

**NAME**

**ovs-tcpdump** – Dump traffic from an Open vSwitch port using **tcpdump**.

**SYNOPSIS**

**ovs-tcpdump** *-i port tcpdump options...*

**DESCRIPTION**

**ovs-tcpdump** creates switch mirror ports in the **ovs-vswitchd** daemon and executes **tcpdump** to listen against those ports. When the **tcpdump** instance exits, it then cleans up the mirror port it created.

**ovs-tcpdump** will not allow multiple mirrors for the same port. It has some logic to parse the current configuration and prevent duplicate mirrors.

The **-i** option may not appear multiple times.

It is important to note that under **Linux** based kernels, tap devices do not receive packets unless the specific tuntap device has been opened by an application. This requires **CAP\_NET\_ADMIN** privileges, so the **ovs-tcpdump** command must be run as a user with such permissions (this is usually a super-user).

**OPTIONS**

**-h**

**--help** Prints a brief help message to the console.

**-V**

**--version**

Prints version information to the console.

**--db-sock**

The Open vSwitch database socket connection string. The default is *unix:/usr/local/var/run/open-vswitch/db.sock*

**--dump-cmd**

The command to run instead of **tcpdump**.

**-i**

**--interface**

The interface for which a mirror port should be created, and packets should be dumped.

**--mirror-to**

The name of the interface which should be the destination of the mirrored packets. The default is *miINTERFACE*

**SEE ALSO**

**ovs-appctl(8)**, **ovs-vswitchd(8)**, **ovs-pcap(1)**, **ovs-tcpundump(1)**, **tcpdump(8)**, **wireshark(8)**.