NSF/IUCRC CAC PROJECT

INTEGRATED VISUALIZING, MONITORING, AND MANAGING HPC SYSTEMS

Jie Li Doctoral Student, TTU 07/17/2020

Advisors:

Mr. Jon Hass, SW Architect, Dell Inc.

Dr. Alan Sill, Managing Director, HPCC, TTU

Dr. Yong Chen, Associate Professor, CS Dept, TTU

Dr. Tommy Dang, Assistant Professor, CS Dept, TTU

- Revisit PAPI (monitoring system status through hardware performance counters)
- Operating System Monitoring Tool

- Interface for using low-level performance counters
- Monitoring performance events, such as cache hit rats, TLB misses, floating-point operations, etc.
- Limited hardware counters (multiplexing), overflow (userdefined handlers)
- Identify the events to track
- Deriving metrics from the raw data

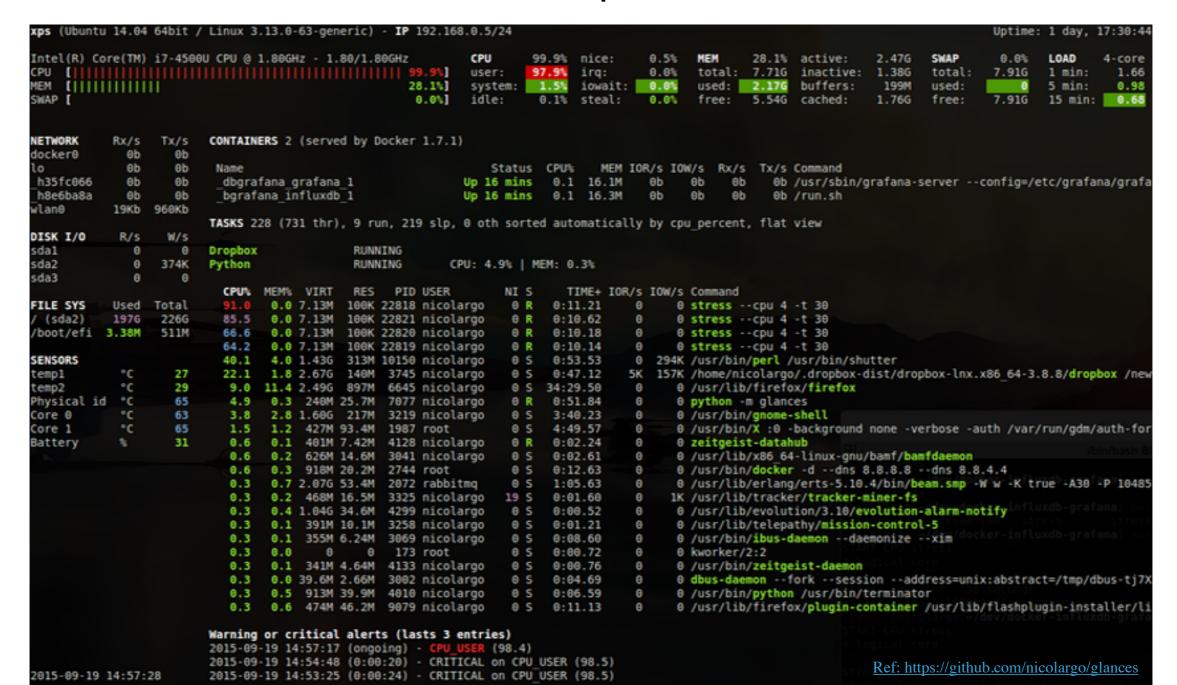
More straightforward way to monitor the operating system?

As easy as using top, vmstat, iostat etc.

- /proc process information pseudo-file system in Linux
- Contains runtime system information, control and information centre for the kernel
- top, vmstat, iostat etc. call to files in this directory

```
/proc/cpuinfo: processor: type, model, performance etc.
/proc/meminfo: memory usage: physical and swap.
/proc/diskstats: I/O statistic
/proc/net: network status
```

- A cross-platform system monitoring tool
- Uses the psutil library to get information from system
- Works in client/server model, provides a RESTful JSON API



Monitoring statistic of every CPU

total: total CPU usage

user: percent time spent in user space

system: percent time spent in kernel space

iowait: percent time spent by the CPU waiting for I/O operations to complete

```
"percpu": [
       "cpu_number": 0,
       "quest": 0.0,
       "guest_nice": 0.0,
       "idle": 89.9,
       "iowait": 0.4,
       "irq": 0.0,
       "key": "cpu_number",
       "nice": 0.0,
       "softirg": 0.1,
       "steal": 0.0,
       "system": 1.3,
       "total": 10.1,
       "user": 8.3
       "cpu_number": 1,
       "guest": 0.0,
       "guest_nice": 0.0,
       "idle": 89.7,
       "iowait": 0.0,
       "irg": 0.0,
       "key": "cpu_number",
       "nice": 0.0,
       "softirg": 0.0,
       "steal": 0.0,
       "system": 1.5,
       "total": 10.3,
       "user": 8.8
```

Monitoring the network interface bit rate

Rx: receive bit rate

Tx: Transmit bit rate

Monitoring the disk I/O throughput write_bytes: write bytes per second

Read_bytes: read bytes per second

Monitoring the used and total file system disk space

Monitoring the temperature of each core

- CPU Usage: ~12.5% when fetching data
- Memory Usage: ~25.43 MiB / 23.37 GiB, 0.11%
- Net I/O: ~117KB/query

DEMO

