NAME

devctl **-** device control utility

SYNOPSIS

devctl attach *device*
devctl clear driver [**-f**] *device*
devctl detach [**-f**] *device*
devctl disable [**-f**] *device*
devctl enable *device*
devctl suspend *device*
devctl resume *device*
devctl set driver [**-f**] *device driver*
devctl rescan *device*
devctl delete [**-f**] *device*
devctl freeze
devctl thaw
devctl reset [**-d**] *device*
devctl getpath *locator device*

DESCRIPTION

The **devctl** utility adjusts the state of individual devices in the kernel’s internal device hierarchy. Each invocation of **devctl** consists of a single command followed by command-specific arguments. Each command operates on a single device specified via the *device* argument. The *device* may be specified either as the name of an existing device or as a bus-specific address. More details on supported address formats can be found in devctl(3).

The following commands are supported:

attach *device*

Force the kernel to re-probe the device. If a suitable driver is found, it is attached to the device.

detach [**-f**] *device*

Detach the device from its current device driver. If the **-f** flag is specified, the device driver will be detached even if the device is busy.

disable [**-f**] *device*

Disable a device. If the device is currently attached to a device driver, the device driver will be detached from the device, but the device will retain its current name. If the **-f** flag is specified, the device driver will be detached even if the device is busy.

enable *device*
Enable a device. The device will probe and attach if a suitable device driver is found. Note that this can re-enable a device disabled at boot time via a loader tunable.

suspend *device*
Suspend a device. This may include placing the device in a reduced power state.

resume *device*
Resume a suspended device to a fully working state.

set driver [**-f**] *device driver*
Force the device to use a device driver named *driver*. If the device is already attached to a device driver and the **-f** flag is specified, the device will be detached from its current device driver before it is attached to the new device driver. If the device is already attached to a device driver and the **-f** flag is not specified, the device will not be changed.

clear driver [**-f**] *device*
Clear a previously-forced driver name so that the device is able to use any valid device driver. After the previous name has been cleared, the device is reprobed so that other device drivers may attach to it. This can be used to undo an earlier **set driver** command. If the device is currently attached to a device driver and the **-f** flag is not specified, the device will not be changed.

rescan *device*
Rescan a bus device checking for devices that have been added or removed.

delete [**-f**] *device*
Delete the device from the device tree. If the **-f** flag is specified, the device will be deleted even if it is physically present. This command should be used with care as a device that is deleted but present can no longer be used unless the parent bus device rediscovers the device via a rescan request.

freeze Freeze probe and attach processing initiated in response to drivers being loaded. Drivers are placed on a "frozen list" and processed when a later "thaw" occurs.

thaw Resume (thaw the freeze) probe and attach initiated in response to drivers being loaded. In addition to resuming, all pending actions that were frozen during the freeze are performed.

reset [**-d**] *device*
Reset the device, using bus-specific reset method. Drivers for the devices being reset are suspended around the reset. If the **-d** option is specified, drivers are detached instead.

Currently, resets are implemented for PCIe buses and PCI devices. For PCIe bus, the link is disabled and then re-trained, causing all children of the bus to reset. Use **-p** option of devinfo(8) tool to report parent bus for the device. For PCI device, if Function-Level Reset is implemented by it, FLR is tried first; if failed or not implemented, power reset is tried.

If you have detached or suspended a child device explicitly and then do a reset, the child device will end up attached.

getpath *locator device*

Prints the full path to the *device* using the *locator* method to get the path name. Currently, only the "UEFI" and "FreeBSD" locators are implemented. The UEFI locator constructs a path to the device using the rules outlines for DEVICE_PATH in the UEFI standard. The FreeBSD locator constructs a path back to the root of the tree with the nodes separated by slashes.

SEE ALSO

devctl(3), devinfo(8)

HISTORY

The **devctl** utility first appeared in FreeBSD 10.3.

BUGS

Currently there is no administrative flag to prevent re-attach or resume of the manually detached or suspended devices after reset. Similarly, there is no flag to prevent un-suspending of the manually suspended devices after system resume.