

# Qubes OS Cheatsheet

## VM Management

**qvm-block** - list/set VM PCI devices

usage:

- `qvm-block -l [options]`
- `qvm-block -a [options] <device> <vm-name>`
- `qvm-block -d [options] <device>`
- `qvm-block -d [options] <vm-name>`

---

`qvm-block -A personal dom0:/home/user/extradisks/data.img` - *attaches an additional storage for the personal-vm*

**qvm-ls** - list VMs and various information about their state

usage: `qvm-ls [options] <vm-name>`

---

`qvm-ls` - *lists all vms*

`qvm-ls -n` - *show network addresses assigned to VMs*

`qvm-ls -d` - *show VM disk utilization statistics*

**qvm-prefs** - list/set various per-VM properties

usage:

- `qvm-prefs -l [options] <vm-name>`
- `qvm-prefs -s [options] <vm-name> <property> [...]`

---

`qvm-prefs win7-copy` - *lists the preferences of the win7-copy*

`qvm-prefs win7-copy -s mac 00:16:3E:5E:6C:05` - *sets a new mac for the network card*

`qvm-prefs lab-win7 -s qrexec_installed true` - *sets the qrexec to installed*

`qvm-prefs lab-win7 -s qrexec_timeout 120` - *usefull for windows hvm based vms*

`qvm-prefs lab-win7 -s default_user joanna` - *sets the login user*

**qvm-run** - runs a specific command on a vm

usage: `qvm-run [options] [<vm-name>] [<cmd>]`

---

`qvm-run personal xterm` - *runs xterm on personal*

`qvm-run personal xterm --pass-io` - *runs xterm and passes all sdtin/stdout/stderr to the terminal*

`qvm-run personal "sudo yum update" --pass-io --nogui` - *pass a specific command directly to the VM*

**qvm-start** - *starts a vm*

usage: qvm-start [options] <vm-name>

---

**qvm-start personal** - *starts the personal-vm*

**qvm-start ubuntu --cdrom personal:/home/user/Downloads/ubuntu-14.04.iso** - *starts the ubuntu-vm with the ubuntu installation CD*

**qvm-sync-appmenus** - *updates desktop file templates for given StandaloneVM or TemplateVM*

usage: qvm-sync-appmenus [options] <vm-name>

---

**qvm-sync-appmenus archlinux-template** - *useful for custom .desktop files or distributions not using yum*

## Dom0

**qubes-dom0-update** - *updates software in dom0*

usage: qubes-dom0-update [--clean][--check-only][--gui] [<yum opts>][<pkg list>]

---

**sudo qubes-dom0-update** - *updates dom0*

**sudo qubes-dom0-update qubes-windows-tools** - *install the windows tools*

**qubes-hcl-report** - *generates a report about the hardware information*

usage: qubes-hcl-report [<vm-name>]

---

**qubes-hcl-report** - *prints the hardware information on the console (terminal)*

**qubes-hcl-report personal** - *sends the hardware information to the personal-vm under /home/user*

**xl** - *Xen management tool, based on LibXenlight*

usage: xl subcommand [args]

---

**xl dmesg** - *Dom0 dmesg output (first place to look for warning or error messages)*

**xl top** - *Monitor host and domains in realtime*

## DomU

**qvm-copy-to-vm** - *Copy file from one VM to another VM*

usage: qvm-copy-to-vm <vm-name> <file> [<file+>] - *file can be a single file or a folder*

---

**qvm-copy-to-vm work Documents** - *copy the Documents folder to the work VM*

**qvm-copy-to-vm personal text.txt** - *copy the text.txt file to the personal VM*

## Example

- Open a terminal in AppVM A (e. g. your personal vm)
- Let's assume we want to copy the Documents folder to AppVM B (e. g. your work VM)
- The command would be: **qvm-copy-to-vm work Documents**

## DomU and Dom0

**List installed qubes packages** —

## Fedora

In VM or Dom0: **rpm -qa \\*qubes-\\*** - *list (qubes-) installed packages*

## Copy from & to Dom0

Copy from: **Dom0** -> **VM**

```
cat /path/to/file_in_dom0 |  
qvm-run --pass-io <dst_domain>  
'cat > /path/to/file_name_in_appvm'
```

Copy from: **VM** -> **Dom0**

```
qvm-run --pass-io <src_domain>  
'cat /path/to/file_in_src_domain' >  
/path/to/file_name_in_dom0
```

## Copy text between VM A and B

*On VM A (source):*

1. CTRL+C
2. CTRL+SHIFT+C

*On VM B (destination):*

3. CTRL+SHIFT+V
4. CTRL+V

## Grow disk

**qvm-grow-private** - *increase private storage capacity of a specified VM*

usage: **qvm-grow-private** <vm-name> <size>

### Example

- In dom0 konsole: **qvm-grow-private personal 40GB**
- In the personal VM: **sudo resize2fs /dev/xvdb**

## VM -> VM Networking

Make sure:

- Both VMs are connected to the same firewall VM
- Qubes IP addresses are assigned to both VMs
- Both VMs are started

Firewall VM's terminal:

```
sudo iptables -I FORWARD 2 -s <IP address of A> -d <IP address of B> -j ACCEPT
```

## Templates

**Fedora Minimal** - *Fedora minimal template*

**sudo qubes-dom0-update qubes-template-fedora-21-minimal** - *installs the fedora-21-minimal template*

**Archlinux Minimal** - *Archlinux minimal template*

1. In a VM:

```
wget http://olivier.medoc.free.fr/rpm/noarch/  
qubes-template-archlinux-minimal-3.0.3-201507281153.noarch.rpm
```

2. Copy RPM-Package to Dom0

3. In Dom0: **sudo rpm -i qubes-template-archlinux-minimal-3.0.3-201507281153.noarch.rpm**

## Create VM from VMware or VirtualBox images

1. Download the image in an AppVM
2. Install `qemu-img` tools - *e. g. `yum install qemu-img` for fedora*
3. Convert the image to a raw format:
  - VMware: `qemu-img convert ReactOS.vmdk -O raw reactos.img`
  - VirtualBox: `qemu-img convert ReactOS.vdi -O raw reactos.img`