## Finding out to which docker container a process belongs to · Max Claus Nunes

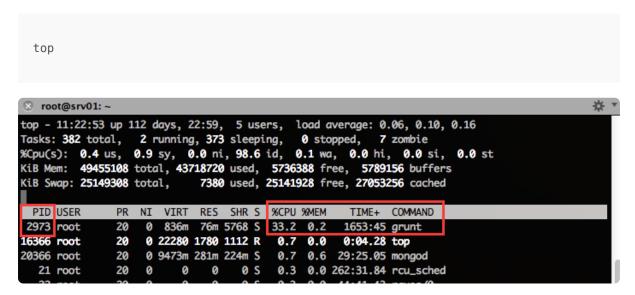
找到进程属于哪一个docker

## Finding out to which docker container a process belongs to

19 Oct 2014

Linux containers run all over the same Linux Kernel. It means all process running inside each container will be visible in the host machine. Due that, how do we know if a process is running in the host or in a container? And how do we discover in each container that process is running? Lets solve that based a real example.

This week I was running some tests in one of the Bravi's server and I realized there was a weird grunt process using 30% of CPU. That was weird because should not exists a grunt process running in a staging/production environment and neither should be using so much from the CPU.



Running this command ps -axfo pid, uname, cmd I got the full list of process running in that machine. In this list I could find that process with id equals to 2973 and its top parent process.

```
2871 root
                   \_ /bin/bash /docker.sh
2911 root

\_ mongod --fork --dbpath /data/db --logpath /

2945 root
                        \_ /opt/node-v0.10.25-linux-x64/bin/node /opt/
2961 root
                            \_ npm
2969 root

    sh −c grunt

2970 root

    grunt

2972 root
2984 root
                                        I \_ node --debug server.js
     root
                                          grunt
```

With the top process id I could finally find the docker container running that grunt process.

```
~ # docker ps | awk '{ print $1}' | xargs docker inspect -f '{{ .State.Pid }} {{ .Config.Hostname }}' | grep 2871 Error: No such image or container: CONTAINER 2871 Error: No such image or container: CONTAINER
```

sudo docker ps | awk '{ print \$1 }' | xargs sudo docker inspect f '{{ .State.Pid }} {{ .Config.Hostname }}' | grep 68317