Rocky Enterprise Linux 9.2 Manual Pages on command 'libinput.4'

$ man libinput.4

LIBINPUT(4) Kernel Interfaces Manual LIBINPUT(4)

NAME

libinput - libinput-based X.Org input driver

SYNOPSIS

```
Section "InputDevice"
  Identifier "devname"
  Driver "libinput"
  Option "Device"   "devpath"
  ...
EndSection
```

NOTE

This is the man page for the X input driver. If you are looking for the library documentation, go to http://wayland.freedesktop.org/libin? put/doc/

DESCRIPTION

libinput is an Xorg input driver based on libinput. It therefore sup? ports all input devices that libinput can handle, including most mice, keyboards, tablets and touchscreens.

It is recommended that libinput devices are configured through the In? putClass directive (refer to xorg.conf(5)) instead of manual per-device configuration. Devices configured in the xorg.conf(5) are not hot-plug capable.

CONFIGURATION DETAILS

Please refer to xorg.conf(5) for general configuration details and for options that can be used with all input drivers. This section only covers configuration details specific to this driver.

The following driver Options are supported:

Option "AccelProfile" "string"

Sets the pointer acceleration profile to the given profile. Per? mitted values are adaptive, flat. Not all devices support this option or all profiles. If a profile is unsupported, the default profile for this device is used. For a description on the pro? files and their behavior, see the libinput documentation.

Option "AccelSpeed" "float"

Sets the pointer acceleration speed within the range [-1, 1]

Option "ButtonMapping" "string"

Sets the logical button mapping for this device, see XSetPoint? erMapping(3). The string must be a space-separated list of but? ton mappings in the order of the logical buttons on the device, starting with button 1. The default mapping is "1 2 3 ... 32".

A mapping of 0 deactivates the button. Multiple buttons can have the same mapping. Invalid mapping strings are discarded and the default mapping is used for all buttons. Buttons not specified in the user's mapping use the default mapping. See section BUT? TON MAPPING for more details.

Option "CalibrationMatrix" "string"

A string of 9 space-separated floating point numbers, in the or? der "a b c d e f g h i". Sets the calibration matrix to the 3x3 matrix where the first row is (abc), the second row is (def) and the third row is (ghi).

Option "ClickMethod" "string"

Enables a click method. Permitted values are none, buttonareas, clickfinger. Not all devices support all methods, if an option is unsupported, the default click method for this device is used.

Option "DisableWhileTyping" "bool"

Indicates if the touchpad should be disabled while typing on the keyboard (this does not apply to modifier keys such as Ctrl or Alt).

Option "Device" "string"

Specifies the device through which the device can be accessed. This will generally be of the form "/dev/input/eventX", where X is some integer. When using InputClass directives, this option is set by the server. The mapping from device node to hardware is system-dependent. Property: "Device Node" (read-only).

Option "DragLockButtons" "L1 B1 L2 B2 ..."

Sets "drag lock buttons" that simulate a button logically down even when it has been physically released. To logically release a locked button, a second click of the same button is required.

If the option is a single button number, that button acts as the "meta" locking button for the next button number. See section BUTTON DRAG LOCK for details.

If the option is a list of button number pairs, the first number of each number pair is the lock button, the second number the logical button number to be locked. See section BUTTON DRAG LOCK for details.

For both meta and button pair configuration, the button numbers are device button numbers, i.e. the ButtonMapping applies after drag lock.

Option "HorizontalScrolling" "bool"

Disables horizontal scrolling. When disabled, this driver will discard any horizontal scroll events from libinput. Note that this does not disable horizontal scrolling, it merely discards the horizontal axis from any scroll events.

Option "LeftHanded" "bool"

Enables left-handed button orientation, i.e. swapping left and right buttons.

Option "MiddleEmulation" "bool"

Enables middle button emulation. When enabled, pressing the left and right buttons simultaneously produces a middle mouse button click.

Option "NaturalScrolling" "bool"

Enables or disables natural scrolling behavior.

Option "RotationAngle" "float"

Sets the rotation angle of the device to the given angle, in de? grees clockwise. The angle must be between 0.0 (inclusive) and 360.0 (exclusive).

Option "ScrollButton" "int"

Designates a button as scroll button. If the ScrollMethod is button and the button is logically down, x/y axis movement is converted into scroll events.

Option "ScrollButtonLock" "bool"

Enables or disables the scroll button lock. If enabled, the ScrollButton is considered logically down after the first click and remains down until the second click of that button. If dis? abled (the default), the ScrollButton button is considered logi? cally down while held down and up once physically released.

Option "ScrollMethod" "string"

Enables a scroll method. Permitted values are none, twofinger, edge, button. Not all devices support all options, if an option is unsupported, the default scroll option for this device is used.

Option "SendEventsMode" "(disabled|enabled|disabled-on-external-mouse)"

Sets the send events mode to disabled, enabled, or "disable when an external mouse is connected".

Option "TabletToolPressureCurve" "x0/y0 x1/y1 x2/y2 x3/y3"

Set the pressure curve for a tablet stylus to the bezier formed

by the four points. The respective x/y coordinate must be in the [0.0, 1.0] range. For more information see section TABLET STYLUS PRESSURE CURVE.

Option "TabletToolAreaRatio" "w:h"

Sets the area ratio for a tablet tool. The area always starts at the origin (0/0) and expands to the largest available area with the specified aspect ratio. Events outside this area are cropped to the area. The special value "default" is used for the default mapping (i.e. the device-native mapping). For more information see section TABLET TOOL AREA RATIO.

Option "Tapping" "bool"

Enables or disables tap-to-click behavior.

Option "TappingButtonMap" "(lrm|lmr)"

Set the button mapping for 1/2/3-finger taps to left/right/mid? dle or left/middle/right, respectively.

Option "TappingDrag" "bool"

Enables or disables drag during tapping behavior ("tap-and-drag"). When enabled, a tap followed by a finger held down causes a single button down only, all motions of that finger thus translate into dragging motion. Tap-and-drag requires op? tion Tapping to be enabled.

Option "TappingDragLock" "bool"

Enables or disables drag lock during tapping behavior. When en? abled, a finger up during tap-and-drag will not immediately re? lease the button. If the finger is set down again within the timeout, the dragging process continues.

For all options, the options are only parsed if the device supports that configuration option. For all options, the default value is the one used by libinput. On configuration failure, the default value is applied.

SUPPORTED PROPERTIES

libinput exports runtime-configurable options as properties. If a prop? erty listed below is not available, the matching configuration option

is not available on the device. This however does not imply that the feature is not available on the device. The following properties are provided by the libinput driver.

libinput Accel Profiles Available

2 boolean values (8 bit, 0 or 1), in order "adaptive", "flat". Indicates which acceleration profiles are available on this de? vice.

libinput Accel Profile Enabled

2 boolean values (8 bit, 0 or 1), in order "adaptive", "flat". Indicates which acceleration profile is currently enabled on this device.

libinput Accel Speed

1 32-bit float value, defines the pointer speed. Value range -1, 1

libinput Button Scrolling Button

1 32-bit value. Sets the button number to use for button scrolling. This setting is independent of the scroll method, to enable button scrolling the method must be set to button-scrolling and a valid button must be set.

libinput Button Scrolling Button Lock Enabled

1 boolean value. If true, the scroll button lock is enabled. This setting is independent of the scroll method or the scroll button, to enable button scrolling the method must be set to button-scrolling and a valid button must be set.

libinput Calibration Matrix

9 32-bit float values, representing a 3x3 calibration matrix, order is row 1, row 2, row 3

libinput Click Methods Available

2 boolean values (8 bit, 0 or 1), in order "buttonareas", "clickfinger". Indicates which click methods are available on this device.

libinput Click Methods Enabled

2 boolean values (8 bit, 0 or 1), in order "buttonareas",

"clickfinger". Indicates which click methods are enabled on this device.

libinput Drag Lock Buttons

Either one 8-bit value specifying the meta drag lock button, or a list of button pairs. See section BUTTON DRAG LOCK for de? tails.

libinput Horizontal Scrolling Enabled

1 boolean value (8 bit, 0 or 1). Indicates whether horizontal scrolling events are enabled or not.

libinput Left Handed Enabled

1 boolean value (8 bit, 0 or 1). Indicates if left-handed mode is enabled or disabled.

libinput Middle Emulation Enabled

1 boolean value (8 bit, 0 or 1). Indicates if middle emulation is enabled or disabled.

libinput Natural Scrolling Enabled

1 boolean value (8 bit, 0 or 1). 1 enables natural scrolling

libinput Rotation Angle

1 32-bit float value [0.0 to 360.0). Sets the rotation angle of the device, clockwise of its natural neutral position.

libinput Scroll Methods Available

3 boolean values (8 bit, 0 or 1), in order "two-finger", "edge", "button". Indicates which scroll methods are available on this device.

libinput Scroll Method Enabled

3 boolean values (8 bit, 0 or 1), in order "two-finger", "edge", "button". Indicates which scroll method is currently enabled on this device.

libinput Send Events Modes Available

2 boolean values (8 bit, 0 or 1), in order "disabled" and "dis? abled-on-external-mouse". Indicates which send-event modes are available on this device.

libinput Send Events Mode Enabled

2 boolean values (8 bit, 0 or 1), in order "disabled" and "dis?abled-on-external-mouse". Indicates which send-event modes is currently enabled on this device.

libinput Tablet Tool Pressurecurve

4 32-bit float values [0.0 to 1.0]. See section TABLET TOOL PRESSURE CURVE

libinput Tablet Tool Area Ratio

2 32-bit values, corresponding to width and height. Special value 0, 0 resets to the default ratio. See section TABLET TOOL AREA RATIO for more information.

libinput Tapping Enabled

1 boolean value (8 bit, 0 or 1). 1 enables tapping

libinput Tapping Button Mapping Enabled

2 boolean value (8 bit, 0 or 1), in order "lrm" and "lmr". Indi?cates which button mapping is currently enabled on this device.

libinput Tapping Drag Lock Enabled

1 boolean value (8 bit, 0 or 1). 1 enables drag lock during tap?ping

libinput Disable While Typing Enabled

1 boolean value (8 bit, 0 or 1). Indicates if disable while typ?ing is enabled or disabled.

Most properties have a libinput <property name> Default equivalent that indicates the default value for this setting on this device.

BUTTON MAPPING

X clients receive events with logical button numbers, where 1, 2, 3 are usually interpreted as left, middle, right and logical buttons 4, 5, 6, 7 are usually interpreted as scroll up, down, left, right. The fourth and fifth physical buttons on a device will thus send logical buttons 8 and 9. The ButtonMapping option adjusts the logical button mapping, it does not affect how a physical button is mapped to a logical button. Traditionally, a device was set to left-handed button mode by applying a button mapping of "3 2 1 ..." On systems using the libinput Xorg in?put driver it is recommended to use the LeftHanded option instead.

The libinput Xorg input driver does not use the button mapping after setup. Use XSetPointerMapping(3) to modify the button mapping at run? time.

BUTTON DRAG LOCK

Button drag lock holds a button logically down even when the button it? self has been physically released since. Button drag lock comes in two modes.

If in "meta" mode, a meta button click activates drag lock for the next button press of any other button. A button click in the future will keep that button held logically down until a subsequent click of that same button. The meta button events themselves are discarded. A sepa? rate meta button click is required each time a drag lock should be ac? tivated for a button in the future.

If in "pairs" mode, each button can be assigned a target locking but? ton. On button click, the target lock button is held logically down until the next click of the same button. The button events themselves are discarded and only the target button events are sent.

This feature is provided by this driver, not by libinput.

TABLET TOOL PRESSURECURVE

The pressure curve affects how stylus pressure is reported. By default, the hardware pressure is reported as-is. By setting a pressure curve, the feel of the stylus can be adjusted to be more like e.g. a pencil or a brush.

The pressure curve is a cubic Bezier curve, drawn within a normalized range of 0.0 to 1.0 between the four points provided. This normalized range is applied to the tablet's pressure input so that the highest pressure maps to 1.0. The points must have increasing x coordinates, if x0 is larger than 0.0 all pressure values lower than x0 are equivalent to y0. If x3 is less than 1.0, all pressure values higher than x3 are equivalent to y3.

The input for a linear curve (default) is "0.0/0.0 0.0/0.0 1.0/1.0 1.0/1.0"; a slightly depressed curve (firmer) might be "0.0/0.0 0.05/0.0 1.0/0.95 1.0/1.0"; a slightly raised curve (softer) might be

"0.0/0.0 0.0/0.05 0.95/1.0 1.0/1.0".

This feature is provided by this driver, not by libinput.

TABLET TOOL AREA RATIO

By default, a tablet tool can access the whole sensor area and the tablet area is mapped to the available screen area. For external tablets like the Wacom Intuos series, the height:width ratio of the tablet may be different to that of the monitor, causing the skew of in? put data.

To avoid this skew of input data, an area ratio may be set to match the ratio of the screen device. For example, a ratio of 4:3 will reduce the available area of the tablet to the largest available area with a ratio of 4:3. Events within this area will scale to the tablet's announced axis range, the area ratio is thus transparent to the X server. Any events outside this area will send events equal to the maximum value of that axis. The area always starts at the device's origin in it's cur? rent rotation, i.e. it takes left-handed-ness into account.

This feature is provided by this driver, not by libinput.

BUGS

This driver does not work with Option "Device" set to an event node in /dev/input/by-id and /dev/input/by-path. This can be usually be worked by using Section "InputClass" with an appropriate Match* statement in the xorg.conf(5).

This driver does not know about the display pixel density and submits motion events assuming an approximate display density of 96dpi. On high-dpi screens this results in a slower physical motion of the cursor (a one-pixel movement is a smaller physical movement on the screen). This can make interaction with the desktop difficult.

Option "DPIScaleFactor" float

This is a temporary solution. The factor should be set to the approximate ratio of the host display compared to the default 96dpi. For example, a display with 200dpi should set a factor of 2.0.

If set, x/y motion will be unconditionally multiplied by this factor,

resulting in faster movement of the cursor. Note that this may make some pixels unadressable and should be used with caution.

This option is a temporary solution. It may be removed in any future update of this driver.

AUTHORS

Peter Hutterer

SEE ALSO

Xorg(1), xorg.conf(5), Xserver(1), X(7)