

Using Text Processing Tools
and other utilities

To Nicely get the system resources
utilization in printable format

Prepared by Mohamed Toure

Introduction

- Importance of Performance and Scalability of the server.
- Several Tools available either free or proprietary
- Usage of sar command to collect system resources utilization
- Process the data to get a tabulated format for gnuplot
- Use gnuplot to generate a postscript file
- Use gimp to interactively save the output of the postscript to .jpeg format

Sar = program that collects, stores, reports system resources utilization at a specific time.

Syntax ---> look at man or info sar for more information

Want to collect info. about cpu, memory utilization, and network flow

Command I use to collect and store information about the system every 5s for the next 5 minutes:

```
$sar -o filename.dat 5 60
```

Extract the information needed in this binary format filename.dat

CPU

```
$sar -f filename.dat -u > cpu.txt
```

Memory

```
$sar -f filename.dat -r > mem.txt
```

Net

```
$sar -f filename.dat -n DEV > net.txt
```

cpu.txt

Linux 2.4.20-13.9 (server.oscc.net) 09/16/2003

01:31:33 PM	CPU	%user	%nice	%system	%idle
01:31:38 PM	all	0.40	0.00	1.20	98.40
01:31:43 PM	all	2.00	0.00	1.00	97.00
01:31:48 PM	all	1.40	0.00	1.00	97.60
01:31:53 PM	all	1.00	0.00	2.20	96.80
01:31:58 PM	all	4.40	0.00	3.60	92.00
01:32:03 PM	all	3.00	0.00	3.00	94.00
01:32:08 PM	all	0.60	0.00	1.00	98.40

```
sar -f filename.dat -u | head
```



```
sar -f filename.dat -r | head
```



mem.txt

Linux 2.4.20-13.9 (server.oscc.net) 09/16/2003

01:31:33 PM	kbmemfree	kbmemused	%memused	kbmemshrd	kbbuffers	kbcached	kbswpfree	kbswpused	%swpused
01:31:38 PM	4904	121024	96.11	0	3312	48488	386752	30896	7.40
01:31:43 PM	4908	121020	96.10	0	3328	48488	386752	30896	7.40
01:31:48 PM	4908	121020	96.10	0	3348	48528	386752	30896	7.40
01:31:53 PM	3208	122720	97.45	0	3580	50404	386752	30896	7.40
01:31:58 PM	2172	123756	98.28	0	4156	50344	386676	30972	7.42
01:32:03 PM	2440	123488	98.06	0	4204	49128	386476	31172	7.46
01:32:08 PM	2452	123476	98.05	0	4180	49140	386440	31208	7.47

net.txt

Linux 2.4.20-13.9 (server.oscc.net) 09/16/2003

01:31:33 PM	IFACE	rxpck/s	txpck/s	rxbyt/s	txbyt/s	rxcmp/s	txcmp/s	rxmcs/s
01:31:38 PM	lo	7.60	7.60	519.40	519.40	0.00	0.00	0.00
01:31:38 PM	eth0	1.00	0.00	72.40	0.00	0.00	0.00	0.00
01:31:43 PM	lo	7.60	7.60	519.40	519.40	0.00	0.00	0.00
01:31:43 PM	eth0	0.80	0.00	108.00	0.00	0.00	0.00	0.00
01:31:48 PM	lo	7.60	7.60	519.40	519.40	0.00	0.00	0.00
01:31:48 PM	eth0	0.40	0.00	24.00	0.00	0.00	0.00	0.00
01:31:53 PM	lo	7.60	7.60	519.40	519.40	0.00	0.00	0.00

↑
`sar -f filename.dat -n DEV | head`

Process for cpu

```
split -l 3 cpu.txt
rm -r xaa xav
cat x* > a
rm -f x*
seq 0 5 295 > time
paste time a > cpusample
rm -f a
```

Process for memory

```
split -l 3 mem.txt
rm -f xaa xav
cat x* > a
rm -f x*
paste time a > memsample
```

Process for network flow

```
split -l 3 net.txt
rm -f xaa xbp -- (extreme files)
cat x* > a ; rm x*;
sort -k 3 a > aa
mv aa a;    split -l 60 a;
rm -f a xab;
sort -k 1 xaa > a; rm -f xaa;
paste time a > netsample
```

0	01:31:38 PM	all	0.40	0.00	1.20	98.40
5	01:31:43 PM	all	2.00	0.00	1.00	97.00
10	01:31:48 PM	all	1.40	0.00	1.00	97.60
15	01:31:53 PM	all	1.00	0.00	2.20	96.80
20	01:31:58 PM	all	4.40	0.00	3.60	92.00
25	01:32:03 PM	all	3.00	0.00	3.00	94.00
30	01:32:08 PM	all	0.60	0.00	1.00	98.40
35	01:32:13 PM	all	0.40	0.00	1.00	98.60
40	01:32:18 PM	all	1.60	0.00	1.00	97.40
45	01:32:23 PM	all	0.20	0.00	1.20	98.60

← cpusample

0	01:31:38 PM	4904	121024	96.11	0	3312	48488	386752	30896	7.40
5	01:31:43 PM	4908	121020	96.10	0	3328	48488	386752	30896	7.40
10	01:31:48 PM	4908	121020	96.10	0	3348	48528	386752	30896	7.40
15	01:31:53 PM	3208	122720	97.45	0	3580	50404	386752	30896	7.40
20	01:31:58 PM	2172	123756	98.28	0	4156	50344	386676	30972	7.42
25	01:32:03 PM	2440	123488	98.06	0	4204	49128	386476	31172	7.46
30	01:32:08 PM	2452	123476	98.05	0	4180	49140	386440	31208	7.47
35	01:32:13 PM	2456	123472	98.05	0	4088	49248	386360	31288	7.49
40	01:32:18 PM	2388	123540	98.10	0	3880	49464	386316	31332	7.50
45	01:32:23 PM	2380	123548	98.11	0	3880	49488	386236	31412	7.52

← memsample

0	01:31:38 PM	eth0	1.00	0.00	72.40	0.00	0.00	0.00	0.00
5	01:31:43 PM	eth0	0.80	0.00	108.00	0.00	0.00	0.00	0.00
10	01:31:48 PM	eth0	0.40	0.00	24.00	0.00	0.00	0.00	0.00
15	01:31:53 PM	eth0	1.20	0.00	128.40	0.00	0.00	0.00	0.00
20	01:31:58 PM	eth0	0.40	0.00	24.00	0.00	0.00	0.00	0.00
25	01:32:03 PM	eth0	0.60	0.00	36.00	0.00	0.00	0.00	0.00
30	01:32:08 PM	eth0	0.80	0.00	54.00	0.00	0.00	0.00	0.00
35	01:32:13 PM	eth0	0.80	0.00	61.20	0.00	0.00	0.00	0.00
40	01:32:18 PM	eth0	0.60	0.00	76.40	0.00	0.00	0.00	0.00
45	01:32:23 PM	eth0	1.20	0.00	128.40	0.00	0.00	0.00	0.00

← netsample

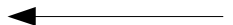
```
set terminal postscript eps color solid
set output "cpu.eps"
set key below
set title "cpu activity"
set ylabel "% of cpu used"
set xlabel "5s interval"
plot "cpusample" using 1:5 t "%user" with lines, "cpusample" using 1:7
t "%sys" with lines, "cpusample" using 1:8 t "%idle" with lines
```

gnuplot_cpu



```
set terminal postscript eps color solid
set output "mem.eps"
set key below
set title "mem activity"
set ylabel "amount of memory in KB"
set xlabel "5s interval"
plot [0:300] [0:131072] "memsample" using 1:4 t "kbmemfree" with
lines, "memsample" using 1:5 t "kbmemused" with lines,
"memsample" using 1:11 t "kbswapused" with lines
```

gnuplot_mem



```
set terminal postscript eps color solid
set output "net.eps"
set key below
set title "net activity booting 1 \n client(s)"
set ylabel "network flow in bytes per second"
set xlabel "5s interval"
plot "netsample" using 1:7 t "total # of bytes received/s" with lines,
"netsample" using 1:8 t "total # of bytes transmitted/s" with lines
```

gnuplot_net

