NAME

udevadm − udev management tool

SYNOPSIS

udevadm [**−−debug**] [**−−version**] [**−−help**]

udevadm info [options] [devpath]

udevadm trigger [options] [devpath]

udevadm settle [options]

udevadm control *option*

udevadm monitor [options]

udevadm test [options] *devpath*

udevadm test−builtin [options] *command devpath*

DESCRIPTION

udevadm expects a command and command specific options. It controls the runtime behavior of **systemd−udevd**, requests kernel events, manages the event queue, and provides simple debugging mechanisms.

OPTIONS

−d, **−−debug**
Print debug messages to standard error. This option is implied in **udevadm test** and **udevadm test−builtin** commands.

−h, **−−help**
Print a short help text and exit.

udevadm info [*options*] [*devpath*|*file*|*unit*...]
Query the udev database for device information.

Positional arguments should be used to specify one or more devices. Each one may be a device name (in which case it must start with /dev/), a sys path (in which case it must start with /sys/), or a systemd device unit name (in which case it must end with ".device", see **systemd.device**(5)).

−q, **−−query=***TYPE*
Query the database for the specified type of device data. Valid *TYPE*s are: **name**, **symlink**, **path**, **property**, **all**.

−p, **−−path=***DEVPATH*
The /sys/ path of the device to query, e.g. [/sys/]/class/block/sda. This option is an alternative to the positional argument with a /sys/ prefix. **udevadm info −−path=/class/block/sda** is equivalent to **udevadm info /sys/class/block/sda**.

−n, **−−name=***FILE*
The name of the device node or a symlink to query, e.g. [/dev/]/sda. This option is an alternative to the positional argument with a /dev/ prefix. **udevadm info −−name=sda** is equivalent to **udevadm info /dev/sda**.

−r, **−−root**
Print absolute paths in **name** or **symlink** query.

−a, **−−attribute−walk**
Print all sysfs properties of the specified device that can be used in udev rules to match the specified device. It prints all devices along the chain, up to the root of sysfs that can be used in udev rules.

−x, **−−export**
Print output as key/value pairs. Values are enclosed in single quotes. This takes effects only when **−−query=property** or **−−device−id−of−file=***FILE* is specified.

−P, **−−export−prefix=***NAME*

Add a prefix to the key name of exported values. This implies **--export**.

-d, **--device-id-of-file=***FILE*
Print major/minor numbers of the underlying device, where the file lives on. If this is specified, all positional arguments are ignored.

-e, **--export-db**
Export the content of the udev database.

-c, **--cleanup-db**
Cleanup the udev database.

-w[SECONDS], **--wait-for-initialization[=SECONDS]**
Wait for device to be initialized. If argument *SECONDS* is not specified, the default is to wait forever.

-h, **--help**
Print a short help text and exit.

The generated output shows the current device database entry in a terse format. Each line shown is prefixed with one of the following characters:

Table 1. udevadm info output prefixes

Prefix	Meaning
"P:"	Device path in /sys/
"N:"	Kernel device node name
"L:"	Device node symlink priority
"S:"	Device node symlink
"E:"	Device property

udevadm trigger [*options*] [*devpath*|*file*|*unit*]
Request device events from the kernel. Primarily used to replay events at system coldplug time.

Takes device specifications as positional arguments. See the description of **info** above.

-v, **--verbose**
Print the list of devices which will be triggered.

-n, **--dry-run**
Do not actually trigger the event.

-q, **--quiet**
Suppress error logging in triggering events.

-t, **--type=***TYPE*
Trigger a specific type of devices. Valid types are: **devices**, **subsystems**. The default value is **devices**.

-c, **--action=***ACTION*
Type of event to be triggered. Possible actions are "add", "remove", "change", "move", "online", "offline", "bind", and "unbind". Also, the special value "help" can be used to list the possible actions. The default value is "change".

-s, **--subsystem-match=***SUBSYSTEM*
Trigger events for devices which belong to a matching subsystem. This option supports shell style pattern matching. When this option is specified more than once, then each matching result is ORed, that is, all the devices in each subsystem are triggered.

-S, **--subsystem-nomatch=***SUBSYSTEM*
Do not trigger events for devices which belong to a matching subsystem. This option supports shell style pattern matching. When this option is specified more than once, then each matching result is ANDed, that is, devices which do not match all specified subsystems are triggered.

-a, **--attr-match=***ATTRIBUTE=VALUE*

Trigger events for devices with a matching sysfs attribute. If a value is specified along with the attribute name, the content of the attribute is matched against the given value using shell style pattern matching. If no value is specified, the existence of the sysfs attribute is checked. When this option is specified multiple times, then each matching result is ANDed, that is, only devices which have all specified attributes are triggered.

−A, **−−attr−nomatch=***ATTRIBUTE=VALUE*
Do not trigger events for devices with a matching sysfs attribute. If a value is specified along with the attribute name, the content of the attribute is matched against the given value using shell style pattern matching. If no value is specified, the existence of the sysfs attribute is checked. When this option is specified multiple times, then each matching result is ANDed, that is, only devices which have none of the specified attributes are triggered.

−p, **−−property−match=***PROPERTY=VALUE*
Trigger events for devices with a matching property value. This option supports shell style pattern matching. When this option is specified more than once, then each matching result is ORed, that is, devices which have one of the specified properties are triggered.

−g, **−−tag−match=***TAG*
Trigger events for devices with a matching tag. When this option is specified multiple times, then each matching result is ANDed, that is, devices which have all specified tags are triggered.

−y, **−−sysname−match=***NAME*
Trigger events for devices for which the last component (i.e. the filename) of the /sys/ path matches the specified *PATH*. This option supports shell style pattern matching. When this option is specified more than once, then each matching result is ORed, that is, all devices which have any of the specified *NAME* are triggered.

−−name−match=*NAME*
Trigger events for devices with a matching device path. When this option is specified more than once, then each matching result is ORed, that is, all specified devices are triggered.

−b, **−−parent−match=***SYSPATH*
Trigger events for all children of a given device. When this option is specified more than once, then each matching result is ORed, that is, all children of each specified device are triggered.

−w, **−−settle**
Apart from triggering events, also waits for those events to finish. Note that this is different from calling **udevadm settle**. **udevadm settle** waits for all events to finish. This option only waits for events triggered by the same command to finish.

−−uuid
Trigger the synthetic device events, and associate a randomized UUID with each. These UUIDs are printed to standard output, one line for each event. These UUIDs are included in the uevent environment block (in the "SYNTH_UUID=" property) and may be used to track delivery of the generated events.

−−wait−daemon[=*SECONDS***]**
Before triggering uevents, wait for systemd−udevd daemon to be initialized. Optionally takes timeout value. Default timeout is 5 seconds. This is equivalent to invoke invoking **udevadm control −−ping** before **udevadm trigger**.

−h, **−−help**
Print a short help text and exit.

In addition, optional positional arguments can be used to specify device names or sys paths. They must start with /dev/ or /sys/ respectively.

udevadm settle [*options*]
Watches the udev event queue, and exits if all current events are handled.

−t, **−−timeout=***SECONDS*

Maximum number of seconds to wait for the event queue to become empty. The default value is 120 seconds. A value of 0 will check if the queue is empty and always return immediately. A non−zero value will return an exit code of 0 if queue became empty before timeout was reached, non−zero otherwise.

−E, **−−exit−if−exists=***FILE*
Stop waiting if file exists.

−h, **−−help**
Print a short help text and exit.

See **systemd-udev-settle.service**(8) for more information.

udevadm control *option*
Modify the internal state of the running udev daemon.

−e, **−−exit**
Signal and wait for systemd−udevd to exit. No option except for **−−timeout** can be specified after this option. Note that systemd−udevd.service contains **Restart=always** and so as a result, this option restarts systemd−udevd. If you want to stop systemd−udevd.service, please use the following:

systemctl stop systemd−udevd−control.socket systemd−udevd−kernel.socket systemd−udevd.service

−l, **−−log−level=***value*
Set the internal log level of systemd−udevd. Valid values are the numerical syslog priorities or their textual representations: **emerg**, **alert**, **crit**, **err**, **warning**, **notice**, **info**, and **debug**.

−s, **−−stop−exec−queue**
Signal systemd−udevd to stop executing new events. Incoming events will be queued.

−S, **−−start−exec−queue**
Signal systemd−udevd to enable the execution of events.

−R, **−−reload**
Signal systemd−udevd to reload the rules files and other databases like the kernel module index. Reloading rules and databases does not apply any changes to already existing devices; the new configuration will only be applied to new events.

−p, **−−property=***KEY=value*
Set a global property for all events.

−m, **−−children−max=***value*
Set the maximum number of events, systemd−udevd will handle at the same time.

−−ping
Send a ping message to systemd−udevd and wait for the reply. This may be useful to check that systemd−udevd daemon is running.

−t, **−−timeout=***seconds*
The maximum number of seconds to wait for a reply from systemd−udevd.

−h, **−−help**
Print a short help text and exit.

udevadm monitor [*options*]
Listens to the kernel uevents and events sent out by a udev rule and prints the devpath of the event to the console. It can be used to analyze the event timing, by comparing the timestamps of the kernel uevent and the udev event.

−k, **−−kernel**
Print the kernel uevents.

−u, **−−udev**
Print the udev event after the rule processing.

−p, **−−property**
Also print the properties of the event.

−s, **−−subsystem−match=***string[/string]*
Filter kernel uevents and udev events by subsystem[/devtype]. Only events with a matching subsystem value will pass. When this option is specified more than once, then each matching result is ORed, that is, all devices in the specified subsystems are monitored.

−t, **−−tag−match=***string*
Filter udev events by tag. Only udev events with a given tag attached will pass. When this option is specified more than once, then each matching result is ORed, that is, devices which have one of the specified tags are monitored.

−h, **−−help**
Print a short help text and exit.

udevadm test [*options*] [*devpath*]
Simulate a udev event run for the given device, and print debug output.

−a, **−−action=***ACTION*
Type of event to be simulated. Possible actions are "add", "remove", "change", "move", "online", "offline", "bind", and "unbind". Also, the special value "help" can be used to list the possible actions. The default value is "add".

−N, **−−resolve−names=early|late|never**
Specify when udevadm should resolve names of users and groups. When set to **early** (the default), names will be resolved when the rules are parsed. When set to **late**, names will be resolved for every event. When set to **never**, names will never be resolved and all devices will be owned by root.

−h, **−−help**
Print a short help text and exit.

udevadm test−builtin [*options*] [*command*] [*devpath*]
Run a built−in command *COMMAND* for device *DEVPATH*, and print debug output.

−h, **−−help**
Print a short help text and exit.

SEE ALSO

udev(7), **systemd-udevd.service**(8)