

OpenVZ is nice name-space virtualization, creating chroot jails on steroids, similar in spirit to Solaris zones. It ideal if you want to run single kernel and allocate resources using bean counters as opposed to hard-limits (20% of CPU as opposed to one core). Each slice is called VE.

fa0x09565d48 [0, 0]

Contents: [Dobrica PavlinuÅjiÄ 's random unstructured stuff]

- Dobrica PavlinuÅjiÄ 's random unstructured stuff (Disk speed)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (Add disk space to VE)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (VE management)
 - ◆ Dobrica PavlinuÅjiÄ 's random unstructured stuff (Update Debian)
 - ◆ Dobrica PavlinuÅjiÄ 's random unstructured stuff (Quick reporting)
 - ◆ Dobrica PavlinuÅjiÄ 's random unstructured stuff (find getty processes)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (devices inside VE)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (Links)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (vz-tools)
- Dobrica PavlinuÅjiÄ 's random unstructured stuff (Related posts on my blog)

Disk speed

```
dpavlin@zut:~$ sudo hdparm -tT /dev/cciss/c1d0 /dev/sda
```

```
/dev/cciss/c1d0:
```

```
Timing cached reads:   2184 MB in  2.00 seconds = 1092.39 MB/sec  
Timing buffered disk reads:  324 MB in  3.02 seconds = 107.40 MB/sec
```

```
/dev/sda:
```

```
Timing cached reads:   2144 MB in  2.00 seconds = 1071.89 MB/sec  
Timing buffered disk reads:  136 MB in  3.02 seconds =  45.02 MB/sec
```

Insert joke about enterprise storage

Add disk space to VE

We are using normal Linux LVM with single logical volume for all VEs.

First, resize logical volume:

```
root@koha-hw:~# vgextend -L +80G /dev/vg/vz  
vgextend: invalid option -- L  
Error during parsing of command line.
```

```
root@koha-hw:~# lvextend -L +80G /dev/vg/vz  
Extending logical volume vz to 100.00 GB  
Logical volume vz successfully resized
```

```
root@koha-hw:~# resize2fs /dev/vg/vz  
resize2fs 1.40-WIP (14-Nov-2006)
```

Filesystem at /dev/vg/vz is mounted on /vz; on-line resizing required
old desc_blocks = 2, new_desc_blocks = 7
Performing an on-line resize of /dev/vg/vz to 26214400 (4k) blocks.
The filesystem on /dev/vg/vz is now 26214400 blocks long.

```
root@koha-hw:~# df -h /vz/
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/vg-vz  99G   20G   79G   21% /vz
```

Then, take a look how much space does VEs take:

```
root@koha-hw:~# vzlist -o veid,diskspace,diskspace.s,diskspace.h,disknodes,disknodes.s,diskspace
  VEID  DQBLOCKS DQBLOCKS.S DQBLOCKS.H  DQINODES DQINODES.S DQBLOCKS.H
  212052  11717220  15728640  20971520    61001   286527  20971520
  212226   6407804  10485760  12582912    69011   435472  12582912
```

alternatively, you can also execute df inside VEs:

```
root@koha-hw:~# vzlist -o veid -H | xargs -i sh -c "echo --{}-- ; vzctl exec {} df -h"
--212052--
Filesystem      Size  Used Avail Use% Mounted on
simfs           15G   12G   3.9G   75% /
tmpfs           2.0G    0 2.0G    0% /lib/init/rw
tmpfs           2.0G    0 2.0G    0% /dev/shm
--212226--
Filesystem      Size  Used Avail Use% Mounted on
simfs           10G   6.2G   3.9G   62% /
tmpfs           2.0G    0 2.0G    0% /lib/init/rw
tmpfs           2.0G    0 2.0G    0% /dev/shm
```

next, we will set diskpace on both VEs (because we want them to share all available resources) to new logical volume size:

```
root@koha-hw:~# vzlist -o veid -H | xargs -i vzctl set {} --diskspace 100G:100G --save
Saved parameters for VE 212052
Saved parameters for VE 212226
```

This VEs are not in production, and one is development version of another. When we move to production, we want to enforce more strict limit on disk usage, to protect production machine from running out of disk space in case the development one goes wild.

VE management

We usually want to do some operations on bunch of VEs at once. This can be done using `vzctl exec` in one sweep like this:

Update Debian

```
vzlist -H -o veid | xargs -i vzctl exec {} 'apt-get update && apt-get -y upgrade' 2>&1 | tee ~/lo
```

Quick reporting

You can read more about [groupby.pl](#) and [sum.pl](#) on my blog.

```
# install dependencies which are not part of standard lenny (sorry!)
cpanp i IPC::System::Simple

dpavlin@mjesecc:~$ vzps -E axv --no-headers \
  | groupby.pl 'sum:($7+$8+$9*1024),1,count:1' --join 'sudo vzlist -H -o veid,hostname' --on 2 \
  | sort -rn | align | sum.pl -h
webgui.rot13.org 23      1026M 0000000000000      1026M
0                385    855M 0000000000----- 1882M
saturn.ffzg.hr  32      544M 000000-----      2427M
eprints.ffzg.hr 18      351M 0000-----      2778M
arh.rot13.org   20      224M 00-----      3003M
```

find getty processes

```
root@mljac:~# ps ax | grep getty | cut -c-5 | xargs vzpid
Pid      VEID      Name
5668     0         getty
5670     0         getty
5672     0         getty
5673     0         getty
5674     0         getty
5675     0         getty
9503     207016    getty
9504     207013    getty
9505     207013    getty
9534     207016    getty
9535     207015    getty
9536     207013    getty
9537     207013    getty
9538     207015    getty
9539     207015    getty
9540     207015    getty
9541     207016    getty
9542     207015    getty
9543     207016    getty
9545     207013    getty
9546     207013    getty
9547     207015    getty
9548     207016    getty
```

devices inside VE

For example, [fuse](#)

```
dpavlin@brr:/dev$ vzctl set 100 --devices c:10:229:rw --save
```

Links

- [Recent Advances in the Linux Kernel resource management](#)

vz-tools

vz-tools

Suite of perl scripts in spirit of [xen-tools](#) but for [OpenVZ](#)

svn co svn://svn.rot13.org/vz-tools/trunk vz-tools

Contents: [[Dobrica Pavlinu's random unstructured stuff](#)]

- [Dobrica Pavlinu's random unstructured stuff \(Installation\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(Install perl dependencies from Debian packages\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(Install utilities from Debian packages\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(Checkout source\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(Check and install perl modules from CPAN\)](#)
- [Dobrica Pavlinu's random unstructured stuff \(Usage\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(vz-create.pl\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(vz-optimize.pl\)](#)
 - ◆ [Dobrica Pavlinu's random unstructured stuff \(vz-clone.pl\)](#)
- [Dobrica Pavlinu's random unstructured stuff \(Source\)](#)

Installation

Install perl dependencies from Debian packages

This step is optional. If you don't want to use perl modules from packages provided by your distribution, skip this step, and modules will be automatically installed in next one.

```
sudo apt-get install libio-prompt-perl libregexp-common-perl libdata-dump-perl
```

Install utilities from Debian packages

```
sudo apt-get install host
```

Checkout source

```
svn co svn://svn.rot13.org/vz-tools/trunk vz-tools
```

Check and install perl modules from CPAN

```
cd vz-tools
perl Makefile.PL
make
```

Please note that there is no need to run `make install`

Tools are runnable from current directory. This will probably change in later versions.

Usage

This is quick hand-on overview of commands to get you started.

All commands must be started with `root` priviledges

vz-create.pl

This will perform following steps:

- Create new virtual machine bootstrapped using `debootstrap`
- Change root password
- Create single user
- Make small customization like installing `vim` and [apt-iselect](#)

All commands will be echoed on screen, **even passwords**. However, if you want to learn steps in creating OpenVZ VE, this is very helpful.

To run interactive session which asks questions use:

```
./vz-create.pl
```

Other alternative is to just enter hostname (defined in `/etc/hosts` for example)

```
./vz-create.pl my-new-ve.example.com
```

or by specifying IP adress

```
./vz-create.pl 192.168.42.42
```

vz-optimize.pl

vz-clone.pl

```
root@black:~/vz-tools# time ./vz-clone.pl create 1001
Clone VE 1001 -> 101001
found LV /dev/vg/vz for /vz
vzquota : (warning) Quota is running, so data reported from quota file may not reflect current va
quota for 1001 | 10485760 < 20971520 | usage: 7826792
using existing /dev/vg/vz-clone-101001
Mounting /dev/vg/vz-clone-101001 to /tmp/vz-clone-101001
rsync /vz/private/1001 -> /tmp/vz-clone-101001/private
101001 new IP number: 10.42.42.42
101001 new hostname: clone-42.example.com
```

```
Please review config file: /etc/vz/conf/101001.conf
Add NAT for new VE with: iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
Start clone of 1001 with: vzctl start 101001
```

```
real    1m57.347s
user    0m2.252s
sys     0m8.591s
```

Source

fetchrss: <http://svn.rot13.org/index.cgi/vz-tools/rss/trunk>

- There was an error: 404 Not Found

Related posts on my blog

fetchrss: <http://blog.rot13.org/mt/mt-search.cgi?search=openvz&Template=feed&IncludeBlogs=1>

- There was an error: 500 read failed: error:14094410:SSL routines:SSL3_READ_BYTES:sslv3 alert handshake failure | error:140940E5:SSL routines:SSL3_READ_BYTES:ssl handshake failure