OnetSwitch Test for OpenFlow Switch Specifiation 1.0.1

### Test 40: Flow Modification Massages

将主机1（mac地址：*00:13:3b:0f:42:1c*）、主机2（mac地址：*78:24:af:6c:41:c9*）、主机3、主机4分别接到交换机的eth1、eth2、eth3、eth4口，设置主机的ip分别为：192.168.100.1、192.168.100.2、192.168.100.3、192.168.100.4。让主机1和主机2互相ping，主机3和主机4互相ping。启动交换机，待交换机启动完成以后，运行dpctl下发流表：

1. 测试40.10：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

此时主机1和2能够ping通，再下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15,flags=0x2 apply:output=1*

此时交换机反馈FLOW\_MOD\_FAILED,OVERLAP信息，主机1和2能够ping通。

第二条流表设置了flags标志位为2，代表overlap，由于第二条流表的优先级和第一条相同，匹配域为全域匹配，所以和第一条流表存在重叠，此时由于OVERLAP，应该下发失败，测试结果也确实如此。

1. 测试40.20：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

此时主机1和2可以相互ping通，再下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=3*

此时交换机会打印replace信息，主机1和2不通。说明第三条流表下发成功，将第一条流表覆盖。由此可以说明no overlap，重叠的流表是可以成功下发的。

1. 测试40.30:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

此时从2口进入流表的数据包被送到1口,济将上面的流表重新下发一次,发现新的流表覆盖了原先的流表,counter也被清零.

1. 测试40.40：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=13 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=12 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=11 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=10 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=9 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=8 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=7 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=6 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=5 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=4 in\_port=2 apply:output=1*

此时下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=3 in\_port=2 apply:output=1*

交换机反馈FLOW\_MOD\_FAILED，TABLE\_FULL信息。Table 0实际可用entry数为12，所以下发第13条流表时会反馈table full信息。

1. 测试40.50：

./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=0

此时交换机反馈BAD\_ACTION，BAD\_OUT\_PORT信息。由于目前ONetSwitch只有四个网口用于数据通路，映射的四个端口为1、2、3、4，还有四个DMA逻辑端口，映射的四个逻辑端口为5、6、7、8。所以输出端口如果不是这八个端口，都会反馈BAD\_OUT\_PORT的信息。

1. 测试40.60：
2. 执行ifconfig eth1 down将交换机的端口1关闭，下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

此时主机1和2不通，ifconfig eth1 up 此时主机1和2能够ping通，说明流表下发成功，当端口1关闭时，通过流表的数据包被丢弃。

1. 修改交换机启动脚本， udatapath只映射eth1、eth2、eth3，交换机启动完

成以后，下发流表：0

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=4*

此时交换机反馈BAD\_ACTION，BAD\_OUT\_PORT信息。

1. 测试40.70：

由于of1.3不支持。

1. 测试40.80：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=mod,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=mod,table=0,prio=14 in\_port=1 apply:output=2*

此时主机1和主机2不通，流表没有下发，也没有error返回，与协议一致，测试通过。

1. 40.90：

下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

此时在端口2发送数据包，查看此条流的counter不为零，此时下发流表：

./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=mod,table=0,prio=15 in\_port=2 apply:output=1

查看此条流表的counter，计数没有被清零。

1. 40.100：

下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

此时在端口2发送数据包，查看此条流的counter不为零，此时下发流表：

./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=mods,table=0,prio=15 in\_port=2 apply:output=1

查看此条流表的counter，计数没有被清零。

1. 40.110：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=dels,table=0,prio=15 in\_port=2*

此时交换机正常工作，反馈无此条entry存在，符合协议规范。

1. 40.120：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15,flags=0x1 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

主机1和2能够ping通，此时下发流表:

./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=del,table=0

流表被删除，主机不通，第一条流表被删除时反馈FLOW\_REMOVED消息，与协议一致，测试通过。

1. 40.130：

Of1.3 不支持emergency mode。

1. 40.140：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

在主机1发目的mac为00:13:3b:0f:42:1c的数据包，在主机2用wireshark抓包，能收到目的mac地址为*00:13:3b:0f:42:1c*的数据包。此时，下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=dels,table=0,prio=15 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

删除流表，此时主机2无法收到来自主机1的目的mac为*00:13:3b:0f:42:1c*数据包，结果符合协议规范，测试通过。

1. 40.150：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

此时下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=del,table=0 in\_port=1 apply: output=2*

两条流表都被删除，结果符合协议规范，测试通过。

1. 40.160：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

此时下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=dels,table=0,prio=15 in\_port=1*

第一条流表被删除，结果符合协议规范，测试通过。

1. 40.170:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=1,eth\_dst=00:13:3b:0f:42:1c apply: output=2*

下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=del,table=0 in\_port=1 apply: output=2*

所有流表都被删除，结果符合协议规范，测试通过。

1. 40.180:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=1apply: output=2*

下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=dels,table=0,prio=15 in\_port=1 apply: output=2*

第一条流表被删除，结果符合协议规范，测试通过。

1. 40.190：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=3 apply:output=4*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=4 apply:output=3*

此时主机1和2能够ping通，主机3和4能够ping通，下发命令：

./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=del,table=0,out\_port=1

此时主机1和2不通，主机3和4能够ping通，说明，将输出端口为1的流表删除了，与协议一致。

1. 40.200：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15,out\_port=2*

*in\_port=2 apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15,out\_port=1*

*in\_port=1 apply:output=2*

此时主机1和2能够ping通，说明out\_port对add没有意义符合协议规范，此时下发

消息：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=mod,table=0,prio=15,out\_port=1*

*in\_port=1 apply:output=3*

此时主机1和2无法ping通，说明，out\_port对mod没有意义，符合协议规范。

#### Test 50: Flow Matching

将主机1（mac地址：*00:13:3b:0f:42:1c*）、主机2（mac地址：*78:24:af:6c:41:c9*）、主机3、主机4分别接到交换机的eth1、eth2、eth3、eth4口，设置主机的ip分别为：192.168.100.1、192.168.100.2、192.168.100.3、192.168.100.4。让主机1和主机2互相ping，主机3和主机4互相ping。启动交换机，待交换机启动完成以后，运行dpctl下发流表：

1. 50.10:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 apply:output=2*

此时在主机2用wireshark抓包，发现主机1，3，4的数据包发送到了主机2，符合协

议规定，测试通过。

1. 50.20：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

此时在主机2用wireshark抓包，发现主机1的数据包发送到了主机2，主机3、4的数据包通过packet\_in消息送到了控制器。结果符合协议规范，测试通过。

1. 50.30：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15 eth\_src=00:13:3b:0f:42:1c apply: output=2*

在主机2用wireshark抓包，能收到源mac地址为*00:13:3b:0f:42:1c*的数据包，源mac地址不符的数据包通过packet\_in消息被发送到控制器。结果符合协议规范，测试通过。

1. 50.40:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*eth\_dst=78:24:af:6c:41:c9 apply: output=2*

在主机2用wireshark抓包，能收到目的mac地址为*78:24:af:6c:41:c9*的数据包，目的

mac地址不符的数据包通过packet\_in消息被发送到控制器。结果符合协议规范，测试通过。

1. 50.50：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*eth\_type=0x0806 apply: output=1,output=2*

此时主机1能够收到主机2的arp包，主机2能够收到主机的arp包，主机1和2都发送icmp包，但是到达不了目的主机，此时下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*eth\_type=0x0800 apply: output=1,output=2*

此时主机1和2能够相互ping通，结果符合协议规范，测试通过。

1. 50.60：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=0 meta=0x0000000300000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15*

*vlan\_vid=2,vlan\_pcp=7 apply: output=1*

通过xcap发vlan\_vid为2，vlan\_pcp=7的数据包，在网口1能够收到此数据包。

1. 50.70:

参考50.60.

1. 50.80:

参考L3

1. 50.90：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15*

*ip\_dst=192.168.100.2,eth\_type=0x0800 apply: output=2*

在主机2通过wireshark抓包，目的ip为192.168.100.2的数据包被送到主机2.目的ip

不是192.168.100.2的数据包被送到控制器。结果符合协议规范，测试通过。

1. 50.100：

参考L4

1. 50.110：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15*

*Ip\_dscp=2 apply: output=2*

通过xcap发ip\_dscp为2的数据包，在网口1能够收到此数据包。

1. 50.120、50.130：

在主机1通过VLC视频播放器新建http协议的视频流，在主机2通过VLC请求主机1

的视频流，此时请求失败，下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*tcp\_src=8080,ip\_proto=6,eth\_type=0x0800 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*tcp\_dst=8080,ip\_proto=6,eth\_type=0x0800 apply: output=1*

此时重新请求主机1的http视频流，请求成功，在主机2能够流畅播放主机1的http

视频流。在测试中也测试了tcp\_src或tcp\_dst不等于8080的情况，主机2请求不成功，由此确定结果符合协议规范，测试通过。

1. 50.140:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=0*

*meta=0x0000000300000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=0*

*meta=0x0000000400000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15*

*in\_port=3,eth\_dst=00:13:3b:0f:42:1c meta:0x2 goto:1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15*

*eth\_src=00:00:00:00:00:33,vlan\_vid=2,meta=0x02 meta:0x2 goto:2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*eth\_type=0x0800,meta=0x12 apply:output=1*

1. 50.150:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=0*

*meta=0x0000000100000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=0*

*meta=0x0000000400000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 meta:0x2*

*goto:1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15*

*ip\_dst=192.168.200.1/0,meta=0x2 meta:0x3 goto:2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15*

*ip\_src=192.168.200.33/0,eth\_type=0x0800,meta=0x13 apply:output=1*

1. 50.160:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=0 meta=0x0000000100000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=0 meta=0x0000000400000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=3 meta:0x2 goto:1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15 udp\_src=161,udp\_dst=162,meta=0x2 meta:0x3 goto:2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15 meta=0x13,eth\_type=0x0800,ip\_proto=17 apply:output=1*

1. 50.170:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1*,eth\_dst=00:13:3b:0f:42:1c *apply:output=2*

在主机1发送目的mac为00:13:3b:0f:42:1c的数据包，在端口2能够收到此数据包，在端口1发送目的mac不为00:13:3b:0f:42:1c的数据包，在控制器能够收到packet\_in消息。

1. 50.180：

OF1.3不支持。

1. 50.190：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2*

*apply:output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply:output=3*

此时在主机1用wireshark抓包，能够抓到来自主机2的数据包，在主机3抓包，不能

抓到来自主机2的数据包，结果符合高优先级流表优先匹配，测试通过。

1. 50.200：

在主机1通过VLC视频播放器新建http协议的视频流，在主机2通过VLC请求主机1的视频流，此时请求失败，下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=15 tcp\_src=8079,ip\_proto=6,eth\_type=0x0800 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=14 tcp\_dst=8079,ip\_proto=6,eth\_type=0x0800 apply: output=1*

此时，请求还是失败，说明数据包没有匹配，下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=13 tcp\_src=8080,ip\_proto=6,eth\_type=0x0800 apply: output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=2,prio=12 tcp\_dst=8080,ip\_proto=6,eth\_type=0x0800 apply: output=1*

此时重新请求主机1的http视频流，请求成功，在主机2能够流畅播放主机1的http视频流。由此确定结果符合协议规范，测试通过。

1. 50.210：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=0*

*meta=0x0000000400000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 ip\_src=192.168.100.2, eth\_type=0x0806 apply:output=1*

此时在主机1能够收到来自主机2的arp包，主机3和4的通过packet\_in消息送给控

制器。

1. 50.220：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=0*

*meta=0x0000000100000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=0*

*meta=0x0000000200000000*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 ip\_dst=192.168.100.2 meta:0x1 goto:1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=1,prio=15 eth\_type=0x0806,meta=0x01 apply:output=2*

此时，在主机2用wireshark抓包，能够抓到主机1发送的目的ip为192.168.100.2的arp包，不能抓到主机3和4的arp请求包，同时3和4主机发送的数据包被通过packet\_in消息送往控制器。结果符合协议规范，测试通过。

#### Test 60: Counters

1. 60.10、60.20：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply:output=1*

主机1 ping主机2，在主机2用wireshark抓包，下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 stats-flow table=0*

查看pkt\_count和byte\_count与wireshark抓包一致。

1. 60.30、60.40:

时间不是太准，由软件实现，但是基本能够满足要求。

1. 60.50、60.70：

主机1发送数据包到端口1，在主机1用wireshark分析发送的数据包个数和字节数，下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 stats-port 1*

查看端口一接受的数据包个数和字节数和主机1发送的一致。

1. 60.60、60.80：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:output=2*

在主机1发包，下发消息：

*./utilities/dpctl tcp:127.0.0.1:6632 stats-port 1*

查看接受的数据包个数和字节数与传输的一致。

#### Test 70: Actions

将主机1（mac地址：*00:13:3b:0f:42:1c*）、主机2（mac地址：*78:24:af:6c:41:c9*）、主机3、主机4分别接到交换机的eth1、eth2、eth3、eth4口，设置主机的ip分别为：192.168.100.1、192.168.100.2、192.168.100.3、192.168.100.4。让主机1和主机2互相ping，主机3和主机4互相ping。启动交换机，待交换机启动完成以后，运行dpctl下发流表：

1. 70.10:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1*

此时在控制器端还是能够收到来自主机1的packet\_in消息，结果不符合协议规范，测

试不通过。

1. 70.30：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:output=all*

在主机1、3、4用wireshark抓包，无法抓到来自主机2的数据包。与协议要求不符，测试不能通过。

1. 70.40：

在控制器与交换机连接成功以后，会立即发送流表，用于将匹配不成功的数据包送到控

制器，此时控制器会收到来自各个网口发的数据包产生的packet\_in消息，结果符合协议规范，测试通过。

1. 70.70：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1*

*apply:output=in\_port*

此时在主机1发包，在主机1能够接受到主机1的包

1. 70.100:

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2*

*apply:output=1,output=3*

此时在主机1、主机3、主机4分别用wireshark抓包，主机1和3分别能够抓到来自主机2的数据包，主机4不能抓到来自主机2的数据包，结果符合协议规范，测试通过。

1. 70.110：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:queue=2,output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=3 apply:queue=3,output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2,output=3*

在主机1运行iperf server，在主机2和3分别运行iperf client发送1000Mbps/sec的数据流量，主机3的数据包在主机1的接受速率远高于主机2,的数据包被接受速率，结果符合协议规范，测试通过。

1. 70.130：

下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:set\_field=vlan\_vid:2,output=1*

在主机2发送vlan id不为2的数据包，在主机1用wireshark抓包，发现来自主机1的

数据包的vlan id值变为2.

1. 70.160：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2 apply:pop\_vlan,output=1*

在主机2发送vlan包，在主机1抓包，发现抓到的数据包没有vlan包头。

1. 70.170：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2*

*apply:set\_field=eth\_src:aa:bb:cc:dd:ee:11,output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=1 apply:set\_field=eth\_src: aa:bb:cc:dd:ee:22,output=2*

打开主机1的wireshark抓包发现来自主机2的包的src\_mac值变为aa:bb:cc:dd:ee:11，打开主机2的wireshark抓包发现来自主机1的包的src\_mac值变为aa:bb:cc:dd:ee:22，说明mac地址修改成功。结果符合协议规范，测试通过。

1. 70.180：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=2*

*apply:set\_field=eth\_dst:aa:bb:cc:dd:ee:11,output=1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=1 apply:set\_field=eth\_dst: aa:bb:cc:dd:ee:22,output=2*

打开主机1的wireshark抓包发现来自主机2的包的dst\_mac值变为aa:bb:cc:dd:ee:11，打开主机2的wireshark抓包发现来自主机1的包的dst\_mac值变为aa:bb:cc:dd:ee:22，说明mac地址修改成功。结果符合协议规范，测试通过。

1. 70.190：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:set\_field=ip\_dst:192.168.100.106,output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply:set\_field=ip\_dst:192.168.100.105,output=1*

打开主机1的wireshark抓包发现来自主机2的包的dst\_ip值变为192.168.100.105，打开主机2的wireshark抓包发现来自主机1的包的dst\_ip值变为192.168.100.106，说明ip修改成功。结果符合协议规范，测试通过。

1. 70.200：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply:set\_field=ip\_src:192.168.100.106,output=2*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply:set\_field=ip\_src:192.168.100.105,output=1*

打开主机1的wireshark抓包发现来自主机2的包的src\_ip值变为192.168.100.105，打开主机2的wireshark抓包发现来自主机1的包的src\_ip值变为192.168.100.106，说明ip修改成功。结果符合协议规范，测试通过。

1. 70.210：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1*

*apply:set\_field=ip\_dscp:3,output=2*

主机1发dscp不为3的数据包，主机2抓包，发现dscp的值为2.

### Test Meter

将主机1（mac地址：*00:13:3b:0f:42:1c*）、主机2（mac地址：*78:24:af:6c:41:c9*）、主机3、主机4分别接到交换机的eth1、eth2、eth3、eth4口，设置主机的ip分别为：192.168.100.1、192.168.100.2、192.168.100.3、192.168.100.4。启动交换机，待交换机启动完成以后，运行dpctl下发流表：

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=15 in\_port=1 apply: output=2 meter:1*

*./utilities/dpctl tcp:127.0.0.1:6632 flow-mod cmd=add,table=0,prio=14 in\_port=2 apply: output=1 meter:1*

在主机1通过VLC视频播放器新建udp协议的视频流，在主机2通过VLC请求主机1的视频流，此时主机2能够流畅播放视频，修改meter：

*./utilities/dpctl tcp:127.0.0.1:6632 meter-mod cmd=mod,meter=1 drop:rate=10000*

此时视频卡顿比较严重，修改meter：

*./utilities/dpctl tcp:127.0.0.1:6632 meter-mod cmd=mod,meter=1 drop:rate=100000*

此时视频恢复流畅，结果符合协议规范，测试通过。