System Monitoring

René Serral-Gracià$^1$

$^1$Universitat Politècnica de Catalunya (UPC)

February 21, 2022
Lectures

1. System administration introduction
2. Operating System installation
3. User management
4. Application management
5. System monitoring
6. Filesystem Maintenance
7. Local services
8. Network services
9. Security and Protection
10. Virtualization
1. Introduction
   - Goals

2. System Monitoring

3. Process management

4. User monitoring

5. Network monitoring
Goals

Knowledge

- Monitoring commands
- Meaning of the different signals

Abilities

- Obtain information about the system’s behavior
  - CPU activity
  - Memory activity
  - Disk activity
- Process status monitoring
  - Priority change
  - Stop and Continue processes
Outline

1. Introduction

2. System Monitoring
   - CPU
   - Memory
   - Disk
   - Network
   - Users
   - Other monitoring tasks

3. Process management

4. User monitoring

5. Network monitoring
System Monitoring

Why monitoring?
- Proactively control the resource status
- Control service status
- Security

Actions
- Automatic
- Manual
System Monitoring

What do we monitor?

- CPU
- Memory
- I/O
- Network
- Users
- Services
- Logs
Other factors

- When a resource is monitored?
- Who do we contact in case there is a problem?
- Which is the criteria to notify a warning?
- And for a critical issue?
CPU Activity

**Monitoring**
- Inactive processors
- Monopolized processors
  - By a single process
  - By a single user

**Tools**
uptime, top, ps
Memory activity

**Monitoring**
- Lack of memory
- Memory monopolization
  - By a single process
  - By a single user
- Swap

**Tools**
- `free`, `vmstat`, `top`
I/O Activity

Monitoring

- Filesystem
- Anomalous I/O activity
- Virtual memory
  - Excessive Pagination
  - Free Space

Tools

vmstat, df, iostat
Network Activity

**Monitoring**
- Bandwidth
- Local and remote services
- Incoming/outgoing connections
- Traffic profile

**Tools**
- `ip -s -d`
- `netstat`
- `tcpdump`
- `nmap`
- Logs del sistema
User activity

Monitoring

- Active sessions
  - Locally
  - Remotely
- Connected users
- What are they doing?

Tools

w, last, finger, fuser, lsof
Other monitoring tasks

Service and server activity

- Web server load
- E-mail queues
  - Input
  - Output
- Printer queues

Registry files (logs)

- System errors
- Anomalous activity (security)
Outline

1. Introduction
2. System Monitoring
3. Process management
   - Priority change
   - Signals
4. User monitoring
5. Network monitoring

R. Serral-Gracià, et. al
Tasks and process management

Process identification
- Who is the owner of the process?
- Which is its purpose?
  - Is it important?
  - Is it an attack? ... or an error?

Actions on the process
- Priority changes
- Stop and reactivation of a process
- Killing a process
Priority change

- When executing the process
  - `nice +10` command...
- Once it is already running
  - `renice +10 <pid>`
- Only root can increase the priority

Negative values indicate higher priorities
Some advise

High priority Shell
- Higher priority than swap
  - Allows a more efficient detection/solving of a memory issue
- The child processes inherit the priority of the parent

Relative priorities
- Priority is a relative term
- Not useful if all the processes have high priority
# Sending signals to processes

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>kill &lt;signal&gt; &lt;pid&gt;</code></td>
<td>- <strong>KILL</strong>: immediately stops the process</td>
</tr>
<tr>
<td></td>
<td>- <strong>TERM</strong>: ask a process to gracefully finish (kill, by default)</td>
</tr>
<tr>
<td></td>
<td>- <strong>INT</strong>: interrupt a process (kill, by default)</td>
</tr>
<tr>
<td></td>
<td>- <strong>STOP</strong>: stop a process</td>
</tr>
<tr>
<td></td>
<td>- Do not allow it to be enqueued in the ready queue</td>
</tr>
<tr>
<td></td>
<td>- <strong>CONT</strong>: reactivate the selected process</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>killall &lt;signal&gt; &lt;command name&gt;</code></td>
<td>Sends the signal to <strong>ALL</strong> the processes matching the name</td>
</tr>
</tbody>
</table>
Outline

1. Introduction
2. System Monitoring
3. Process management
4. User monitoring
   • Examples
5. Network monitoring
User monitoring

User activity

- `w [user]`
  - List of connected users and the command being executed
  - Given a username, it lists his/her connections

- `last [user]`
  - Lists the last established connections... either finished or not

- `finger [user]`
  - Lists all the sessions or the ones belonging to an user
File monitoring

**File activity monitoring**

- `fuser <filename>`
  - Identifies the processes being used by a file
- `lsof [filename | directory name]`
  - Lists open files
Disk activity

**Used space**
- `du [filename | directory name]`
- Indicates used space per directory (including subdirs)

**Free space**
- `df [filename | directory name]`
- Free space on each partition

**I/O activity**
- `vmstat`
- `iostat`
### Example: `top`

```
top - 10:01:50 up 4 days, 8:40, 5 users, load average: 1.77, 1.51, 1.56
Tasks: 281 total, 1 running, 279 sleeping, 0 stopped, 1 zombie
%Cpu0 : 13.2 us, 3.3 sy, 0.0 ni, 82.9 id, 0.3 wa, 0.0 hi, 0.3 si, 0.0 st
%Cpu1 : 10.2 us, 1.5 sy, 0.0 ni, 87.3 id, 0.3 wa, 0.0 hi, 0.6 si, 0.0 st
%Cpu2 : 12.7 us, 1.5 sy, 0.0 ni, 84.6 id, 0.6 wa, 0.0 hi, 0.6 si, 0.0 st
%Cpu3 : 16.3 us, 1.7 sy, 0.0 ni, 81.6 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
KiB Mem : 16314076 total, 5436464 free, 3590272 used, 7287340 buff/cache
KiB Swap: 16360444 total, 16318936 free, 41508 used, 10859404 avail Mem

<table>
<thead>
<tr>
<th>PID</th>
<th>USER</th>
<th>PR</th>
<th>NI</th>
<th>VIRT</th>
<th>RES</th>
<th>SHR</th>
<th>S</th>
<th>%CPU</th>
<th>%MEM</th>
<th>TIME+</th>
<th>COMMAND</th>
</tr>
</thead>
<tbody>
<tr>
<td>17901</td>
<td>rserral</td>
<td>1</td>
<td>0</td>
<td>1429512</td>
<td>265436</td>
<td>126648</td>
<td>S</td>
<td>16.5</td>
<td>1.6</td>
<td>4:51.75</td>
<td>slack</td>
</tr>
<tr>
<td>17115</td>
<td>rserral</td>
<td>5</td>
<td>0</td>
<td>2640856</td>
<td>349772</td>
<td>137352</td>
<td>S</td>
<td>9.6</td>
<td>2.1</td>
<td>5:00.66</td>
<td>gnome-shell</td>
</tr>
<tr>
<td>17340</td>
<td>rserral</td>
<td>1</td>
<td>0</td>
<td>1667320</td>
<td>157220</td>
<td>91880</td>
<td>S</td>
<td>4.6</td>
<td>1.0</td>
<td>0:33.14</td>
<td>slack</td>
</tr>
<tr>
<td>444</td>
<td>root</td>
<td>-51</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>2.0</td>
<td>0.0</td>
<td>17:17.13</td>
<td>irq/17-i2c_desi</td>
</tr>
<tr>
<td>17133</td>
<td>rserral</td>
<td>1</td>
<td>0</td>
<td>562520</td>
<td>236400</td>
<td>201880</td>
<td>S</td>
<td>1.7</td>
<td>1.4</td>
<td>0:51.53</td>
<td>Xwayland</td>
</tr>
<tr>
<td>17343</td>
<td>rserral</td>
<td>1</td>
<td>0</td>
<td>471912</td>
<td>48636</td>
<td>30472</td>
<td>S</td>
<td>1.7</td>
<td>0.3</td>
<td>0:00.92</td>
<td>python2</td>
</tr>
<tr>
<td>18210</td>
<td>rserral</td>
<td>1</td>
<td>0</td>
<td>3021200</td>
<td>577976</td>
<td>253764</td>
<td>S</td>
<td>1.3</td>
<td>3.5</td>
<td>4:42.75</td>
<td>firefox</td>
</tr>
<tr>
<td>286</td>
<td>root</td>
<td>-51</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>1.0</td>
<td>0.0</td>
<td>8:01.12</td>
<td>irq/17-idma64.1</td>
</tr>
<tr>
<td>20211</td>
<td>rserral</td>
<td>6</td>
<td>0</td>
<td>46988</td>
<td>3904</td>
<td>3044</td>
<td>R</td>
<td>1.0</td>
<td>0.0</td>
<td>0:00.33</td>
<td>top</td>
</tr>
<tr>
<td>19472</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.7</td>
<td>0.0</td>
<td>0:11.71</td>
<td>kworker/u8:2</td>
</tr>
<tr>
<td>6</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>13:19.49</td>
<td>ksoftirqd/0</td>
</tr>
<tr>
<td>7</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>2:02.42</td>
<td>rcu_preempt</td>
</tr>
<tr>
<td>17</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>13:23.78</td>
<td>ksoftirqd/1</td>
</tr>
<tr>
<td>23</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>14:30.76</td>
<td>ksoftirqd/2</td>
</tr>
<tr>
<td>29</td>
<td>root</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>16:11.32</td>
<td>ksoftirqd/3</td>
</tr>
<tr>
<td>445</td>
<td>root</td>
<td>-51</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>3:06.32</td>
<td>irq/51-DLL075B:</td>
</tr>
<tr>
<td>621</td>
<td>message+</td>
<td>1</td>
<td>0</td>
<td>48732</td>
<td>6700</td>
<td>3072</td>
<td>S</td>
<td>0.3</td>
<td>0.0</td>
<td>4:09.41</td>
<td>dbus-daemon</td>
</tr>
</tbody>
</table>
```
Exercise

We have a database server with 1 CPU (and hyperthreading)

- Which is the problem present on the server if any?
- Which actions would you take?

```
top - 09:38:09 up 1 day, 18:29,  6 users, load average: 4.08, 4.93, 4.39
Tasks: 425 total, 12 running, 413 sleeping, 0 stopped, 0 zombie
%Cpu(s): 91.0 us,  6.8 sy,  0.9 ni,  1.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 16355660 total, 125088 free, 6559812 used, 9670760 buff/cache
KiB Swap: 33691644 total, 33689476 free, 2168 used. 8286212 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
4102 pcomp 20  0 2920500 1.029g 98884 S 46.1  6.6 103:32.24 firefox-esr
12802 pcomp 20  0 102332 68188 14164 R 30.6  0.4  0:00.93 chrome-bg-proc
12818 pcomp 20  0 80856 51980 17732 R 22.4  0.3  0:00.68 chrome-bg-proc
12835 pcomp 20  0 88840 49892 10524 R 17.1  0.3  0:00.52 chrome-bg-proc
3947 pcomp 20  0 2207552 505540 69276 S 14.5  3.1 49:25.10 gnome-shell
12861 pcomp 20  0 75972 37808 10480 R 12.2  0.2  0:00.37 chrome-bg-proc
12834 pcomp 20  0 65460 25816 8488 R 11.2  0.2  0:00.34 chrome-bg-proc
12873 pcomp 20  0 69680 32032 10508 R  9.2  0.2  0:00.28 chrome-bg-proc
12858 pcomp 20  0 59056 18824 8452 R  7.6  0.1  0:00.23 chrome-bg-proc
12833 pcomp 20  0 14312 11436 1356 R  6.9  0.1  0:00.21 mysqld
```
Exercise

We have a server with 32 logical CPU
- Which is the problem present on the server?
- How would you solve it?

```bash
top - 16:31:15 up 3:04, 20 users, load average: 29.76, 17.88, 10.19
Tasks: 1016 total, 2 running, 1013 sleeping, 1 stopped, 0 zombie
Cpu(s):  2.5%us, 1.2%sy, 0.0%ni, 86.8%id, 9.4%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 65969572k total, 33193236k used, 32776336k free, 8656k buffers
Swap: 16777208k total, 7635416k used, 9141792k free, 31292k cached
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
3164 tst8 20 0 23.1g 21g 584 R 100.0 34.1 7:44.76 emacs
4576 tst8 20 0 104m 1080 476 S 53.3 0.0 2:17.90 genarray.sh
1010 root 20 0 0 0 0 D 2.0 0.0 2:07.06 kmirrord
3342 g_users 20 0 15868 1528 476 R 1.0 0.0 1:43.80 top
168 root 20 0 0 0 0 S 0.3 0.0 0:02.09 events/21
2568 tst6 20 0 101m 376 240 S 0.3 0.0 1:27.30 sshd
```
Outline

1. Introduction
2. System Monitoring
3. Process management
4. User monitoring
5. Network monitoring
Network monitoring

Integrated systems

- Centralized information for various servers
  - Resources
  - Services
  - Uptime
  - Connectivity
  - Logs
- Ease the issue detection
- NagiOS, Splunk
Example: Nagios XI

Image source: http://www.nagios.com/
Personal homework

- Backup tools
  - dump
  - tar
  - gzip, bzip2, zip, rar, partimage, Norton Ghost