instance 的网卡是如何被配置并拉起的？这是理解和用好 cloud-init 非常关键的一步。我们先讨论一个最简单基础的场景：镜像中没有安装 cloud-init。

此时 instance 启动时网卡能不能被拉起来完全 **靠运气**！是的，就是运气。

因为这种情况下网卡的配置是死的，完全依赖于镜像中 /etc/network/interfaces 原有的配置。比如原镜像中的配置是：  
  
auto eth0  
iface eth0 inet dhcp  
  
instance 只有满足下面所有条件网卡才能被拉起来：

1. 正好只有一块网卡
2. 正好网卡就叫 eth0
3. 正好 subnet 开了 DHCP

只要出现下面任意一种情况就会失败：

1. 还有其他网卡，比如 eth1，或者
2. 网卡不叫 eth0 ，比如 ens3，或者
3. 没有 DHCP

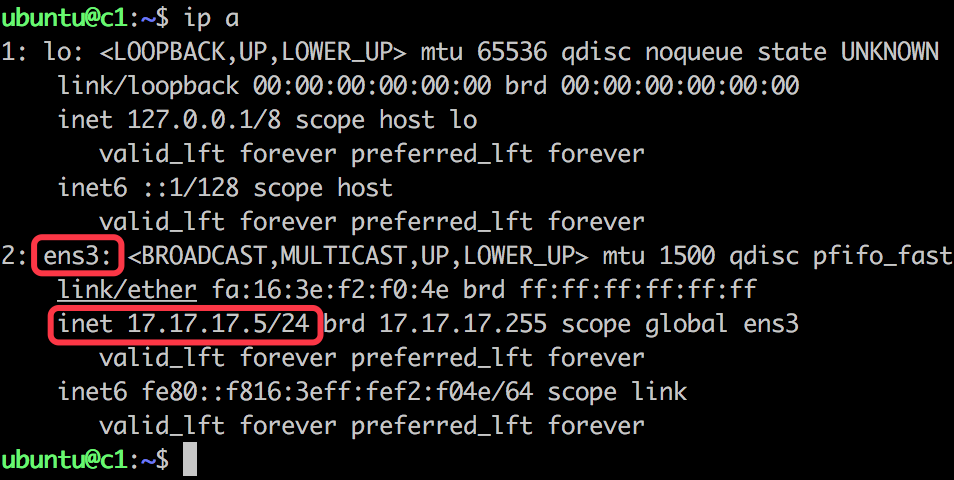
不同 instance 的网络配置差别很大，在 image 中写死的方法几乎是无效的，只能依靠 cloud-init 动态写入，接下来我们详细分析 cloud-init 的解决方案。

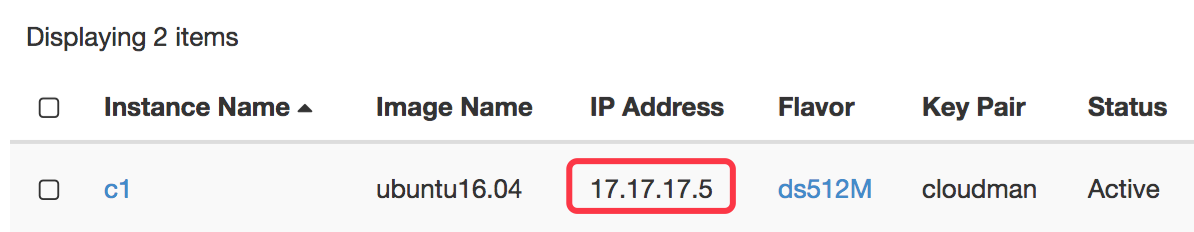
**dhcp**

先考虑 subnet 有 DHCP 服务的情况。

我们使用的镜像是 ubuntu 的 cloud image，已经预装的 cloud-init，下载地址为 <http://cloud-images.ubuntu.com/> ，国内镜像<http://mirrors.ustc.edu.cn/ubuntu-cloud-images/>

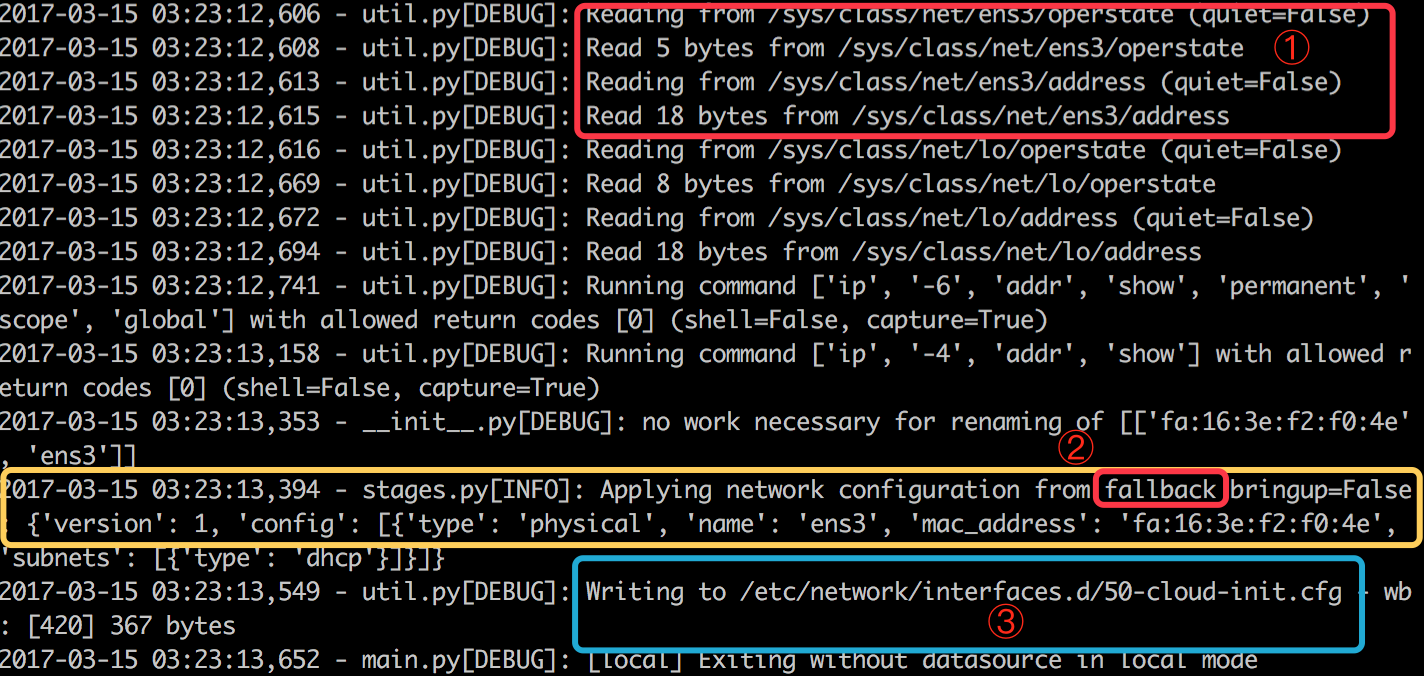
部署成功后，登录 instance，ip a 显示网卡 ens3 已经正确配置。





下面分析这个 IP 是怎样配置上去的。

上一节我们讨论到，cloud-init 是在 local 阶段完成网络配置的，cloud-init 的执行过程被详细记录在 /var/log/cloud-init.log 中，让我们找找相关操作。

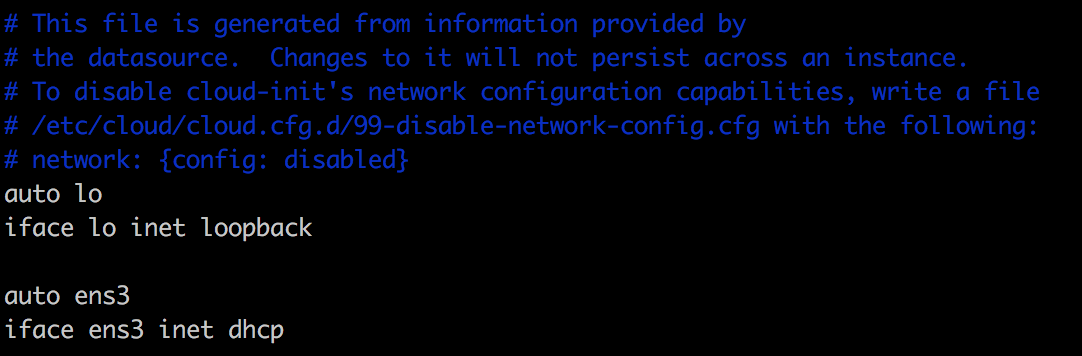


这里可以看到，cloud-init 会做如下工作：

① 扫描出 instance 中的所有网卡（这里是 ens3）

② 获取该网卡的配置信息。 因为没有 config drive，无法得知网卡的详细配置信息，只能采用默认的 fallback 配置，即 dhcp 配置。

③ 将配置信息写入 /etc/network/interfaces.d/50-cloud-init.cfg，内容为：



这样网卡就以 dhcp 模式拉起来，正好与 subnet 的 dhcp 服务对接上，IP、网关等信息就配上去了。

几点说明：

1. instance 上的每一块网卡都会被 cloud-init 扫描出来。
2. 如果没有 config drive 将采用 fallback 配置，将扫描出来的**第一块** （只有这一块）网卡配置成 dhcp 模式。请注意：这是 cloud-init 默认行为，跟这块网卡对应的 subnet 是否开启了 DHCP 没有任何关系。
3. cloud-init 会根据 instance 操作系统类型生成网卡配置文件。例如操作系统是 centos 的话则会将配置写到 /etc/sysconfig/network-scripts 目录下。

现在请大家思考一个问题：如果 subnet 没有开 DHCP，会是怎样一个情况？下节将分析这个问题。