Parallels Virtuozzo Containers 4.6 for Windows

Reference Guide

Parallels IP Holdings GmbH.
Vordergasse 59
CH8200 Schaffhausen
Switzerland
Tel: + 41 526320 411
Fax: + 41 52672 2010
www.parallels.com

Contents

Preface 5

About Parallels Virtuozzo Containers 4.6 5
About This Guide 6
Organization of This Guide 7
Documentation Conventions 7
Getting Help 8
Feedback 9

Parallels Virtuozzo Containers Utilities Overview 10

General Utilities 12

vzctl 12
vzctl create 13
vzctl delete, vzctl destroy 14
vzctl mount, vzctl umount 15
vzctl start, vzctl stop, vzctl restart, and vzctl status 15
vzctl set 15
vzctl exec, vzctl exec2, and vzctl enter 21
vzctl mountext, vzctl umountext 21
vzctl partadd, vzctl partdel 22
vzctl reinstall 22
vzctl shrink 23
vzctl defrag 23
vzctl addrole, vzctl delrole, vzctl enumrole 24
vzlist 24
vzlist Output Parameters and Their Modifiers 25
vzquota 28
vzquota setlimit 28
vzquota stat, vzquota show 29

Licensing Utilities 30

vzlicload 30
vzlicview 31

Migration Utilities 32

vzmigrate 32
vzp2v 33
vzmlocal 34

Backup Utilities 36

vzabackup 36
vzarestore 38
vzvssctl 40

Template Management Utilities 41

vzpkgls 41
vzpkgdeploy 42
vzpkgadd 42
vzpkgrm 43

Supplementary Utilities 44

vzkeygen 44
vzcache 45
vzlscache 46
vzuncache 47
vznetcfg 48
vznetstat 49
vzdevctl 50
vzcpucfg 52
vzquery 53
vzwinupdatecmd 53

Glossary 55

Index 56

CHAPTER 1

Preface

In This Chapter

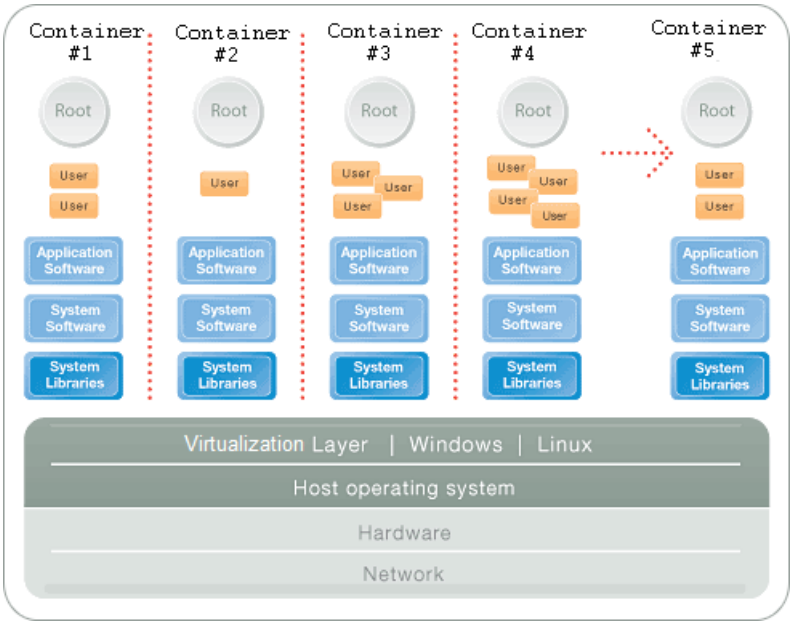
About Parallels Virtuozzo Containers 4.6 5
About This Guide 6
Getting Help 8
Feedback 9

About Parallels Virtuozzo Containers 4.6

Parallels Virtuozzo Containers 4.6 is a patented OS virtualization solution. It creates isolated partitions or Containers on a single physical server and OS instance to utilize hardware, software, data center and management effort with maximum efficiency. The basic Parallels Virtuozzo Containers capabilities are:

- **Intelligent Partitioning**—Division of a server into as many as hundreds of Containers with full server functionality.
- **Complete Isolation**—Containers are secure and have full functional, fault and performance isolation.
- **Dynamic Resource Allocation**—CPU, memory, network, disk and I/O can be changed without rebooting.
- **Mass Management**—Suite of tools and templates for automated, multi-Container and multi-server administration.

The diagram below represents a typical model of the Parallels Virtuozzo Containers system structure:



The Parallels Virtuozzo Containers OS virtualization model is streamlined for the best performance, management, and efficiency. At the base resides a standard Host operating system which can be either Windows or Linux. Next is the virtualization layer with a proprietary file system and a kernel service abstraction layer that ensure the isolation and security of resources between different Containers. The virtualization layer makes each Container appear as a standalone server. Finally, the Container itself houses the application or workload.

The Parallels Virtuozzo Containers OS virtualization solution has the highest efficiency and manageability making it the best solution for organizations concerned with containing the IT infrastructure and maximizing the resource utilization. The Parallels Virtuozzo Containers complete set of management tools and unique architecture makes it the perfect solution for easily maintaining, monitoring, and managing virtualized server resources for consolidation and business continuity configurations.

About This Guide

This guide is a complete reference on all Parallels Virtuozzo Containers configuration files and Hardware Node command-line utilities. It familiarizes you with the way to configure Parallels Virtuozzo Containers to meet your requirements and to perform various tasks by using the corresponding Parallels command line utilities.

The primary audience for this guide is anyone who is looking for an explanation of a particular configuration option, does not understand a Parallels file format, needs help for a particular command, or is seeking for a command to perform a certain task.

Organization of This Guide

This guide is organized as follows:

- Chapter 1, Introduction, gives an overview of Parallels Virtuozzo Containers and this guide.
- Chapter 2, Parallels Virtuozzo Containers Utilities Overview, lists all utilities supported in the current version of Parallels Virtuozzo Containers.
- Chapter 3, General Utilities, describes utilities for performing day-to-day maintenance tasks.
- Chapter 4, Licensing Utilities, provides information on utilities for managing Parallels Virtuozzo Containers licenses.
- Chapter 5, Migration Utilities, focuses on utilities for migrating Containers between or within Hardware Nodes.
- Chapter 6, Backup Utilities, describes utilities for backing up and restoring Container private areas and configuration files.
- Chapter 7, Template Management Utilities, focuses on utilities for creating and managing templates.
- Chapter 8, Miscellaneous Utilities, concentrates on utilities for performing miscellaneous tasks in the Hardware Node and Container context.

Documentation Conventions

Before you start using this guide, it is important to understand the documentation conventions used in it. For information on specialized terms used in the documentation, see the glossary at the end of this document.

Typographical Conventions

The following kinds of formatting in the text identify special information.

Formatting convention	Type of Information	Example
`Preformatted`	On-screen computer output in your command-line sessions; source code in XML, C++, or other programming languages.	`Saved parameters for Container 101`
`Preformatted Bold`	What you type, as contrasted with on-screen computer output.	**`C:\>vzlist -a`**
`Monospace`	The names of commands, files, and directories.	Use `vzctl start` to start a Container.
`Monospace Italics`	Designates a command line placeholder, which is to be replaced with a real name or value.	To delete a Container, type `vzctl delete` *`CT_ID`*.
Special Bold	All elements of the graphical user interface (GUI): menu items, menu options, menu buttons, etc.	Go to the Resources tab.

	Titles of chapters, sections, and subsections.	Read the Basic Administration chapter.
Italics	Used to emphasize the importance of a point or to introduce a term.	*Host operating system* is an operating system installed on the Hardware Node.
CAPITALS	Names of keys on the keyboard.	SHIFT, CTRL, ALT
KEY+KEY	Key combinations for which the user must press and hold down one key and then press another.	CTRL+P, ALT+F4

General Conventions

Be aware of the following conventions used in this book.

- Chapters in this guide are divided into sections, which, in turn, are subdivided into subsections. For example, Documentation Conventions is a section, and General Conventions is a subsection.
- When following steps or using examples, be sure to type double-quotes (") and single-quotes (') exactly as shown.

Getting Help

In addition to this guide, there are a number of other resources available for Parallels Virtuozzo Containers 4.6 which can help you use the product more effectively. These resources include:

- *Getting Started With Parallels Virtuozzo Containers 4.6*. This guide provides basic information on how to install Parallels Virtuozzo Containers 4.6 on your server, create new Containers, and perform main operations on them.
- *Parallels Virtuozzo Containers 4.6 Installation Guide*. This guide provides exhaustive information on the process of installing, configuring, and deploying your Parallels Virtuozzo Containers system. Unlike the *Getting Started With Parallels Virtuozzo Containers 4.6* guide, it contains a more detailed description of all the operations needed to install and set Parallels Virtuozzo Containers 4.6 to work including planning the structure of your Parallels Virtuozzo Containers network, performing the Parallels Virtuozzo Containers unattended installation, etc. Besides, it does not include the description of any Container-related operations.
- *Parallels Virtuozzo Containers 4.6 User's Guide*. This guide provides comprehensive information on Parallels Virtuozzo Containers 4.6 covering the necessary theoretical conceptions as well as all practical aspects of working with Parallels Virtuozzo Containers. However, it does not deal with the process of installing and configuring your Parallels Virtuozzo Containers system.
- *Parallels Virtuozzo Containers 4.6 Templates Management Guide*. This guide is meant to provide complete information on Parallels Virtuozzo Containers templates, an exclusive Parallels Virtuozzo Containers technology allowing you to efficiently deploy standard Windows applications inside your Containers and to greatly save the Hardware Node resources (physical memory, disk space, etc.).

- *Parallels Management Console Help*. This help system provides detailed information on Parallels Management Console, a graphical user interface tool for managing Hardware Nodes and their Containers.
- *Parallels Virtual Automation Online Help*. This help system shows you how to work with Parallels Virtual Automation, a tool providing you with the ability to manage Hardware Nodes and their Containers with the help of a standard Web browser on any platform.
- *Parallels Power Panel Online Help*. This help system deals with Parallels Power Panel, a means for administering individual Containers through a common Web browser on any platform.

Feedback

If you spot a typo in this guide, or if you have an opinion about how to make this guide more helpful, you can share your comments and suggestions with us by completing the Documentation Feedback form on our website (http://sp.parallels.com/en/support/usersdoc/).

CHAPTER 2

Parallels Virtuozzo Containers Utilities Overview

This chapter documents the utilities supported in the current version of Parallels Virtuozzo Containers. For every utility, all available command-line options are described.

The Parallels Virtuozzo Containers command-line utilities can be subdivided into the following categories: general utilities, licensing utilities, Container migration utilities, Container backup utilities, template management utilities, and miscellaneous utilities.

General utilities for performing day-to-day maintenance tasks:

Name	Description
vzctl	Container management utility.
vzlist	Displays a list of Containers existing on the Node with additional information.
vzquota	Manages Parallels Virtuozzo Containers disk quotas.

Licensing utilities for managing Parallels Virtuozzo Containers licenses:

Name	Description
vzlicview	Shows the Parallels Virtuozzo Containers license status and parameters.
vzlicload	Installs Parallels Virtuozzo Containers licenses.

Container migration utilities for migrating Containers between or within Hardware Nodes:

Name	Description
vzmigrate	Migrates Containers to another Hardware Node.
vzmlocal	Migrates Containers within the same Hardware Node.
vzp2v	Migrates physical servers to Containers.

Container backup utilities for backing up and restoring Container private areas and configuration files:

Name	Description
vzabackup	Backs up Containers.
vzarestore	Restores Containers.
vzvssctl	Suspends/resumes Container activity during backups.

Template management utilities allow you to create new templates, install them on Hardware Nodes, and other operations:

Name	Description
vzpkgls	Displays templates installed on the Hardware Node and/or applied to Containers.
vzpkgdeploy	Install/removes templates on/from the Hardware Node.
vzpkgadd	Adds application templates to Containers.
vzpkgrm	Removes templates from Containers.

Miscellaneous utilities perform different tasks in the Hardware Node and Container context:

Name	Description
vzkeygen	Show the Hardware Node ID.
vzcache	Caches common files across Containers.
vzlscache	List folders created by `vzcache`.
vzuncache	Detaches Containers from their caches, copies cached files back to the Container private area.
vznetcfg	Manages virtual networks and network classes on the Hardware Node.
vznetstat	Displays traffic usage statistics for Containers.
vzdevctl	Forwards hardware devices (SCSI, iSCSI, USB flash drives, etc.) from the Hardware Node to Containers.
vzcpucfg	Manages CPU pools on the Hardware Node.
vzquery	Determines Container IDs using process and session IDs.
vzwinupdatecmd	Manages Windows Server updates inside Containers.

CHAPTER 3

General Utilities

This chapter describes utilities for performing day-to-day maintenance tasks.

In This Chapter

vzctl 12
vzlist 24
vzquota 28

vzctl

The the primary tool for Container management. To use it, you have to log in to the Hardware Node as the administrator.

Syntax

```
vzctl [--quiet|--verbose] command <CT_ID>
vzctl --version
vzctl --help
```

Commands

Name	Description
create	Creates a Container.
delete	Removes a Container.
destroy	Removes a Container.
mount	Mounts the Container private area.
umount	Unmounts the Container private area.
start	Starts a Container.
stop	Stops a Container.
restart	Restarts a Container.
status	Displays the Container status.
set	Sets Container parameters.
enter	Provides a way for the Hardware Node administrator to “enter” a Container without knowing the Container administrator password.

exec, exec2	Run arbitrary commands inside a Container without logging in to the Container. The difference between the command is their returned status.
mountext	Mounts folders and drives of the Hardware Node to Containers.
umountext	Unmounts external folders and drives of the Hardware Node from Containers.
partadd	Creates a new Container drive by mounting it to a loopback file on the Hardware Node.
partdel	Unmounts the Container drive mounted by means of the `vzctl partadd` command.
reinstall	Restores the original state of the Container system and application files.
shrink	Shrinks compact virtual disk drives inside Containers.
defrag	Defragments Container virtual disks.
addrole	Adds a new server role to the Container.
delrole	Removes a server role from the Container.
enumroles	Lists the server roles installed or ready to be installed inside Containers.

Verbosity Options

Name	Description
--quiet	Disables logging to the display and to the log file.
--verbose	Sets the log level to the maximum possible value.

Miscellaneous Options

Name	Description
--version	Displays the Parallels Virtuozzo Containers version installed on the Hardware Node.
--help	Displays the `vzctl` usage information.

vzctl create

Creates new Containers.

Syntax

```
vzctl create <CT_ID> --pkgset <name> [options]
```

A Container ID is required for this command and shall be unique within a Hardware Node. Container IDs from 1 to 100 are reserved for internal Parallels Virtuozzo Containers needs. Do not use IDs from 1 to 100 for your Containers.

Options

Name	Description

--pkgset <*name*>	Mandatory. Denotes the OS template to base the Container on.
--ipadd <*IP_address*>	Optional. Adds the specified IP address to the list of IP addresses the Container can use and brings up the network interface with this address inside the Container.
--hostname <*hostname*>	Optional. The hostname to assign to the Container.
--config <*filename*>	Optional. Applies the resource parameters from the specified configuration file to the Container. Configuration files are located in the *X*:\Program Files\Parallels\Containers\Configs folder on the Hardware Node and have the form ve-<*name*>.conf-sample.
--name <*name*>	An arbitrary name to assign to the Container. This name can be used, along with the Container ID, to refer to the Container while performing Container-related operations.
--private <*path*>	Optional. Specifies the location of the Container private area. The Container private area can be placed anywhere on your hard disk. While choosing a path for this folder, keep in mind the following: ▪ This folder cannot be a mount point, i.e. you cannot mount external disk partitions to this folder. ▪ This folder cannot be a network share, i.e. it cannot be located on a server network drive. ▪ The hard disk partition where this folder will be located must have no less than 10 Gb of free disk space.
--diskspace <*size*>	The total size of disk space to allocate to the Container, in 1 Kb blocks.
--disktype <plain | compact>	Optional. The type of the Container virtual disk drive. It can be one of the following: ▪ plain: in this case, the size of a virtual hard disk is fixed, i.e. all disk space is allocated during the virtual disk creation. ▪ compact (default): in this case you set only the maximal size of a virtual hard disk. A Container hard disk grows in size each time new data is written to the hard disk and can increase up to the maximum size specified during the hard disk creation.

vzctl delete, vzctl destroy

Deletes a Container from the Hardware Node.

Syntax

```
vzctl delete <CT_ID>
vzctl destroy <CT_ID>
```

When executed, vzctl delete or vzctl destroy physically removes all the files located in the Container private area with the default path of C:\vz\private*CT_ID*. These commands do not take any additional arguments and require the Container to be stopped and its private area to be unmounted.

vzctl mount, vzctl umount

Mounts or unmounts Container's private area to Container's root folder (*X*:\vz\root\<*CT_ID*>) without starting it and Container's registry branch to HKEY_LOCAL_MACHINE\<*CT_ID*> in the Node's registry.

Syntax

```
vzctl mount <CT_ID>
vzctl umount <CT_ID>
```

Note: Normally, you do not have to use these commands, because `vzctl start` and `vzctl stop` mount and unmount Container's private area and registry automatically.

vzctl start, vzctl stop, vzctl restart, and vzctl status

Starts, stops, restarts a Container, or queries its status.

Syntax

```
vzctl start <CT_ID>
vzctl stop <CT_ID>
vzctl restart <CT_ID>
vzctl status <CT_ID>
```

Note: Certain commands can only be performed after using `vzctl stop` (for example, `vzctl delete`).

vzctl set

Sets Container parameters.

Syntax

```
vzctl set <CT_ID> [options] [--save]
```

The optional `--save` switch instructs `vzctl` to save changes to the Container configuration file. Most of the Container settings can be changed dynamically without Container reboot.

General Options

Name	Description
--onboot <yes | no>	This setting requires the `--save` switch. If you set it to "yes", Parallels Virtuozzo Containers will automatically start the Container on the next system start-up.
--userpasswd <*user*:*password*>	This setting is used to set a new password for the specified user inside the Container (this user must already exist).

	Omitting the user name (e.g., `:XXXXXXXX`) will set the password for the built-in Administrator account inside the Container, even though it was renamed.
--offline_management <yes | no>	This setting enables/disables the direct managing of the Container through a common Internet browser by means of Parallels Power Panel. To enable/disable the offline management feature for the Hardware Node and, consequently, for all Containers, set this parameter for Container 0.
--showctid <yes | no>	This setting defines the appearance of the Container ID string in the top right corner of the Container desktop: ▪ If set to `yes`, the ID string is displayed in the top right corner of the Container desktop helping you identify the ID of the Container you are currently logged in to. ▪ If set to `no`, the ID string is not shown on the Container desktop. By default, this option is set to `yes`. Keep in mind that you must log off from the Container and log in to it anew for the changes to take effect. To disable the appearance of the ID string for all Containers that you will create on the Hardware Node, specify `0` as the Container ID when executing the `vzctl set` command.
--showhostname <yes | no>	This setting defines the appearance of the Container hostname string in the top right corner of the Container desktop: ▪ If set to `yes`, the hostname string is displayed in the top right corner of the Container desktop helping you identify the hostname of the Container you are currently logged in to. ▪ If set to `no`, the string is not shown on the Container desktop. By default, this option is set to `no`. Keep in mind that you must log off from the Container and log in to it anew for the changes to take effect. To disable the appearance of the hostname string for all Containers that you will create on the Hardware Node, specify `0` as the Container ID when executing the `vzctl set` command.
--name <*name*>	An arbitrary name to assign to the Container. This name can then be used, along with the Container ID, to refer to the Container while performing Container-related operations.
--description <*desc*>	Sets the Container description. You are allowed to use only symbols in the 'A -z' and '0-9' ranges in your descriptions.
--bootorder <*number*>	Sets the start-up priority for Containers, starting from 1. The lower number is assigned to the Container, the higher priority it has. Specifying `0` after `--bootorder` restores the default start order of the Container (defined by its ID).
--regowner <*name*>	Sets the registered owner name for the Container. By default, this name is set to `User`.

--regorganization <*name*>	Sets the registered organization name for the Container. By default, this name is set to `Organization`.

Resource Management Options

These options allow you to control resources a Container can consume.

Name	Description
-p, --numproc <*number*>	The number of processes allowed to simultaneously run inside the Container. Upon hitting this limit, the Container will not be able to start a new process.
-t, --numsessions <*number*>	The maximum number of concurrent terminal sessions to a Container. **Note:** Windows Server 2008 and newer operating systems maintain an extra terminal session to each Container. To provide for this, increase the option value by 1.
--cpuunits <*units*>	CPU power. This is a positive integer number that defines how much CPU time one Container will receive in comparison with the other Containers on the Hardware Node in case all the CPUs of the Node are fully used.
--cpuguarantee <*percent*>	CPU guarantee. This is a positive integer number that determines the minimal guaranteed share of the CPU time, in percent, the Container is guaranteed to receive.
--cpulimit <*percent*>	CPU limit. This is a positive number indicating the CPU time, in percent, the Container is not allowed to exceed. By default, this parameter is disabled for all Containers on the Hardware Node, i.e. any application inside any Container can use all the free CPU power of the Node.
--numa <yes | no>	Enables/disables the Non-Uniform Memory Access (NUMA) support inside the Container.
--diskspace <*number*>	The total size of disk space that can be consumed by the Container, in 1 Kb blocks. You can additionally use the `--drive` option to specify for which Container drive the disk space is to be set. If the `--drive` option is omitted, the disk space limit is set for the `C:\` drive.
--drive <*name*>	The name of the drive inside the Container for which the disk space limit is to be set. This option should be used in conjunction with the `--diskspace` option.
--vprvmem <*number*>	The size of private or potentially private memory that can be allocated to all applications inside the Container, in megabytes. Shared or potentially shared memory (e.g., memory mapped files) is not included in this resource parameter.
--disktype <plain | compact>	The type of the Container virtual disk drive. You can set the disk type to one of the following: ▪ `plain`: in this case, the size of a virtual hard disk is fixed, i.e.

	all disk space is allocated during the virtual disk creation. ▪ `compact` (default): in this case, you set only the maximal size of a virtual hard disk. A Container hard disk grows in size each time new data is written to the disk and can increase up to the maximum size specified during the disk creation.
--pagedpoollimit <*number*>	The amount of paged-pool memory, in megabytes, that can be allocated to the Container.
--nonpagedpoollimit <*number*>	The amount of non-paged-pool memory, in megabytes, that can be allocated to the Container.
--cpus <*number*>	If the Hardware Node has more than one CPU installed, sets the number of CPUs to be available to the Container. The changes made to the Container will take effect on the next Container start.
--rate <*class*:*Kbits*>	If traffic shaping is turned on, this parameter specifies the bandwidth limit for the Container. The format is *class*:*Kbits* where *class* is the network class (group of IP addresses) and *Kbits* is the traffic bandwidth.

Network-related Options

These options allow you to manage Container's hostname, IP address(es), DNS server address(es), etc.

Note: Most of the network options listed below can be used to configure both the default and additional network adapters inside a Container. In the latter case, you need to additionally specify the `--netif` option when running the `vzctl set` command.

Name	Description
--hostname <*name*>	Sets the hostname for the Container.
--ipadd <*addr*>	Adds the IP address to the list of IP addresses the Container can use and brings up the network interface with this address inside the Container. You can set several IP addresses and separate them by spaces.
--ipadd <*addr*/*netmask*>	Along with adding the IP address to the Container, sets its subnet mask.
--ipdel <*addr* | all>	Removes the specified IP address from the Container. Specifying `all` after the option, removes all IP addresses assigned to the Container.
--nameserver <*addr*>	Sets the DNS server for the Container. More than one server can be specified in the space-separated format.
--searchdomain <*domain*>	Sets the DNS search domain for the Container. You can specify several domains and separate them by spaces.
--psched <on | off>	Enables/disables the Quality of Service packet scheduler inside the Container. By default, the scheduler is disabled. **Note:** If you have enabled the QoS packet scheduler for

	a running Container, restart the Container for the changes to take effect.
--preferred_adapter <*adapter_ID*>	If several network adapter cards are installed on the Hardware Node, you can use this option to specify the preferred network adapter for a Container. Specifying the preferred adapter for a Container means that this network adapter will be used to connect the Container to the network and handle the whole network traffic of the Container. In Parallels-based systems, network adapters are identified by the Media Access Control (MAC) addresses assigned to them. To set a preferred adapter for a Container, specify its MAC address as the value of this option. While working with network adapters in Parallels Virtuozzo Containers-based systems, keep in mind the following: ▪ After the Parallels Virtuozzo Containers installation, all active adapters on the Hardware Node are used to control network traffic for all Containers on this Node. ▪ You can specify several network adapters for a Container and separate them by spaces. In this case, the Container will be able to access all networks to which the specified adapters are connected. ▪ If you use the option without specifying any value (i.e. execute the `vzctl set` *`CT_ID`* `--preferred_adapter ""` command), the network adapters of Container 0 (i.e. of the Node itself) will be used to control network flows for the corresponding Container. If there is no adapter set for Container 0, then all the adapters currently active on the Node will handle the Container network traffic.
--nettype <bridged | routed>	Sets the operating mode (either bridged or host-routed) for the Container virtual network adapter.
--network <*network_ID*>	Connects the Container virtual network adapter to the specified Virtual Network.
--mac <*mac*>	The MAC address to be assigned to the Container virtual network adapter.
--gateway <*addr*>	Specifies the IP address of the device to be used for routing the traffic from the Container virtual network adapter to external networks.
--winsserver <*addr*>	The IP address of the WINS server to be used by the Container virtual network adapter.
--vpn <on | off>	Enables/disables the Virtual Private Network (VPN) support by the Container virtual network adapter. By default, the VPN support is disabled.
--nat <on | off>	Enables/disables the Network Address Translation (NAT) functionality for the Container virtual network adapter. By default, NAT is disabled for any newly created Container network adapter.

--cluster_ip <*addr*>	Sets the virtual (Network Load Balancing cluster) IP address for the Container virtual network adapter. Can be set in one of these forms: ▪ `IP_address` ▪ `IP_address:subnet_mask` ▪ `IP_address/subnet_mask`
--nlb <yes | no>	Enables/disables the support for the Network Load Balancing (NLB) feature inside the Container. By default, the NLB support is disabled.
--nlb_mode <multicast | unicast>	Sets the operating mode of Network Load Balancing inside the Container. By default, the mode is set to `multicast`.
--tsmode <admin | app_device | app_user>	Sets the Terminal Services mode for the Container: ▪ If set to `admin`, enables the Remote Desktop for Administration mode for the Container. ▪ If set to `app_device` (or `app`), enables the device-based licensing scheme of the Terminal Server mode for the Container. ▪ If set to `app_user`, enables the user-based licensing scheme of the Terminal Server mode for the Container. By default, all newly created Containers is set to work in the Remote Desktop for Administration mode during its creation. Container restart is needed for the new setting to come into effect.
--tslicservers <*name*>	The NETBIOS name or the IP address of the Terminal Server License (TSL) server to be used by the Container. You can specify several TSL servers and separate them by spaces.
--port_mapping <tcp | udp>:<*ext_port*>-<*int_port*> [`...`]	Sets port mapping.
--promiscuous <yes | no>	Enables/disables the promiscuous mode for the Container network adapter.
--broadcasts <on | off>	Enables/disables the Container network adapter to receive network broadcasts. This option is valid only for Containers operating in the host-routed mode.
--failover_cluster <yes | no>	Enables/disables the support for the Microsoft clustering software inside the Container. To enable/disable the possibility of using the clustering software for all Containers on the Hardware Node at once, specify `0` as <*CT_ID*>.
--scsi <yes | no>	Enables/disables SCSI-aware drivers to load inside the Container. To enable/disable the loading of SCSI-aware drivers for all Containers on the Hardware Node at once, specify `0` as <*CT_ID*>.

--dhcp <on | off>	When set to `on`, enables the Container network adapter to get IPv4 configuration settings via the DHCP protocol. By default, this option is set to `off`.
--dhcp_ipv6 <on | off>	When set to `on`, enables the Container network adapter to get IPv6 configuration settings via the DHCP protocol. By default, this option is set to `off`.
--dns_suffix <*suffix*>	Sets the Domain Name System (DNS) suffix for the Container.
--iplock <on | off>	Enables/disables the possibility of configuring IP address settings and DHCP support from inside the Container.
--netcfglock <on | off>	Enables/disables the possibility of configuring WINS server and DNS suffix settings from inside the Container.
--netif_add <*name*>	Creates a new virtual network adapter for the Container with the name specified.
--netif_del <*name*/*GUID*>	Removes the virtual network adapter with the specified name or GUID (globally unique identifier) from the Container.
--netif <*name*>	Specifies the name of the Container virtual network adapter whose settings you want to configure. Omit this option if you have only one network adapter inside a Container.

vzctl exec, vzctl exec2, and vzctl enter

Runs arbitrary commands inside a Container.

Syntax

```
vzctl <exec|exec2> <CT_ID> <command>
vzctl enter <CT_ID>
```

where *command* is a string to be executed in the Container.

The difference between `exec` and `exec2` is the exit code. `vzctl exec` returns 0 in case `vzctl` has been able to launch the command and does not take into account the exit code of the command itself. `vzctl exec2` returns the exit code of the command executed in the Container.

`vzctl enter` is similar to `vzctl exec`. The difference between the two is that `vzctl enter` makes the command line believe that it is connected to a terminal. As such, you receive a command line prompt and are able to execute multiple commands as if you were logged in to the Container.

vzctl mountext, vzctl umountext

Mounts a drive on the Hardware Node to a drive or folder inside a Container.

Syntax

```
vzctl mountext <CT_ID> --srcdir <path> --dstdir <path>
```

Options

Name	Description
<CT_ID>	The ID of the Container where to mount the drive/folder.
--srcdir *<path>*	The full path to the drive/folder inside the Container where to mount the drive. The drive from the Node can be mounted to any drive/folder inside the Container. If the specified drive/folder does not exist, it is automatically created.
--dstdir *<path>*	The full path to the drive on the Node.

vzctl partadd, vzctl partdel

The `vzctl partadd` command creates a new drive inside a Container by mounting it to a loopback file on the Hardware Node. The `vzctl partdel` command unmounts the Container drive mounted with `vzctl partadd` from a file on the Hardware Node.

Syntax

```
vzctl partadd <CT_ID> --drive <CT_drive> [--size <drive_size>] [--file <filename>]
vzctl partdel <CT_ID> --drive <CT_drive> [--delete]
```

Options

Name	Description
--drive *<CT_drive>*	Assigned drive name in one of these formats: *X*, *X*:, *X*:\.
--size *<drive_size>*	Drive size, in kilobytes.
--file *<filename>*	File to create on the Node. Upon the command completion, it will have the `.efd` extension and will be located in the *X*:\vz\private*CT_ID* folder on the Node. If you omit this option, the disk will be assigned a name in the `lpbkXXXX.efd` format.
--delete	Removes the file the Container drive has been mounted to from the *X*:\vz\private*CT_ID* folder on the Node. If this option is omitted, the file remains.

vzctl reinstall

Restores the original state of the Container system and application files. The Container is restored to the state it has been in at the time of creation and/or when other applications were added to the Container afterwards.

Syntax

```
vzctl reinstall <CT_ID> [options]
```

Options

Name	Description
--resetpwdb	Removes the `Administrator` credentials from the users' database and creates a clean database as for any new installation. **Note:** During Container reinstallation, the username is always reset to `Administrator` whether this option is used or not.
--skipbackup	The contents of the old private area are not saved in the `reinstall` folder.
--skipapps	Skips reinstalling the application templates inside the Container.

When executed, the `vzctl reinstall` command creates a new private area for the Container and rewrites the Container from scratch using its configuration file (thus retaining the Container IP address, hostname, resource control parameters, and all the other settings). The contents of the Container old private area are then copied to the `C:\reinstall` folder in the new private area, to retain the user files.

vzctl shrink

Removes unused space from the specified virtual disk, reducing its footprint on the Hardware Node.

```
vzctl shrink <CT_ID> --drive <CT_drive>
```

Options

Name	Description
<CT_ID>	Container ID.
--drive *<CT_drive>*	Drive name in one of these formats: *X*, *X*`:`, *X*`:\`.

When using this command, keep in mind the following:

- You can shrink virtual disks inside both running and stopped Containers.
- Shrinking a Container virtual disk does not reduce the maximum capacity of the virtual disk itself.
- You cannot shrink plain virtual disks.

vzctl defrag

Checks virtual disk fragmentation and defragments it if necessary.

Syntax

```
vzctl defrag <CT_ID> --drive <CT_drive> [--force] [--analysis]
```

Options

Name	Description
<*CT_ID*>	Container ID.
--drive <*drive*>	The drive to defragment/analyze. Can be specified in one of the following formats: *X*, *X*:, *X*:\.
--force	Force defragmentation. You can use this option if you do not have enough free space on the virtual disk. Keep in mind, however, that forcing defragmentation when there is insufficient disk space may result in partial defragmentation.
--analysis	Analyze the virtual disk without defragmenting it.

vzctl addrole, vzctl delrole, vzctl enumrole

Adds, removes, or lists server roles, role services, and features inside Containers running the Windows Server 2008 or Windows Server 2008 R2 operating system.

Syntax

```
vzctl addrole <CT_ID> --role <role_name> [...] [--restart]
vzctl delrole <CT_ID> --role <role_name> [...] [--restart]
vzctl enumroles <CT_ID>
```

Options

Name	Description
<*CT_ID*>	Container ID.
--role <*role_name*> [...]	Role(s) to install or remove. To specify multiple roles, separate them by white spaces.
--restart	Restart the Container once installation or removal is complete.

vzlist

Lists Containers existing on the Hardware Node and displays additional information about them.

Syntax

```
vzlist [--all] [--stopped] [--name [CT_ID]] [-o <specifier> [...]] [--no-header]
       [CT_ID [...]] [--sort [-]parameter[.specifier]] [--hostname <pattern>]
       [--name_filter <pattern>] [--description <desc>] [--quiet] [--verbose]
vzlist --list
```

```
vzlist --help
```

Options

Name	Description
-a, --all	List the Containers existing on the Hardware Node. By default, only running Containers are shown.
-S, --stopped	List only stopped Containers.
-o <*specifier*>	Display only particular information about the Containers. The parameters that can be used after the `-o` option are listed in the following subsection. To display a number of parameters in a single output, separate them with commas.
-H, --no-header	Do not display column headers.
-L, --list	List all parameters for `vzlist`.
--help	Display command usage information.
-s, --sort *[-]parameter[.specifier]*	Sort the Containers in the list by the specified parameter. If "`-`" is given before the name of the parameter, the sorting order is reversed.
-h, --hosthame <*pattern*>	Display only those Containers whose hostnames correspond to the specified pattern. The following wildcards can be used: `*`.
-n, --name	If used without any parameters, displays information on all Containers on the Node together with their names. If you indicate the Container ID after this option, displays information on the specified Container only.
--netif, -i <*fname*>	Displays the IP properties of the network interface `fname` instead of the default one.
-N, --name_filter <*pattern*>	Displays only the Container that corresponds to the specified pattern.
-d, --description <*desc*>	Displays only the Container whose description corresponds to the specified pattern.
--quiet	Disables logging to the display and to the log file.
--verbose	Sets the log level to the maximum possible value.

vzlist Output Parameters and Their Modifiers

Some parameters that can be used after the `-o` and `-s` switches of the `vzlist` utility can be specified by the "dot+letter" combination following the parameter and denoting one of the following things:

Specifier	Description
.b	The barrier on using the corresponding resource set for the given Container.
.h	The hard limit on using the corresponding resource set for the given Container.

The following parameters are available for using with the utility:

Parameter	Output Column	Description
ctid	CTID	The Container ID.
hostname	HOSTNAME	The Container hostname.
ip	IP_ADDR	The Container IP address.
status	STATUS	Specifies whether the Container is running or stopped.
templates	TEMPLATES	Displays the OS and application templates applied to the Container. If the `ctid` parameter is not specified, all templates available on the Hardware Node are shown.
rates	RATES	The bandwidth limit set for the Container.
name	NAME	Container name.
description	DESCRIPTION	Container description.
numproc	NPROC	The number of processes allowed to simultaneously run in the Container. Can be used with the `.b` specifier.
numsessions	SESSIONS	The number of terminal sessions opened to the Container. Can be used with the `.b` specifier. **Note:** Windows Server 2008 and newer operating systems maintain an extra terminal session to each Container. As a result, this counter remains non-zero even when no user-initiated sessions are open.
diskspace	DQBLOCKS	The total size of disk space consumed by the Container, in 1 Kb blocks. Can be used with the `.h` specifier defining the hard limit on using disk space.
cpuunits	CPUUNI	Allowed CPU power. This is a positive integer number that defines how much CPU time one Container will receive in comparison with the other Containers on the Hardware Node in case all the CPUs of the Node are fully used.
cpuguarantee	CPUGUARANTEE	CPU guarantee. This is a positive integer number that determines the minimal guaranteed share of the CPU time, in percent, the Container is guaranteed to receive.
cpulimit	CPULIMIT	CPU limit. This is a positive number indicating the CPU time, in percent, the Container is not allowed to exceed.
cpus	CPUS	The number of CPUs available to the Container.
cpupool	CPUPOOL	The CPU pools assigned to the Container.
vprvmem	VPRVMEM	The size of private or potentially private memory that can be allocated to all applications inside the Container, in megabytes. Shared or potentially shared memory (e.g., memory mapped files) is not included in this resource parameter.
pagedpool	PAGEDPOOL	The amount of paged-pool memory, in megabytes, that

		can be allocated to the Container.
disktype	DISKTYPE	The type of the Container virtual disk drive. It can be one of the following: ▪ `plain`. The size of a virtual hard disk is fixed and all disk space is allocated during the virtual disk creation. ▪ `compact`. Only the maximal size of a virtual hard disk is set. The Container hard disk grows in size each time new data is written to the hard disk and can increase up to the maximum size specified during the hard disk creation.
bootorder	BOOTORDER	The start-up priority set for the Container. The lower number is assigned to the Container, the higher priority it has.
regowner	REGOWNER	The registered owner name set for the Container.
regorganization	REGORGANIZATION	The registered organization name set for the Container.
flags	FLAGS	Displays miscellaneous options (offline management, the support for using the Microsoft Cluster Server software and loading SCSI-aware drivers, etc.) currently enabled for the Container.
network	NETWORK	The Virtual Network to which the Container virtual adapter is connected.
nettype	NETTYPE	The network operating mode set for the default Container.
gateway	GATEWAY	The IP address of the device used for routing the traffic from the Container to external networks.
nameserver	NAMESERVER	The DNS servers set for the Container.
winsserver	WINSSERVER	The WINS server set for the Container.
clusterip	CLUSTERIP	The virtual (Network Load Balancing cluster) IP address assigned to the Container.
ifname	INTERFACES	The list of virtual network adapters available inside the Container.
tsmode	TSMODE	The Terminal Services mode set for the Container: ▪ `admin`: the Remote Desktop for Administration mode is currently enabled for the Container. ▪ `app_device` (or `app`): the device-based licensing scheme of the Terminal Server mode is currently enabled for the Container. ▪ `app_user`: the user-based licensing scheme of the Terminal Server mode is currently enabled for the Container.

If a parameter that can be used with a specifier is used without any specifier in the command line; the current usage of the corresponding resource is shown by default.

vzquota

Configures Container disk quotas, shows Container disk quota statistics.

Note: `vzctl` uses `vzquota` to configure Container quotas and you usually do not have to use `vzquota` itself except for checking the current quota statistics.

Syntax

```
vzquota [--quiet|--verbose] [-b] setlimit <CT_ID> -B <num> [--drive <drive_name>]
vzquota [--quiet|--verbose] [-b] <stat|show> <CT_ID>
vzquota --help
```

Commands

Name	Description
setlimit	Configure the quota limit for the running quota.
stat	Show quota statistics for the running quota.
show	Show quota usage from the quota file.

Options

Name	Description
-B, --block-hardlimit <*num*>	Specifies the disk quota block hard limit, in 1KB blocks.
--drive <*drive_name*>	Sets the disk drive inside the VPS for which the disk quota limit is to be set.
-t, --type	Shows loopback drive type column.
-s, --size	Shows loopback drive size column.
-b	Enables batch mode.
-v, --verbose	Enables verbose mode. `vzquota` will print output messages about operation progress. Up to two such options are supported to increase verbosity.
-q, --quiet	Enables quiet mode. Suppress warning and diagnostic messages, display only fatal errors.
--help	Displays command usage information.

vzquota setlimit

Updates limits for the running Container quota. The command requires at least one limit to be specified.

Syntax

```
vzquota setlimit <CT_ID> -B <num> [--drive <name>]
```

Options

Name	Description
-B, --block-hardlimit <*num*>	Specifies the disk quota block hard limit, in 1 KB blocks. This limit cannot be exceeded by the Container.
--drive <*name*>	Specifies the disk drive inside the Container for which the disk quota limit will be set.

For its execution, `vzquota setlimit` requires only the ID of the Container and the disk quota limit to be specified. However, you can additionally use the `--drive` option to specify to what Container disk drive the defined quota limit is to be applied.

Note: The disk quota limit for all Container disk drives except for the `C:\` drive is set on the fly. To make the disk quota limit changes apply to the `C:\` drive, restart the Container where the drive is located.

vzquota stat, vzquota show

`vzquota stat` shows quota statistics for the running quota. `vzquota show` shows quota usage from the quota file.

Syntax

```
vzquota show <CT_ID>
vzquota stat <CT_ID>
```

CHAPTER 4

Licensing Utilities

This chapter provides the information on utilities for managing Parallels Virtuozzo Containers licenses.

In This Chapter

vzlicload 30
vzlicview 31

vzlicload

Parallels Virtuozzo Containers license management utility.

Syntax

```
vzlicload [options]
```

Options

Name	Description
-p, --product-key <*key*>	Installs the Parallels Virtuozzo Containers license on the Hardware Node.
-i, --stdin	Makes `vzlicload` use standard input while installing the Parallels Virtuozzo Containers license on the Hardware Node.
-f, --license-file <*path*>	The full path to the license file containing the Parallels Virtuozzo Containers license to be installed on the Hardware Node.
-u, --update	Connects to the Parallels Key Authentication (KA) server and updates the license installed on the Hardware Node.
-n, --no-check	Skips license validity check and forcibly connects to the Parallels Key Authentication (KA) server during license update.
-t, --transfer	Transfers the Parallels Virtuozzo Containers license from one Hardware Node (Source Node) to another (Destination Node). You should use this option along with the `-p` option when running the `vzlicload` utility on the Destination Node.
-r, --remove <*key*>	Removes the Parallels Virtuozzo Containers license with the specified serial number from the Hardware Node. You can find out the license serial number using the `vzlicview` utility (see vzlicview (p. 31)).
-A, --proxy-server <*addr*|*hostname*>	The IP address or hostname of the proxy server, if you use any, to

	connect to the Internet. By default, the Internet Explorer proxy settings are used.
-U, --username <*username*>	Username for the proxy server.
-P, --password <*password*>	Password for the username specified with the `-U` option.
-h, --help	Displays the utility usage and exits.

vzlicview

Displays license information.

Syntax

```
vzlicview [options]
```

Options

Name	Description
-f, --license-file <*file*>	Displays the license information from the specified Parallels Virtuozzo Containers license file.
-p, --product-key <*key*>	Displays the license information contained in the specified Parallels Virtuozzo Containers product key.
-i, --stdin	Makes `vzlicview` use standard input as a license and display its information.
-v, --validate	Displays the total number of Containers which may simultaneously run on the Hardware Node in accordance with the license currently installed on this Node.
-h, --help	Displays command usage information.

License Statuses

Name	Description
ACTIVE	The license is valid.
INVALID	The license is invalid (for example, because of the Hardware Node ID mismatch) or corrupt.
EXPIRED	The license file matches the Hardware Node ID, but has expired.
GRACED	The license has expired and is currently on the grace period (i.e. it is active till the end of the grace period).

CHAPTER 5

Migration Utilities

This chapter focuses on utilities for migrating Containers between Hardware Nodes or within one Hardware Node.

In This Chapter

vzmigrate 32
vzp2v 33
vzmlocal 34

vzmigrate

Moves Containers from a source Node to a destination Node with minimal downtime.

Syntax

```
vzmigrate [--remove-area <yes|no>] [--nostart] --srv_addr <addr> --srv_addr <username>
          --srv_pswd <passwd> <source_CTID>[:dest_CTID] [...]
vzmigrate --help
```

Options

Name	Description
-r, --remove-area <yes | no>	If “yes” is specified, then the private area will be deleted after the corresponding Container is successfully migrated to the destination Node. If “no” is specified, the private area will not be deleted. In this case the Container configuration file will be renamed from *CT_ID*.conf to *CT_ID*.conf_migrated. By default, the Container private area is removed from the source Node.
-n, --nostart	Do not start the Container on the destination Node after migration. This option does not have any effect if the Container was not running on the source Node.
-A, --srv_addr <*addr*>	The IP address or the hostname of the destination Node, i.e. of the Node where the Container is to be migrated.
-U, --srv_user <*username*>	The username to log in to the destination Node with. Must have administrator rights.
-P, --srv_pswd <*passwd*>	The password for the username provided with the --srv_user option.

-h, --help	Displays command usage information.
--quiet	Disables logging to the screen and to the log file.
--verbose	Sets the log level to the maximum possible value for this `vzmigrate` session.

To migrate the Container, execute the `vzmigrate` command on the source Node and pass the corresponding options to it. A different `dest_CTID` parameter is needed in case the source Node (the one where you run `vzmigrate`) and the destination Node have Containers with the same ID. You can specify multiple Containers for migration.

If the Container is running on the source Node, `vzmigrate` copies its private files, registry settings, etc. to the destination Node, stops the Container on the source Node, copies the files changed after the first copying to the destination Node again, and starts the Container on the destination Node (if the `-n` option is not specified).

Since a Container consists of thousands of files, copying all of them to a new Hardware Node may take considerable time. However, `vzmigrate` stops the Container only after all the files have been copied to the new Hardware Node and resynchronizes only those files that have been changed while being copied. This brings down a typical Container downtime to the time required for restarting a Container.

vzp2v

Migrates a physical server to a Container on the Hardware Node.

Syntax

```
vzp2v <CT_ID> --src_addr <addr> --src_user <username> --src_pswd <passwd>
      [--src_stop] [--ve_start] [--exclude <disk_drive> [...]] [--bridged <vn_ID>]
vzp2v --help
```

Options

Name	Description
-A, --src_addr <*addr*>	Mandatory. The IP address or hostname of the physical server you are going to migrate to a Container on your Node.
-U, --src_user <*username*>	Mandatory. The username to log in to the physical server. The specified user should have the Administrator rights to the server.
-P, --src_pswd <*passwd*>	Mandatory. The password of the user specified as the value of the `-U` option.
-s, --src_stop	Optional. Stop the physical server after its successful migration. If this option is omitted, the physical server continues running upon the migration completion.
-v, --ve_start	Optional. Start the Container on the Hardware Node after the physical server has been successfully migrated. If the option is not specified, the Container is not started after the physical server migration.

-x, --exclude <*drive*>	Optional. The name of the disk drive that you do not wish to move to the Container. The formats that can be used for specifying the drive name are *X*, *X*: or *X*:\ where *X* denotes the name of the drive that is not to be moved to the Container. You can specify several drives to be excluded from the migration process and separate them by spaces. If you omit this option, all disk drives available on the physical server will be migrated to the Container.
-b, --bridged <*virt_net_ID*>	Optional. If one or more network adapters on the physical server use the DHCP protocol to receive their IP address, you should specify this option to denote the Virtual Network on the Hardware Node where these network adapters are to be connected. If you omit this option, you will need to manually configure the Container network adapter parameters for it to be visible on your network. Detailed information on Virtual Networks is provided in the *Parallels Virtuozzo Containers 4.6 User's Guide*.
--help, /?	Displays command usage information.

To migrate a physical server, you should execute the `vzp2v` command on the Hardware Node and pass the corresponding options to it. Before starting the migration process, please make sure that there are no Containers on your Node with the ID you are going to specify as *CT_ID*; otherwise, the migration will fail.

vzmlocal

Moves/copies Containers within the same Hardware Node. Essentially, this command changes the existing Container ID and private area path to move that Container, or creates a copy of an existing Container with a different ID and private path.

Syntax

```
vzmlocal [options] <source_CTID>:<dest_CTID>[:<dest_private>] [...]
vzmlocal --help
```

Options

Name	Description
<*source_CTID*>	Source Container ID.
<*dest_CTID*>	Destination Container ID.
<*dest_private*>	Destination Container's private path. Overrides the default path (*X*:\vz\private*CT_ID*).
-C, --copy	Clones the source Container instead of moving it.
-s, --fast-sid	Speeds up Container cloning by doing a faster SID scan (checks only the registry and files subset).
-d, --destroy-source	Destroys the source Container after cloning.

-n, --disable-network	Disables offline management for the source Container after cloning.
-#, --custom-storage	Clone to custom storage.
-l, --skiplock	Allows you to clone locked Containers.
-S, --sid-change <block | newvpsid | none>	Allows you to specify the SID change method. Default: `block`. ▪ `block`: Changes the SID throughout the new Container's virtual disk. ▪ `newvpsid`: Changes the SID at preset locations inside the new Container. ▪ `none`: Does not change the SID.
--verbose	Sets log level to maximum possible value for this `vzmlocal` session.
--help	Displays command usage information.

CHAPTER 6

Backup Utilities

This chapter describes utilities for backing up and restoring Container private areas and configuration files.

In This Chapter

vzabackup 36
vzarestore 38
vzvssctl 40

vzabackup

Creates backups of Containers or entire Hardware Nodes. The `vzabackup` utility can be run on:

- The Source Node where the Container to be backed up is residing.
- A Backup Node intended for storing Container backups.
- Any other physical server on your network running Parallels Virtuozzo Containers.

The only requirements that should be met to execute `vzabackup` is to have a server with Parallels Virtuozzo Containers and the Parallels Agent software installed on it and to provide the network connectivity for this server to be able to establish connections to the Source and Backup Nodes, if necessary. The created Container backups are then stored on the Backup Node which can also be presented by any Node with running the Parallels Virtuozzo Containers and Parallels Agent software.

Syntax

```
vzabackup [backup_options] <source_node> [...] [CT_options]
```

where `source_node` is the Source Node IP address and credentials as follows: [*user*[**:***passwd*]**@**]<*source_node_addr*>. If you omit the credentials, you will be asked to provide them during `vzabackup` execution.

Backup Options

Name	Description
-F, -l, --TFull	Make a full backup. By default, `vzabackup` creates a full backup of Containers and Hardware Nodes.
-i, --Tinc	Make an incremental backup or, if no full backups are available, a full backup. If this option is omitted, the full backup is created.

--Tdiff	Make a differential backup or, if no full backups are available, a full backup. If this option is omitted, the full backup is created.
-D *<desc>*	The description of the backup archive.
-o, --rm-old	Create a new backup and then remove the oldest backup of the specified Node/Container.
--rm-tag *<backup_ID>*	Create a backup and then remove the backup with the specified ID. You can learn what ID is assigned to what Container backup using the `-l` and `-f` options of the `vzarestore` utility.
-C<n | g | b>	Indicates the level of compression for the resulting Container backup archive. In the current version of Parallels Virtuozzo Containers, you can set this level to one of the following: ▪ `n`: create the Container backup without any compression. Using this level of compression may speed up the backing up time; however, it may significantly increase the size of the resulting backup file. ▪ `g`: compress the resulting backup with the normal level of compression. This is the default level of compression used to back up all Nodes/Containers. ▪ `b`: compress the resulting backup with the maximum level of compression. In this case the backup file size is the smallest; however, it may take much time to create the backups. The optimal data compression level depends on the type of files to be stored in the backup archive. For example, it is advisable to use the 'normal' and 'none' compression types if most of the files to be backed up are already compressed (e.g. the files with the `.zip` and `.rar` extensions) or can be compressed with a low degree of efficiency (e.g. all executable files with the `.exe` extension or image files with the `.jpg`, `.jpeg`., and `.gif` extensions).
-J	If several Source Nodes are specified, back up these Nodes and/or their Containers simultaneously. If the option is omitted, the Nodes are backed up sequentially one after another.
--force	Force the process of backing up the Hardware Node(s)/Container(s). Can be used when backing up several HNs/Containers to continue the backup process even if some errors occur when creating a backup of a certain Node/Container.
--view-folder	Shows current backup storage configuration.
--set-folder-creds *<user>*[:*passwd*]	Set backup storage login credentials. Required for Samba shares.
--backup-folder-path *<path>*	Path to a custom backup storage location.
--backup-folder-login *<user>*	Username for a custom backup storage on a Samba share.
--backup-folder-passwd *<passwd>*	Password for a custom backup storage on a Samba share.
--set-folder	Use `--backup-folder-*` option values to change backup storage configuration.
--storage [*user*[:*passwd*]@]*<addr>*	The IP address and the credentials of the Backup Node. If you do not indicate the password to log in to the Backup Node, you will be asked to do so during

	the `vzabackup` execution. If this option is omitted, `vzabackup` puts the created backup(s) to the backup folder on the Source Node (by default, this folder is *X*`:\vz\Backups`).
-q, --no-progress	Disables logging to the screen during the `vzabackup` operation.

Container Options

Name	Description
-e <*CT_ID*> `[...]`	The Containers to back up on the Source Node. If this option is omitted, all Containers on the given Node will be backed up. Containers can be specified using both their IDs (e.g. `101` or `102`) and their names (e.g. `comp1` or `comp2`).
-x <*CT_ID*> `[...]`	The Containers that need not be backed up (Containers to exclude). If this option is omitted, all Containers on the given Source Node will be backed up. Containers can be specified using both their IDs (e.g. `101` or `102`) and their names (e.g. `comp1` or `comp2`).
--include-files <*list*>	Only the specified files and folders will be included in the Container backup. **Note:** If you create a Container backup using this option, you will be able to restore only separate files from the resulting Container backup, but not the Container as a whole.
--exclude-files <*list*>	The path to the files and folders inside the Container to be excluded from the backup.

vzarestore

Manages Container backups: restores Containers or Container files/folders from the Container backup, lists backups on the Backup Node, removes backups, etc. The `vzarestore` command can be run on any Hardware Node that has Parallels Agent installed.

Syntax

```
vzarestore [CT_ID | -e <CT1>[,...] [-x <CT1>[,...]]] [restore_options] [backup_node]
vzarestore -l,--list [list_options] [backup_node]
vzarestore -r,--remove <backup_ID> [...]
vzarestore --browse <backup_ID> [-d <path>] [browse_options] [backup_node]
vzarestore --print-ct-config <backup_ID> [backup_node]
vzarestore --help
```

Options

Name	Description
-e <*CTID*> [,...]	The Containers to be restored on the Destination Node. Any Container can be specified using both its IDs (e.g., `101` or `102`) and its names (e.g., `comp1` or `comp2`).
-x <*CTID*> [,...]	The Containers that need not be restored (the Containers to exclude). Any Container can be specified using both its IDs (e.g., `101` or `102`) and its names (e.g., `comp1` or `comp2`).
-b <*backup_ID*>	The ID assigned to the Container backup. This ID can be used to manage the backup archive (e.g., to delete the Container backup from the Backup Node or restore the files from the Container backup with the specified ID). If this option is omitted, the latest Container backup is used.
--force	Do not stop on errors during the `vzarestore` execution. Can be used when you are restoring more than one Container. This option allows you to continue the restoring process if some errors occur when restoring a certain Container.
--skip-ct-config	Do not restore the Container configuration file. Can be used only if you are restoring a single Container. **Note:** The Container configuration file is not changed when restoring separate Container files.
--files <*path*>	The full path to the files/folders inside the Container to be restored. This options is incompatible wit the `-e` and `-x` options.
--backup-folder-path <*path*>	Path to a custom backup storage location.
--skip-locked	Do not stop on errors even if some of the files to be restored are in the 'locked' state.
-B	Handle the values after the `-e` and `-x` options as backup IDs.
--storage [username[:passwd]@]<*addr*>	The IP address and the credentials of the Backup Node. f this option is omitted, `vzarestore` looks for the Container backup on the local Node.

List Options

Name	Description
f, --full	Display detailed information on the backed up Containers.
--latest	Display the latest Container backup.
-e <*CTID*> [,...]	Display the information on the backups for the specified Containers only.
-B	Handle the values after the `-e` option as backup IDs.

Miscellaneous Options

Name	Description
-d <*path*>	The path to the folder inside the Container backup archive whose contents is to be shown. Used only with the `vzarestore --browse <backup_ID>` command.
--print-ct-config <*backup_ID*>	Displays the parameters set in the Container configuration file at the moment of creating the backup with the specified backup ID.
-r, --remove <*backup_ID*>	Removes the Container backup with the specified backup ID.
--help	Shows command usage information.

vzvssctl

Suspends or resumes Container activity (e.g., during backup).

Syntax

```
vzvssctl <-s|-r> [CT_ID] [...]
vzvssctl --help
```

Options

Name	Description
-s	Suspends the Container activity.
-r	Resumes the Container activity.
--help, /?	Displays the utility usage and exits.

CHAPTER 7

Template Management Utilities

This chapter focuses on utilities for creating and managing templates.

In This Chapter

vzpkgls 41
vzpkgdeploy 42
vzpkgadd 42
vzpkgrm 43

vzpkgls

Lists OS and application templates installed on the Hardware Node or applied to a Container. If you specify multiple Container IDs, the utility will list templates applied to these Containers. Without a Container ID as an argument, the utility lists all templates available for Containers on the Hardware Node.

Syntax

```
vzpkgls [options] [CT_ID] [...]
```

Options

Name	Description
-d, --deployed	Skips templates which are cached but not installed.
-o, --os	Lists all OS templates installed on the Hardware Node.
-a, --application	Lists all application templates installed on the Hardware Node.
-s, --separate	By default, `vzpkgls` outputs the template and all its updates on a single line separated by spaces. This option changes the output and prints a separate line for each available version of the template.
-q, --quiet	Disables logging to the display and to the log file.
-h, --help	Displays the usage information and exits.

vzpkgdeploy

Installs/removes OS and application templates on the Hardware Node.

Syntax

```
vzpkgdeploy [-q|-v] --install|--uninstall <package> [...]
vzpkgdeploy --help
```

Options

Name	Description
-i, --install <*package*> [. . .]	Installs the template on the Hardware Node.
-u, --uninstall <*package*> [. . .]	Removes the template from the Hardware Node.
-h, --help	Displays the usage information and exit.
--quiet	Disables logging to the display and to the log file.
--verbose	Sets the log level to the maximum possible value for this `vzpkgdeploy` session.

vzpkgadd

Adds application templates to a running Container.

Syntax

```
vzpkgadd [options] <CT_ID> <template>[/version] [...]
```

The <*version*> parameter specifies the template version to use if there are available upgrades. By default, the latest available version is used.

Options

Name	Description
-h, --help	Display the usage info and exit.
-f, --force	Force template installation.
--quiet	Disables logging to the display and to the log file.
--verbose	Sets the log level to the maximum possible value for this `vzpkgadd` session.

vzpkgrm

Uninstalls application templates from a running Container.

Syntax

```
vzpkgrm [options] <CT_ID> <template>[/version] [...]
```

Options

Name	Description
-d, --nodeps	Removes templates with unresolved dependencies.
-h, --help	Displays command usage information.
-f, --force	Forces uninstallation of templates.
-q, --quiet	Disables logging to the display and to the log file.
-v, --verbose	Sets the log level to the maximum possible value for this `vzpkgrm` session.

CHAPTER 8

Supplementary Utilities

This chapter concentrates on utilities for performing different tasks in the Hardware Node and Container context.

In This Chapter

vzkeygen 44
vzcache 45
vzlscache 46
vzuncache 47
vznetcfg 48
vznetstat 49
vzdevctl 50
vzcpucfg 52
vzquery 53
vzwinupdatecmd 53

vzkeygen

Generates a unique Hardware Node ID. To obtain a Parallels Virtuozzo Containers license, send your Node ID together with a license request to the Parallels sales department.

Syntax

```
vzkeygen [--company-name <name>] [--output-file <filename>]
vzkeygen --help
```

Options

Name	Description
-n, --company-name <*name*>	Your company name
-o, --output-file <*filename*>	Saves output to a file instead of stdout. If a filename is provided without a path, the output is saved to `C:\Documents and Settings\Administrator` by default.
-h, --help	Displays command usage information.

vzcache

There may be situations when one and the same application or application update is installed not as a template, but separately inside several Containers. A good example of this is the Adobe Acrobat Reader application that can be installed inside a number of Containers thus having a vast amount of identical files throughout the Containers. This fact tells dramatically on the Container quotas, which may be avoided by putting all the identical files to the Hardware Node template area and creating links instead of real files inside the affected Containers.

The problem like the one described above is solved by using the `vzcache` utility. The utility scans the specified Containers for common files and caches these files in the Hardware Node template area (in the *X*`:\vz\templates\__vzcache` folder), replacing the real files inside the Containers with links to the template area (*X*`:\vz\templates\__vzcache`). In case of a significant number of identical files, using `vzcache` results in a notable disk space gain.

Syntax

```
vzcache [options] <CT_ID> [...]
```

Options

Name	Description
-h, --help	Print usage information.
--version	Display the utility version.
-v, --verbose	Verbose mode. Causes `vzcache` to print debugging messages about its progress. Multiple `-v` options increase verbosity. The maximal number of allowed options is 2.
-q, --quiet	Quiet mode. Causes all warning and diagnostic messages to be suppressed. Only fatal errors are displayed.
-n, --cache-name <*name*>	The name of the folder in *X*`:\vz\templates\__vzcache` where you want to store the cached files. If you omit this option, `vzcache` creates a subfolder with a random name. If used with the `-a` option, tells `vzcache` to look for similar files in the specified cache folders.
-a, --append <*name*>	Look for similar files inside the specified Containers and compare them with the files in the indicated caches and templates. `-a` requires one of the following options: ▪ `-n`. In this case `vzcache` compares the files inside the specified Containers with those in the specified caches and replaces them with links, if necessary. ▪ `-T`. In this case `vzcache` compares the files inside the specified Containers with those in the specified templates and replaces them with links, if necessary.

	▪ `-H`. In this case `vzcache` compares the files inside the specified Containers with those in the system partition on the Node and replaces them with links, if necessary. ▪ `-A`. In this case `vzcache` compares the files inside the specified Containers with those in all available caches and templates. It does not, however, look for similar files in the system partition on the Node.
-s, --size-limit <*N*>	Do not process files smaller than `N` bytes. By default, only empty files are not processed.
-m, --min-links <*M*>	Cache files found at least `M` times. The default value is 2.
-T, --template-name <*name*>	Look for similar files in the specified OS and application templates. Must be used with the `-a` option. You can indicate several templates at a time and separate them by commas (e.g., `Template1`, `Template2`).
-H, --host	Look for similar files in the system partition on the Hardware Node. Must be used with `-a` and one of the following options: ▪ `-A` ▪ `-T` *`OS_Name`* (where *`OS_Name`* denotes the name of the OS template installed on the Node)
-A, --all	Look for similar files in all caches and templates and in the system partition on the Hardware Node. Must be used with the `-a` option.
-f, --force	Force the `vzcache` execution.
-t, --test	Do not perform any caching. Just collect and display the statistics on identical files.

Here are some examples on using the `vzcache` utility:

- To create a cache named `cache1` for Containers 153 and 154 caching only those files which are greater than 1024 bytes by using the `vzcache` utility, you should issue the following command:

```
vzcache -s 1024 -n cache1 153 154
```

- To append Container 155 to the existing `cache1` cache:

```
vzcache -a -n cache1 155
```

vzlscache

Displays a list of cache directories created by the `vzcache` utility. Shows either all the cache directories on the given Hardware Node, or those used by the specified Container(s).

Syntax

```
vzlscache [options] [CT_ID] [...]
```

Options

Name	Description
-h, --help	Print utility usage information.
-q, --quiet	Quiet mode. Causes all warning and diagnostic messages to be suppressed. Only fatal errors are displayed.

vzuncache

Whereas the `vzcache` utility helps effectively gain disk space both in the Hardware Node and within Containers, there may be situations when it is necessary to detach a Container from its cache and copy the cached files back to the Container private area. A typical example of this is migrating a Container to another Hardware Node. The migration is not possible if there are links in the Container private area pointing to the `C:\vz\templates\__vzcache` folder on the Hardware Node.

The `vzuncache` utility is used to copy the regular files from the specified cache directory on the Hardware Node (located in the `C:\vz\templates\__vzcache` folder) back to the private area of the specified Container, replacing the corresponding links or stubs inside the Container with the real files and thus detaching the Container from its cache.

Syntax

```
vzuncache [options] <CT_ID> [cache_dir] [...]
```

Options

Name	Description
-h, --help	Print usage information.
--version	Display the utility version.
-q, --quiet	Quiet mode. Causes all warning and diagnostic messages to be suppressed. Only fatal errors are displayed.
-v, --verbose	Verbose mode. Causes `vzuncache` to print debugging messages about its progress. Multiple `-v` options increase verbosity.
-a, --all	Detach the specified Container from all the caches. The list of cache directories is not needed if this option is specified.
-t, --test	Do not copy files from the cache. Just collect and display the relevant statistics.
-f, --force	Detach the specified Container from its cache even in case the cache does not exist.

vznetcfg

The `vznetcfg` utility is used to manage Virtual Networks and network classes on the Hardware Node.

Syntax

```
vznetcfg <command> [command_options]
vznetcfg --help
```

Commands

Name	Description
net new <*net_ID*> <*MAC*>[:VLAN_ID]	Creates a new Virtual Network with the ID of `net_ID` and associates it with either: ▪ a physical adapter on the Hardware Node with the MAC address of `MAC`, or ▪ a Virtual Local Area Network (VLAN) adapter bound to the physical adapter with the MAC address of `MAC`.
net change <*net_ID*> <*MAC*>[:VLAN_ID]	Associates the specified Virtual Network with the physical or VLAN adapter on the Hardware Node.
net del <*net_ID*>	Deletes the specified Virtual Network from the Hardware Node.
net list	Lists Virtual Networks existing on the Hardware Node.
class add <*class_ID*> <*IP_range1*> [IP_range2 ... IP_rangeN]	Creates a new network class with the ID of `class_ID` and the specified IP address range(s). An IP address range should be specified as an IP address, a slash, and a subnet mask or a CIDR suffix. For example, specify `10.0.0.0/255.0.0.0` or `10.0.0.0/8` to cover IP addresses from `10.0.0.0` to `10.255.255.255`.
class remove <*class_ID*> [IP_range1 ... IP_rangeN]	Removes the specified IP address range(s) from the specified network class. If no IP address ranges are specified, removes the entire network class. An IP address range should be specified as an IP address, a slash, and a subnet mask or a CIDR suffix. For example, specify `10.0.0.0/255.0.0.0` or `10.0.0.0/8` to cover IP addresses from `10.0.0.0` to `10.255.255.255`.
class rate <*class_ID*> <*rate*>	Sets the maximal network bandwidth, in Kilobits per second, any Container on the Hardware Node is guaranteed to have for outgoing traffic within the specified network class.
class list	Lists network classes existing on the Hardware Node.
shaper <on | off | status>	Enables, disables, or shows the status of the traffic shaper. The traffic shaper can be managed with `vzctl` and `vznetcfg class rate`

	commands.
privnet add <*privnet_ID*> <*IP_range1*> [IP_range2 ... IP_rangeN]	Adds a private Virtual Network with one or more IP ranges. An IP address range should be specified as an IP address, a slash, and a subnet mask or a CIDR suffix. For example, specify `10.0.0.0/255.0.0.0` or `10.0.0.0/8` to cover IP addresses from `10.0.0.0` to `10.255.255.255`.
privnet remove <*privnet_ID*> [IP_range1 ... IP_rangeN]	Removes the specified IP range(s) from a private Virtual Network. If no IP ranges are specified, removes the entire private Virtual Network. An IP address range should be specified as an IP address, a slash, and a subnet mask or a CIDR suffix. For example, specify `10.0.0.0/255.0.0.0` or `10.0.0.0/8` to cover IP addresses from `10.0.0.0` to `10.255.255.255`.
privnet list	Lists private Virtual Networks existing on the Hardware Node.
privnet weak <yes | no | show>	Controls private Virtual Network weakness: ▪ `yes` — Allows passing packets from private Virtual Networks to external hosts. ▪ `no` — Forbids passing packets from private Virtual Networks to external hosts. ▪ `show` — Shows the current weakness status.
privnet status [on | off]	Enables or disables private Virtual Networks on the Hardware Node. If no options are specified, shows the private Virtual Network status.
firewall status [on | off]	Enables or disables the firewall on the Hardware Node. If no options are specified, shows the firewall status.
firewall status-all-adapters [on | off]	Enables or disables for all adapters rules set with `vznetcfg firewall policy`. These rules add to existing policies for specific adapters set with `vzctl`. If no options are specified, shows the current firewall status.
firewall policy <default | private | public> [accept | drop accept | drop]	Sets a firewall policy of the specified type. The first `accept|drop` option is for incoming packets, the second is for outgoing packets. If no `accept|drop` options are provided, shows the specified firewall policy status.
--help	Displays command usage information.

vznetstat

Outputs traffic usage statistics for Containers.

Syntax

```
vznetstat [-v <CT_ID>] [-c <net_class_ID>] [-a] [-r <K|M|G>]
vznetstat --help
```

Options

Name	Description
-v <*CT_ID*>	Displays statistics for a Container with the ID of <*CT_ID*>.
-c <*class*>	Shows statistics for the <*class*> class only.
-a	Displays statistics for all classes.
-r <K|M|G>	Displays the network statistics, which is shown in bytes by default, in the following measurement units: ▪ K: display the network statistics in kilobytes; ▪ M: display the network statistics in megabytes; ▪ G: display the network statistics in gigabytes.
--help	Displays utility usage information.

When executed without any options, `vznetstat` displays network statistics, in bytes, for all defined network classes of all running Containers on the Node.

vzdevctl

Forwards hardware devices existing on the Hardware Node (SCSI, iSCSI, USB flash drives, etc.) to Containers.

Syntax

```
vzdevctl [--quiet|--verbose] add <CT_ID> --deviceid <name> [--alias <name>]
         [--exclusive] [--connect] [--onboot]
vzdevctl [--quiet|--verbose] remove <CT_ID> --deviceid <name>|--alias <name>
vzdevctl [--quiet|--verbose] connect <CT_ID> --deviceid <name>|--alias <name>
vzdevctl [--quiet|--verbose] dconnect <CT_ID> --deviceid <name>|--alias <name>
vzdevctl [--quiet|--verbose] set <CT_ID> [--deviceid <name>|--alias <name>
         [--newalias <name>]|--group <name>] [--onboot <yes|no>]
vzdevctl [--quiet|--verbose] devtree [--deviceid <name>|--all|--forwarded]
vzdevctl [--quiet|--verbose] status <CT_ID> [--deviceid <name>|--alias <name>|--all]
vzdevctl [--quiet|--verbose] linkadd <CT_ID> [--symlink <name>] --object <name>
vzdevctl [--quiet|--verbose] linkdel <CT_ID> --symlink <name>|--object <name>
vzdevctl [--quiet|--verbose] linkquery <CT_ID>
vzdevctl --version
vzdevctl --help
```

Commands

Name	Description
add	Forwards the hardware device to the specified Container.
remove	Removes the hardware device from the specified Container. If the device is in the active state, it is first disconnected from the Container.
connect	Connects the hardware device to the Container. Before connecting the device to the Container, you must first forward it to this Container.
dconnect	Disconnects the hardware device from the Container.
set	Sets device info. (Forwarded/shared state can be changed after disconnect-connect cycle only)
devtree	Shows all hardware devices available on the Hardware Node.
status	Shows the hardware device status for the specified Container.
linkadd	Adds a symbolic link to the Hardware Node object.
linkdel	Deletes a symbolic link from the specified Container.
linkquery	Displays the symbolic links for the specified Container.

Options

Name	Description
--deviceid <*ID*>	The ID to be assigned to the hardware device. It should be set in the following form: *Enumerator**Hardware_ID**Instance_ID* where: ▪ *Enumerator* denotes a system component responsible for discovering hardware devices on your Hardware Node (usually, the Plug-an-Play manager). ▪ *Hardware_ID* is a vendor-defined identification string used by the Setup program to match a hardware device on your Hardware Node to an INF file. ▪ *Instance_ID* is a hardware device identification string that distinguishes the device from other devices of the same type on your Hardware Node. Detailed information on device identification strings is provided at http://msdn.microsoft.com/en-US/library/ff541224.aspx.
--alias <*alias*>	A short name (alias) to be assigned to the hardware device.
--exclusive	Sets the hardware device forwarding mode to 'exclusive' instead of 'shared' which is set by default. In this case the device cannot be forwarded to any other Container on the Hardware Node. This command must be used for any SCSI or USB-flash device which you are going to forward to a Container.
--connect	Specifies whether the hardware device being forwarded to the Container is also to be connected to the Container.
--onboot <yes|no>	Sets whether the hardware device is to be automatically connected to the Container on its startup (yes) or should be manually joined after the Container is up and running (no).
--newalias <name>	Sets new alias for device.

--class	Sets class name filter, may contains a wildcard.
--fname	Sets friendly name filter, may contains a wildcard.
--wizard	Turns on interactive mode which allows to change state of every listed device. For devtree using this mode without any filtering options is strongly unrecomended (too many entries to skip).
--force	Forces the device disconnection for busy devices. This option is strongly unrecommended.
--forwarded	Displays only those hardware devices on the Hardware Node that are currently forwarded to some Containers.
--all	Can be indicated in one of the following cases: ▪ when used with the `devtree` command, displays all SCSI devices available on the Hardware Node (connected, not connected, denied); ▪ when used with the `status` command, displays all Container devices including the ones that are currently not connected to the Container.
--symlink	The full name to be assigned to the symbolic link.
--object	The full name of the object for which you are creating a symlink. If used without the `--symlink` option, the name specified after this option is also used to denote the Container symbolic link.
--version	Shows the Parallels Virtuozzo Containers version.
--quiet	Disables logging to the log file and screen. Can be used with any commands.
--verbose	Sets the logging level to the maximum value. Can be used with any commands.
--help, /?	Gets the utility usage information and exits.

vzcpucfg

Manages CPU pools on the Hardware Node. You can do the following:

- create a new CPU pool,
- remove an existing CPU pool,
- list the CPU pools currently existing on the physical server,
- configure the number of CPUs in a CPU pool,
- assign CPU pools to Containers.

Syntax

```
vzcpucfg pool set <pool_ID> <CPU_range>
vzcpucfg pool del <pool_ID>
vzcpucfg pool list
```

Options

Name	Description
set <*pool_ID*> <*CPU_range*>	Creates a new CPU pool or configures the number of CPUs in an existing CPU pool.
del <*pool_ID*>	Removes a CPU pools from the physical server.
list	Lists the CPU pools existing on the physical server.
<*pool_ID*>	The ID to be assigned to the CPU pool or the ID of the CPU pool to be removed from the physical server.
<*CPU_range*>	The CPU range to be included in the CPU pool. A range may contain one or several CPUs (for example, `1` or `1-3`). You can also specify multiple comma-separated ranges at once (e.g., `0-2,3-6,8`).

vzquery

Determines the Container ID using either a process ID or a session ID.

Syntax

```
vzquery s2v <session_ID>
vzquery v2s <CT_ID>
vzquery p2v <process_ID>
```

Options

Name	Description
s2v <*session_ID*>	Displays the ID of the Container to which the Terminal Services session with the specified ID is opened.
v2s <*CT_ID*>	Lists the IDs of all Terminal Services sessions opened to the specified Container.
p2v <*process_ID*>	Displays the ID of the Container where the process with the specified ID is running.

vzwinupdatecmd

Lists Windows Server updates installed in Containers.

Syntax

```
vzwinupdatecmd /listctupd <CT_ID> [...] [/all]
```

```
vzwinupdatecmd /help
```

Options

Name	Description
/listctupd <*CT_ID*> [. . .]	Lists Windows updates currently installed inside the Container(s).
/all	Applies the operation to all Containers on the Hardware Node.
/help, /?	Displays command usage information.

CHAPTER 9

Glossary

Application template is a template used to install a set of applications in *Containers*. See also *Template*.

Container is a virtual private server, which is functionally identical to an isolated standalone server, with its own IP addresses, processes, files, user database, configuration files, applications, system libraries, and so on. While sharing the same *Hardware Node* and OS kernel, *Containers* are isolated from each other. A *Container* is a kind of 'sandbox' for processes and users.

Hardware Node (or Node) is the server where *Parallels Virtuozzo Containers* is installed.

Host Operating System (or Host OS) is an operating system installed on the *Hardware Node*.

OS template (or Operating System template) is used to create new *Containers* with a preinstalled operating system. See also *Template*.

Parallels Virtuozzo Containers is a complete server automation and virtualization solution that allows you to create multiple isolated *Containers* on a single physical server to share hardware, licenses, and management effort with maximum efficiency.

Parallels Virtuozzo Containers license is a special license which you must install on a *Hardware Node* to be able to use *Parallels Virtuozzo Containers*. Every *Hardware Node* must have a license installed.

Virtuozzo File System (VZFS) is a virtual file system for mounting to Container private areas. VZFS symlinks are seen as real files inside *Containers*.

Parallels Management Console is a *Parallels Virtuozzo Containers* management and monitoring tool with a graphical user interface. It is used to control individual *Hardware Nodes* and their *Containers*. The console is cross-platform and runs on both Microsoft Windows and Linux workstations.

Parallels Power Panel is a tools for managing personal *Containers* via a standard Web browser.

Parallels Virtual Automation is a tool for managing *Hardware Nodes* and *Containers* residing on them via a standard Web browser.

Private area is a location where *Container* files which are not shared with other *Containers* are stored.

Template is a set of original application files (packages) repackaged for using inside *Containers*. There are two types of templates: *OS Templates* are used to create new *Containers* with a preinstalled operating system, *application templates* are used to install applications or sets of applications in *Containers*.

Parallels Agent (or Parallels Agent Protocol) is an XML-based protocol used to monitor and manage *Hardware Nodes*. It is a backend for the *Parallels Management Console*.

Index

A

About Parallels Virtuozzo Containers 4.6 - 5
About This Guide - 6
Administrator
  Container - 22
Applications - 45

B

Backup
  full - 36
  incremental - 36
Backup Utilities - 36

C

Container
  destroying - 14
  listing - 24
  migrating - 32
  mounting - 15
  private area - 14, 15, 32
  restoring - 22, 38
  starting/stopping - 15

D

Documentation Conventions - 7

F

Feedback - 9

G

General Utilities - 12
Getting Help - 8
Glossary - 55

H

Host OS - 55
Hostname
  Hardware Node - 32

I

Internet Explorer - 30, 55
IP Address
  Hardware Node - 36

L

Licensing Utilities - 30

M

Migration
  Container to Container - 32
  physical server to Container - 33
Migration Utilities - 32
Mozilla - 55

N

Node
  Destination - 32
  Source - 32, 36

O

Offline Management - 34
Organization of This Guide - 7

P

Parallels Agent - 36
Parallels Power Panel - 55
Parallels Virtual Automation - 55
Parallels Virtuozzo Containers License - 55
  checking - 31
  installing - 30
  overview - 55
Parallels Virtuozzo Containers Utilities
  Overview - 10
Preface - 5

R

Registry - 15
Resources
  disk space - 28

S

Service Container - 55
Supplementary Utilities - 44

T

Template
application - 41, 42, 43
area - 45
installing - 42
listing - 41
removing - 43
Template Management Utilities - 41

U

Utilities
backup management - 36, 38, 40
Container management - 14, 15, 21, 22, 23, 24
license management - 30, 31
migration management - 32, 33, 34
resources management - 28, 29
supplementary - 44, 45, 46, 47, 48, 49
template management - 41, 42, 43

V

vzabackup - 36
vzarestore - 38
vzcache - 45
vzcpucfg - 52
vzctl - 12
vzctl addrole, vzctl delrole, vzctl enumrole - 24
vzctl create - 13
vzctl defrag - 23
vzctl delete, vzctl destroy - 14
vzctl exec, vzctl exec2, and vzctl enter - 21
vzctl mount, vzctl umount - 15
vzctl mountext, vzctl umountext - 21
vzctl partadd, vzctl partdel - 22
vzctl reinstall - 22
vzctl set - 15
vzctl shrink - 23
vzctl start, vzctl stop, vzctl restart, and vzctl status - 15
vzdevctl - 50
vzkeygen - 44
vzlicload - 30
vzlicview - 31
vzlist - 24
vzlist Output Parameters and Their Modifiers - 25
vzlscache - 46
vzmigrate - 32
vzmlocal - 34
vznetcfg - 48
vznetstat - 49
vzp2v - 33
vzpkgadd - 42
vzpkgdeploy - 42
vzpkgls - 41
vzpkgrm - 43
vzquery - 53
vzquota - 28
vzquota setlimit - 28
vzquota stat, vzquota show - 29
vzuncache - 47
vzvssctl - 40
vzwinupdatecmd - 53

X

XML - 55