cloud-init 是 linux 的一个工具，当系统启动时，cloud-init 可从 nova metadata 服务或者 config drive 中获取 metadata，完成包括但不限于下面的定制化工作：

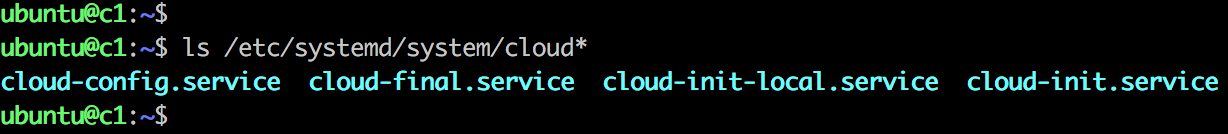
1. 设置 default locale
2. 设置 hostname
3. 添加 ssh keys到 .ssh/authorized\_keys
4. 设置用户密码
5. 配置网络
6. 安装软件包

为了实现 instance 定制工作，cloud-init 会按 4 个阶段执行任务：

1. local
2. init
3. config
4. final

cloud-init 安装时会将这 4 个阶段执行的任务以服务的形式注册到系统中，比如在 systemd 的环境下，我们能够看到这4个阶段分别对应的服务：

1. local - cloud-init-local.service
2. init - cloud-init.service
3. config - cloud-config.service
4. final - cloud-final.service



**local 阶段**

作为 cloud-init 执行的第一个阶段，此时 instance 还不知道该如何配置网卡，cloud-init 的任务就是从 config drive 中获取配置信息，然后写入 /etc/network/interfaces 文件（如果是 centos 则写入 /etc/sysconfig/network-scripts/ifcfg-xxx）。

**如果没有 config drive，则将所有网卡配置成 dhcp 模式**。这是非常关键的一步，只有当网卡正确配置后，才能获取到 metadata。

关于 local 阶段下一节会通过实验详细分析。

**init, config 和 final 阶段**

正常情况下，在这三个阶段执行之前 instance 网络已经配置好了，并且已经成功获取到 metadata。cloud-init 的配置文件 /etc/cloud/cloud.cfg 定义了三个阶段分别要执行的任务，任务以 module 形式指定。



instance 真正的定制工作就是由这些 module 完成的。module 决定做哪些定制化工作，而 metadata 则决定最终定制化的结果。

举个例子，如果 cloud.cfg 中指定了 set\_hostname 这个 module，则意味着 cloud-int 会设置 instance 的主机名，而具体设置成哪个主机名则由 metadata 中 hostname 参数决定。

有些 module 是有默认行为的，比如 growpart，如果 metadata 中没有特别指定，它会自动扩展 / 分区。

由于篇幅限制，这里就不一一讨论每个 module 了，具体可参看文档 <https://cloudinit.readthedocs.io/en/latest/topics/modules.html>

后面我们会讨论 cloud-init 典型的使用场景，其中也会涉及常用 module 的示例。