

- [Home](#)

## As if it mattered.

Tech-oriented howtos & musings

### Setting up custom buttons and scrolling for Kensington trackballs under Linux part 2 (Wayland version)

01 Jul 2024 at 18:04

admin

[Gentoo](#), [Linux](#)

In an [earlier article](#) I documented how I set up a custom configuration for my trackball's buttons and scroll direction on Linux when using the X11 windows server. KDE Plasma 6 brings the Wayland window server to my Linux installation, replacing X11, so I've had to think again how to implement my custom config.

The general design philosophy of Wayland seems to be that it's up to the desktop environment (such as KDE Plasma or Gnome) to provide configuration options – Wayland itself is very light. Plasma 6's System Settings supports left-handed mode (that is, switching the behaviour of the left and right buttons) and rotating the ball to scroll works out of the box, though it is not possible to alter the scroll direction. The deal-breaker though was lack of support for Click Lock AKA Drag Lock (a button to 'pick up' items so they can be moved without holding down a trackball button).

#### First solution using input-remapper

I initially achieved Drag Lock using [input-remapper](#), writing this macro as the output for the selected button :

```
if_eq($a, 0, key_down(BTN_LEFT).set(a,1),key_up(BTN_LEFT).set(a,0))
```

It defines a variable 'a' to keep state. If a==0 then send the 'left button down' event and set a=1. If a==1 then send the 'left button up' event and set a=0.

Input-remapper however entirely disregards the behaviour configured in Plasma System Settings, so I needed to additionally configure swapping left and right buttons. and I was unsuccessful in using it to reverse the scroll direction. Additionally, input-remapper depends on systemd to configure it as a background service whereas I still use SysV init, so this solution saw me needing to manually reapply the mapping every time I switched the trackball to the Gentoo machine which would get very old very quickly.

#### Better solution using evsieve

As I kept googling around for solutions I came across the rather superb [evsieve](#) project. On Linux, all input devices send input events via the evdev protocol. evsieve is a pure command-line utility for filtering and altering that stream of events. Its command set is very extensive and very well thought-through, covering all manner of special and edge cases. My solution was as follows:

```
evsieve --input /dev/input/by-id/usb-047d_Kensington_Slimblade_Trackball-event-mouse grab
persist=reopen \
--map yield btn:left btn:right \
--map btn:right btn:left \
--hook btn:middle toggle \
--toggle btn:middle btn:left:0 btn:left:1 \
--output
```

Explaining each element in turn:

- the `--input` line selects the device we wish to monitor for events. ‘grab’ means that evsieve is the exclusive consumer of that device’s events; the desktop environment only sees evsieve’s output. Events that aren’t part of the mapping pass through unchanged, whilst events that match the mapping get altered. ‘persist=reopen’ is a killer feature – it means that the configuration remains in place even if the input device disappears. In my case, where I use KVM switch, this saves me having to get into udev event detection to set up evsieve every time I switch devices.
- the first `--map` line turns left button clicks into right button clicks. Since evsieve works such that, by default, the output of each parameter act as input to the following one, ‘yield’ breaks that chain. Without this, the subsequent mapping would just reverse this one.
- the second `--map` line turns right button clicks into left button clicks.
- the `--hook` line is used to flip the toggle (described in the next parameter) whenever the middle button is clicked (the top-left button on Kensington trackballs sends the middle button event).
- the `--toggle` parameter sends either a left button down or left button up event when the middle button is clicked, depending upon the state of the toggle.
- the `--output` line is unconfigured, which means the complete stream of output events gets sent from a virtual input device that evsieve sets up automatically.

Since altering the event stream requires root privileges, and because I want this running all the time, I set up [a SysV init wrapper](#) and installed it as a service to call the command line above during boot. A systemd equivalent is left as an exercise for the reader :)

## Conclusion

This configuration is surprisingly simpler and more reliable than [the solution I’d arrived at for X11](#), even though Wayland offers no features to assist at all. I have given in on the scroll wheel direction since it was not obvious how to achieve my previous preference, but the muscle memory reprogramming only took a couple of days to kick in. I am a happy camper :)

← [Offsite backup with TrueNAS Core, Duplicacy and an SFTP destination with public key authentication](#)  
[Achieving flow-state whilst working](#) →

## 1 Comment

- by **Ben in Seattle**

**09 Mar 2025** at [03:04](#)

[Reply](#)

Thank you! Finally a solution that doesn't just say "Wayland doesn't do that; ask your Desktop Environment for support." I was just about to switch back to X11 before I found your article on `evsieve`.

Now, if only I could find a way to setup acceleration for my trackball the way I used to do with xinput. (I know one can set the DPI in `/etc/udev/hwdb.d`, but libinput only uses it to scale if your device's DPI is over 1,000.)

## Leave a Reply

Name (required)

Mail (will not be published) (required)

Website

Submit Comment

Time limit is exhausted. Please reload CAPTCHA.

six +  = 13 

Search

**Privoxy**

[Main site](#)  
[Downloads](#)

## BASH shell scripts

[Sync playlists to Android](#)  
[Gentoo emerge tools](#)  
[Tomcat access log parser](#)

## macOS shell scripts

[Firewall hardening](#)  
[Email virus-scanning](#)  
[Scheduled virus scans](#)  
[Configuring clamd](#)  
[iTunes backup](#)

## Categories

- [Android](#) (4)
- [Apple](#) (5)
- [Audio](#) (1)
- [Firefox](#) (2)
- [FreeBSD](#) (4)
- [FreeNAS](#) (7)
- [Gentoo](#) (10)
- [iPhone](#) (6)
- [Linux](#) (14)
- [Networking](#) (3)
- [Nook](#) (3)
- [OS X](#) (11)
- [Privacy](#) (6)
- [TrueNAS Core](#) (7)
- [Uncategorized](#) (3)

## Archives

- [January 2025](#)
- [December 2024](#)
- [July 2024](#)
- [May 2024](#)
- [March 2024](#)
- [April 2023](#)
- [March 2023](#)
- [October 2022](#)
- [February 2022](#)
- [November 2021](#)

- [October 2021](#)
- [September 2021](#)
- [April 2021](#)
- [October 2020](#)
- [July 2020](#)
- [April 2020](#)
- [January 2020](#)
- [July 2019](#)
- [May 2019](#)
- [April 2019](#)
- [January 2019](#)
- [September 2018](#)
- [June 2018](#)
- [April 2018](#)
- [May 2016](#)
- [April 2016](#)
- [November 2014](#)
- [February 2014](#)
- [October 2013](#)
- [September 2013](#)
- [June 2013](#)
- [March 2013](#)
- [December 2012](#)
- [September 2012](#)
- [March 2012](#)
- [November 2011](#)
- [January 2011](#)
- [December 2010](#)
- [May 2010](#)
- [April 2010](#)

- **Blogroll**

- [Asbestos Salamander](#)
- [Mack Diesel's Musings](#)
- [Rixstep](#)
- [Technically Sound](#)
- [Virtual Hiker](#)

As if it mattered. is proudly powered by [WordPress](#). Theme by [Benoît «LeBen» Burgener](#) for [My Tapestry](#)

[RSS → Entries](#) | [RSS → Comments](#)