

Virtualisierung unter Linux

Linux Workshop Köln

20. Juli 2010

Harald Weidner
hweidner@gmx.net

Übersicht

- Virtualisierungsziele und -techniken
- Kernel based Virtual Machine (KVM)
- Linux Containers (LXC)
- Virtualization Library (libvirt)

Was ist Virtualisierung?

If something is there and you can see it, it's **real**.

If something is there but you can't see it, it's **hidden**.

If something isn't there but you can see it, it's **virtual**.

And if something isn't there and you can't see it, it's **gone**.

[aus dem Usenet, Verfasser unbekannt]

Virtualisierung

- Zielsetzung: Betrieb eines oder mehrere Gast-Betriebssysteme auf einer Maschine
- Emulation
 - Simulation eines vollständigen Rechners (CPU, Speicher, Festplatte, Netzwerkkarte, ...)
- Virtualisierung
 - Aufteilung der CPU- und Speicherressourcen
 - Ggf. Simulation der Hardware (Festplatte, Netzwerkkarte, Grafikkarte, ...)
 - Realisierung in Hardware und/oder Software

Virtualisierungstechniken (1)

- Hardware-Virtualisierung
 - Prozessoren und MMUs mit Virt'unterstützung
 - Hypervisor als Verwaltungsinstanz
 - IBM zSeries (LPAR, VM), pSeries (LPAR), Intel VT (Vanderpool), AMD SVM (Pacifica)
- Paravirtualisierung
 - Performanceoptimierte Treiber-API statt HW-Emulation
 - I/O über Host-Betriebssystem
 - Angepasstes Gast-Betriebssystem mit paravirt. Treibern für Storage und Netzwerk

Virtualisierungstechniken (2)

- (Voll-)Virtualisierung
 - Gemeinsame Nutzung der CPU
 - Unmodifiziertes Gast-Betriebssystem
 - Emulation der Hardware
- Container
 - Virtualisierungsfunktionen in den einzelnen Subsystemen des Linux-Kernels
 - Prozessverwaltung, Speicherverwaltung, User, Filesysteme, Netzwerk, IPC, etc.
- Applikationsvirtualisierung
 - Virtuelle Maschine für Applikationskontext (z.B. Java)

Qemu (1)

- Qemu (2003)
 - Hersteller: Qumranet
 - Emulator für komplette PCs
 - CPUs: x86, PPC, M68k, Alpha, ARM, Sparc etc.
 - Harddisk, CDrom, Netzwerkkarte
 - Video mit SDL, serielle Konsole, USB
- KQEMU (2005-2009)
 - Virtualisierungsschicht für Qemu als Kernelmodul
 - Ursprünglich proprietär, seit 2007 unter GPL
 - Seit 2009 unterstützt Qemu kqemu nicht mehr
 - Alternatives OSS Projekt: QVM86 (eingestellt 2007)

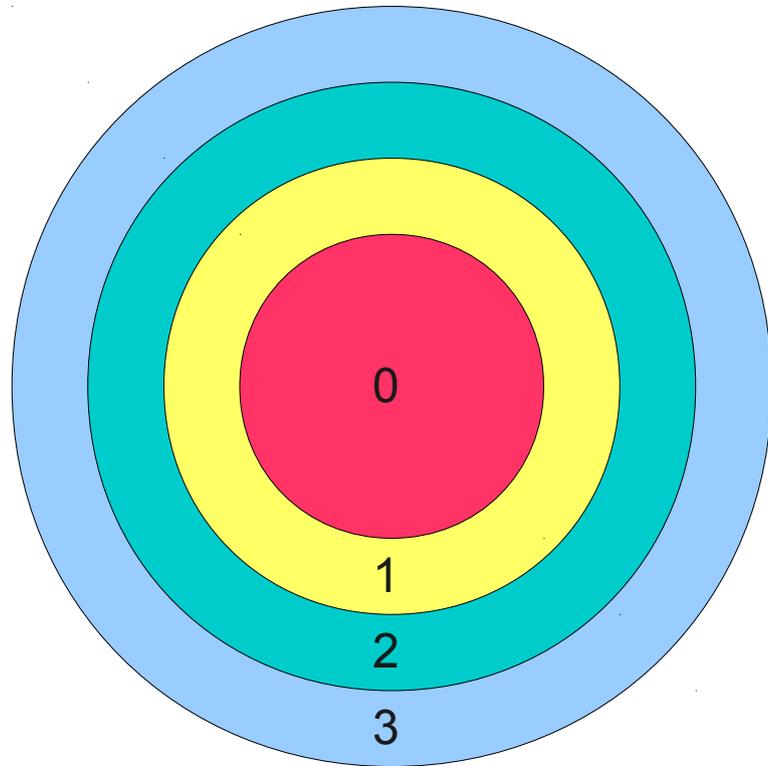
Qemu (2)

- Emulationsschicht für Xen, KVM
- Emuliert
 - IDE/SCSI Controller
 - Netzwerkkarten
 - Grafikhardware
 - BIOS
 - etc.
- Basis für die Entwicklung von VirtualBox

KVM

- Vollvirtualisierung mit Hardware-Unterstützung
- Kernelmodule + Userspace Tools basierend auf Qemu
- HW-Unterstützung: i386, amd64 (Intel VT, AMD-V)
- Host-OS: Linux (Bestandteil des offiziellen Kernels seit 2.6.20)
- Gast-OS: Linux, Windows (NT4 – 7), *BSD, (Open-)Solaris, etc.
- Optional paravirtualisierte Treiber für Harddisk und Netzwerk (virtio)
- Management über Kommandozeile oder libvirt

CPU Ringe (Intel/AMD)



Intel (seit 80286)

- 0: Supervisor Mode
- 1-3: User Mode

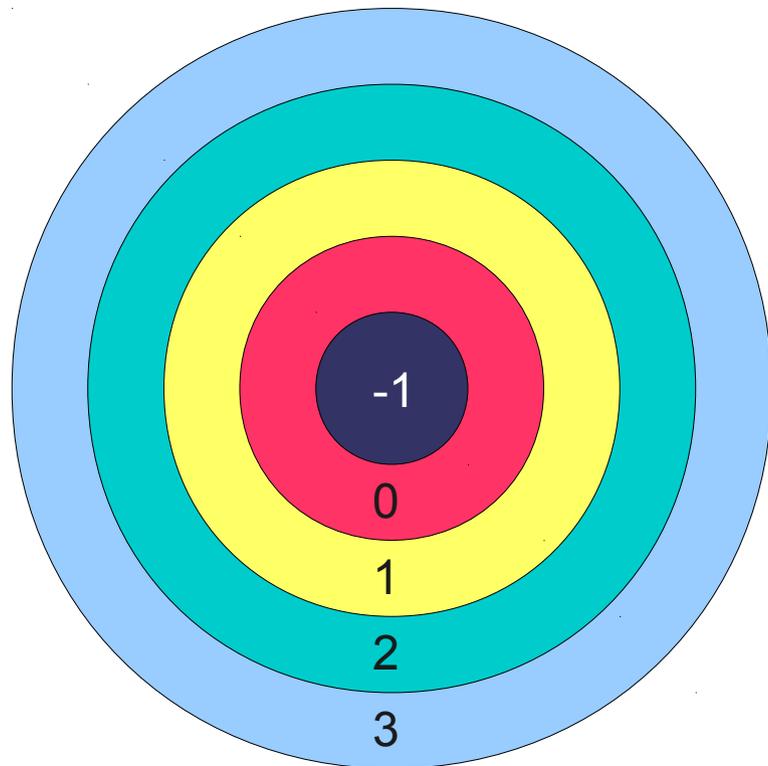
Linux

- 0: Kernel Mode
- 1+2: ungenutzt
- 3: User Mode

VirtualBox, Xen etc.

- 0: Hostsystem, Hypervisor
- 1: Gastsystem-Kernel
- 3: User Mode (Host+Gast)

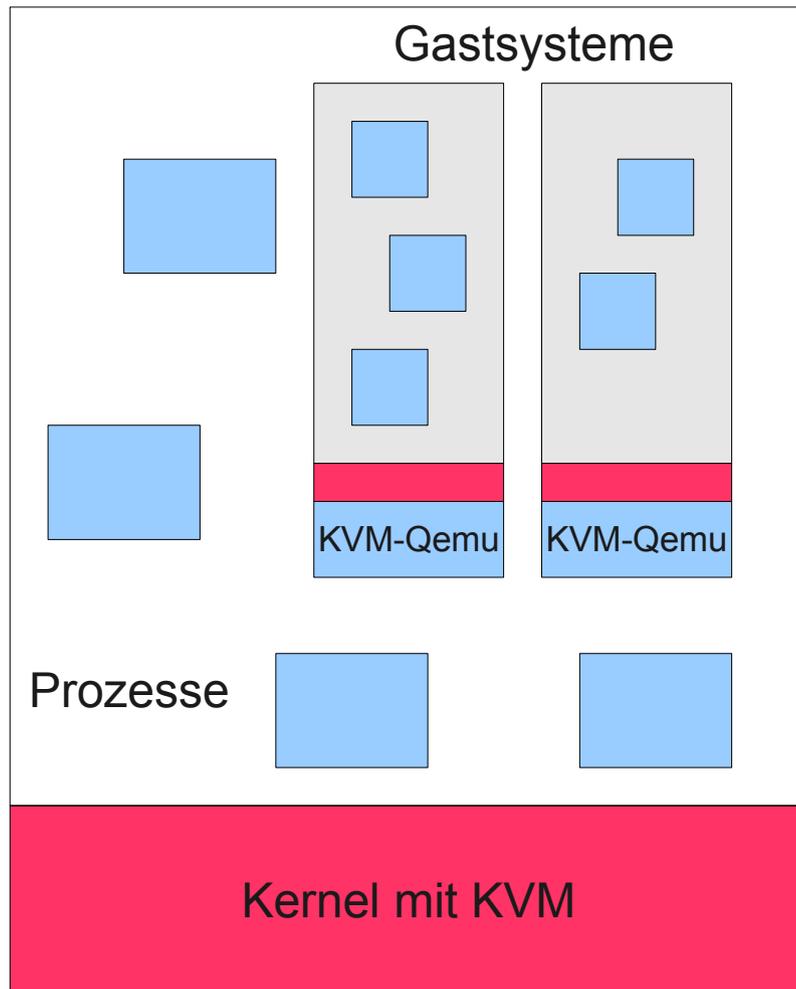
CPU-Ringe mit HVM



KVM, Xen-HV

- -1: Hypervisor
- 0: Kernel (Host+Guest)
- 3: User Mode (Host+Guest)

KVM – Architektur



- Host-Kernel als Hypervisor
- Gastssysteme erscheinen als Prozesse auf Host
- Steuerung über kill, nice, taskset, ...
- VM benötigt keine root-Rechte

KVM – Bedienung

```
kvm --help
```

```
kvm vm1.img
```

ist Abkürzung für

```
kvm -sdl -drive file=vm1.img -boot c  
-m 128 -smp 1 -net nic -net user
```

KVM - Laufwerke

- fda/-fdb file file als Floppy Image
- hd[a-d] file file als IDE Harddisk Image
- cdrom file file als CD Image
- drive file=file,if=type,...,boot=on|off
generisches Interface für Disks
- sd file SD Karten
- boot [acdn] Boot (Floppy, HD, CD, Netz)

KVM – Anzeige

- `-nographic` keine Grafik, serielle Konsole
- `-sdl` Anzeige über SDL
VM terminiert, wenn Fenster geschlossen wird
KVM Konsole über C-A-1/2/3
- `-vnc display` Anzeige über VNC
VM läuft auch ohne aktive Anzeige
- `-k de` Deutsche Tastatur

KVM - Netzwerk

`-net nic[,vlan=n,macaddr=mac,model=type,...]`

Netzwerkinterface im Gast

`-net user[,vlan=n,net=addr/mask,...]`

User Mode Networking (Qemu)

`-net tap[...]` Mit TAP Device verbinden

(z.B. Verbindung zu Bridge)

`-net none` Kein Netzwerk

KVM – Netzwerk

Bridged Networking

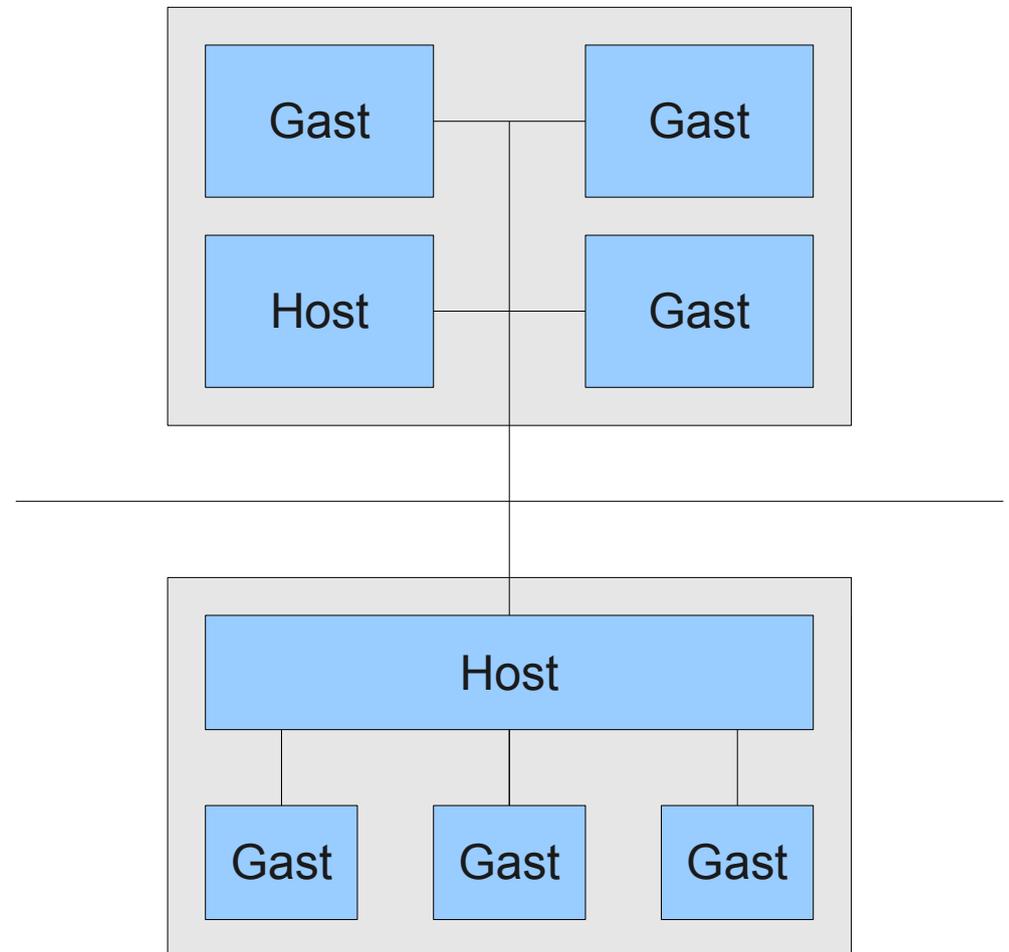
- Über Linux Bridging
- Mit oder ohne externem Interface

Routing

- Hostrouten zwischen Host u. Gast
- Mit oder ohne NAT

Mischformen

- Interne Bridge, nach außen geroutet



KVM – weitere Optionen

- `-smp n` Anzahl Prozessoren
- `-m n` Speicher (in MB)
- `-audio-...` Soundausgabe
- `-usb` USB Treiber aktivieren
- `-usbdevice ...` USB Geräte durchreichen
- `-pcidevice ...` PCI Geräte durchreichen

KVM - Paravirtualisierung

- Paravirtualisierte Treiber im Gast (Linux, Windows) kommunizieren mit API in KVM/Qemu
- Harddisk
`kvm -drive file=vm1.img,if=virtio ...`
(Devices im Gast: /dev/vda...!)
- Netzwerk
`kvm -net nic,model=virtio -net ...`
- Performance nicht optimal!
Projekt AlacrityVM soll Abhilfe schaffen.

KVM - Xen

KVM

- Im offiziellen Kernel
- Einfacher
- Mischung von VM und anderen Prozessen
- Setzt HVM voraus

Xen

- Höhere Performance, insbes. bei Paravirtualisierung
- Paravirtualisierung auch ohne HVM
- Patch beruht auf veraltetem Kernel

KVM - VirtualBox

KVM

- Etwas performanter
- Keine 3D Grafik
- HVM Voraussetzung
- Freie Software
- Zielgruppe: Server, Entwicklung, Test

VirtualBox

- GUI Bedienung
- Schnelle 3D Grafik
- HVM nicht nötig
- Kommerzielles Produkt, eingeschränkte OSS Version
- Zielgruppe: Desktop

Linux Containers (LXC)

- Container: isolierte Umgebung innerhalb des Systems
- Minimal: Isolation von Filesystem, Prozessliste und IPC
- Optional: CPUsets, Load, max. Memory
- System- oder Anwendungscontainer
- Verfügbar ab Kernel 2.6.27, volle Funktionen ab 2.6.29

LXC – System Container

- Debian / Ubuntu

```
debootstrap lenny /srv/lxc/con1 \  
http://ftp2.de.debian.org/debian
```

- Mit LXC Templates

```
lxc-create -n con1 \  
-f /path/to/config -t debian
```

LXC – Application Container

Bind Mounts

/bin → /srv/lxc/con2/bin (ro,nosuid)

/lib → /srv/lxc/con2/lib (ro,nosuid)

/usr → /srv/lxc/con2/usr (ro,nosuid)

/var/www → /srv/lxc/con2/var/www (ro,nosuid)

Minimales /etc und /var innerhalb /srv/lxc/con2

LXC – Konfiguration

```
lxc.utsname      = con1
lxc.network.type = veth
lxc.network.flags = up
lxc.network.link = br0
lxc.network.ipv4 = 192.168.1.16/24
lxc.network.name = eth0
lxc.mount        = /etc/lxc/con1/fstab
lxc.rootfs       = /srv/lxc/con1
```

```
lxc-create -n con1 -f /etc/lxc/con1.conf
lxc-destroy -n con1
```

LXC - Bedienung

`lxc-start -n con1 init` System-Container starten

`lxc-start -n con2 /root/my_app.sh`
App-Container starten

`lxc-info -n con1` Container-Info

`lxc-stop -n con1` Container stoppen

`lxc-freeze -n con2` Container einfrieren

`lxc-unfreeze -n con2` Container auftauen

`lxc-wait -n con2 -s FROZEN`
auf Zustand warten

LXC - Linux-VServer

LXC

- Bestandteil des offiziellen Linux Kernels
- Bridging, dadurch eigenes Routing
- Live Migration (geplant)

Linux-VServer

- Sehr stabil und ausgereift
- Netzvirtualisierung durch Aliase, gemeinsame Routingtabelle
- Etwas bessere Performance
- Zukunft unklar

LXC - OpenVZ

LXC

- Generische Lösung
- Im offiziellen Kernel
- Einfachere Administration

OpenVZ

- Spezialisiert auf Provider/Hosting
- Feine Ressourcenlimits
- Kommerzielles Produkt Virtuozzo
- Kernelpatch sehr invasiv und meist veraltet

KVM - LXC

KVM

- Server und Desktop
- Vollständiger Rechner
- Kernelmodule
- Distributionstests
- Andere Betriebssysteme

LXC

- Serverdienste
- Ablaufumgebung für Prozesse
- Volle Host-Performance
- Schneller Start
- Linux only

Virtualization Library (libvirt)

- Einheitliche Verwaltung virtueller Maschinen
- Library und CLI Tools
- Bindings für diverse Programmiersprachen
- Unterstützung für diverse Virt'lösungen:
 - Xen, QEMU, KVM, VirtualBox
 - LXC, Linux-VServer, OpenVZ
 - User Mode Linux
 - VMware Server, Virtual Infrastructure (ESX)

Informationsquellen

- www.linux-kvm.org
- www.qemu.org
- qemu-buch.de
- en.wikipedia.org/wiki/Comparison_of_platform_virtual_machines
- lxc.sourceforge.net
- www.ibm.com/developerworks/linux/library/l-lxc-containers
- libvirt.org