NAME

vpsctl — manage Virtual Private Systems instances

SYNOPSIS

```
vpsctl start id [config-file]
vpsctl stop id [config-file]
vpsctl list
vpsctl show id
vpsctl shell id
vpsctl console id
vpsctl ifmove id ifname [ifnewname]
vpsctl execin id command [arguments ...]
vpsctl execwt id command [arguments ...]
vpsctl suspend id
vpsctl resume id
vpsctl snapshot id output-file
vpsctl abort id
vpsctl restore id input-file
vpsctl migrate id remote-host [norsync|onersync]
vpsctl argshow id ipnet id add adress/network, ...
vpsctl ipnet id rem adress/network, ...
vpsctl priv id allow privilege-number, ...
vpsctl priv id deny privilege-number, ...
vpsctl priv id nosys privilege-number, ...
vpsctl limit id resource:softlimit:hardlimit, ...
vpsctl showdump file
```

DESCRIPTION

The **vpsctl** utility is used to manage Virtual Private System (VPS) instances, e.g. starting, stopping, migrating and reading status information.

Basically VPS works in a hierarchical way, so you can use the **vpsctl** utility on the main host system (which in turn is a VPS instance too) and inside child VPS instances as well. You can only see and manage instances that are children of the current instance.

For a general description of what VPS is, see vps(4).

The following commands are available:

start *id* [*config-file*]

Creates and runs a new VPS instance named *id* using settings from *config-file*.

If no *config-file* is given, **vpsctl** tries to read settings from /etc/vps/vps_<ID>.conf.

See the **FILES** sections for the config file syntax.

stop *id* [*config-file*]

Stops and destroys VPS instance *id*.

If no *config-file* is given, **vpsctl** tries to read settings from /etc/vps/vps_<ID>.conf.

If you perform ’shutdown -h ...’ from inside a VPS instance, you still have to run **stop** *id* to free it entirely.

list Prints a listing of all currently existing VPS instances. Status is one of running, suspended or dead. After stopping an instance, it may reside for a while as dead instance in the system. A typical cause is TCP sockets in TIME_WAIT state. After all references are gone, it will be automatically deleted.

show *id*
: Shows some details about the VPS instance given by *id*. For a more detailed listing including all run-time properties about a VPS instance see **argshow** *id*.

shell *id*
: Allocates a pseudo tty and starts an interactive shell in the given VPS instance. This command is equally to 'vpsctl execwt /bin/sh'.

console *id*
: Opens the system console of the given VPS instance.

ifmove *id ifname* [*ifnewname*]
: Moves network interface *ifname* into vps instance *id* and optionally renames it to [*ifnewname*].

execin *id command* [*arguments ...*]
: Starts the given *command* using *arguments* if given in the context of VPS instance *id*. The **vpsctl** utility does not wait on exit of *command* but exits immediately.

execwt *id command* [*arguments ...*]
: Same syntax as for the **execin** command, but a pseudo tty is allocated for *command* in order to have interactive input/output and the **vpsctl** utility waits until exit of *command*.

suspend *id*
: Suspends all processes in VPS instance *id*, all TCP sockets are set to drop incoming data, every other activity is suspended. May be used for creating consistent backups of one VPS instances' filesystem space.

resume *id*
: Resumes previously suspended vps instance *id*.

snapshot *id output-file*
: Writes a snapshot of *id* to *output-file*, including the state of all processes, sockets, network attributes etc. The VPS instance *id* has to be suspended by **suspend** first.

abort *id*
: Ungracefully kill all processes in VPS instance *id*. You still have to run **stop** *id* to get rid of it. If *id* is currently suspended, run **resume** *id* afterwards in order to have effect.

restore *id input-file*
: Restores a VPS instance from a snapshot file previously created by **snapshot** to a suspended state. Use **resume** *id* to get it running again.

The snapshot file can originate from a different host, but care has to be taken to have compatible or better, the same kernel, on both systems. Before the actual restore process starts, sanity and compatibility checks against the snapshot file are performed, but in case of mismatches a kernel crash could still happen.

Instead of *id* an empty string ('') can be given. The *id* the instance had when its snapshot was created is used then.

The right instance config file has to exist.

migrate *id remote-host* [*norsync*|*onersync*]
: Performs a live migration of VPS instance *id* to host *remote-host*. All processes, open files, pipes, sockets, TCP session etc. are preserved.

Specify *norsync* if you don't want the filesystem tree to be synced. Do not use this option unless you are sure nothing was modified or the filesystem is mounted readonly. Specify *onersync* if you only need one sync pass (faster than two-pass but vps is suspended longer).

If the migration process fails or is aborted, the instance is left in suspended state. Use **resume** *id* to get it running again. You might have to set some network settings in the parent system manually, like published arp entries and routes.

SSH is used as transport to *remote-host*. If no ssh key is loaded/available, you will be prompted for a password.

argshow *id*

Shows arguments of VPS instance *id*.

The output show IPv4 and IPv6 networks the instance is allowed to use on its interfaces, as well as a list of privileges `priv`(9.) For *NOSYS* privileges, a 'No such system call' error is returned instead of 'Permission denied'. Some applications wouldn't run otherwise.

Resource limits are shown as: resource type, current utilization, preconfigured soft limit (may be exceeded if resources are still available), hard limit (will never be exceeded) and counters how often the soft or hard limits have been hit.

ipnet *id* **add** *address/network, ...*

Adds an IP address or network to the list of networks the VPS instance is allowed to use on its interfaces.

Following formats are valid:

```
192.168.123.231
192.168.213.0/255.255.255.0
fc00::200:20
fc00::100:0/112
```

Multiple networks can be specified on the command line at once, separated by ',' but without whitespace.

To specify networks in a configuration file, use *IP_NETWORKS.*

ipnet *id* **rem** *address/network, ...*

Removes a previously added IP address or network from the list. See the description of **ipnet** *id* **add** for more information.

priv *id* **allow** *privilege-number, ...*

Adds one or more privileges, specified by name (e.g. PRIV_KLD_LOAD) or its numeric value, to the list of allowed privileges. Separate multiple privileges by ',' without whitespace.

See `priv`(9) for more information about privileges, and *sys/priv.h* for a list of defined privileges.

To specify privileges in a configuration file, use *PRIV_ALLOW.*

priv *id* **deny** *privilege-number, ...*

Removes one or more privileges, specified by name (e.g. PRIV_KLD_LOAD) or its numeric value, from the list of allowed or 'nosys' privileges.

System calls or other operations that depend on the privilege in question, return to userspace with a 'Permission denied' error.

priv *id* **nosys privilege-number, ...**

Adds one ore more privileges, specified by name (e.g. PRIV_KLD_LOAD) or its numeric value, to the list of privileges, that are not allowed, but supposed to return ''No such system call'' instead of

”Permission denied”, to satisfy some applications.

To specify privileges in a configuration file, use *PRIV_NOSYS.*

limit *id resource:softlimit:hardlimit, ...*

Configures a limit on one or more given resources (like virtual memory,
cpu utilization, ...).

The command **argshow** *id* shows you which resources are known.

The *softlimit* is a treshold that will be exceeded in case the current utilization of the given resource allows to do so without affecting other VPS instances’ performance.

The *hardlimit* will never be exceeded.

For CPU utilization ’sysctl kern.fscale’ equals 100% of one cpu (To allow a VPS instance to use 25% of one CPU set the limit to (sysctl kern.fscale) * 0.25).

To specify resource limits in a configuration file, use *LIMITS*.

showdump *file*

Shows information about the snapshot contained in *file*. It generates a lot of output.

EXIT STATUS

The **vpsctl** utility exits 0 on success and -1 if an error occurs.

FILES

`/etc/vps/vps_<ID>.conf` for each VPS instance with id <ID>.

EXAMPLES

Example of a rc script (e.g. put these lines in /etc/rc.local):

```
kldload vps_ddb
kldload vps_dev
kldload if_vps
kldload vpsfs
kldload vps_account
kldload vps_suspend
kldload vps_libdump
kldload vps_snapst
kldload vps_restore

ifconfig vps0 create
ifconfig vps0 up

sysctl -w net.inet.ip.forwarding=1
sysctl -w net.inet6.ip6.forwarding=1
```

Example of a minimal VPS instance config file (/etc/vps/vps_testvps.conf):

```
NAME = testvps
FSROOT = /vps/testvps
NETIF_0_ADDRESS = ’1.2.3.4, 2001:2002::2003’
ROOT_MOUNT = ’true’
ROOT_UMOUNT = ’true’
INIT = ’/sbin/init’
```

See `vps.conf`(5) for a complete description of config file options.

The /vps/testvps directory contains a full FreeBSD userland installation. See `jail`(8) for examples how to install into a directory.

The only crucial configuration bits are setting all terminals to *off* in */etc/ttys*, as well as putting *root_rw_mount=NO* into */etc/rc.conf*.

Don't use nullfs, there are some issues with snapshot/restore.

The IP addresses given are assumed to belong to a subnet on a physically attached ethernet network. The **vpsctl** utility creates a published ARP entry and a local route for each address.

Start the vps instance:

```
vpsctl start testvps
vpsctl list
```

SEE ALSO

`vps`(4), `vps`(9), `vps.conf`(5), `mount_vpsfs`(8), *http://www.7he.at/freebsd/vps/*

HISTORY

Work on VPS was started in February 2009.

AUTHORS

Virtual Private Systems for FreeBSD and this manual page as well, were written by Klaus P. Ohrhallinger.

Development of this software was partly funded by:

TransIP.nl <http://www.transip.nl/>

BUGS

VPS is in an early stage of development and has to be considered as experimental. This means many bugs have to be expected.

Please submit bug reports to *freebsd-vps@7he.at*.

VERSION

$Id: vpsctl.8 130 2013-04-10 09:32:39Z klaus $