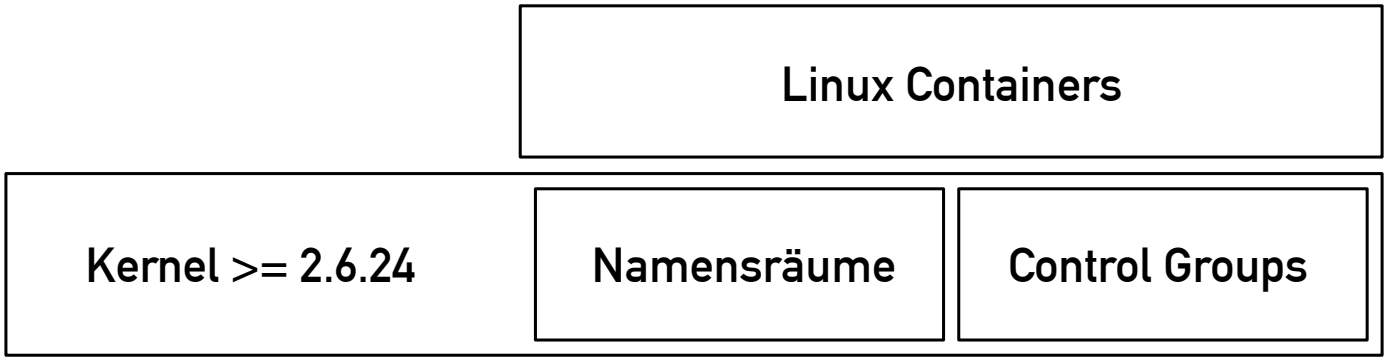
Linux Containers (LXC)

Überblick

1) Namensräume

2) Control Groups

3) Linux Containers

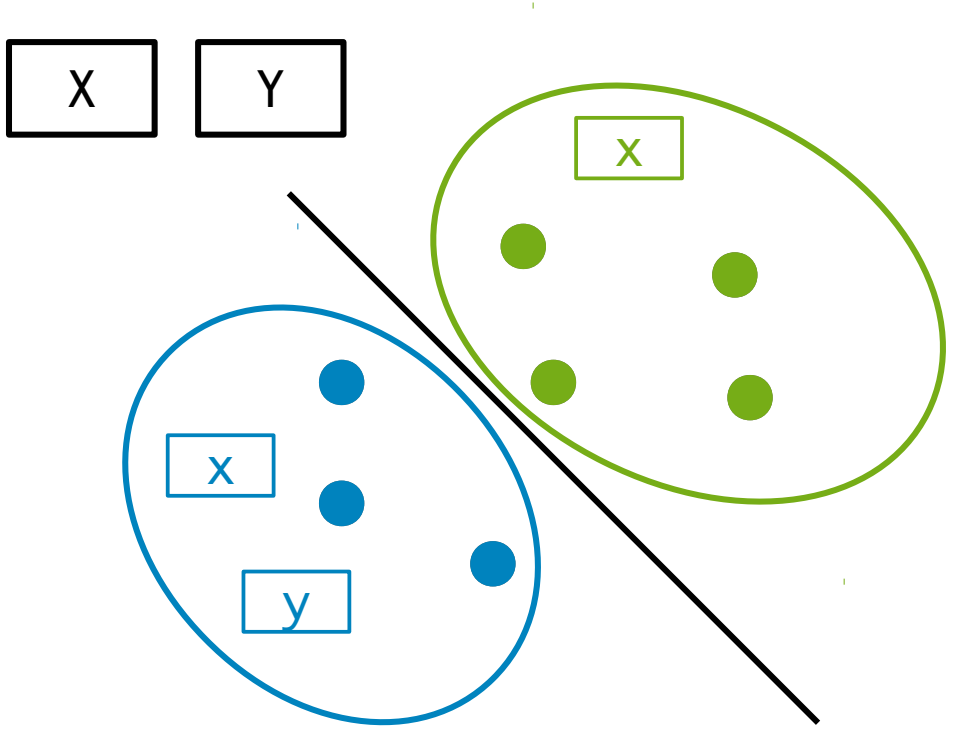


Namensräume

- Zusammenfassen von Prozessen

- Schaffen einer privaten Sicht auf eigentlich geteilte Kernel-Ressourcen

- Isolation

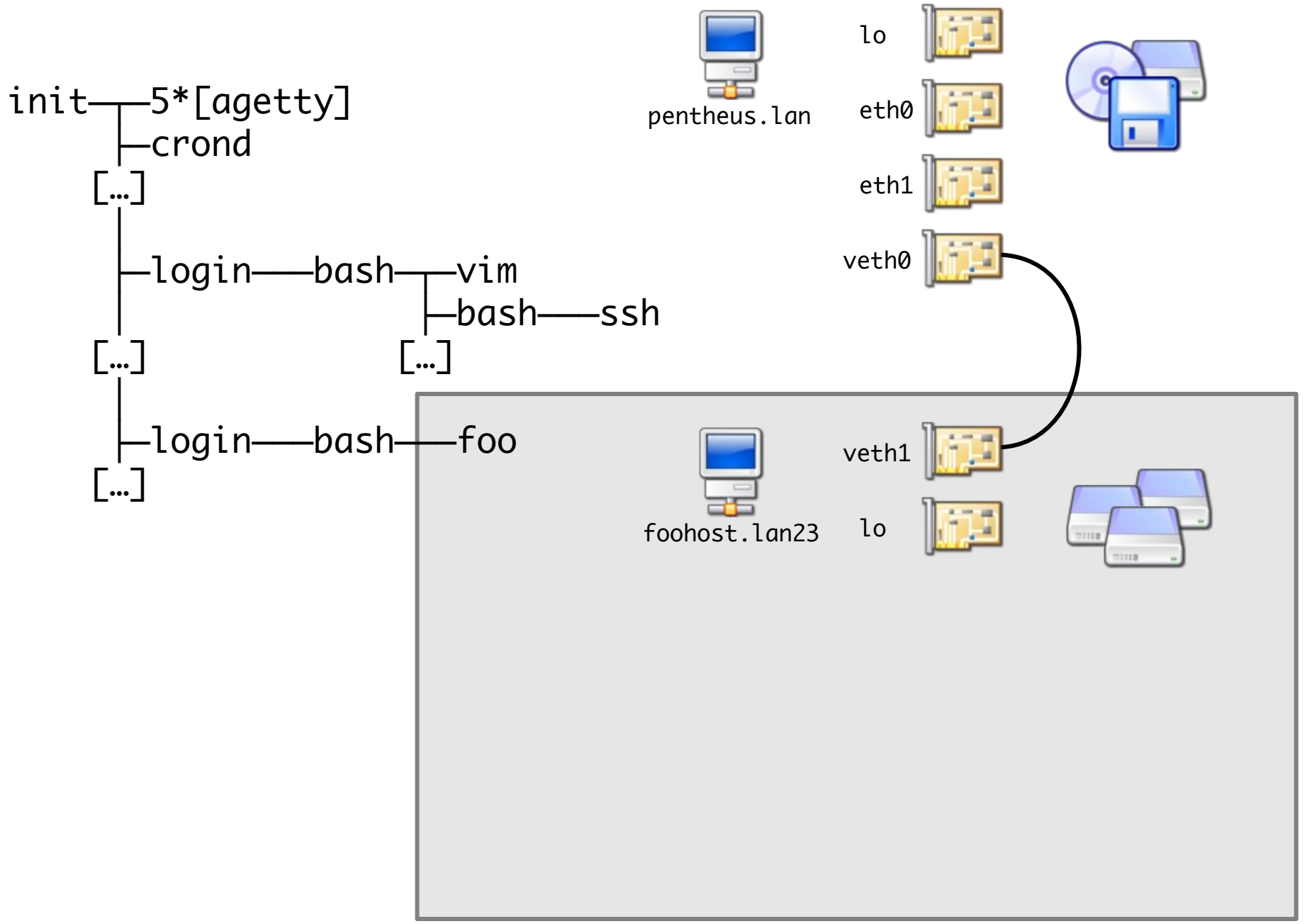


Namensräume

- Kernel-Ressourcen: X
  - Netzwerk-Stack 
  - Hostname/Domain 

  - Mount-Status 

Namensräume



Namensräume

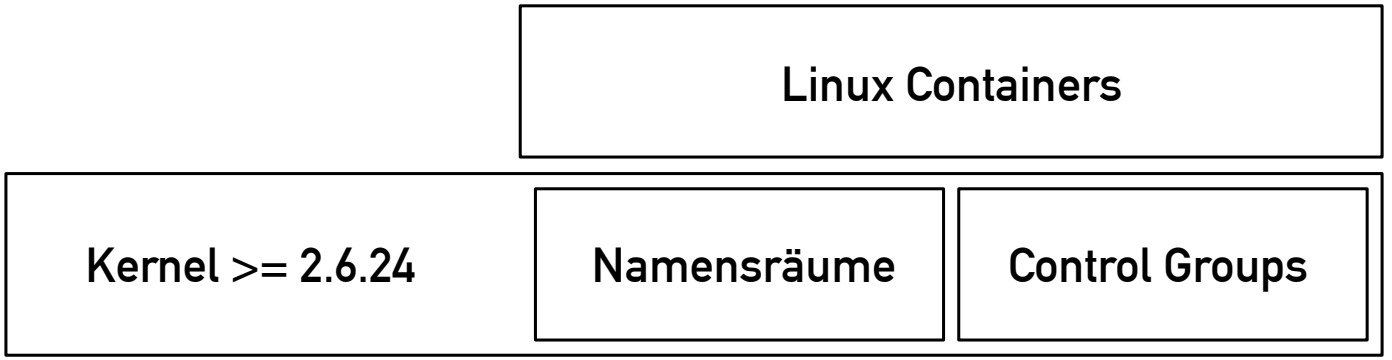
- Demo…

Überblick

1) Namensräume

2) Control Groups

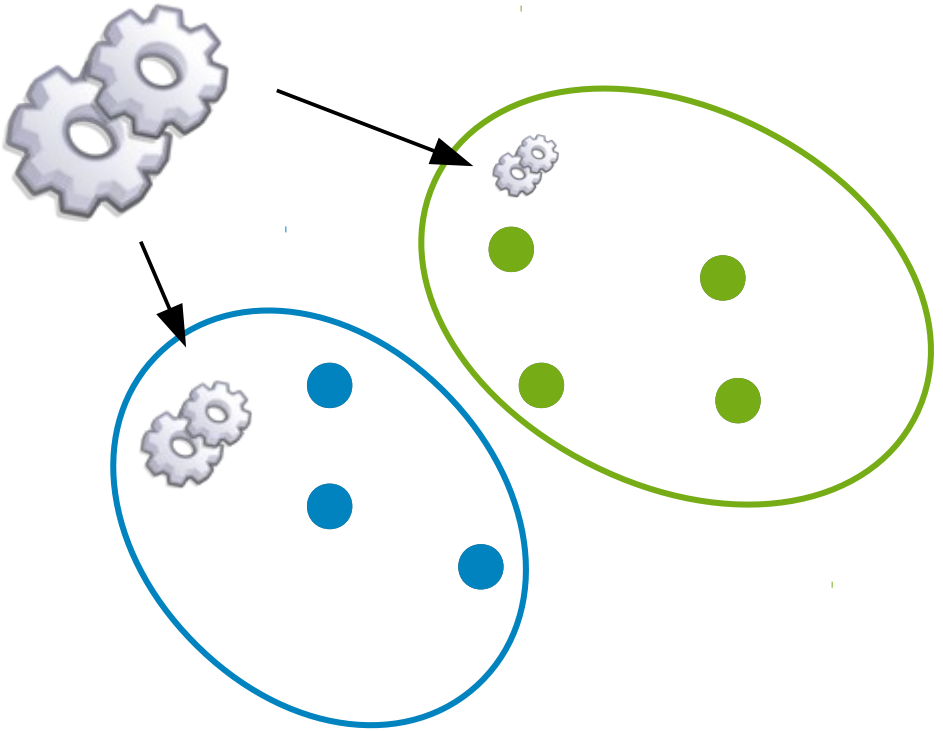
3) Linux Containers



Control Groups

- seit 2.6.24 im Mainline Kernel
- Zusammenfassen von Prozessen zu Prozessgruppen
- Restriktion von Betriebsmitteln pro Prozessgruppe

- schachtelbar
- steuerbar über virtuelles Dateisystem



Control Groups

- Betriebsmittel:

  - CPU
  - Hauptspeicher
  - Device Nodes
  - I/O-Bandbreite

    für Block Devices
  - Netzwerkpakete
  - Prozesszustand

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset          0            1              1
cpu             0            1              1
cpuacct         0            1              1
memory          0            1              1
devices         0            1              1
freezer         0            1              1
net_cls         0            1              1
blkio           0            1              1
```

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@pentei
#subsys_name    hierarchy   num_cgroups   enabled
cpuset          0           1             1
cpu             0           1             1
cpuacct         0           1             1
memory          0           1             1
devices         0           1             1
freezer         0           1             1
net_cls         0           1             1
blkio           0           1             1
```

Zuweisung von CPUs und Speicherknoten

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@pentei
#subsys_name    hierarchy   num_cgroups   enabled
cpuset          0           1             1
cpu             0           1             1
cpuacct         0           1             1
memory          0           1             1
devices         0           1             1
freezer         0           1             1
net_cls         0           1             1
blkio           0           1             1
```

Vergeben von
CPU-Prioritäten

Control Groups

- Betriebsmittel → „Subsysteme“

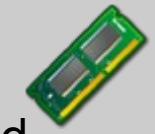
```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy       num_cgroups     enabled
cpuset          0               1               1
cpu             0               1               1
cpuacct         0               1               1
memory          0               1               1
devices         0               1               1
freezer         0               1               1
net_cls         0               1               1
blkio           0               1               1
```

CPU-Statistiken

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset                                      1
cpu                                         1
cpuacct                                     1
memory                                      1
devices         0            1              1
freezer         0            1              1
net_cls         0            1              1
blkio           0            1              1
```



Speicherbegrenzungen und Statistiken

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset          0            1              1
cpu                                         1
cpuacct                                     1
memory                                      1
devices                                     1
freezer         0            1              1
net_cls         0            1              1
blkio           0            1              1
```

Black List und
White List für verfügbare
Device Nodes

Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset          0            1              1
cpu             0            1              1
cpuacct                                     1
memory                                      1
devices                                     1
freezer                                     1
net_cls         0            1              1
blkio           0            1              1
```

Prozessgruppen
einfrieren und auftauen

Control Groups

- Betriebsmittel → „Subsysteme“

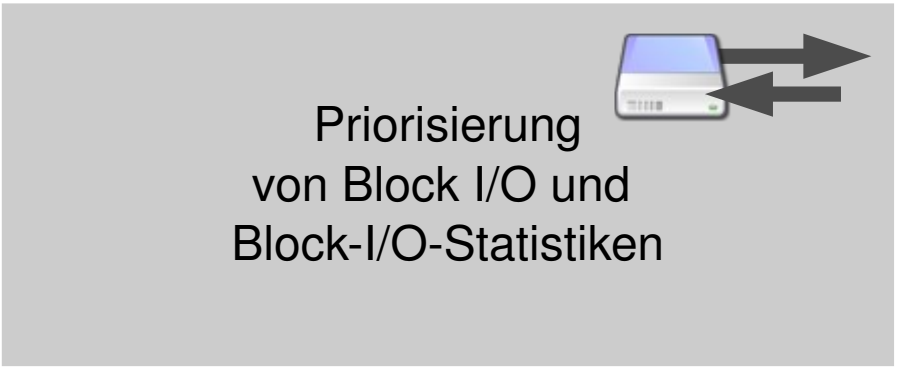
```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset          0            1              1
cpu             0            1              1
cpuacct         0            1              1
memory                                      1
devices                                     1
freezer                                     1
net_cls                                     1
blkio           0            1              1
```

Paket-Markierung
→ kann von tc (traffic control)
ausgewertet werden

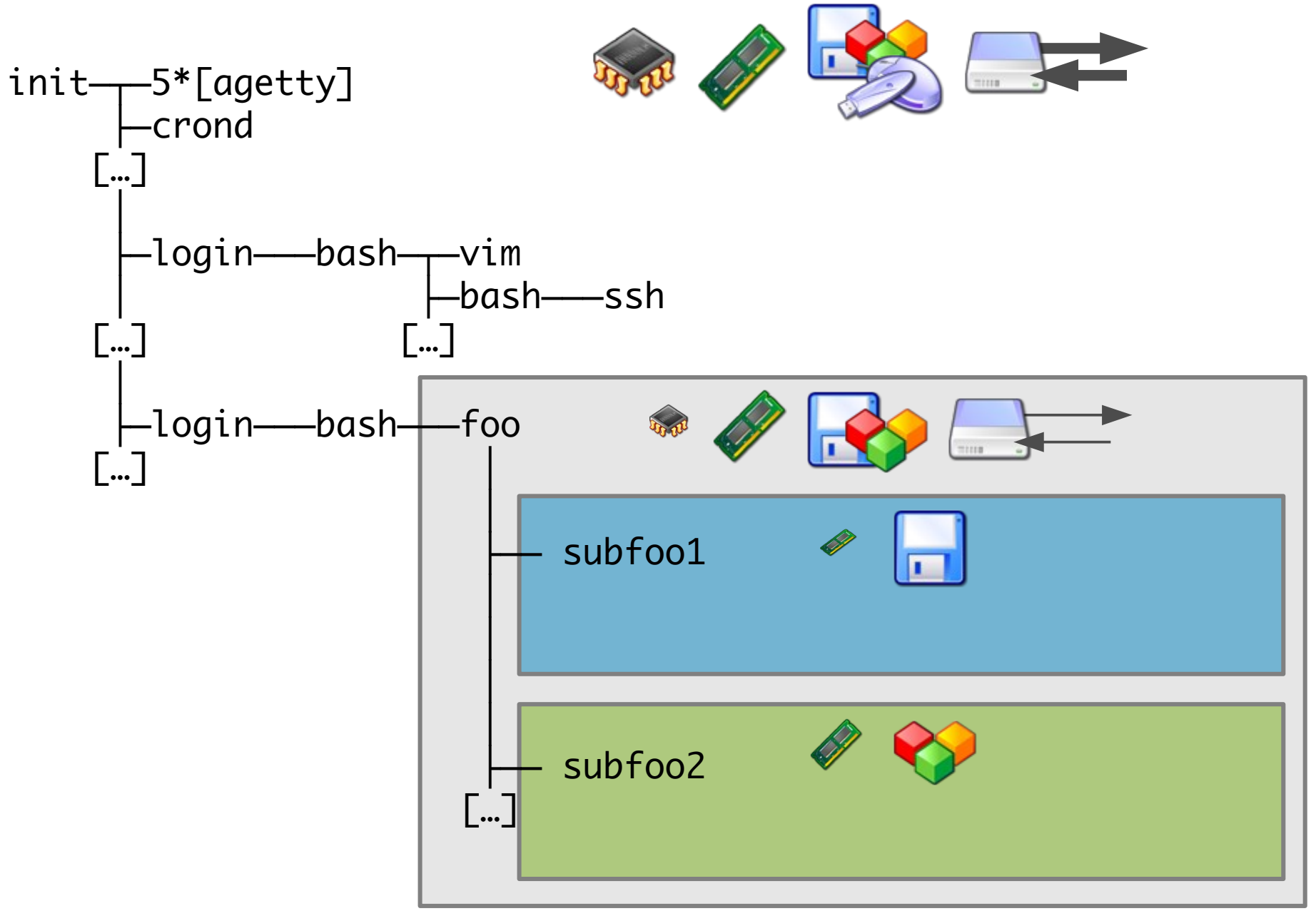
Control Groups

- Betriebsmittel → „Subsysteme“

```
patrick@penteus mnt $ cat /proc/cgroups
#subsys_name    hierarchy    num_cgroups    enabled
cpuset          0            1              1
cpu             0            1              1
cpuacct         0            1              1
memory                                      1
devices                                     1
freezer                                     1
net_cls                                     1
blkio           0            1              1
```



Control Groups



Control Groups

- Verwendung:

```
mount -t cgroup -o [
    cpuset,cpu,cpuacct,memory,devices,freezer,net_cls,blkio
] none /sys/fs/cgroup
```

- Control Group anlegen

```
mkdir /sys/fs/cgroup/lt
```

Control Groups

Ressourcen einstellen

```
root@penteus ~ # cat /sys/fs/cgroup/lt/test03/cpu.shares
1024
root@penteus ~ # echo 512 > /sys/fs/cgroup/lt/test03/cpu.shares
root@penteus ~ # cat /sys/fs/cgroup/lt/test03/cpu.shares
512
```

Control Groups

Prozess einer Control Group hinzufügen

```
root@penteus ~ # ls /sys/fs/cgroup/lt/test03/
blkio.io_merged
blkio.io_queued
blkio.io_service_bytes
blkio.io_serviced
blkio.io_service_time
blkio.io_wait_time
blkio.reset_stats
[…]
memory.kmem.tcp.failcnt
memory.kmem.tcp.limit_in_bytes
memory.kmem.tcp.max_usage_in_bytes
memory.usage_in_bytes
memory.use_hierarchy
net_cls.classid
notify_on_release
tasks
```

Control Groups

Prozess einer Control Group hinzufügen

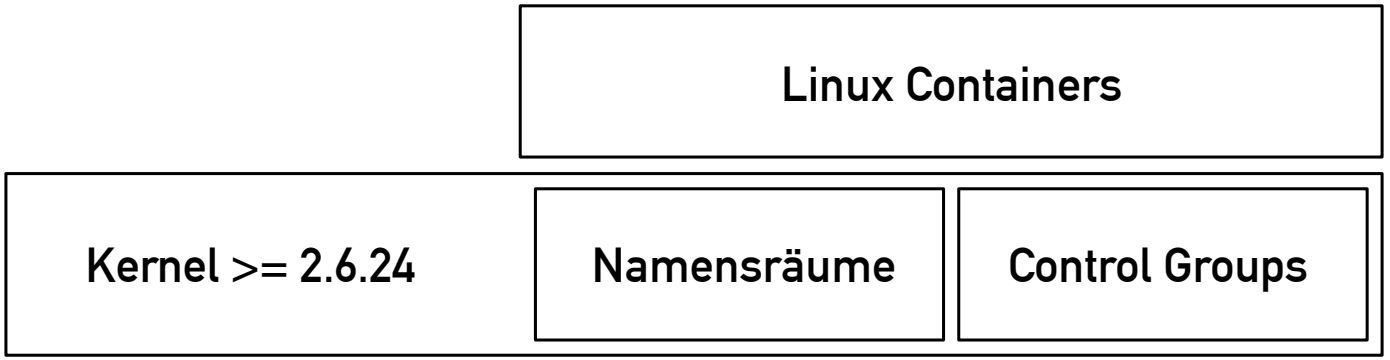
```
root@penteus ~ # cat /sys/fs/cgroup/lt/test03/tasks
root@penteus ~ #
root@penteus ~ # echo $$
597
root@penteus ~ # echo 597 > /sys/fs/cgroup/lt/test03/tasks
root@penteus ~ # cat /sys/fs/cgroup/lt/test03/tasks
597
1930
root@penteus ~ # /bin/bash
root@penteus ~ # cat /sys/fs/cgroup/lt/test03/tasks
597
1931
1932
```

Überblick

1) Namensräume

2) Control Groups

3) Linux Containers



Linux Containers

- seit 2008
- leichtgewichtige Virtualisierungslösung
- Kapselung skalierbar von minimaler Isolierung eines Prozesses bis hin zu Containering
- CLI + umfangreiche Konfigurationsmöglichkeiten
- nutzt bestehende Mechanismen des Linux-Kernels (>= 2.6.24)

Linux Containers

- zwei Betriebsmodi:

`lxc-execute`

→ eigener Namensraum + eigene Control Group

→ z.B. für Webserver mit RAM-Begrenzung und eigenem Netzwerk

`lxc-start`

→ eigener Namensraum + eigene Control Group + chroot

→ z.B. für Container-VMs mit eigener Betriebssystem-umgebung

Linux Containers

Was genau

ist ein Linux Container?

```
root@penteus lxc # tree -L 2 debian
debian
├── config
└── rootfs
    ├── bin
    ├── boot
    ├── dev
    ├── etc
    ├── home
    ├── lib
    ├── lib64
    ├── media
    ├── mnt
    ├── opt
    ├── proc
    ├── root
    ├── run
    ├── sbin
    ├── selinux
    ├── srv
    ├── sys
    ├── tmp
    ├── usr
    └── var
```

Linux Containers

Konfiguration

```
root@penteus lxc # tree -L 2 debian
debian
├── config
└── rootfs
    ├── bin
    ├── boot
    ├── dev
```

```
lxc.utsname=debian
lxc.tty=4
lxc.rootfs=/usr/local/var/lib/lxc/debian/rootfs
lxc.pts=1024
lxc.network.type=veth
lxc.network.ipv4=10.77.1.253/24
lxc.network.hwaddr=00:16:3E:4D:01:FD
lxc.network.veth.pair=veth1-0
lxc.mount.entry=proc /usr/local/var/lib/lxc/debian/rootfs/proc
    proc nodev,noexec,nosuid 0 0
lxc.mount.entry=sysfs /usr/local/var/lib/lxc/debian/rootfs/sys
    sysfs defaults 0 0
lxc.cgroup.memory.limit_in_bytes=134217728

lxc.cgroup.devices.deny=all

lxc.cgroup.devices.allow=c 1:3 rwm # /dev/null
lxc.cgroup.devices.allow=c 1:5 rwm # /dev/zero
[…]
```

```
    └── var
```

Linux Containers

```
root@penteus lxc # tree -L 2 debian
debian
├── config
└── rootfs
    ├── bin
    ├── boot
    ├── dev
    └── var
```

Konfiguration

- → Hostname
- → Netzwerk
  - → Typ
  - → IPv4/6-Adressen
  - → Gateway
  - → MAC-Adressen
  - → Up-Skripte
- → Mount Points
- → Root Filesystem
- → Control Groups
- → Drop Capabilities

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach      lxc-destroy     lxc-netstat     lxc-unfreeze
lxc-cgroup      lxc-execute     lxc-ps          lxc-unshare
lxc-checkconfig lxc-freeze      lxc-restart     lxc-version
lxc-checkpoint  lxc-info        lxc-setcap      lxc-wait
lxc-clone       lxc-kill        lxc-setuid
lxc-console     lxc-ls          lxc-start
lxc-create      lxc-monitor     lxc-stop
```

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

Befehl (vom Host aus) zur Ausführung im Linux Container absetzen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach      lxc-destroy     lxc-netstat     lxc-unfreeze
lxc-cgroup      lxc-execute     lxc-ps          lxc-unshare
lxc-checkconfig lxc-freeze      lxc-restart     lxc-version
lxc-checkpoint  lxc-info        lxc-setcap      lxc-wait
lxc-clone       lxc-kill        lxc-setuid
lxc-console     lxc-ls          lxc-start
lxc-create      lxc-monitor     lxc-stop
```

Control-Group-Einstellungen auslesen und setzen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

System bzgl. der LXC-Tauglichkeit überprüfen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc
lxc
```

```
root@penteus ~ # lxc-checkpoint -n debian --statefile=/tmp/foo
lxc-checkpoint: 'checkpoint' function not implemented
lxc-checkpoint: failed to checkpoint 'debian'
```

checkpoint/restart

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

einen Linux Container clonen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

auf getty-Konsole eines laufenden
Linux Containers zugreifen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

leeren Linux Container mit leerer config erstellen (oder Daten aus angegebenen Template übernehmen) / Linux Container löschen (samt zugehörigen Daten im Dateisystem)

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

einen Linux Container starten/stoppen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

einen Linux Container einfrieren/auftauen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

Linux-Container-Infos anzeigen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

`kill` für Linux Container

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

zeigt laufende und verfügbare Linux Container an

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

Überwachung von Zustandsänderungen eines Linux Containers

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

vom Host aus `netstat` in einem Linux Container aufrufen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach      lxc-destroy     lxc-netstat     lxc-unfreeze
lxc-cgroup      lxc-execute     lxc-ps          lxc-unshare
lxc-checkconfig lxc-freeze      lxc-restart     lxc-version
lxc-checkpoint  lxc-info        lxc-setcap      lxc-wait
lxc-clone       lxc-kill        lxc-setuid
lxc-console     lxc-ls          lxc-start
lxc-create      lxc-monitor     lxc-stop
```

vom Host aus `ps` in einem Linux Container aufrufen

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

nötige Capabilities und Rechte an die entsprechenden lxc-Binaries/-Scripts binden, sodass auch Nutzer mit eingeschränkten Rechten Linux Containers verwalten können

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

Erweitertes unshare **mit Unterstützung für**
MOUNT, PID, UTSNAME, IPC, USER, NETWORK

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

Version der LXC Tools

Linux Containers

CLI

```
root@penteus lxc-0.8.0-rc1 # lxc-
lxc-attach       lxc-destroy     lxc-netstat      lxc-unfreeze
lxc-cgroup       lxc-execute     lxc-ps           lxc-unshare
lxc-checkconfig  lxc-freeze      lxc-restart      lxc-version
lxc-checkpoint   lxc-info        lxc-setcap       lxc-wait
lxc-clone        lxc-kill        lxc-setuid
lxc-console      lxc-ls          lxc-start
lxc-create       lxc-monitor     lxc-stop
```

blockiert bis ein bestimmter Linux Container in den angegebenen Zustand übergegangen ist

Überblick

1) Namensräume

2) Control Groups

3) Linux Containers