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

Rocky Enterprise Linux 9.2 Manual Pages on command 'libinput-debug-events.1'

$ man libinput-debug-events.1

libinput-debug-events(1)   libinput Manual   libinput-debug-events(1)

NAME

    libinput-debug-events - debug helper for libinput

SYNOPSIS

    libinput debug-events [options]

    libinput debug-events [options] --udev <seat>

    libinput debug-events [options] [--device] /dev/input/event0 [/dev/in?

    put/event1...]

DESCRIPTION

    The libinput debug-events tool creates a libinput context and prints

    all events from these devices.

    This is a debugging tool only, its output may change at any time. Do

not rely on the output.

This tool usually needs to be run as root to have access to the
/dev/input/eventX nodes.

OPTIONS

--device /dev/input/event0

Use the given device(s) with the path backend. The --device ar?
gument may be omitted.

--grab Exclusively grab all opened devices. This will prevent events
from being delivered to the host system.

--help Print help

--quiet Only print libinput messages, don't print anything from this
tool. This is useful in combination with --verbose for internal
state debugging.

--show-keycodes

Key events shown by this tool are partially obfuscated to avoid
passwords and other sensitive information showing up in the
output. Use the --show-keycodes argument to make all keycodes
visible.

--udev <seat>

Use the udev backend to listen for device notifications on the
given seat.   The default behavior is equivalent to --udev
"seat0".

--verbose

Use verbose output

libinput configuration options

   --apply-to="pattern"

      Configuration options are only applied where the device name

      matches the pattern. This pattern has no effect on the --dis‐

      able-sendevents option.

   --disable-sendevents="pattern"

      Set the send-events option to disabled for the devices matching

      patterns.   This option is not affected by the --apply-to="pat‐

      tern" option.

   --enable-tap|--disable-tap

      Enable or disable tap-to-click

   --enable-drag|--disable-drag

      Enable or disable tap-and-drag

   --enable-drag-lock|--disable-drag-lock

      Enable or disable drag-lock

   --enable-natural-scrolling|--disable-natural-scrolling

      Enable or disable natural scrolling

   --enable-left-handed|--disable-left-handed

      Enable or disable left handed button configuration

   --enable-middlebutton|--disable-middlebutton

      Enable or disable middle button emulation

   --enable-dwt|--disable-dwt

      Enable or disable disable-while-typing

   --enable-scroll-button-lock|--disable-scroll-button-lock

Enable or disable the scroll button lock

--set-click-method=[none|clickfinger|buttonareas]

Set the desired click method

--set-scroll-method=[none|twofinger|edge|button]

Set the desired scroll method

--set-scroll-button=BTN_MIDDLE

Set the button to the given button code

--set-profile=[adaptive|flat]

Set pointer acceleration profile

--set-speed=<value>

Set pointer acceleration speed. The allowed range is [-1, 1].

--set-tap-map=[lrm|lmr]

Set button mapping for tapping

NOTES

Events shown by this tool may not correspond to the events seen by a
different user of libinput. This tool initializes a separate context.

LIBINPUT

Part of the libinput(1) suite

libinput 1.19.3   libinput-debug-events(1)