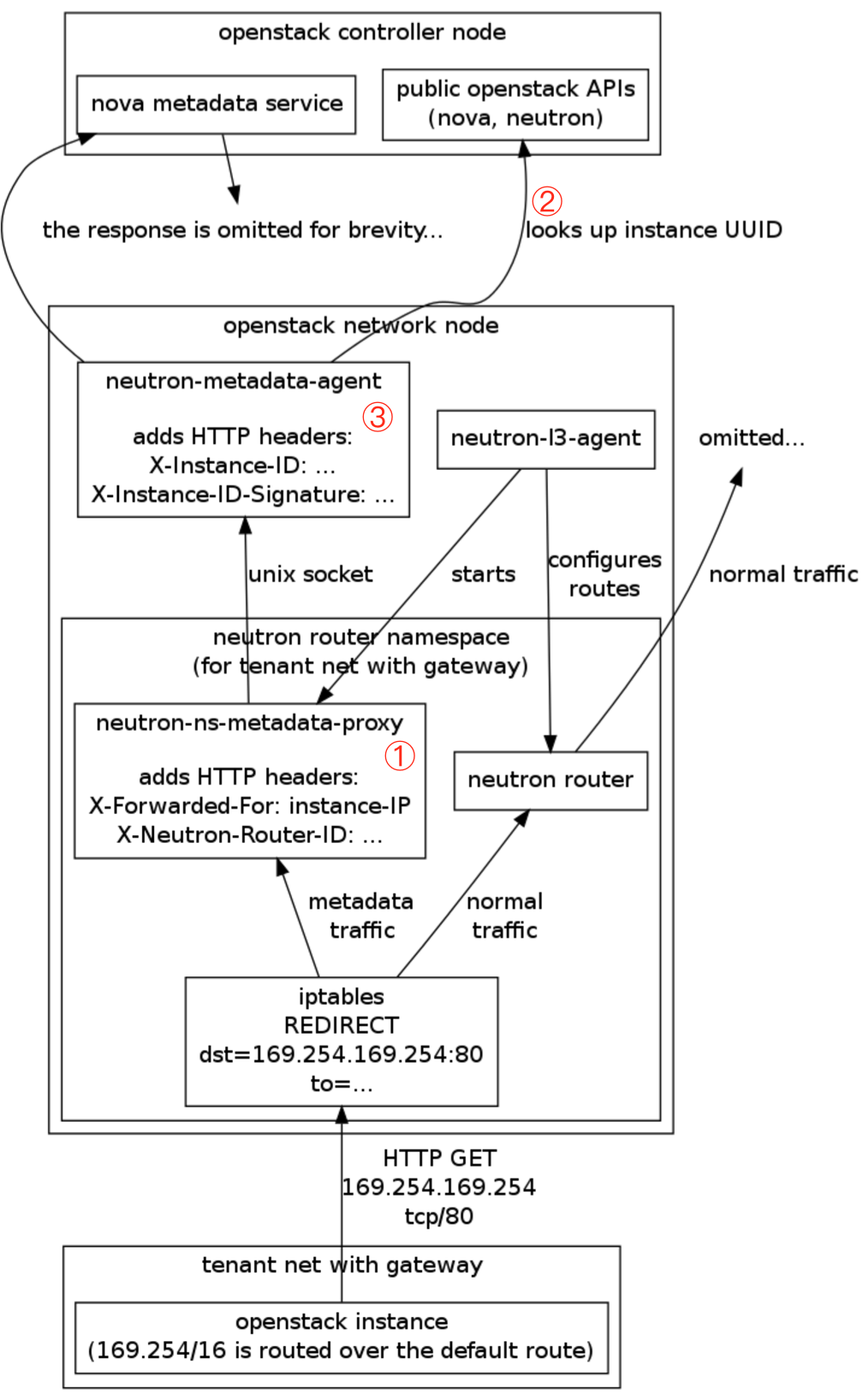
要想从 nova-api-metadata 获得 metadata，需要指定 instance 的 id。但 instance 刚启动时无法知道自己的 id，所以 http 请求中不会有 instance id 信息，id 是由 neutron-metadata-agent 添加进去的。针对 l3-agent 和 dhcp-agent 这两种情况在实现细节上有所不同，下面分别讨论。

**l3-agent**

下面是 l3-agent 参与情况下 metadata http 请求的处理流程图。



大的流程为：instance -> neutron-ns-metadata-proxy -> neutron-metadata-agent -> nova-api-metadata，处理细节说明如下：

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 l3-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。

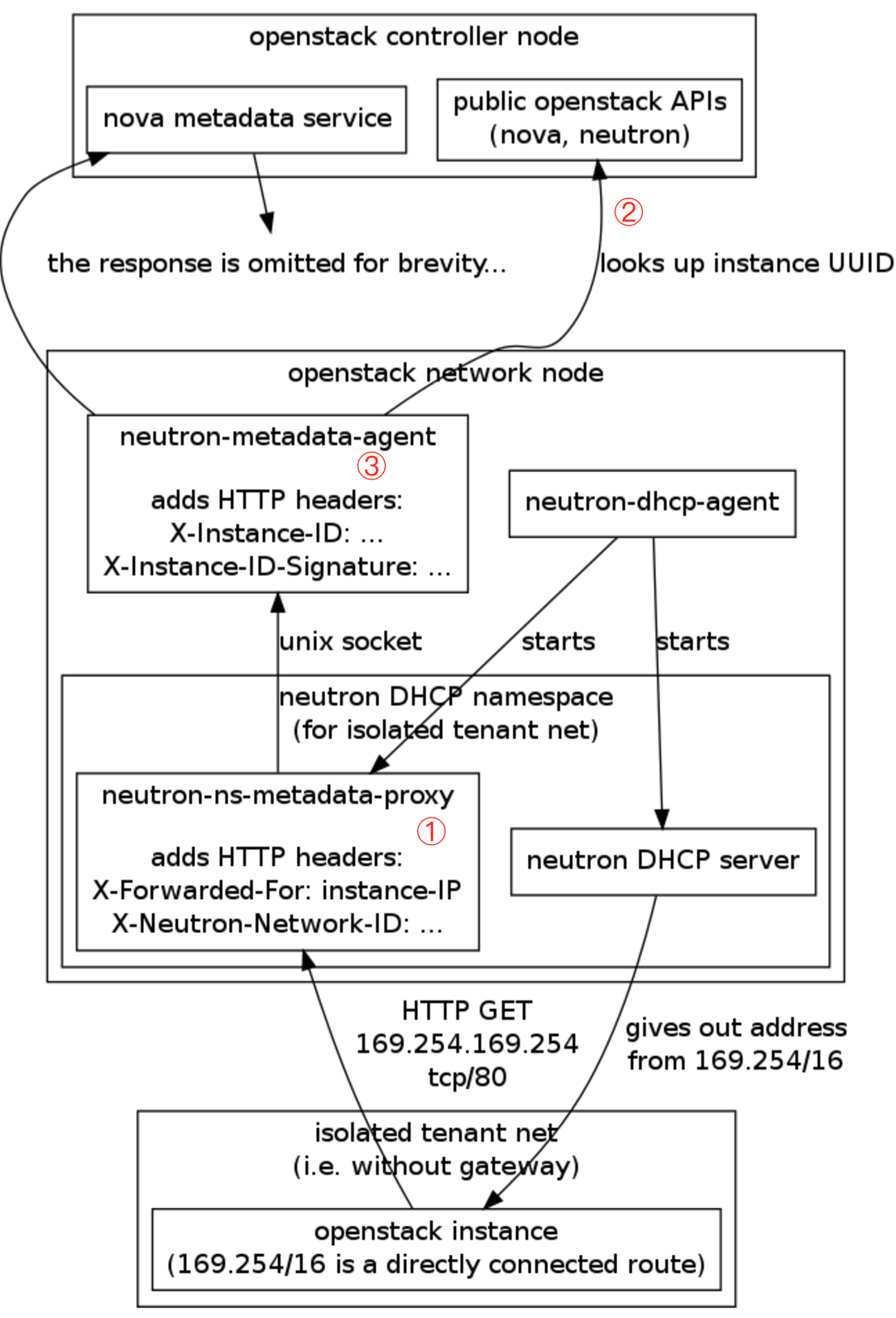
2）在 subnet 中找到 instance ip 对应的 port。

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-api-metadata，这样 nova-api-metadata 就能返回指定 instance 的 metadata 了。

我们再来看 dhcp-agent 的情况。

**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-api-metadata，这样 nova-api-metadata 就能返回指定 instance 的 metadata 了。

这样，不管 instance 将请求发给 l3-agent 还是 dhcp-agent，nova-api-metadata 最终都能获知 instance 的 id，进而返回正确的 metadata。

从获取 metadata 的流程上看，有一步是至关重要的：**instance 必须首先能够正确获取 DHCP IP**，否则请求发送不到 169.254.169.254。但不是所有环境都会启用 dhcp，更极端的，有些环境可能连 nova-api-metadata 服务都不会启用。那么 instance 还能获得 metadata 吗？

这就是下一节我们要讨论的主题：**config drive**。