

Persistent attach of device with varying backend name

[dom0, configuration](#)

[org](#) 1 May 28, 2024, 3:13pm

I am trying to persistently attach a device which has a varying backend name.

[This is the guide](#) that I followed to get the backend name

It returns something like “dm-85”, but it changes with each boot, so I cant do `qvm-block attach -p vmname dom0:dm-85` since it won't exist on next boot.

I can script it to attach it on boot, but then if I restart a VM it won't be attached anymore.

What could be some solutions?

[Extra Volumes](#)

[Mounting other disks \(HDD, SSD, etc.\)](#)

[i3confuze](#) 2 May 28, 2024, 6:07pm

Could you use `cron` or `systemd` to periodically check if the VM is running, and if it is, to then attach the device?

[apparatus](#) 3 May 28, 2024, 6:32pm

Use libxl hook as I've suggested in another topic:

```
sudo mkdir /etc/libvirt/hooks/
cat << 'EOF' | sudo tee /etc/libvirt/hooks/libxl >/dev/null
#!/bin/bash
guest_name="$1"
libvirt_operation="$2"

if [ "$guest_name" = "yourqubename" ] && [ "$libvirt_operation" = "started" ]
(
    exec 0</dev/null
    exec 1>/dev/null
    exec 2>/dev/null
    dev=$(basename $(readlink /dev/YOUR_LVM_VG/YOUR_LVM_IMAGE))
    if ! qvm-block attach "$guest_name" dom0:$dev; then
        qubesdb-write /qubes-block-devices/$dev/desc "YOUR_LVM_IMAGE"
        qvm-block attach "$guest_name" dom0:$dev
    fi
)
```

```
) & disown  
fi  
EOF  
sudo chmod +x /etc/libvirt/hooks/libxl
```

3 Likes

[org](#) 4 May 29, 2024, 9:48am

Thank you, works like a charm

Some tips for future readers, to debug you can restart libvirtd manually instead of rebooting completely, `systemctl restart libvirtd`

`exit 0` as quickly as possible if it's not your target operation/VM

the subshell and redirection of output seems to prevent locking/hanging, if it's not used the VM wouldn't boot properly, probably for the same reason a terminal might hang until you hit cancel. And if that happens you can barely kill the VM because it's in a locked state

look for alternative faster `qvm-` tools like `qubesdb-read` rather than `qubes-block ls`, any time wasted in the hook is time lost on boot

Some useful links here

[libvirt: Hooks for specific system management](#)

libvirt, virtualization, virtualization API

[libvirt: Hooks for specific system management](#)

libvirt, virtualization, virtualization API

<https://wiki.archlinux.org/title/Libvirt#Hooks>

My script is highly personalized with personal device names and VM names, so not useful for the public, but I hope this helps someone else solve the same issue 😊

1 Like

[Johnboy](#) 5 September 6, 2024, 5:07am

Thank you very much, this was exactly what I was looking for.
Maybe this can be merged into a community guideline.