

虚拟机定制化包括两部分：

metadata service 为 cloud-init 提供自定义配置数据，cloud-init 完成配置工作。

有几个方法可以完成虚拟机定制化：

1. 将这些东西统统做到 image 中。

这种方案可以实现，但不现实。image 应该被看着是一个模板，存放的是通用的内容。在 image 中加入

个性化配置的做法要么使 image 变得非常庞杂，要么导致数量众多的 image，不易管理。

2. instance 部署出来之后手工完成个性化配置。

由于需要手工操作，instance 数量多了之后工作量会激增，而且容易出错。

Metadata Service

nova-api-metadata 是 nova-api 的一个子服务，它是 metadata 的提供者，instance 可以通过 novaapi-metadata 的REST API 来获取 metadata 信息。

nova-api-metadata 运行在控制节点上，服务端口是 8775。

nova.conf 通过参数enabled_apis 指定是否启用 nova-api-metadata。

osapi_compute 是常规的 nova-api 服务，metadata 就是 nova-api-metadata 服务。

instance通过nova-api-metadata请求metadata信息，完成开机之后的初始化准备。

网络链路：

```
1 instance --->
2 neutron network (Project 网络) --->
3 neutron-ns-metadataproxy(通过dhcp-agent 和 l3-agent创建) --->
4 unix domain socket (连接instance网络和OpenStack的内部网络) --->
5 neutron-metadata-agent --->
6 nova-api-metadata
```

l3-agent 创建和管理 neutron-ns-metadata-proxy

instance所认为的 metadata 服务地址是 169.254.169.254，端口为 80（固定的）。

172.24.10.9 实际上就是路由在network上的 interface IP（neutron-ns-metadata-proxy

是路由建的，监听端口是9697)。这条路由是 OpenStack 自动添加到instance 中的，这样就将 metadata 的请求转发到 neutron router。

```
[root@vmkvm1qing network-scripts]# ip r
default via 172.24.10.1 dev eth1 proto static metric 100
default via 172.24.10.1 dev eth0 proto static metric 101
169.254.169.254 via 172.24.10.9 dev eth1 proto dhcp metric 100
172.24.10.0/24 dev eth1 proto kernel scope link src 172.24.10.34 metric 100
172.24.10.0/24 dev eth0 proto kernel scope link src 172.24.10.55 metric 101
192.168.122.0/24 dev virbr0 proto kernel scope link src 192.168.122.1
```

具体过程：

1. instance 通过预定义的 169.254.169.254 请求 metadata。
2. 请求被转发到 neutron router。
3. router 将请求转发给 neutron-ns-metadata-proxy。
4. 再后面就简单了：neutron-ns-metadata-proxy 将请求通过 unix domain socket 发

dhcpagent 来创建和管理 neutron-ns-metadata-proxy

配置/etc/kolla/neutron-dhcp-agent/dhcp_agent.ini

force_metadata = true

重启devstack@q-dhcp.service

原理：

dhcp-agent 则是将此 IP 配置到自己的 interface 上，而I3是通过iptables进行端口映射。

instance如何获取instance id信息

要想从 nova-api-metadata 获得 metadata，需要指定 instance 的 id。但 instance 刚启动时无法知道自

己的 id，所以 http 请求中不会有 instance id 信息，id 是由 neutron-metadata-agent 添加进去的。针对

I3-agent 和 dhcp-agent 这两种情况在实现细节上有所不同，下面分别讨论。

I3-agent

1. neutron-ns-metadata-proxy 接收到请求，在转发给 neutron-metadata-agent 之前会将 instance ip和 router id 添加到 http 请求的 head 中，这两个信息对于 I3-agent 来说很容易获得。

2. neutron-metadata-agent 接收到请求后，会查询 instance 的 id，具体做法是：
 - 1) 通过 router id 找到 router 连接的所有 subnet，然后筛选出 instance ip 所在的 subnet。
 - 2) 在 subnet 中找到 instance ip 对应的 port。
 - 3) 通过 port 找到对应的 instance 及其 id。
3. neutron-metadata-agent 将 instance id 添加到 http 请求的 head 中，然后转发给 nova-apimetadata，这样 nova-api-metadata 就能返回指定 instance 的 metadata 了。

dhcp-agent

1. neutron-ns-metadata-proxy 在转发请求之前会将 instance ip 和 network id 添加到 http 请求的 head 中，这两个信息对于 dhcp-agent 来说很容易获得。
2. neutron-metadata-agent 接收到请求后，会查询 instance 的 id，具体做法是：
 - 1) 通过 network id 找到 network 所有的 subnet，然后筛选出 instance ip 所在的 subnet。
 - 2) 在 subnet 中找到 instance ip 对应的 port。
 - 3) 通过 port 找到对应的 instance 及其 id。
3. neutron-metadata-agent 将 instance id 添加到 http 请求的 head 中，然后转发给 nova-apimetadata，这样 nova-api-metadata 就能返回指定 instance 的 metadata 了。

config-drive

config drive 是一个特殊的文件系统，OpenStack 会将 metadata 写到 config drive，并在 instance 启动时挂载给 instance。如过 instance 安装了 cloud-init，config drive 会被自动 mount 并从中读取 metadata，进而完成后续的初始化工作。

config drive 默认是 disable 的，所以首先得启用。有两种方法启用 config drive：

1. 启动 instance 时指定 `--config-drive true`。
2. 在计算节点的 `/etc/nova/nova.conf` 中配置 `force_config_drive = true`，这样部署到此计算节点的 instance 都会使用 config drive。

config drive 支持两种格式，iso9660 和 vfat，默认是 iso9660，但这会导致 instance 无法在线迁移，必须设置成 `config_drive_format=vfat` 才能在线迁移，这一点需要注意。

cloud-init

编辑虚拟机的/etc/cloud/cloud.cfg文件，可以禁用Cloud-init的初始化功能。

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

1. local --- 配置网卡
2. init
3. config
4. final

cloud-init 安装时会把这 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：

5. local - cloud-init-local.service
6. init - cloud-init.service
7. config - cloud-config.service
8. final - cloud-final.service

详细过程参考Cloud Man



每天5分钟玩转OpenStack.pdf
2017-11-03 15:10:42, 53.8 MB