

NAME

usbdump - dump traffic on USB host controller

SYNOPSIS

```
usbdump [-d [ugen]B] [-d [ugen]B.D] [-d [ugen]B.D.E] [-i ifname] [-r file] [-s snaplen] [-v] [-w file]  
[-f filter] [-b file] [-h]
```

DESCRIPTION

The **usbdump** utility provides a way to dump USB packets on host controllers.

The following options are accepted:

-d [*ugen*]*bus*

Shortcut for **-i** option. The argument may be prefixed by "ugen". The option may be specified multiple times, but the bus specified must be the same.

-d [*ugen*]*bus.device*

Shortcut for **-i** and **-f** options. The argument may be prefixed by "ugen". The option may be specified multiple times, but the bus specified must be the same.

-d [*ugen*]*bus.device.endpoint*

Shortcut for **-i** and **-f** options. The argument may be prefixed by "ugen". The option may be specified multiple times, but the bus specified must be the same.

-b file Store data part of the USB trace in binary format to the given *file*. This option also works with the **-r** and **-f** options.

-i ifname

Listen on USB bus interface *ifname*.

-r file Read the raw packets from *file*. This option also works with the **-f** option.

-s snaplen

Snapshot *snaplen* bytes from each packet.

-v Enable debugging messages. When defined multiple times the verbosity level increases.

-w file Write the raw packets to *file*. This option also works with the **-s** and **-v** options.

-f filter The filter argument consists of either one or two numbers separated by a dot. The first

indicates the device unit number which should be traced. The second number which is optional indicates the endpoint which should be traced. To get all traffic for the control endpoint, two filters should be created, one for endpoint 0 and one for endpoint 128. If 128 is added to the endpoint number that means IN direction, else OUT direction is implied. A device unit or endpoint value of -1 means ignore this field. If no filters are specified, all packets are passed through using the default -1,-1 filter. This option can be specified multiple times.

-h This option displays a summary of the command line options.

EXAMPLES

Capture the USB raw packets on usb2:

```
usbdump -i usb2 -s 256 -v
```

Dump the USB raw packets of usb2 into the file without packet size limit:

```
usbdump -i usb2 -s 0 -w /tmp/dump_pkts
```

Dump the USB raw packets of usb2, but only the control endpoint traffic of device unit number 3:

```
usbdump -i usb2 -s 0 -f 3.0 -f 3.128 -w /tmp/dump_pkts
```

Read and display the USB raw packets from previous file:

```
usbdump -r /tmp/dump_pkts -v
```

OUTPUT FORMAT

The output format of **usbdump** is as follows:

```
<time> <bus>.<addr> <ep> <xfertype> <S/D> (<frames>/<length>) <...>
```

The meaning of the output format elements is as follows:

<time> A timestamp preceding all output lines. The timestamp has the format "hh:mm:ss.frac" and is as accurate as the kernel's clock.

<bus> The USB host controller's bus unit number.

<addr> The unique number of the USB device as allocated by the host controller driver.

- <ep> The USB endpoint address that indicates whether the address is OUT or IN.
- <xfer-type> The USB transfer type. Can be CTRL, ISOC, BULK or INTR.
- <S/D> 'S' indicates a USB submit. 'D' indicates a USB transfer done.
- <frames> Numbers of frames in this packets. If this is a USB submit, its value is xfer->nframes which means how many frames are acceptable or registered to transfer. If this is a USB done, xfer->aframes is the actual number of frames.
- <length> Total packet size. If this is a USB submit, its value is xfer->sumlen. If this is a USB done, its value is xfer->actlen.
- <...> Optional field used for printing an error string if the packet is from USB done.

SEE ALSO

usbconfig(8)

AUTHORS

Weongyo Jeong <weongyo@FreeBSD.org>