1、了解Open vSwitch流表的基本概念。  
2、掌握流表的基本命令，学会添加、删除、查看流表，为后续实验做准备。

**任务环境**

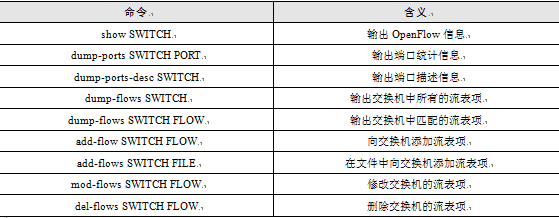
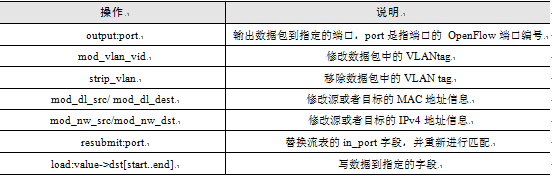
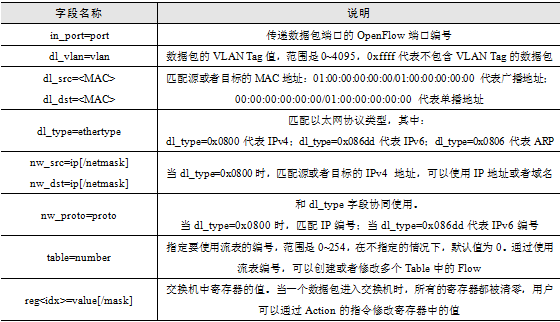
|  |  |  |
| --- | --- | --- |
| **设备名称** | **软件环境** | **硬件环境** |
| 交换机 | Ubuntu 14.04命令行版 Open vSwitch 2.3.1 | CPU：1核 内存：2G 磁盘：20G |

注：系统默认的账户为root/root@openlab，openlab/user@openlab。

**任务内容**

1、学习Open vSwitch流表的基本概念。  
2、学习常用的流表命令，进行流表的添加、删除、查看等操作。

**实验原理**

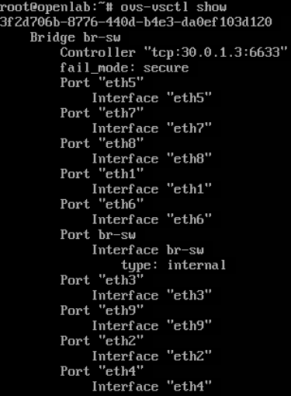
OpenFlow是用于管理交换机流表的协议，ovs-ofctl是Open vSwitch提供的命令行工具。在没有配置OpenFlow控制器的模式下，用户可以使用ovs-ofctl命令通过OpenFlow协议连接Open vSwitch来创建、修改或删除Open vSwitch中的流表项，并对Open vSwitch的运行状况进行动态监控。ovs-ofctl关于流表管理的常用命令如下表所示。  
  
对于add-flow、add-flows和mod-flows这3个命令，还需要指定要执行的动作actions=[target],[target]…，一个流规则中可能有多个动作，按照指定的先后顺序执行。  
常见的流表操作如下表所示。  
  
在OpenFlow白皮书中，Flow被定义为某个特定的网络流量。例如，一个TCP连接就是一个Flow，或者从某个IP地址发出来的数据包，都可以被认为是一个Flow。支持OpenFlow协议的交换机应该包括一个或多个流表，流表中的条目包含：数据包头的信息、匹配成功后要执行的指令和统计信息。当数据包进入OVS后，会将数据包和流表中的流表项进行匹配，如果发现了匹配的流表项，则执行该流表项中的指令集。相反，如果数据包在流表中没有发现任何匹配，OVS会通过控制通道把数据包发到OpenFlow控制器中。在OVS中，流表项作为ovs-ofctl的参数，采用如下的格式：字段=值，如果有多个字段，可以用逗号或空格分开，一些常用的字段列举如下表所示。  


**实验步骤**

**一、实验环境检查**

**步骤1 登录交换机，执行以下命令，查看镜像中原有的网桥，如下图所示。**

ovs-vsctl show



**步骤2 执行以下命令，删除当前网桥，并进行确认，如下图所示。**

ovs-vsctl del-br br-sw

ovs-vsctl show

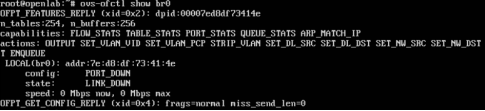


**二、流表管理**

**步骤1 执行以下命令，添加网桥，并查看虚拟交换机的基本信息，如下图所示。**

# ovs-vsctl add-br br0

# ovs-ofctl show br0

  
由上图可知，可以查看到交换机dpid、流表数量、性能参数、动作参数、MAC地址等信息。

**步骤2 执行以下命令，查看虚拟交换机上各端口的状态，如下图所示。**

# ovs-ofctl dump-flows br0

  
由上图可知，输出的结果中包含了各端口上收到的数据包数，字节数，丢包数，错误数据包数等。

**步骤3 执行以下命令，添加一条流表项，设置流表项生命周期为1000s，优先级为17，入端口为3，动作是output:2。**

# ovs-ofctl add-flow br0 idle\_timeout=1000,priority=17,in\_port=3,actions=output:2

**说明：这条流表项的作用是将端口3接收到的数据包从端口2输出。**

**步骤4 执行以下命令，查看交换机上所有流表信息，如下图所示。**

# ovs-ofctl dump-flows br0

**步骤5 执行以下命令，删除入端口为3的的流表项，删除后，再次查看流表信息，如下图所示。**

# ovs-ofctl del-flows br0 in\_port=3

# ovs-ofctl dump-flows br0



窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端

窗体顶端

窗体底端