

Under construction, but this needs to be here as it's mentioned literally *no where else*.

About udev

[udev](#) stands for *userspace / dev* and is a device manager for the Linux kernel. As a user with *root* privileges you have the ability to configure a lot of device specific stuff via udev rules, assuming you know what you are doing. udev was created to respond to hotplug type of events. In a nutshell, udev is being used to do various things whenever a device or a specific (device) event is being detected, no matter whether on boot time, on system shutdown or even on-the-fly (e.g. when a device is being plugged in/out while the system is running). In Linux, udev is running as a service/daemon (called **udev**). The times when udevd is active are:

- At startup, it parses all the config files and rule files and builds a rules database in memory.
- When an event happens, it checks its rule database and performs the appropriate actions.

In a nutshell, udev rules do always have two major parts:

- **Matches** ("If this condition(s) match(es)"...) → The operator for matches are: == and !=
- **Actions** (... "then do this!") → The operator for actions is: =

udev rules in Batocera

Batocera supports the use of udev rules just like any other ordinary Linux based operating system, however the path to use in Batocera is different from most other Linux systems. You can transfer your udev rules over to `/userdata/system/udev/rules.d/` and they will be enabled once Batocera has finished booting.

If you need something activated earlier in the boot process or something more intimate to the system, you can instead use [boot scripts](#).

For example, if you wanted to enable the [Bluetooth passthrough udev rule of Dolphin](#) for your own Bluetooth dongle (list of compatible adapters **for Dolphin, not necessarily Batocera** is [further down that page](#)), you would use this file:

[52-dolphin.rules](#)

```
SUBSYSTEM=="usb", ATTRS{idVendor}=="YOURVID",  
ATTRS{idProduct}=="YOURPID", TAG+="uaccess"
```

Replace the values YOURVID and YOURPID according to your specific device found in the list mentioned above. Then save it and copy it to `/userdata/system/udev/rules.d/52-dolphin.rules` and reboot Batocera.

Deeper explanation of udev rules

As an example, let's take the udev rule `52-dolphin.rules` from above to dive a little deeper into udev rules:

`SUBSYSTEM`, `ATTRS{idVendor}` and `ATTRS{idProduct}` are all variables that do all have a `==` between them and their assigned values. This means they are all *matches* (conditions). On the other hand, the last variable `TAG` does have a `+=`, which adds itself to any other `TAG` action being executed on this specific event (e.g. by an additional udev rule matching the same event). Basically you can read this specific udev rule like this:

*If the detected device is a USB device with the vendor ID 'YOURVID' and the product ID 'YOURPID', then, in addition to any other TAG action on this specific event, tag the device with 'uaccess' (which provides access to the /dev/ file for the logged in user (in case of Batocera, mostly **root** user)).*

Viewing and editing pre-existing udev rules

Once Batocera has booted you can view the existing udev rules for your system in `/etc/udev/rules.d/`. You can do this from the file manager ([F1] on the system list) and going up one level, or [via SSH](#). You cannot access this location from the network share, as it is outside of the userdata partition.

Troubleshooting

If you find that your udev rule isn't working after reboot for any reason, you can alternatively try putting your `*.rules` file directly into `/etc/udev/rules.d/` and running `batocera-save-overlay` afterward. This will call it slightly earlier in the boot process. Remember: Everything saved to the boot partition via `batocera-save-overlay` will get lost after updating Batocera to a new version.

From:

<https://www.wiki.batocera.org/> - **Batocera.linux** - Wiki

Permanent link:

https://www.wiki.batocera.org/udev_rules?rev=1642617401

Last update: **2022/01/19 18:36**

