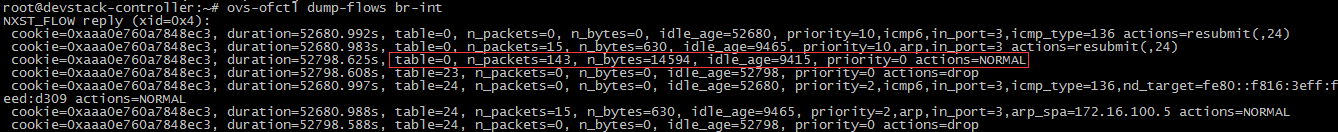


OVS 的数据流向都是由 Flow 规则控制的，今天我们就来分析 VxLAN 的 Flow 规则。  
  
提个醒：这可能是本教程最烧脑的一节，let's rock it !

下面分析控制节点上的 flow rule，计算节点类似。

**br-int 的 flow rule**

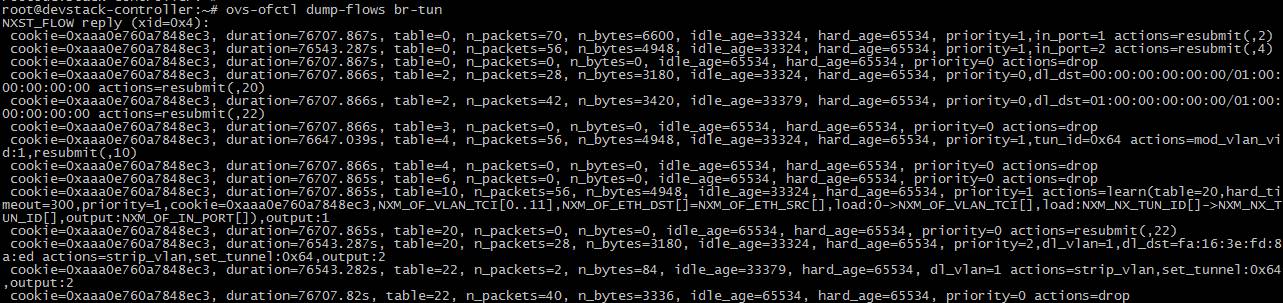


br-int 的 rule 看上去虽然多，其实逻辑很简单，br-int 被当作一个二层交换机，其重要的 rule 是下面这条：

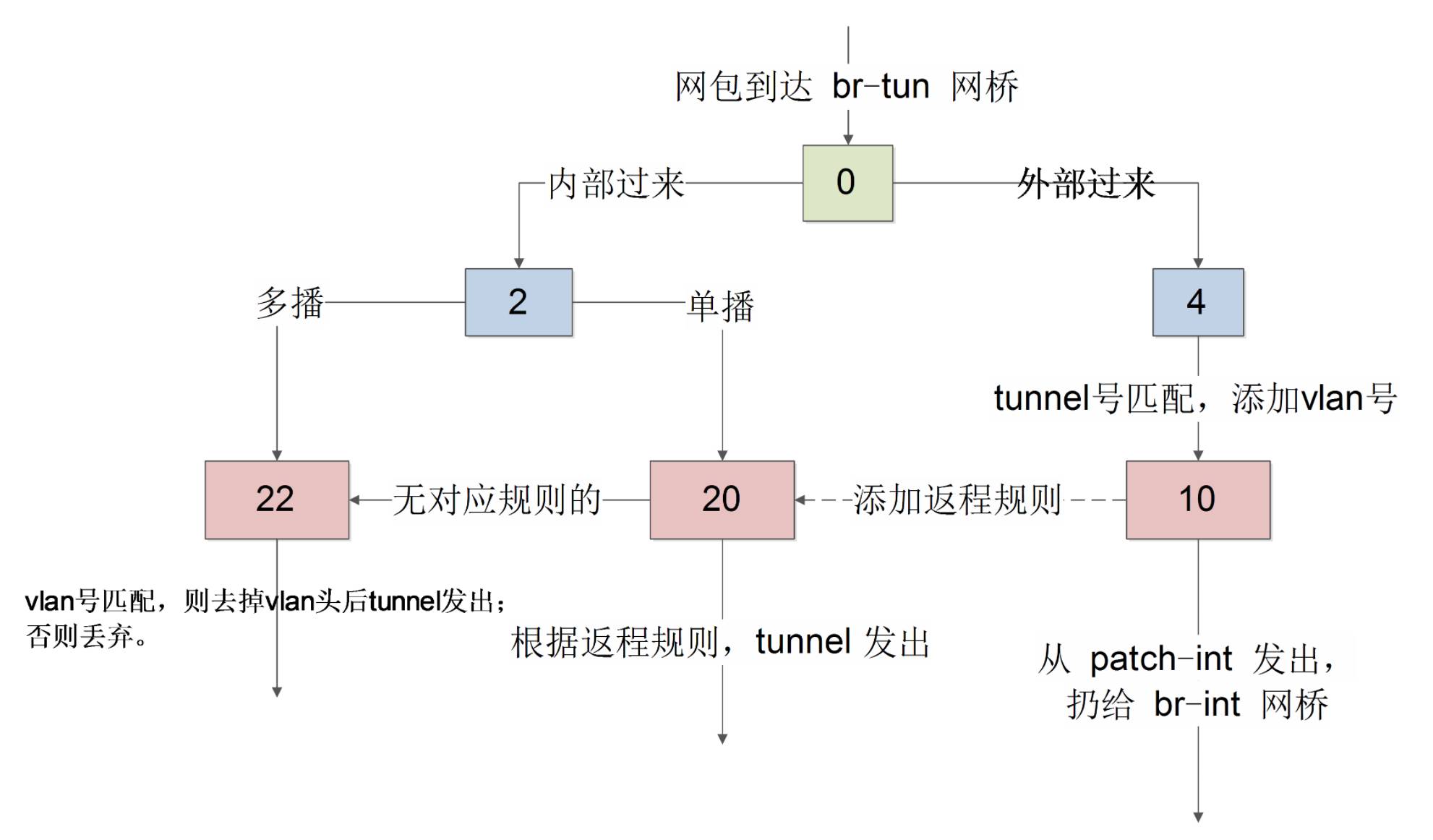
cookie=0xaaa0e760a7848ec3, duration=52798.625s, table=0, n\_packets=143, n\_bytes=14594, idle\_age=9415, priority=0 actions=**NORMAL**

此规则的含义是：根据 vlan 和 mac 进行转发。

**br-tun 的 flow rule**



这些才是真正处理 VXLAN 数据包的 rule，流程如下：



上图各方块中的数字对应 rule 中 table 的序号，比如编号为0的方块对应下面三条 rule。

**table 0**

cookie=0xaaa0e760a7848ec3, duration=76707.867s, **table=0**, n\_packets=70, n\_bytes=6600, idle\_age=33324, hard\_age=65534, priority=1,in\_port=1 actions=resubmit(,2)

cookie=0xaaa0e760a7848ec3, duration=76543.287s, **table=0**, n\_packets=56, n\_bytes=4948, idle\_age=33324, hard\_age=65534, priority=1,in\_port=2 actions=resubmit(,4)

cookie=0xaaa0e760a7848ec3, duration=76707.867s, **table=0**, n\_packets=0, n\_bytes=0, idle\_age=65534, hard\_age=65534, priority=0 actions=drop

结合如下 port 编号：

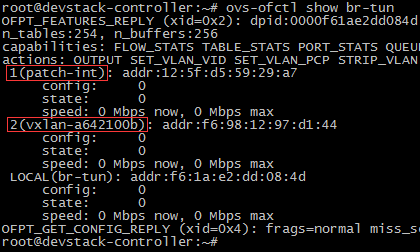


table 0 flow rule 的含义为：

1. 从 port 1（patch-int）进来的包，扔给 table 2 处理：actions=resubmit(,2)
2. 从 port 2（vxlan-a642100b）进来的包，扔给 table 4 处理：actions=resubmit(,4)

即第一条 rule 处理来自内部 br-int（这上面挂载着所有的网络服务，包括路由、DHCP 等）的数据；第二条 rule 处理来自外部 VXLAN 隧道的数据。

**table 4**

cookie=0xaaa0e760a7848ec3, duration=76647.039s, **table=4**, n\_packets=56, n\_bytes=4948, idle\_age=33324, hard\_age=65534, priority=1,tun\_id=0x64 actions=mod\_vlan\_vid:1,resubmit(,10)

table 4 flow rule 的含义为： 如果数据包的 VXLAN tunnel ID 为 100（tun\_id=0x64），action 是添加内部 VLAN ID 1（tag=1），然后扔给 table 10 去学习。

**table 10**

cookie=0xaaa0e760a7848ec3, duration=76707.865s, **table=10**, n\_packets=56, n\_bytes=4948, idle\_age=33324, hard\_age=65534, priority=1 actions=learn(table=20,hard\_timeout=300,priority=1,cookie=0xaaa0e760a7848ec3,NXM\_OF\_VLAN\_TCI[0..11],NXM\_OF\_ETH\_DST[]=NXM\_OF\_ETH\_SRC[],load:0->NXM\_OF\_VLAN\_TCI[],load:NXM\_NX\_TUN\_ID[]->NXM\_NX\_TUN\_ID[],output:NXM\_OF\_IN\_PORT[]),output:1

table 10 flow rule 的含义为： 学习外部（从 tunnel）进来的包，往 table 20 中添加对返程包的正常转发规则，然后从 port 1（patch-int）扔给 br-int。

rule 中下面的内容为学习规则，这里就不详细讨论了。

NXM\_OF\_VLAN\_TCI[0..11],NXM\_OF\_ETH\_DST[]=NXM\_OF\_ETH\_SRC[],load:0->NXM\_OF\_VLAN\_TCI[],load:NXM\_NX\_TUN\_ID[]->NXM\_NX\_TUN\_ID[],output:NXM\_OF\_IN\_PORT[]

**table 2**

cookie=0xaaa0e760a7848ec3, duration=76707.866s, **table=2**, n\_packets=28, n\_bytes=3180, idle\_age=33324, hard\_age=65534, priority=0,dl\_dst=00:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,20)

cookie=0xaaa0e760a7848ec3, duration=76707.866s, **table=2**, n\_packets=42, n\_bytes=3420, idle\_age=33379, hard\_age=65534, priority=0,dl\_dst=01:00:00:00:00:00/01:00:00:00:00:00 actions=resubmit(,22)

table 2 flow rule 的含义为：

1. br-int 发过来数据如果是单播包，扔给 table 20 处理：resubmit(,20)
2. br-int 发过来数据如果是多播或广播包，扔 table 22 处理：resubmit(,22)

**table 20**

cookie=0xaaa0e760a7848ec3, duration=76543.287s, **table=20**, n\_packets=28, n\_bytes=3180, idle\_age=33324, hard\_age=65534, priority=2,dl\_vlan=1,dl\_dst=fa:16:3e:fd:8a:ed actions=strip\_vlan,set\_tunnel:0x64,output:2

cookie=0xaaa0e760a7848ec3, duration=76707.865s, **table=20**, n\_packets=0, n\_bytes=0, idle\_age=65534, hard\_age=65534, priority=0 actions=resubmit(,22)

table 20 flow rule 的含义为：

1. 第一条规则就是 table 10 学习来的结果。内部 VLAN 号为 1（tag=1），目标 MAC 是 fa:16:3e:fd:8a:ed（virros-vm2）的数据包，即发送给 virros-vm2 的包，action 是去掉 VLAN 号，添加 VXLAN tunnel ID 100(十六进制 0x64)，并从 port 2 (tunnel 端口 vxlan-a642100b) 发出。
2. 对于没学习到规则的数据包，则扔给 table 22 处理。

**table 22**

cookie=0xaaa0e760a7848ec3, duration=76543.282s, **table=22**, n\_packets=2, n\_bytes=84, idle\_age=33379, hard\_age=65534, dl\_vlan=1 actions=strip\_vlan,set\_tunnel:0x64,output:2

cookie=0xaaa0e760a7848ec3, duration=76707.82s, **table=22**, n\_packets=40, n\_bytes=3336, idle\_age=65534, hard\_age=65534, priority=0 actions=drop

table 22 flow rule 的含义为： 如果数据包的内部 VLAN 号为 1（tag=1），action 是去掉 VLAN 号，添加 VXLAN tunnel ID 100(十六进制 0x64)，并从 port 2 (tunnel 端口 vxlan-a642100b) 发出。

**VXLAN 的路由和 floating IP 支持**

对于多 VXLAN 之间的 routing 以及 floating IP，实现方式与 vlan 非常类似，这里不再赘述，请参看前面 vlan 相关章节。

**总结**

本章重点讨论 Neutron 的架构，并通过分析 Linux Bridge 和 Open vSwitch 两个 mechnism driver 的技术细节，实践了 local，flat，vlan，vxlan 四种网络类型，同时也讨论了 routing 以及 floating IP 的实现细节。

Linux Bridge 和 Open vSwitch 都支持 Securet Group，Firewall as a Service ，Load Balancing as a Service 等高级功能，其实现方式也大致相同。

通过本章的学习，大家应该能够掌握 Neutron 的理论知识并应用到实践部署中。

到这里，OpenStack 的核心技术全部讨论完了，下一次 CloudMan 会带着大家回顾一下这些知识，给大家一些深入学习的建议，同时也会谈谈对后续新教程的一些想法。