Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'tc-vlan.8' command

$ man tc-vlan.8

VLAN manipulation action in tc(8) Linux VLAN manipulation action in tc(8)

NAME

vlan - vlan manipulation module

SYNOPSIS

tc ... action vlan { pop | pop_eth | PUSH | MODIFY | PUSH_ETH } [CON?

TROL]

PUSH := push [protocol VLANPROTO] [priority VLANPRIO] id VLANID

MODIFY := modify [protocol VLANPROTO] [priority VLANPRIO] id

VLANID

PUSH_ETH := push_eth dst_mac LLADDR src_mac LLADDR

CONTROL := { reclassify | pipe | drop | continue | pass | goto chain

CHAIN_INDEX }

DESCRIPTION

The vlan action allows one to perform 802.1Q en- or decapsulation on a packet, reflected by the operation modes POP, PUSH and MODIFY. The POP mode is simple, as no further information is required to just drop the outer-most VLAN encapsulation. The PUSH and MODIFY modes require at least a VLANID and allow one to optionally choose the VLANPROTO to use. The vlan action can also be used to add or remove the base Ethernet header. The pop_eth mode, which takes no argument, is used to remove the base Ethernet header. All existing VLANs must have been previously dropped. The opposite operation, adding a base Ethernet header, is done with the push_eth mode. In that case, the packet must have no MAC

header (stacking MAC headers is not permitted). This mode is mostly useful when a previous action has encapsulated the whole original frame behind a network header and one needs to prepend an Ethernet header be? fore forwarding the resulting packet.

OPTIONS

pop Decapsulation mode, no further arguments allowed.

push Encapsulation mode. Requires at least id option.

modify Replace mode. Existing 802.1Q tag is replaced. Requires at least id option.

pop_eth

Ethernet header decapsulation mode. Only works on a plain Ether? net header: VLANs, if any, must be removed first.

push_eth

Ethernet header encapsulation mode. The Ethertype is automati? cally set using the network header type. Chaining Ethernet head? ers is not allowed: the packet must have no MAC header when us? ing this mode. Requires the dst_mac and src_mac options.

id VLANID

Specify the VLAN ID to encapsulate into. VLANID is an unsigned 16bit integer, the format is detected automatically (e.g. prefix with '0x' for hexadecimal interpretation, etc.).

protocol VLANPROTO

Choose the VLAN protocol to use. At the time of writing, the kernel accepts only 802.1Q or 802.1ad.

priority VLANPRIO

Choose the VLAN priority to use. Decimal number in range of 0-7.

dst_mac LLADDR

Choose the destination MAC address to use.

src_mac LLADDR

Choose the source MAC address to use.

CONTROL

How to continue after executing this action.

reclassify

Restarts classification by jumping back to the first fil?

ter 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 ICMP packets on eth0 from 10.0.0.2 into VLAN ID 123:

```
#tc qdisc add dev eth0 handle ffff: ingress
#tc filter add dev eth0 parent ffff: pref 11 protocol ip \
    u32 match ip protocol 1 0xff flowid 1:1 \
      match ip src 10.0.0.2 flowid 1:1 \
    action vlan push id 123
```

Here is an example of the pop function: Incoming VLAN packets on eth0 are decapsulated and the classification process then restarted for the plain packet:

```
#tc qdisc add dev eth0 handle ffff: ingress
#tc filter add dev $ETH parent ffff: pref 1 protocol 802.1Q \
    u32 match u32 0 0 flowid 1:1 \
    action vlan pop reclassify
```

For an example of the pop_eth and push_eth modes, see tc-mpls(8).

SEE ALSO

tc(8), tc-mpls(8)

iproute2 12 Jan 2015VLAN manipulation action in tc(8)