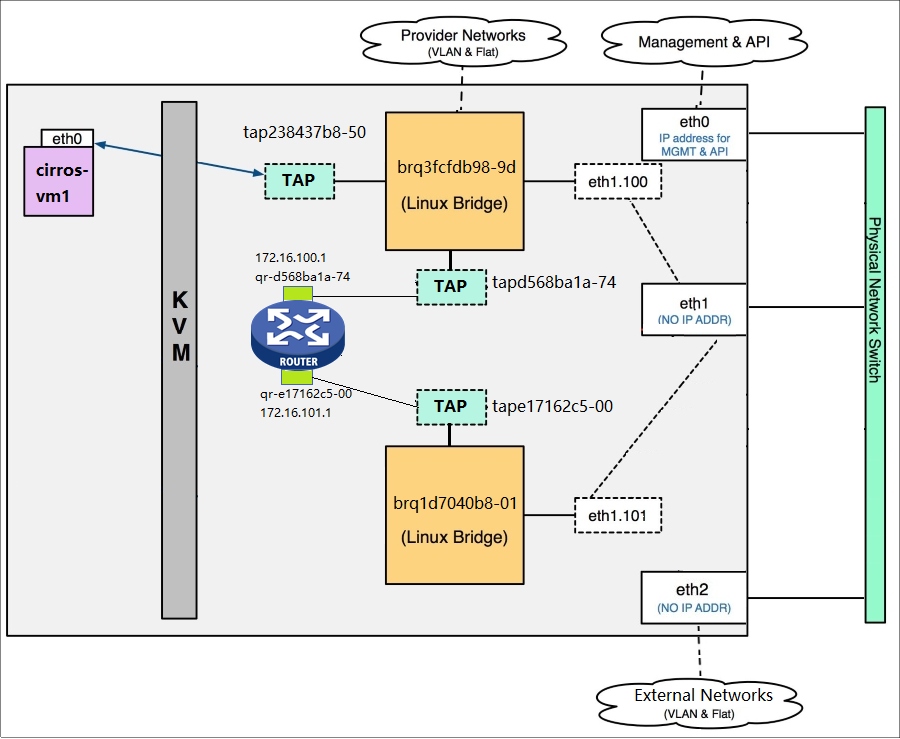


上一节我们讨论了 Neutron 将虚拟 router 放置到 namespace 中实现了不同 subnet 之间的路由。  
今天探讨为什么要用 namespace 封装 router？

回顾一下前面的网络逻辑结构图：

  
我们需要讨论一个深层次的问题：

为什么不直接在 tape17162c5-00 和 tapd568ba1a-74 上配置 Gateway IP，而是引入一个 namespace，在 namespace 里面配置 Gateway IP 呢？

首先考虑另外一个问题：

如果不用 namespace，直接 Gareway IP 配置到 tape17162c5-00 和 tapd568ba1a-74 上，能不能连通 subnet\_172\_16\_100\_0 和 subnet\_172\_16\_101\_0 呢？

答案是可以的，只要控制节点上配置了类似下面的路由。

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

172.16.100.0    \*               255.255.255.0   U     0      0        0 tapd568ba1a-74

172.16.101.0    \*               255.255.255.0   U     0      0        0 tape17162c5-00

既然不需要 namespace 也可以路由，为什么还要加一层 namespace 增加复杂性呢？

其根本原因是：**为了支持网络重叠**。

云环境下，租户可以按照自己的规划创建网络，不同租户的网络是可能重叠的。

将路由功能放到 namespace 中，就能隔离不同租户的网络，从而支持网络重叠。

下面通过例子进一步解释。

Tenant A  vlan100 subnet A-1: 10.10.1.0/24    {"start": "10.10.1.1", "end": "10.10.1.254"}

Tenant A  vlan101 subnet A-2: 10.10.2.0/24    {"start": "10.10.2.1", "end": "10.10.2.254"}

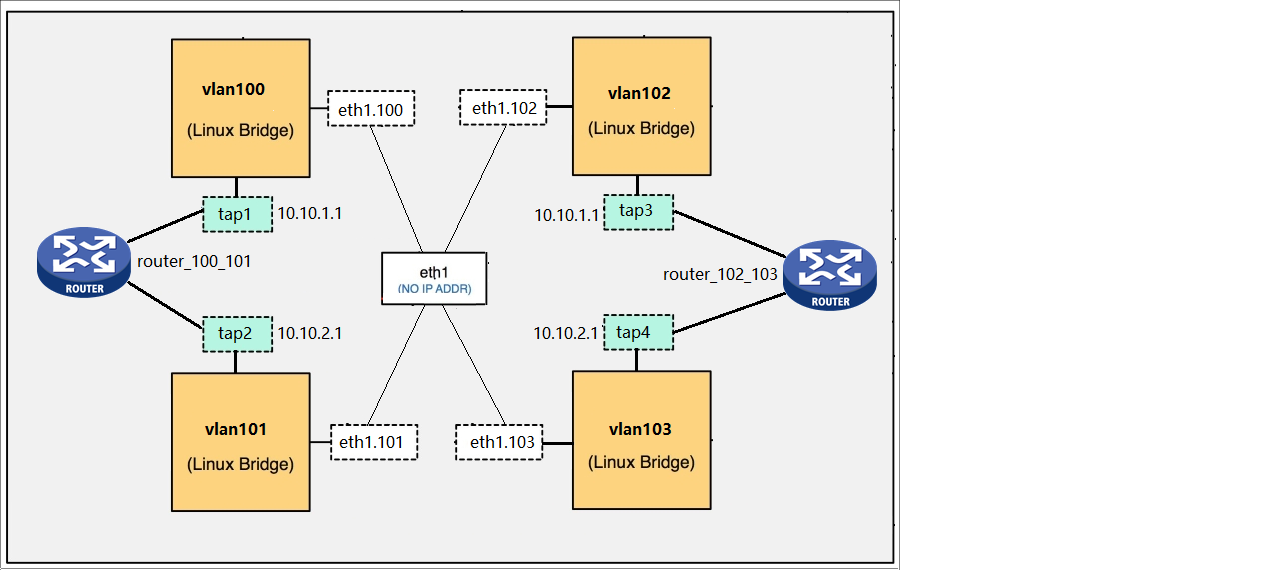
Tenant B  vlan102 subnet B-1: 10.10.1.0/24    {"start": "10.10.1.1", "end": "10.10.1.254"}

Tenant B  vlan103 subnet B-2: 10.10.2.0/24    {"start": "10.10.2.1", "end": "10.10.2.254"}

A，B 两个租户定义了完全相同的两个 subnet，网络完全重叠。

**不使用 namespace 的场景**

如果不使用 namespace，网络结构如下：



其特征是网关 IP 配置在 TAP interface 上。

因为没有 namespace 隔离，router\_100\_101 和 router\_102\_103 的路由条目都只能记录到控制节点操作系统（root namespace）的路由表中，内容如下：

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

 10.10.1.0                 \*               255.255.255.0   U     0      0        0     tap1

 10.10.2.0                 \*               255.255.255.0   U     0      0        0     tap2

 10.10.1.0                 \*               255.255.255.0   U     0      0        0     tap3

 10.10.2.0                 \*               255.255.255.0   U     0      0        0     tap4

这样的路由表是无法工作的。

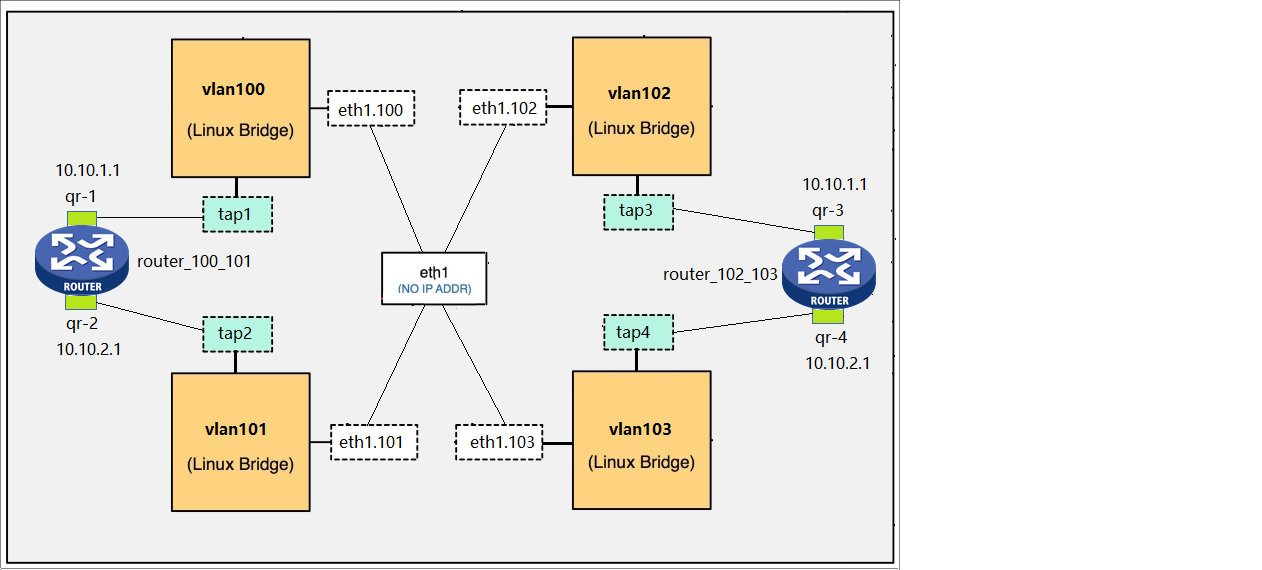
按照路由表优先匹配原则，Tenant B 的数据包总是错误地被 Tenant A 的 router 路由。

例如 vlan102 上有数据包要发到 vlan103。

选择路由时，会匹配路由表的第二个条目，结果数据被错误地发到了 vlan101。

**使用 namespace 的场景**

如果使用 namespace，网络结构如下：



其特征是网关 IP 配置在 namespace 中的 veth interface 上。

每个 namespace 拥有自己的路由表。

router\_100\_101 的路由表内容如下：

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

10.10.1.0                 \*               255.255.255.0   U     0      0        0     qr-1

10.10.2.0                 \*               255.255.255.0   U     0      0        0     qr-2

router\_102\_103 的路由表内容如下：

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

10.10.1.0                 \*               255.255.255.0   U     0      0        0     qr-3

10.10.2.0                 \*               255.255.255.0   U     0      0        0     qr-4

这样的路由表是可以工作的。

例如 vlan102 上有数据包要发到 vlan103。

选择路由时，会查看 router\_102\_103 的路由表, 匹配第二个条目，数据通过 qr-4 被正确地发送到 vlan103。

同样当 vlan100 上有数据包要发到 vlan101时，会匹配 router\_100\_101 路由表的第二个条目，数据通过 qr-2 被正确地发送到 vlan101。

可见，namespace 使得每个 router 有自己的路由表，而且不会与其他 router 冲突，所以能很好地支持网络重叠。

好了，目前我们已经搞清楚了 Neutron 内部 subnet 之间是如何通信的了。  
下一节我们将学习 OpenStack 内部网络如何与外部网络通信。