

Proxmox Cheatsheet

This page presents a list of commonly used commands for proxmox.

Categories: [cheatsheet](#) [proxmox](#)

Access this page by simply typing in "**cs proxmox**" in your browser address bar if you have [bunnylol](#) configured.

Commands

VM Management

Basics

```
# List VMs
qm list

# Create or restore a virtual machine.
qm create <vmid>

# Create or restore a virtual machine with core, memory, disks specified.
qm create <vmid> --name <vm-name> --cores <number-of-cores> --memory <memory-size-in-bytes> --scsi0 file=<vg-name>:<size>

# Start a VM
qm start <vmid>

# Suspend virtual machine.
qm suspend <vmid>

# Shutdown a VM
qm shutdown <vmid>

# Reboot a VM
qm reboot <vmid>

# Reset a VM
qm reset <vmid>

# Stop a VM
qm stop <vmid>

# Destroy the VM and all used/owned volumes.
# Removes any VM specific permissions and firewall rules
qm destroy <vmid>

# Enter Qemu Monitor interface.
qm monitor <vmid>

# Get the virtual machine configuration with both current and pending values.
qm pending <vmid>

# Send key event to virtual machine.
qm sendkey <vmid> <key> [OPTIONS]

# Show command line which is used to start the VM (debug info).
qm showcmd <vmid> [OPTIONS]

# Unlock the VM.
qm unlock <vmid>

# Clone a VM
qm clone <vmid> <newid>

# Clone a VM in full clone mode and also set the name.
qm clone <vmid> <newid> --full --name <name>

# Migrate a VM
qm migrate <vmid> <target-node>

# Show VM status
qm status <vmid>

# Clean up resources for a VM
qm cleanup <vmid> <clean-shutdown> <guest-requested>

# Create a Template.
qm template <vmid> [OPTIONS]

# Set virtual machine options (synchronous API)
qm set <vmid> [OPTIONS]
```

```
# Get automatically generated cloudinit config.
qm cloudinit dump <vmid> <type>

# Get the cloudinit configuration with both current and pending values.
qm cloudinit pending <vmid>

# Regenerate and change cloudinit config drive.
qm cloudinit update <vmid>
```

Copy

Disk

```
# Import an external disk image as an unused disk in a VM.
# The image format has to be supported by qemu-img(1).
qm disk import <vmid> <source> <storage>

# Move volume to different storage or to a different VM.
qm disk move <vmid> <disk> [<storage>] [OPTIONS]

# Rescan all storages and update disk sizes and unused disk images.
qm disk rescan [OPTIONS]

# Extend volume size.
qm disk resize <vmid> <disk> <size> [OPTIONS]

# Unlink/delete disk images.
qm disk unlink <vmid> --idlist <string> [OPTIONS]

# rescan volumes
qm rescan
```

Copy

Snapshot

```
# List all snapshots.
qm listsnapshot <vmid>

# Snapshot a VM
qm snapshot <vmid> <snapname>

# Delete a snapshot.
qm delsnapshot <vmid> <snapname>

# Rollback a snapshot
qm rollback <vmid> <snapname>

# Open a terminal using a serial device
# (The VM need to have a serial device configured, for example serial0: socket)
qm terminal <vmid> [OPTIONS]

# Proxy VM VNC traffic to stdin/stdout
qm vncproxy <vmid>
```

Copy

Others

```
# Execute Qemu Guest Agent commands.
qm guest cmd <vmid> <command>

# Executes the given command via the guest agent
qm guest exec <vmid> [<extra-args>] [OPTIONS]

# Gets the status of the given pid started by the guest-agent
qm guest exec-status <vmid> <pid>

# Sets the password for the given user to the given password
qm guest passwd <vmid> <username> [OPTIONS]
```

Copy

PV, VG & LV Management

```
# Create a PV
pvcreate <disk-device-name>

# Remove a PV
pvremove <disk-device-name>

# List all PVs
pvs

# Create a VG
vgcreate <vg-name> <disk-device-name>

# Remove a VG
vgremove <vg-name>

# List all VGs
vgs

# Create a LV
lvcreate -L <lv-size> -n <lv-name> <vg-name>

# Remove a LV
lvremove <vg-name>/<lv-name>

# List all LVs
lvs
```

Copy

Storage Management

```
# Create a new storage.
pvesm add <type> <storage> [OPTIONS]

# Allocate disk images.
pvesm alloc <storage> <vmid> <filename> <size> [OPTIONS]

# Delete volume
pvesm free <volume> [OPTIONS]

# Delete storage configuration.
pvesm remove <storage>

# List storage content.
pvesm list <storage> [OPTIONS]

# An alias for pvesm scan lvm.
pvesm lvmscan

# An alias for pvesm scan lvmthin.
pvesm lvmthinscan

# List local LVM volume groups.
pvesm scan lvm

# List local LVM Thin Pools.
pvesm scan lvmthin <vg>

# Get status for all datastores.
pvesm status [OPTIONS]
```

[Copy](#)

Template Management

```
# list all templates
pveam available

# list all templates
pveam list <storage>

# Download appliance templates
pveam download <storage> <template>

# Remove a template.
pveam remove <template-path>

# Update Container Template Database.
pveam update
```

[Copy](#)

Container Management

Basics

```
# List containers
pct list

# Create or restore a container.
pct create <vmid> <ostemplate> [OPTIONS]

# Start the container.
pct start <vmid> [OPTIONS]

# Create a container clone/copy
pct clone <vmid> <newid> [OPTIONS]

# Suspend the container. This is experimental.
pct suspend <vmid>

# Resume the container.
pct resume <vmid>

# Stop the container.
# This will abruptly stop all processes running in the container.
pct stop <vmid> [OPTIONS]

# Shutdown the container.
# This will trigger a clean shutdown of the container, see lxc-stop(1) for details.
pct shutdown <vmid> [OPTIONS]

# Destroy the container (also delete all user files).
pct destroy <vmid> [OPTIONS]

# Show CT status.
pct status <vmid> [OPTIONS]

# Migrate the container to another node. Creates a new migration task.
pct migrate <vmid> <target> [OPTIONS]

# Get container configuration.
pct config <vmid> [OPTIONS]

# Print the list of assigned CPU sets.
pct cpusets

# Get container configuration, including pending changes.
pct pending <vmid>

# Reboot the container by shutting it down, and starting it again. Applies pending changes.
pct reboot <vmid> [OPTIONS]

# Create or restore a container.
pct restore <vmid> <ostemplate> [OPTIONS]

# Set container options.
pct set <vmid> [OPTIONS]

# Create a Template.
pct template <vmid>

# Unlock the VM.
pct unlock <vmid>
```

Disk

```
# Get the container's current disk usage.
pct df <vmid>

# Run a filesystem check (fsck) on a container volume.
pct fsck <vmid> [OPTIONS]

# Run fstrim on a chosen CT and its mountpoints.
pct fstrim <vmid> [OPTIONS]

# Mount the container's filesystem on the host.
# This will hold a lock on the container and is meant for emergency maintenance only
# as it will prevent further operations on the container other than start and stop.
pct mount <vmid>

# Move a rootfs-/mp-volume to a different storage or to a different container.
pct move-volume <vmid> <volume> [<storage>] [<target-vmid>] [<target-volume>] [OPTIONS]

# Unmount the container's filesystem.
pct unmount <vmid>

# Resize a container mount point.
pct resize <vmid> <disk> <size> [OPTIONS]

# Rescan all storages and update disk sizes and unused disk images.
pct rescan [OPTIONS]

# Launch a console for the specified container.
pct console <vmid> [OPTIONS]

# Launch a shell for the specified container.
pct enter <vmid>

# Launch a command inside the specified container.
pct exec <vmid> [<extra-args>]

# Copy a file from the container to the local system.
pct pull <vmid> <path> <destination> [OPTIONS]

# Copy a local file to the container.
pct push <vmid> <file> <destination> [OPTIONS]
```

Copy

Snapshot

```
# Snapshot a container.
pct snapshot <vmid> <snapname> [OPTIONS]

# List all snapshots.
pct listsnapshot <vmid>

# Rollback LXC state to specified snapshot.
pct rollback <vmid> <snapname> [OPTIONS]

# Delete a LXC snapshot.
pct delsnapshot <vmid> <snapname> [OPTIONS]
```

Copy

Web GUI

```
# Restart web GUI
service pveproxy restart
```

Copy

Important File/Dir Path

PVE

<code>/etc/pve/authkey.pub</code>	Public key used by the ticket system
<code>/etc/pve/ceph.conf</code>	Ceph configuration file (note: <code>/etc/ceph/ceph.conf</code> is a symbolic link to this)
<code>/etc/pve/corosync.conf</code>	Corosync cluster configuration file (prior to Proxmox VE 4.x, this file was called <code>cluster.conf</code>)
<code>/etc/pve/datacenter.cfg</code>	Proxmox VE data center-wide configuration (keyboard layout, proxy, ...)
<code>/etc/pve/domains.cfg</code>	Proxmox VE authentication domains
<code>/etc/pve/firewall/cluster.fw</code>	Firewall configuration applied to all nodes
<code>/etc/pve/firewall/<NAME>.fw</code>	Firewall configuration for individual nodes
<code>/etc/pve/firewall/<VMID>.fw</code>	Firewall configuration for VMs and containers
<code>/etc/pve/ha/crm_commands</code>	Displays HA operations that are currently being carried out by the CRM
<code>/etc/pve/ha/manager_status</code>	JSON-formatted information regarding HA services on the cluster
<code>/etc/pve/ha/resources.cfg</code>	Resources managed by high availability, and their current state
<code>/etc/pve/nodes/<NAME>/config</code>	Node-specific configuration
<code>/etc/pve/nodes/<NAME>/lxc/<VMID>.conf</code>	VM configuration data for LXC containers
<code>/etc/pve/nodes/<NAME>/openvz/</code>	Prior to PVE 4.0, used for container configuration data (deprecated, removed soon)
<code>/etc/pve/nodes/<NAME>/pve-ssl.key</code>	Private SSL key for <code>pve-ssl.pem</code>
<code>/etc/pve/nodes/<NAME>/pve-ssl.pem</code>	Public SSL certificate for web server (signed by cluster CA)
<code>/etc/pve/nodes/<NAME>/pveproxy-ssl.key</code>	Private SSL key for <code>pveproxy-ssl.pem</code> (optional)
<code>/etc/pve/nodes/<NAME>/pveproxy-ssl.pem</code>	Public SSL certificate (chain) for web server (optional override for <code>pve-ssl.pem</code>)
<code>/etc/pve/nodes/<NAME>/qemu-server/<VMID>.conf</code>	VM configuration data for KVM VMs
<code>/etc/pve/priv/authkey.key</code>	Private key used by ticket system
<code>/etc/pve/priv/authorized_keys</code>	SSH keys of cluster members for authentication
<code>/etc/pve/priv/ceph*</code>	Ceph authentication keys and associated capabilities
<code>/etc/pve/priv/known_hosts</code>	SSH keys of the cluster members for verification
<code>/etc/pve/priv/lock/*</code>	Lock files used by various services to ensure safe cluster-wide operations
<code>/etc/pve/priv/pve-root-ca.key</code>	Private key of cluster CA
<code>/etc/pve/priv/shadow.cfg</code>	Shadow password file for PVE Realm users
<code>/etc/pve/priv/storage/<STORAGE-ID>.pw</code>	Contains the password of a storage in plain text
<code>/etc/pve/priv/tfa.cfg</code>	Base64-encoded two-factor authentication configuration
<code>/etc/pve/priv/token.cfg</code>	API token secrets of all tokens
<code>/etc/pve/pve-root-ca.pem</code>	Public certificate of cluster CA
<code>/etc/pve/pve-www.key</code>	Private key used for generating CSRF tokens
<code>/etc/pve/sdn/*</code>	Shared configuration files for Software Defined Networking (SDN)
<code>/etc/pve/status.cfg</code>	Proxmox VE external metrics server configuration
<code>/etc/pve/storage.cfg</code>	Proxmox VE storage configuration
<code>/etc/pve/user.cfg</code>	Proxmox VE access control configuration (users/groups/...)
<code>/etc/pve/virtual-guest/cpu-models.conf</code>	For storing custom CPU models
<code>/etc/pve/vzdump.cron</code>	Cluster-wide <code>vzdump</code> backup-job schedule

Debug

<code>/etc/pve/.version</code>	file versions (to detect file modifications)
<code>/etc/pve/.members</code>	Info about cluster members
<code>/etc/pve/.vmlist</code>	List of all VMs
<code>/etc/pve/.clusterlog</code>	Cluster log (last 50 entries)
<code>/etc/pve/.rrd</code>	RRD data (most recent entries)

OpenVZ Section

<code>/etc/vz/conf/xxx.conf</code>	config
<code>/var/lib/vz/root/xxx</code>	data
<code>/var/lib/vz/template/cache</code>	template
<code>/var/lib/vz/dump</code>	snapshot
<code>/etc/vz/vz.conf</code>	OpenVZ config

KVM Section

<code>/var/lib/vz/images/xxx</code>	data
<code>/var/lib/vz/template/iso</code>	template
<code>/var/lib/vz/dump</code>	snapshot

LXC Section

<code>/var/lib/lxc/xxx/config</code>	config
<code>/var/lib/vz/images/xxx</code>	data
<code>/var/lib/vz/template/cache</code>	template
<code>/var/lib/vz/dump</code>	snapshot