System Administration

Performance Monitoring

For a server, it is crucial to monitor the health of the machine

You need not only real time data collection and presentation but “offline” statistical analysis as well

Characteristics of a computer's health:

- Load
- Disk usage
- Process accounting

System Administration

Load

Load average measures the number of processes in the *run queue*, i.e. the number of processes waiting to use the CPU

Ideally, load should be 1 or less

Several ways to obtain load averages:

- **uptime, w, ps, top** all report load averages
- **cat** */proc/load*
- **tload, xload** present graphical displays of load

System Administration

Performance Monitoring

Load

Exercise

Run xload and start an intensive process

xload -update 1 -scale 2 &

ooimpress &

System Administration

Performance Monitoring

Disk Usage

Disks do no good if they are full

This is especially true for server processes.

If a log file cannot be written, some daemons may crash

- **du** gives a report of disk usage. It will descend all directories unless the **-s** option is given

- **df** gives a report of disk usage and the amount of free space available

df [*file* or *file system]*

System Administration

Process Accounting

THE most important thing to monitor is what is running on your computer

Processes are tasks being executed by your CPU

Each process is allocated space in the *process table* which tracks information the CPU needs to schedule and swap a task for CPU and memory usage

Information for processes currently in the process table can be retrieved with the **ps** and **top** commands

pstree is also very useful for following the process ownership

System Administration

Process Accounting

ps gives a variety of output depending on the options selected.

Most useful options

- **A** – show all processes
- **a** – show only processes connected with a tty
- **x** – show only processes disconnected from a tty
- **u** – display the effective user name
- **o** – user defined format

System Administration

Process Accounting

Example **ps** output:

```
<lister> ps xua

root         1  0.0  0.0  1320  484 ?        S    Jan13   0:04 init
edsall    5047  0.0  0.0  3876  564 pts/8    S    14:03   0:00 sleep 300
edsall    5066  0.0  0.0  2872  980 pts/10   R    14:07   0:00 ps xua
```

giving the following information

- Username
- Process ID (PID)
- Percent CPU being used
- Percent memory being used
- Virtual memory size
- Resident memory usage
- Associated TTY
- Current process status
- Date/time the process was started
- System tme used by the process
- Name of the command

State codes

First letter:

- R – running
- S – interruptible sleep
- D – uninterruptible sleep
- T – stopped
- Z -- “zombie”

Second letter:

- + -- foreground process
- s – session leader
- N – low priority (“nice”)
- < -- high priority (not nice)
- l -- multithreaded

System Administration

Performance Monitoring

Process Accounting

ps only gives a one-time snapshot of the processes in the process table

top gives process information which updates at regular intervals.

top also allows you to interactively sort the information being displayed

top presents other useful system information including the system load average and memory usage

System Administration

Performance Monitoring

Process Accounting

Example **top** output:

```
  8:05pm  up 8 days,  2:31,  1 user,  load average: 0.54, 0.55, 0.50
57 processes: 56 sleeping, 1 running, 0 zombie, 0 stopped
CPU states:  1.9% user,  2.3% system,  0.0% nice, 95.6% idle
Mem:  776780K av, 664392K used, 112388K free,  46804K shrd, 270344K buff
Swap: 530104K av,   4124K used, 525980K free                195596K cached

  PID    USER      PRI  NI  SIZE  RSS  SHARE STAT LIB %CPU %MEM   TIME COMMAND
29370 root       16   0  1028 1028   824 R       0  1.5  0.1   0:00 top
27682 nobody      2   0 15780  14M  4492 S       0  0.5  1.9   0:06 httpd
28409 nobody     14   0 14900  13M  4532 S       0  0.3  1.8   0:03 httpd
28895 nobody      1   0 15008  14M  4512 S       0  0.3  1.8   0:03 httpd
27666 nobody      1   0 15156  14M  4500 S       0  0.1  1.8   0:06 httpd
27687 nobody     17   0 16648  15M  4532 S       0  0.1  2.0   0:06 httpd
27829 nobody      7   0 14984  14M  4496 S       0  0.1  1.8   0:03 httpd
28211 nobody      1   0 14684  13M  4504 S       0  0.1  1.8   0:03 httpd
29341 root        1   0  1112 1112   832 S       0  0.1  0.1   0:00 in.rlogind
```

System Administration

Performance Monitoring

Process Accounting

Exercises

Run ps aux

ps aux

Run top

top

System Administration

/proc File System

On Linux systems (and many other operating systems), **ps** and **top** actually read their data from the */proc* file system

/proc is a mirror of the kernel configuration and what is currently memory resident

As a result we can not only read the state of the system but modify its configuration in real time

Information in */proc* is stored in files in a hierarchical manner based on the aspect of the system they describe

System Administration

/proc File System

Examples:

- loadavg- file containing System load average
- meminfo – file containing the number of total, used and free bytes of memory and swap area
- cpuinfo – file containing recognized processor parameters
- net – Directory containing descriptions about the network layer(s)

System Administration

/proc File System

Information on current processes is stored in directories corresponding to the PID of the process.

Example:

The **init** command, corresponding to PID = 1 has the following subdirectories in */proc/1*:

cmdline cwd environ exe fd maps mem mounts root stat statm status

# cat /proc/1/environ
HOME=/ TERM=linux

System Administration

Monitoring the health of your computer can be enhanced by collecting data and analyzing later to look for trends.

There are many ways to do this:

- Write your own packages
- Use an Open Source package
- Use a commercial package

Examples:

- Tivoli
- Ecotools
- Candle

System Administration

Before writing your own package, look to see if someone else has already done the work for you.

Many Open Source packages available

- psacct
- sysstat
- Hobbit

System Administration

psacct

The **psacct** package provides simple process accounting and process summary information.

Data collection is initiated with the command **accton**

These data can later be analyzed with the commands **ac, lastcomm** and **sa**, the most useful of which is **sa**

System Administration

Performance Analysis

sysstat

The **sysstat** package, which comes by default with most recent Red Hat Linux distributions, collects an enormous amount of data on system resources including

- paging activity
- interrupts
- network activity
- memory

System Administration

Performance Analysis

sysstat

Several times a day, cron runs the commands in the */etc/cron.d/sysstat* crontab to collects the data.

The same crontab analyzes the collected data once a day.

Results can be found in */var/log/sa/* and reports can be generated with the **sar** command.

System Administration

Performance Analysis

sysstat

Exercise

Look at the contents of */var/log/sa*

ls -l /var/log/sa

Run the sar command to get a report

/usr/bin/sar -A | more

System Administration

Hobbit

Hobbit is an open source pacakge available to most non-profit groups for free.

Hobbit not only generates history graphs, it also, and most prominently, is used as a real-time performance monitor

IT Services uses Hobbit:

http://kosh.its.iastate.edu

More information can be found at

http://hobbitmon.sourceforge.net/

System Administration

Log Files

Many processes running on the system generate log files

Linux stores all of its system log files in a special directory - */var*

Log files can quickly fill the partition on which they are stored

Tip – create a separate partition for */var*

System Administration

Log Files

One important system logging process – **syslogd.** Many processes make a syslog call to write information to the system log

```
void openlog(char *ident, int option, int facility);

void syslog(int priority, char *format, ...);

void closelog(void);
```

The processes are always writing. Whether you wish to see it depends on how you configure **syslogd**

System Administration

Log Files

syslogd is configured with the file */etc/syslog.conf*

Each line consists of a selector and an action for that selector

```
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;cron.none    /var/log/messages
```

System Administration

Log Files

```
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;cron.none    /var/log/messages
```

The *selector* identifies the *facility* which may request logging and the *priority* level at which it may want logging. See the correlation?

```
void openlog(char *ident, int option, int facility);

void syslog(int priority, char *format, ...);
```

For each selector an action is specified. In most cases, the action is the name of a file to which the messages will be logged

System Administration

Remote Log Files

syslogd can log its information to a remote server:

. @remotelog1.iastate.edu
. @remotelog2.iastate.edu

In this case, **syslogd** will send everything to the **syslog** daemon on the remote servers *remotelog1* and *remotelog2*

If the bad guys tamper with log files on your server, you have a pristine copy elsewhere that might show their break-in

System Administration

Monitoring Your Log Files

logwatch

- Daily analysis of logfiles – can be e-mailed to you
- Global and local configuration
- Can analyze on a per-service basis

In */etc/log.d*:

Scripts for services and filters they use

Config files for services

System Administration

Monitoring Your Log Files

logwatch

######## LogWatch 4.3.2 (02/18/03)########
Processing Initiated: Mon Mar 26 11:33:53 2007
Date Range Processed: yesterday
Detail Level of Output: 0
Logfiles for Host: webmail-1.iastate.edu
##

---------- Connections (secure-log) Begin ----------

Connections:
Service shell:
129.186.140.67: 4 Time(s)

---------- Connections (secure-log) End ----------

System Administration

Reining in Your Log Files

logrotate

Wouldn't it be nice if we could delete old log files or move them elsewhere? Otherwise, our disks may fill.

logrotate can:

- Back-up
- Compress
- Mail

logfiles subject to a variety of constraints you specify

System Administration

Reining in Your Log Files

logrotate

logrotate is configured via a set of configuration files

Generally one main config file (*/etc/logrotate.conf)* is used to “include” other config files in */etc/logrotate.d*

The config files contain the name of the file to be acted on and directives which specify the actions to be taken, how often to take these actions, etc

System Administration

Reining in Your Log Files

logrotate

```
# Logrotate file for psacct RPM

/var/account/pacct {
prerotate
        /usr/sbin/accton
endscript
        compress
        notifempty
        daily
        rotate 31
        create 0600 root root
postrotate
        /usr/sbin/accton /var/account/pacct
endscript
}
```