

Linux Virtualization

*Kir Kolyskin <kir@openvz.org>
OpenVZ project manager*

What is virtualization?

Virtualization is a technique for deploying technologies. Virtualization creates a level of indirection or an abstraction layer between a physical object and the managing or using application.

<http://www.aarohi.net/info/glossary.html>

Virtualization is a framework or methodology of dividing the resources of a computer into multiple execution environments...

<http://www.kernelthread.com/publications/virtualization/>

A key benefit of the virtualization is the ability to run multiple operating systems on a single physical server and share the underlying hardware resources – known as **partitioning**.

<http://www.vmware.com/pdf/virtualization.pdf>

Ways to Virtualize

- Hardware Emulation
- Para-Virtualization
- Virtualization on the OS level
- Multi-server virtualization

Hardware Emulation

a.k.a. VM (Virtual Machine)

– VMware



– QEmu



– Bochs



Pros:

- Can run arbitrary OS, unmodified

Cons:

- Low density/scalability
- Slow/complex management
- Low performance

Para-virtualization

- Xen
- UML
(User Mode Linux)



Multiple (modified) OSs run under a hypervisor (a.k.a. Virtual Machine Monitor), which shares the hardware resources between guests.

Pros:

- Better performance

Cons:

- Needs modified guest OS
- Static resource allocation, bad scalability, bad manageability

OS Level Virtualization

(OS == kernel)

- OpenVZ
- FreeBSD jails
- Linux-VServer
- Solaris Zones



Most applications running on a server can easily share a machine with others, if they could be isolated and secured. OS Virtualization provides the required isolation and security to run multiple applications or copies of the same OS on the same server.

Pros:

- Native performance
- Dynamic resource allocation, best scalability

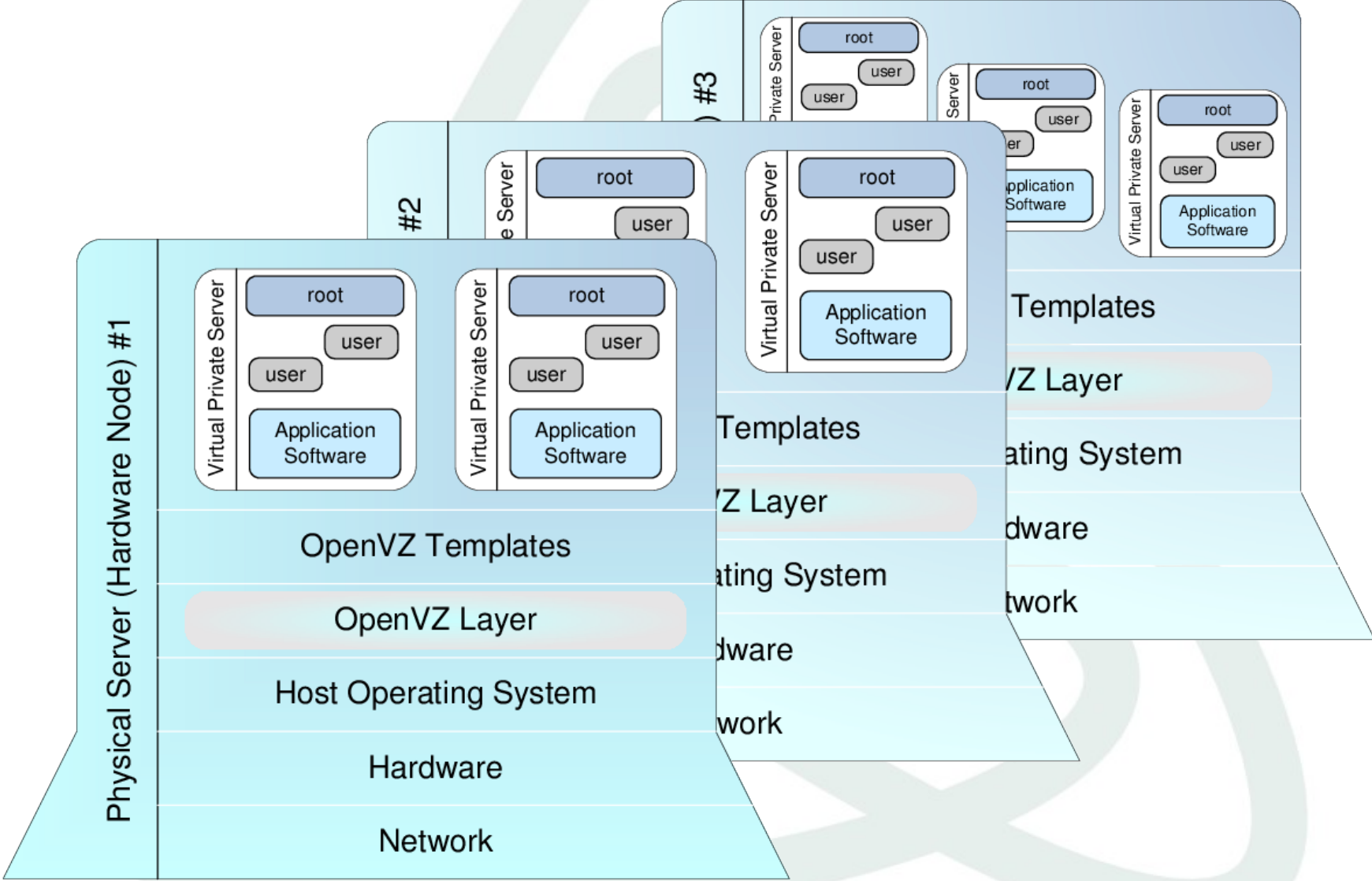
Cons:

- Single (same) kernel per physical server

OSs evolution

- **Multitask**
many processes
- **Multiuser**
many users
- **Multiple execution environments**
many Virtual Private Servers
(VPSs, containers, guests, partitions...)

OpenVZ design approach



OpenVZ: components

Kernel

- Isolation
- Virtualization
- Resource Management

Tools

- vzctl: Virtual Private Server (VPS) control utility
- vzpkg: VPS software package management

Templates

- precreated VPS images for fast VPS creation

Kernel: Virtualization & Isolation

Each VPS has its own

- **Files**
System libraries, applications, virtualized /proc and /sys, virtualized locks etc.
- **Process tree**
Featuring virtualized PIDs, so that the init PID is 1
- **Network**
Virtual network device, its own IP addresses, set of netfilter and routing rules
- **Devices**
If needed, any VPS can be granted access to real devices like network interfaces, serial ports, disk partitions, etc.
- **IPC objects**
shared memory, semaphores, messages
- ...

Kernel: Resource Management

Managed resource sharing and limiting.

- **User Beancounters** is a set of per-VPS resource counters, limits, and guarantees (kernel memory, network buffers, phys pages, etc.)
- **Fair CPU scheduler** (SFQ with shares and hard limits)
- **Two-level disk quota** (first-level: per-VPS quota; second-level: ordinary user/group quota inside a VPS)

Resource management is what makes OpenVZ different from other technologies.

Tools: VPS control

```
# vzctl create 101 --ostemplate fedora-core-4
# vzctl set 101 --ipadd 192.168.4.45 --save
# vzctl start 101
# vzctl exec 101 ps ax
  PID TTY          STAT       TIME COMMAND
    1 ?            Ss          0:00 init
 11830 ?            Ss          0:00 syslogd -m 0
 11897 ?            Ss          0:00 /usr/sbin/sshd
 11943 ?            Ss          0:00 xinetd -stayalive -pidfile ...
 12218 ?            Ss          0:00 sendmail: accepting connections
 12265 ?            Ss          0:00 sendmail: Queue runner@01:00:00
 13362 ?            Ss          0:00 /usr/sbin/httpd
 13363 ?            S           0:00 \_ /usr/sbin/httpd
.....
 13373 ?            S           0:00 \_ /usr/sbin/httpd
 6416 ?            Rs          0:00 ps axf
# vzctl enter 101
bash# logout
# vzctl stop 101
# vzctl destroy 101
```

Tools: Templates

vzpkgls

fedora-core-4-i386-default
centos-x86_64-minimal

vzpkgcache

(creates templates from metadata/updates existing templates)

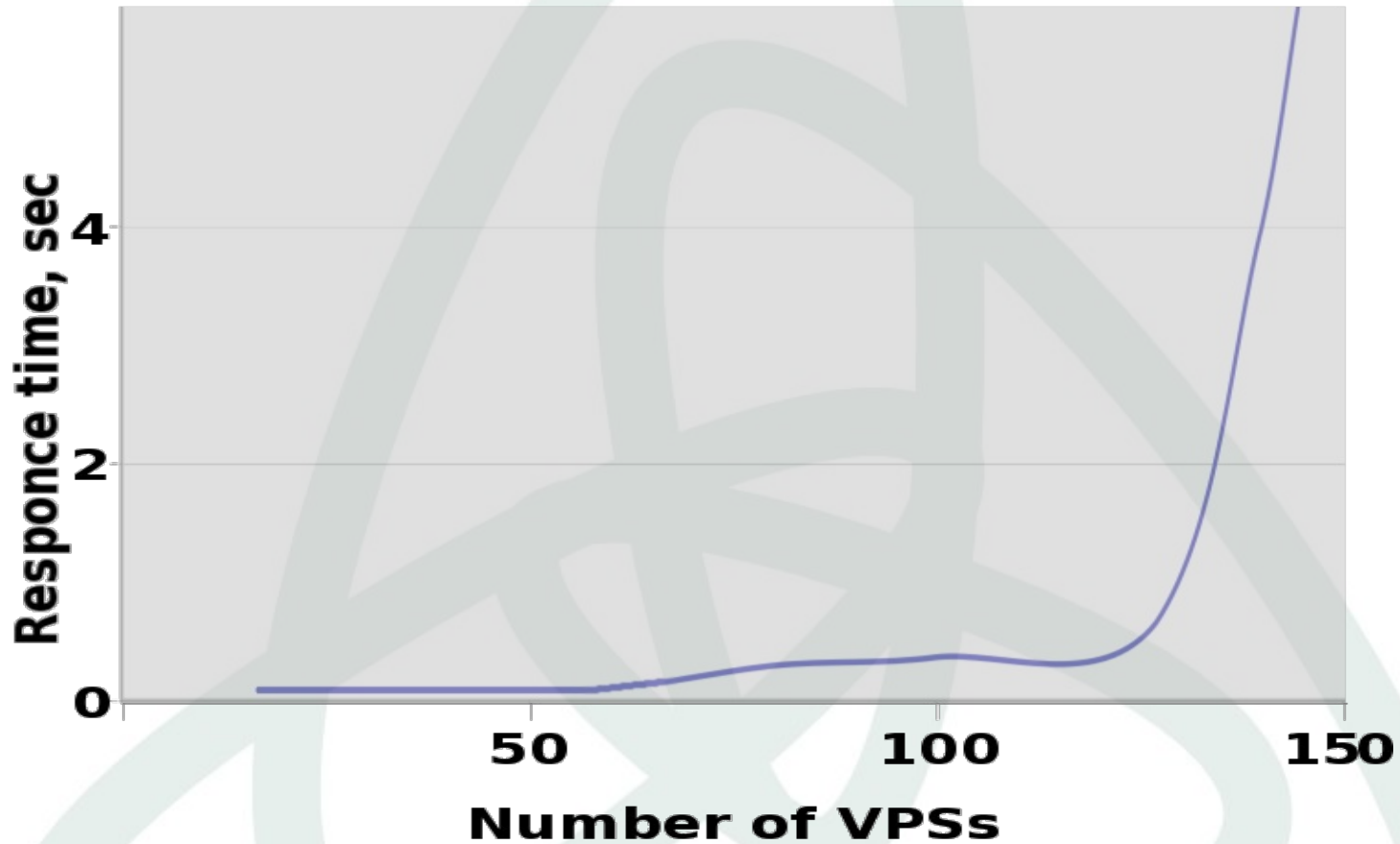
vzyum 101 install gcc

(installs gcc and its deps to VPS 101)

Live Migration

- A VPS can be migrated between physical servers
- No need to shutdown
- Network connections are preserved
- Users will not notice the migration
- No special hardware requirements: works with non-shared storage, normal NICs

Scalability



768 ($\frac{3}{4}$) MB RAM - up to 120 VPSs
2GB RAM - up to 320 VPSs


Users Feedback

Hello all, just downloaded and installed OpenVZ, and i must say its a big improvement over other VPS systems that i have tested IMHO.

http://forum.openvz.org/index.php?t=msg&goto=646#msg_646

I use virtuozzo in my day job and openvz is very much the same. Just no windows GUI which I hate using anyway! Virtuozzo and openvz are wonderful - I don't know why more people aren't using them. I hear a lot of hype for  and but virtuozzo/openvz is so great for many common needs. I'm very happy to be using openvz - very good for my side projects that I can't afford real virtuozzo for.

http://forum.openvz.org/index.php?t=msg&goto=650#msg_650

Last week when we were in limbo about what to do, it was decided to try out Virtualization. From what is written in the press the  system has alot of promise, <...> but was far too complicated to get working in our configuration. OpenVZ was the only virtual server system that was simple to install and get working.

http://forum.openvz.org/index.php?t=msg&goto=568#msg_568

Usage Scenarios

- Server Consolidation
- Hosting
- Development and Testing
- Security
- Educational

Server Consolidation

A bunch of servers:

- harder to manage
- upgrade is a pain
- eats up rack space
- high electricity bills

A bunch of VPSs:

- uniform management
- easily upgradeable and scalable
- fast migration

Hosting

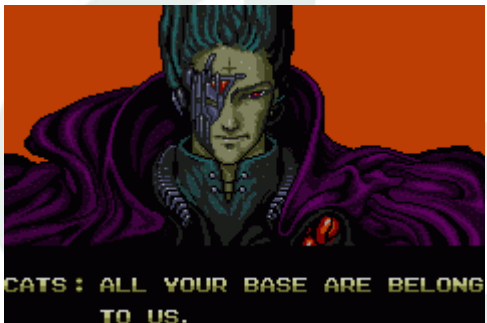
- Web server serving hundreds of virtual hosts
- Users see each other processes etc
- DoS attacks
- Unable to change/upgrade hardware
- Users are isolated from each other
- VPS is like a real server, just cheap
- Much easier to admin

Development & Testing

- A lot of hardware
- Zoo: many different Linux distros
- Frequent reinstalls take much time
- Fast provisioning
- Different distros can co-exist on one box
- Cloning, snapshots, rollbacks
- VPS is a sandbox – work and play, no fear

Security

- Several network services are running
- One of them has a hole
- Cracker gets through
- Whoops... "all your base are belong to us"
- Put each service into a separate VPS
- OpenVZ creates walls between applications
- Added benefit: dynamic resource management



Educational

- No root access
- Frequent reinstalls
- DoS attacks
- Everybody and his dog can have a root access
- Different Linux distros
- No need for a lot of hardware

Future plans

- Inject into Linux distros: Novell, Red Hat, Debian etc.
- IP v6 support
- Merge into mainstream kernel

OpenVZ Project Role

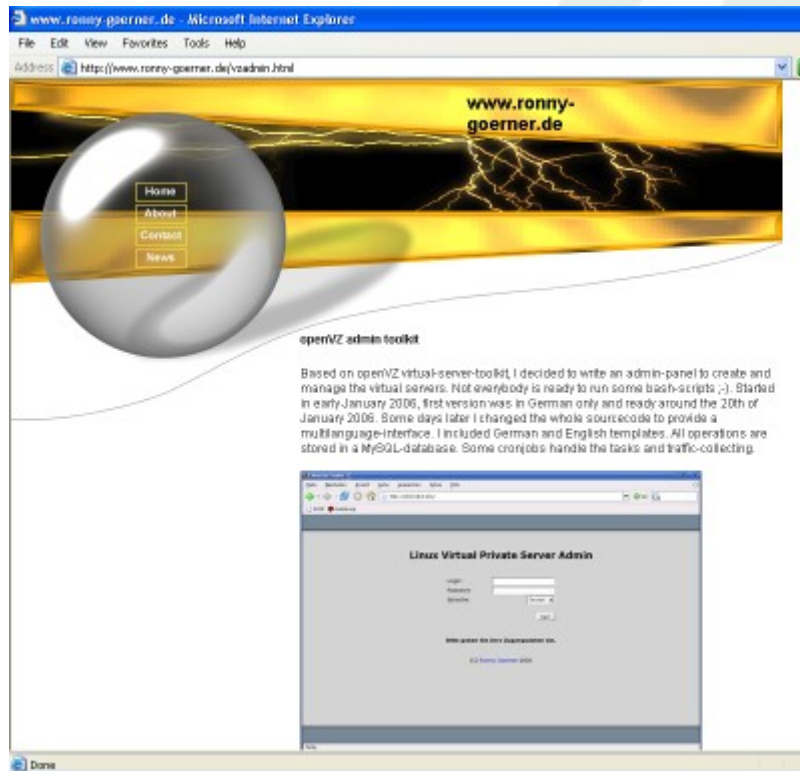
- Freely distribute and offer support to make virtualization technology accessible
- Serve the needs of the community developers, testers, documentation experts, and other technology enthusiasts who wish to participate in and accelerate the technology development process
- We hope many, many users will benefit from OpenVZ software technology, which helps increase server utilization
- The OpenVZ website is an open door to operating system virtualization software built on Linux

Your role

- Use OpenVZ
- Contribute to OpenVZ, be a part of community:
 - Programmer
 - fixes
 - enhancements
 - new functionality
 - Non-programmer
 - bug reports
 - documentation, how-tos
 - answer support questions

One example

Web Control Panel for OpenVZ



Project Links

- Main site: <http://openvz.org/>
- Downloads: <http://ftp.openvz.org/>
- GIT source repo: <http://git.openvz.org/>
- Forum: <http://forum.openvz.org/>
- Bug Tracking: <http://bugzilla.openvz.org/>
- Blog: <http://blog.openvz.org/>
- Mailing lists:
 - users@openvz.org
 - devel@openvz.org
 - announce@openvz.org