CPU的统计信息，共输出5次。
# mpstat -P 1 2 5
```

mpstat 输出项说明

CPU	在多CPU系统里，每个CPU有一个ID号，第一个CPU为0。all表示统计信息为所有CPU的平均值。
%user	显示在用户级别运行所占用CPU总时间的百分比。
%nice	显示在用户级别，用于nice操作，所占用CPU总时间的百分比。
%sys	显示在kernel级别运行所占用CPU总时间的百分比。注意：这个值并不包括服务中断和softirq。
%iowait	显示用于等待I/O操作时，占用CPU总时间的百分比。
%irq	显示用于中断操作，占用CPU总时间的百分比。
%soft	显示用于softirq操作，占用CPU总时间的百分比。
%steal	管理程序（hypervisor）为另一个虚拟进程提供服务而等待虚拟 CPU 的百分比。
%idle	显示CPU在空闲状态，占用CPU总时间的百分比。
intr/s	显示CPU每秒接收到的中断总数。

参考

- 在UNIX5.0.5 系统中巧用sar 命令进行系统维护 [<http://www.proxyservice.net/index.php?q=aHR0cDovL3d3dy5nYnVuaXguY29tL2h0bWxkYXRhLzIwMDRfMTIvMS85L2FydGljbGVfOITYxXzEuaHRtbA%3D%3D>]
- 系统状态查看工具Sysstat [<http://www.proxyservice.net/index.php?q=aHR0cDovL3d3dy5saW51eHNpci5vcmcvbwFpbi8%2FcT1ub2RlZlXM%3D%3D>]
- 显示源文件
- 登录