实训8：限制docker容器对cpu的使用

1. **实验目的**

学习如何限制docker容器使用CPU

**二、实验描述**

1. 限制docker容器对cpu的使用
2. 实验时长：15分钟
3. 主要步骤：
   * 下载或导入progrium/stress镜像
   * 使用progrium/stress镜像查看CPU限制效果

**三、实验环境**

* 系统版本：ubuntu 18.04
* docker版本： 18.06.1-ce

**四、实验知识点**

* linux系统基础配置
* docker基础命令
* progrium/stress 镜像使用

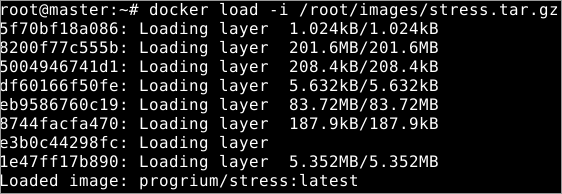
**五、实验步骤**

**1. 下载或导入已保存在本地的**stress**基础镜像。**

docker pull progrium/stress

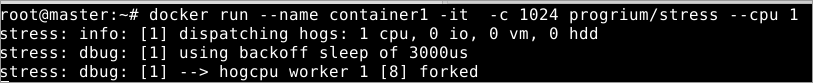
或

docker -i load /root/images/stress.tar.gz



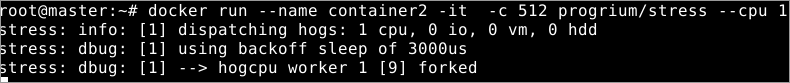
**2. docker容器限制CPU与限制内存不同，限制CPU只能限制先对权重值。改值默认是1024。现在开启两个容器，名字分别为container1和container2。container1的CPU权重值为container2的2倍。开启一个linux终端，启动container1：**

docker run --name container1 -it -c 1024 progrium/stress --cpu 1



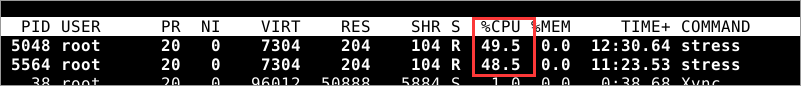
**3. 开启一个新的linux终端，启动container2：**

docker run --name container2 -it -c 512 progrium/stress --cpu 1



**4. 开启第三个终端，使用top命令查看容器对CPU的使用情况，但是看到CPU使用情况几乎是一样的，并不是一个容器是另一个容器的2倍。这是因为该限制只有当CPU紧张时才会生效。而实验环境所在的物理服务器通常有十几个甚至几十个CPU核心。如果想看到效果，可以自己开一个只有1个CPU核的vmware或virtualbox虚拟机测试。可以按Q键退出top命令。**

top



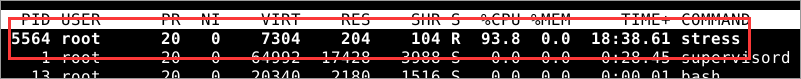
**5. 现在暂停container1容器。**

docker pause container1



**6. 再次执行top命令，可以看到container1暂停的情况下，container2将用满cpu。**

top



**六、总结**

本章介绍了如何限制docker容器使用CPU，但是因为实验环境的特殊性，看不到实际的效果。