

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.

top

```
root@srv01: ~
top - 11:22:53 up 112 days, 22:59, 5 users, load average: 0.06, 0.10, 0.16
Tasks: 382 total, 2 running, 373 sleeping, 0 stopped, 7 zombie
%Cpu(s): 0.4 us, 0.9 sy, 0.0 ni, 98.6 id, 0.1 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 49455108 total, 43718720 used, 5736388 free, 5789156 buffers
KiB Swap: 25149308 total, 7380 used, 25141928 free, 27053256 cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU  %MEM    TIME+  COMMAND
 2973 root        20   0  836m  76m  5768 S  33.2   0.2   1653:45 grunt
16366 root        20   0 22280  1780  1112 R   0.7   0.0    0:04.28 top
20366 root        20   0 9473m 281m  224m S   0.7   0.6   29:25.05 mongod
    21 root        20   0     0     0     0 S   0.3   0.0   262:31.84 rcu_sched
 23 root        20   0     0     0     0 S   0.2   0.0    14:15.13 rcu_sched
```

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 /c
2945 root      | \ /opt/node-v0.10.25-linux-x64/bin/node /opt/t
2961 root      | | \ npm
2969 root      | | \ sh -c grunt
2970 root      | | \ grunt
2972 root      | | \ grunt
2984 root      | | | \ node --debug server.js
2973 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 bravi.com.br

```

```

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

```