

**NAME**

mpls - mpls manipulation module

**SYNOPSIS**

**tc ... action mpls** { *POP* | *PUSH* | *MODIFY* | *dec\_ttl* } [ *CONTROL* ]

*POP* := **pop** protocol *MPLS\_PROTO*

*PUSH* := **push** [ protocol *MPLS\_PROTO* ] [ **tc** *MPLS\_TC* ] [ **ttl** *MPLS\_TTL* ] [ **bos** *MPLS\_BOS* ] **label** *MPLS\_LABEL*

*MODIFY* := **modify** [ **label** *MPLS\_LABEL* ] [ **tc** *MPLS\_TC* ] [ **ttl** *MPLS\_TTL* ]

*CONTROL* := { **reclassify** | **pipe** | **drop** | **continue** | **pass** | **goto chain** *CHAIN\_INDEX* }

**DESCRIPTION**

The **mpls** action performs mpls encapsulation or decapsulation on a packet, reflected by the operation modes *POP*, *PUSH*, *MODIFY* and *DEC\_TTL*. The *POP* mode requires the ethertype of the header that follows the MPLS header (e.g. IPv4 or another MPLS). It will remove the outer MPLS header and replace the ethertype in the MAC header with that passed. The *PUSH* and *MODIFY* modes update the current MPLS header information or add a new header. *PUSH* requires at least an *MPLS\_LABEL*. *DEC\_TTL* requires no arguments and simply subtracts 1 from the MPLS header TTL field.

**OPTIONS**

**pop** Decapsulation mode. Requires the protocol of the next header.

**push** Encapsulation mode. Requires at least the **label** option.

**modify** Replace mode. Existing MPLS tag is replaced. **label**, **tc**, and **ttl** are all optional.

**dec\_ttl** Decrement the TTL field on the outer most MPLS header.

**label** *MPLS\_LABEL*

Specify the MPLS LABEL for the outer MPLS header. *MPLS\_LABEL* is an unsigned 20bit integer, the format is detected automatically (e.g. prefix with '0x' for hexadecimal interpretation, etc.).

**protocol** *MPLS\_PROTO*

Choose the protocol to use. For push actions this must be **mpls\_uc** or **mpls\_mc** (**mpls\_uc** is the default). For pop actions it should be the protocol of the next header. This option cannot be used with modify.

**tc** *MPLS\_TC*

Choose the TC value for the outer MPLS header. Decimal number in range of 0-7. Defaults to 0.

**ttl** *MPLS\_TTL*

Choose the TTL value for the outer MPLS header. Number in range of 0-255. A non-zero default value will be selected if this is not explicitly set.

**bos** *MPLS\_BOS*

Manually configure the bottom of stack bit for an MPLS header push. The default is for TC to automatically set (or unset) the bit based on the next header of the packet.

*CONTROL*

How to continue after executing this action.

**reclassify**

Restarts classification by jumping back to the first filter attached to this action's parent.

**pipe** Continue with the next action, this is the default.

**drop** Packet will be dropped without running further actions.

**continue**

Continue classification with next filter in line.

**pass** Return to calling qdisc for packet processing. This ends the classification process.

**EXAMPLES**

The following example encapsulates incoming IP packets on eth0 into MPLS with a label 123 and sends them out eth1:

```
#tc qdisc add dev eth0 handle ffff: ingress
#tc filter add dev eth0 protocol ip parent ffff: flower \
    action mpls push protocol mpls_uc label 123 \
    action mirred egress redirect dev eth1
```

In this example, incoming MPLS unicast packets on eth0 are decapsulated and to ip packets and output to eth1:

```
#tc qdisc add dev eth0 handle ffff: ingress
#tc filter add dev eth0 protocol mpls_uc parent ffff: flower \
    action mpls pop protocol ipv4 \
    action mirred egress redirect dev eth0
```

**SEE ALSO**

**tc(8)**