Dom0 Tools

qvm-remove

NAME

qvm-remove - remove a VM

SYNOPSIS

qvm-remove [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

-q, –quiet
Be quiet

–just-db
Remove only from the Qubes Xen DB, do not remove any files

–force-root
Force to run, even with root privileges

qvm-firewall

NAME

qvm-firewall

SYNOPSIS

qvm-firewall [-n] <vm-name> [action] [rule spec]

Rule specification can be one of:
1. address|hostname[/netmask] tcp|udp port[-port] 2. address|hostname[/netmask] tcp|udp service_name 3. address|hostname[/netmask] any

OPTIONS

-h, –help
Show this help message and exit

-l, –list
List firewall settings (default action)

-a, –add
Add rule

-d, –del
Remove rule (given by number or by rule spec)

-P SET_POLICY, –policy=SET_POLICY
Set firewall policy (allow/deny)

-i SET_ICMP, –icmp=SET_ICMP
Set ICMP access (allow/deny)

-D SET_DNS, –dns=SET_DNS
Set DNS access (allow/deny)

-Y SET_YUM_PROXY, –yum-proxy=SET_YUM_PROXY
Set access to Qubes yum proxy (allow/deny). *Note:* if set to "deny", access will be rejected even if policy set to "allow"

-n, –numeric
Display port numbers instead of services (makes sense only with –list)

qvm-create-default-dvm

NAME

qvm-create-default-dvm - creates a default disposable VM

SYNOPSIS

qvm-create-default-dvm templatename|–default-template|–used-template [scriptname|–default-script]

OPTIONS

templatename
Base DispVM on given template. The command will create AppVM named after template with "-dvm" suffix. This VM will be used to create DispVM

savefile. If you want to customize DispVM, use this VM - take a look at https://wiki.qubes-os.org/wiki/UserDoc/DispVMCustomization

–default-template
Use default template for the DispVM

–used-template
Use the same template as earlier

–default-script
Use default script for seeding DispVM home.

qvm-run

NAME

qvm-run - run a command on a specified VM

SYNOPSIS

qvm-run [options] [<vm-name>] [<cmd>]

OPTIONS

-h, –help
Show this help message and exit

-q, –quiet
Be quiet

-a, –auto
Auto start the VM if not running

-u USER, –user=USER
Run command in a VM as a specified user

–tray
Use tray notifications instead of stdout

–all
Run command on all currently running VMs (or all paused, in case of –unpause)

–exclude=EXCLUDE_LIST
When –all is used: exclude this VM name (might be repeated)

–wait
Wait for the VM(s) to shutdown

–shutdown
(deprecated) Do ‘xl shutdown’ for the VM(s) (can be combined this with –all and –wait)

–pause
Do ‘xl pause’ for the VM(s) (can be combined this with –all and –wait)

–unpause
Do ‘xl unpause’ for the VM(s) (can be combined this with –all and –wait)

-p, –pass-io
Pass stdin/stdout/stderr from remote program

–localcmd=LOCALCMD
With –pass-io, pass stdin/stdout/stderr to the given program

–force
Force operation, even if may damage other VMs (eg. shutdown of NetVM)

qvm-ls

NAME

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

SYNOPSIS

qvm-ls [options] <vm-name>

OPTIONS

-h, –help
Show help message and exit

-n, –network
Show network addresses assigned to VMs

-c, –cpu
Show CPU load

-m, –mem
Show memory usage

-d, –disk
Show VM disk utilization statistics

-i, –ids
Show Qubes and Xen id

-k, –kernel
Show VM kernel options

-b, –last-backup
Show date of last VM backup

–raw-list
List only VM names one per line

qvm-clone

NAME

qvm-clone - clones an existing VM by copying all its disk files

SYNOPSIS

qvm-clone [options] <src-name> <new-name>

OPTIONS

-h, –help
Show this help message and exit

-q, –quiet
Be quiet

-p DIR_PATH, –path=DIR_PATH
Specify path to the template directory

qvm-prefs

NAME

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

SYNOPSIS

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

OPTIONS

-h, –help
Show this help message and exit

-l, –list
List properties of a specified VM

-g, –get
Get a single property of a specified VM

-s, –set
Set properties of a specified VM

PROPERTIES

include_in_backups
Accepted values: `True`, `False`

Control whenever this VM will be included in backups by default (for now works only in qubes-manager). You can always manually select or deselect any VM for backup.

pcidevs
PCI devices assigned to the VM. Should be edited using qvm-pci tool.

pci_strictreset
Accepted values: `True`, `False`

Control whether prevent assigning to VM a device which does not support any reset method. Generally such devices should not be assigned to any VM, because there will be no way to reset device state after VM shutdown, so the device could attack next VM to which it will be assigned. But in some cases it could make sense - for example when the VM to which it is assigned is trusted one, or is running all the time.

label
Accepted values: `red`, `orange`, `yellow`, `green`, `gray`, `blue`, `purple`, `black`

Color of VM label (icon, appmenus, windows border). If VM is running, change will be applied at first VM restart.

netvm
Accepted values: netvm name, `default`, `none`

To which NetVM connect. Setting to `default` will follow system-global default NetVM (managed by qubes-prefs). Setting to `none` will disable networking in this VM.

dispvm_netvm
Accepted values: netvm name, `default`, `none`

Which NetVM should be used for Disposable VMs started by this one. `default` is to use the same NetVM as the VM itself.

maxmem
Accepted values: memory size in MB

Maximum memory size available for this VM. Dynamic memory management (aka qmemman) will not be able to balloon over this limit. For VMs with qmemman disabled, this will be overridden by *memory* property (at VM startup).

memory
Accepted values: memory size in MB

Initial memory size for VM. This should be large enough to allow VM startup - before qmemman starts managing memory for this VM. For VM with qmemman disabled, this is static memory size.

kernel
Accepted values: kernel version, `default`, `none`

Kernel version to use (only for PV VMs). Available kernel versions will be listed when no value given (there are in /var/lib/qubes/vm-kernels). Setting to `default` will follow system-global default kernel (managed via qubes-prefs). Setting to `none` will use "kernels" subdir in VM directory - this allows having VM-specific kernel; also this the only case when /lib/modules is writable from within VM.

template
Accepted values: TemplateVM name

TemplateVM on which VM base. It can be changed only when VM isn't running.

vcpus
Accepted values: no of CPUs

Number of CPU (cores) available to VM. Some VM types (eg DispVM) will not work properly with more than one CPU.

kernelopts
Accepted values: string, `default`

VM kernel parameters (available only for PV VMs). This can be used to workaround some hardware specific problems (eg for NetVM). Setting to `default` will use some reasonable defaults (currently different for VMs with PCI devices and without). For VM without PCI devices `default` option means inherit this value from the VM template (if any). Some helpful options (for debugging purposes): `earlyprintk=xen`, `init=/bin/bash`

name
Accepted values: alphanumerical name

Name of the VM. Can be only changed when VM isn't running.

drive
Accepted values: [hd:|cdrom:][backend-vm:]path

Additional drive for the VM (available only for HVMs). This can be used to attach installation image. `path` can be file or physical device (eg. /dev/sr0). The same syntax can be used in qvm-start –drive - to attach drive only temporarily.

mac
Accepted values: MAC address, `auto`

Can be used to force specific of virtual ethernet card in the VM. Setting to `auto` will use automatic-generated MAC - based on VM id. Especially useful when some licencing depending on static MAC address. For template-based HVM `auto` mode means to clone template MAC.

default_user
Accepted values: username

Default user used by qvm-run. Note that it make sense only on non-standard template, as the standard one always have "user" account.

debug
Accepted values: `on`, `off`

Enables debug mode for VM. This can be used to turn on/off verbose logging in many qubes components at once (gui virtualization, VM kernel, some other services). For template-based HVM, enabling debug mode also disables automatic reset root.img (actually volatile.img) before each VM startup, so changes made to root filesystem stays intact. To force reset root.img when debug mode enabled, either change something in the template (simple start+stop will do, even touch its root.img is enough), or remove VM's volatile.img (check the path with qvm-prefs).

qrexec_installed
Accepted values: `True`, `False`

This HVM have qrexec agent installed. When VM have qrexec agent installed, one can use qvm-run to start VM process, VM will benefit from Qubes RPC services (like file copy, or inter-vm clipboard). This option will be automatically turned on during Qubes Windows Tools installation, but if you install qrexec agent in some other OS, you need to turn this option on manually.

guiagent_installed
Accepted values: `True`, `False`

This HVM have gui agent installed. This option disables full screen GUI virtualization and enables per-window seemless GUI mode. This option will be automatically turned on during Qubes Windows Tools installation, but if you install qubes gui agent in some other OS, you need to turn this option on manually. You can turn this option off to troubleshoot some early HVM OS boot problems (enter safe mode etc), but the option will be automatically enabled at first VM normal startup (and will take effect from the next startup).

Notice: when Windows GUI agent is installed in the VM, SVGA device (used to full screen video) is disabled, so even if you disable this option, you will not get functional full desktop access (on normal VM startup). Use some other means for that (VNC, RDP or so).

autostart
Accepted values: `True`, `False`

Start the VM during system startup. The default netvm is autostarted regardless of this setting.

timezone
Accepted values: `localtime`, time offset in seconds

Set emulated HVM clock timezone. Use `localtime` (the default) to use the same time as dom0 have. Note that HVM will get only clock value, not the timezone itself, so if you use `localtime` setting, OS inside of HVM should also be configured to treat hardware clock as local time (and have proper timezone set).

qvm-shutdown

NAME

qvm-shutdown

SYNOPSIS

qvm-shutdown [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

-q, –quiet
Be quiet

–force
Force operation, even if may damage other VMs (eg. shutdown of NetVM)

–wait
Wait for the VM(s) to shutdown

–all
Shutdown all running VMs

–exclude=EXCLUDE_LIST
When –all is used: exclude this VM name (might be repeated)

qvm-template-commit

NAME

qvm-template-commit

SYNOPSIS

qvm-template-commit [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

qvm-block

NAME

qvm-block - list/set VM PCI devices.

SYNOPSIS

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

OPTIONS

-h, –help
Show this help message and exit

-l, –list
List block devices

-a, –attach
Attach block device to specified VM

-d, –detach
Detach block device

-f FRONTEND, –frontend=FRONTEND
Specify device name at destination VM [default: xvdi]

–ro
Force read-only mode

–no-auto-detach
Fail when device already connected to other VM

qvm-backup

NAME

qvm-backup

SYNOPSIS

qvm-backup [options] <backup-dir-path>

OPTIONS

-h, –help
Show this help message and exit

-x EXCLUDE_LIST, –exclude=EXCLUDE_LIST
Exclude the specified VM from backup (might be repeated)

qvm-pci

NAME

qvm-pci - list/set VM PCI devices

SYNOPSIS

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

OPTIONS

-h, –help
Show this help message and exit

-l, –list
List VM PCI devices

-a, –add
Add a PCI device to specified VM

-d, –delete
Remove a PCI device from specified VM

qubes-prefs

NAME

qubes-prefs - display system-wide Qubes settings, such as:

- clock VM
- update VM
- default template
- default firewallVM
- default kernel
- default netVM

SYNOPSIS

qubes-prefs

qvm-sync-appmenus

NAME

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

SYNOPSIS

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

OPTIONS

-h, –help
Show this help message and exit

-v, –verbose
Run in verbose mode

qvm-start

NAME

qvm-start - start a specified VM

SYNOPSIS

qvm-start [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

-q, –quiet
Be quiet

–no-guid
Do not start the GUId (ignored)

–console
Attach debugging console to the newly started VM

–dvm
Do actions necessary when preparing DVM image

–custom-config=CUSTOM_CONFIG
Use custom Xen config instead of Qubes-generated one

qubes-dom0-update

NAME

qubes-dom0-update - update software in dom0

SYNOPSIS

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

OPTIONS

–clean
Clean yum cache before doing anything

–check-only
Only check for updates (no install)

–gui
Use gpk-update-viewer for update selection

<pkg list>
Download (and install if run by root) new packages in dom0 instead of updating

Besides above options, when no –gui or –check-only given, all other options are passed to yum call. So for example –enablerepo/–disablerepo options works as well.

qvm-revert-template-changes

NAME

qvm-revert-template-changes

SYNOPSIS

qvm-revert-template-changes [options] <template-name>

OPTIONS

-h, –help
Show this help message and exit

–force
Do not prompt for comfirmation

qvm-service

NAME

qvm-service - manage (Qubes-specific) services started in VM

SYNOPSIS

qvm-service [-l] <vmname> qvm-service [-e|-d|-D] <vmname> <service>

OPTIONS

-h, –help
Show this help message and exit

-l, –list
List services (default action)

-e, –enable
Enable service

-d, –disable
Disable service

-D, –default
Reset service to its default state (remove from the list). Default state means "lets VM choose" and can depend on VM type (NetVM, AppVM etc).

SUPPORTED SERVICES

This list can be incomplete as VM can implement any additional service without knowlege of qubes-core code.

meminfo-writer
Default: enabled everywhere excluding NetVM

This service reports VM memory usage to dom0, which effectively enables dynamic memory management for the VM.

Note: this service is enforced to be set by dom0 code. If you try to remove it (reset to defult state), will be recreated with the rule: enabled if VM have no PCI devices assigned, otherwise disabled.

qubes-dvm
Default: disabled

Used internally when creating DispVM savefile.

qubes-firewall
Default: enabled only in ProxyVM

Dynamic firewall manager, based on settings in dom0 (qvm-firewall, firewall tab in qubes-manager). This service is not supported in netvms.

qubes-network
Default: enabled only in NetVM and ProxyVM

Expose network for other VMs. This includes enabling network forwarding, MASQUERADE, DNS redirection and basic firewall.

qubes-netwatcher
Default: enabled only in ProxyVM

Monitor IP change notification from NetVM. When received, reload qubes-firewall service (to force DNS resolution). This service makes sense only with qubes-firewall enabled.

qubes-update-check
Default: enabled

Notify dom0 about updates available for this VM. This is shown in qubes-manager as 'update-pending' flag.

cups
Default: enabled only in AppVM

Enable CUPS service. The user can disable cups in VM which do not need printing to speed up booting.

cron
Default: disabled

Enable CRON service.

network-manager
Default: enabled in NetVM

Enable NetworkManager. Only VM with direct access to network device needs this service, but can be useful in ProxyVM to ease VPN setup.

ntpd
Default: disabled

Enable NTPD service. By default Qubes calls ntpdate every 6 minutes in selected VM (aka ClockVM), then propagate the result using qrexec calls. Enabling ntpd *do not* disable this behaviour.

qubes-yum-proxy
Deprecated name for qubes-updates-proxy.

qubes-updates-proxy
Default: enabled in NetVM

Provide proxy service, which allow access only to yum repos. Filtering is done based on URLs, so it shouldn't be used as leak control (pretty easy to bypass), but is enough to prevent some erroneous user actions.

yum-proxy-setup
Deprecated name for updates-proxy-setup.

updates-proxy-setup
Default: enabled in AppVM (also in templates)

Setup yum at startup to use qubes-yum-proxy service.

Note: this service is automatically enabled when you allow VM to access yum proxy (in firewall settings) and disabled when you deny access to yum proxy.

disable-default-route
Default: disabled

Disables the default route for networking. Enabling this service will prevent the creation of the default route, but the VM will still be able to reach it's direct neighbors. The functionality is implemented in /usr/lib/qubes/setup-ip.

disable-dns-server
Default: disabled

Enabling this service will result in an empty /etc/resolv.conf. The functionality is implemented in /usr/lib/qubes/setup-ip.

qvm-backup-restore

NAME

qvm-backup-restore - restores Qubes VMs from backup

SYNOPSIS

qvm-backup-restore [options] <backup-dir>

OPTIONS

-h, –help
Show this help message and exit

–skip-broken
Do not restore VMs that have missing templates or netvms

–ignore-missing
Ignore missing templates or netvms, restore VMs anyway

–skip-conflicting
Do not restore VMs that are already present on the host

–force-root
Force to run, even with root privileges

–replace-template=REPLACE_TEMPLATE
Restore VMs using another template, syntax: old-template-name:new-template-name (might be repeated)

-x EXCLUDE, –exclude=EXCLUDE
Skip restore of specified VM (might be repeated)

–skip-dom0-home
Do not restore dom0 user home dir

–ignore-username-mismatch
Ignore dom0 username mismatch while restoring homedir

qvm-add-appvm

NAME

qvm-add-appvm - add an already installed appvm to the Qubes DB

WARNING: Noramlly you would not need this command, and you would use qvm-create instead!

SYNOPSIS

qvm-add-appvm [options] <appvm-name> <vm-template-name>

OPTIONS

-h, –help
Show this help message and exit

-p DIR_PATH, –path=DIR_PATH
Specify path to the template directory

-c CONF_FILE, –conf=CONF_FILE
Specify the Xen VM .conf file to use(relative to the template dir path)

qvm-add-template

NAME

qvm-add-template - adds an already installed template to the Qubes DB

SYNOPSIS

qvm-add-template [options] <vm-template-name>

OPTIONS

-h, –help
Show this help message and exit

-p DIR_PATH, –path=DIR_PATH
Specify path to the template directory

-c CONF_FILE, –conf=CONF_FILE
Specify the Xen VM .conf file to use(relative to the template dir path)

–rpm
Template files have been installed by RPM

qvm-kill

NAME

qvm-kill - kills the specified VM

SYNOPSIS

qvm-kill [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

qvm-create

NAME

qvm-create - creates a new VM

SYNOPSIS

qvm-create [options] <vm-name>

OPTIONS

-h, –help
Show this help message and exit

-t TEMPLATE, –template=TEMPLATE
Specify the TemplateVM to use

-l LABEL, –label=LABEL
Specify the label to use for the new VM (e.g. red, yellow, green, . . .)

-p, –proxy
Create ProxyVM

-n, –net
Create NetVM

-H, –hvm
Create HVM (standalone, unless –template option used)

–hvm-template
Create HVM template

-R ROOT_MOVE, –root-move-from=ROOT_MOVE
Use provided root.img instead of default/empty one (file will be MOVED)

-r ROOT_COPY, –root-copy-from=ROOT_COPY
Use provided root.img instead of default/empty one (file will be COPIED)

-s, –standalone
Create standalone VM - independent of template

-m MEM, –mem=MEM
Initial memory size (in MB)

-c VCPUS, –vcpus=VCPUS
VCPUs count

-i, –internal
Create VM for internal use only (hidden in qubes-manager, no appmenus)

–force-root
Force to run, even with root privileges

-q, –quiet
Be quiet

qvm-grow-private

NAME

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

SYNOPSIS

qvm-grow-private <vm-name> <size>

OPTIONS

-h, –help
Show this help message and exit

VM Tools

qvm-run

NAME

qvm-run - run a specified command in a specified VM

SYNOPSIS

qvm-run vmname command [aguments]

OPTIONS

–dispvm
Pass this option instead of vmname to start new DisposableVM

qvm-open-in-vm

NAME

qvm-open-in-vm - open a specified file in other VM

SYNOPSIS

qvm-open-in-vm vmname filename

OPTIONS

qvm-open-in-dvm

NAME

qvm-open-in-dvm - open a specified file in disposable VM

SYNOPSIS

qvm-open-in-dvm filename

OPTIONS

qvm-copy-to-vm

NAME

qvm-copy-to-vm - copy specified files to specified destination VM

SYNOPSIS

qvm-copy-to-vm [–without-progress] dest_vmname file [file]+

OPTIONS

–without-progress
Don’t display progress info