Leichtgewichtige Virtualisierung mit LXC-Containern

Henry Jensen

2021-10-27

Über mich

- Henry Jensen, 51 Jahre alt, verheiratet, Asperger
- Erster Computer: 1990, Linux seit 1999
- 2003 FiSi
- Netzwerk- und Systemadminstrator mit Schwerpunkt GNU/Linux
- Seit August 21 bei der Zentralstelle für Weiterbildung im Handwerk (ZWH) in Düsseldorf

Werbeblock:

- Wir suchen Linux Systemadministratoren und Anwendungsentwickler (PHP, JavaScript)
- https://zwh.de/ueber-uns/karriere/

Was sind Container?

- Ein Container stellt eine Umgebung bereit, um einen oder mehrere Prozesse isoliert vom Hauptsystem zu betreiben
- Es wird der Kernel des Hauptsystems genutzt
- Am Anfang war chroot (Unix Version 7, 1979)
- FreeBSD jails (2000)
- Solaris Zones (2005)
- **Linux Containers/LXC** (2008)
- Docker (2013)

LXC Übersicht

- Entwicklung seit 2008
- Linux-natives System zur Containerisierung
- Es wird ein gesamtes Betriebssystem (exklusive Kernel) containerisiert
- Lieferumfang:
  - liblxc
  - Standard LXC Tools
  - Diverse Bindings: python3, Lua, Go, Ruby, Haskell
  - Container templates
- Container-Image Repo: https://images.linuxcontainers.org/
  - Alma, Alpine, Alt, Amazon, Apertis, Arch, Busybox, CentOS, Debian, Devuan, Fedora, Funtoo, Gentoo Kali, Mint Opensuse, Openwrt, Oracle, Plamo, PLD, Sabayon, Spingdale, Ubuntu, Void

Virtuelle Maschinen

Vorteile:

- Komplettes Betriebssystem inklusive Kernel
  - Hardwarenahe Operationen möglich
- Unabhängig vom Host-System
  - Fast uneingeschränkte Betriebssytemauswahl
- Starke Isolierung
  - Eine einzelne kompromittierte VM hat keine Auswirkungen auf das Hostsystem und andere VM
- Portabilität
  - Einfache Migration auf andere Hostsysteme, teilw. sogar unter anderen Hyoervisoren lauffähig

Nachteile:

- Ressourcen-Overhead
  - Emulation der Hardware durch Hypervisor/Emulator
  - Benötigt zusätzliche Ressourcen bzw. ist nicht so performant wie nativ
- Benötigt CPU-Virtualisierung

Container

Vorteile:

- Native Performance
- I.d.R. sehr klein
- Sehr schnell einzurichten (z.B. mit Automatisierung oder entsprechenden Tools)
- Viele adminstrative Dinge können von Host aus erlegigt werden (z.B. Backup)

Nachteile:

- Schwächere Isolierung als bei VM
  - Bei unsicher konfigurierten Containern ist Ausbruch relativ trivial
- Eingeschränkte Wahl des OS und der Architektur
  - Nur Linux -> Linux
  - x86 Container unter x64 Host geht jedoch

privilegiert

Der Container läuft mit Root-Rechten, der Account “root” im Container entspricht auch “root” auf dem Host. Dies macht es relativ einfach, aus dem Container auszubrechen. Diese Art von Containern gelten daher als ungeeignet, wenn der Zweck des Betriebs von Containern sein soll, Systeme voneinander zu isolieren

unprivilegiert

Der root Account im Container entspricht nicht dem root Account des Hosts (wie auch andere Useraccounts vom Host separiert sind). Dies wird erreicht durch UID-Mapping, wobei die UIDs und GIDs im Container aus Host-Sicht eine sehr hohe Nummer haben. Z.B. kann der root-Account im Container der UID 1000000 auf dem Host entsprechen. Wenn Isolationssicherheit eine Rolle spielt sollten immer unprivilegierte Container verwendet werden.

LXC-Container absichern

- Privilegierte Container: Mandatory Access Control (MAC), z.B. AppArmor SELinux
- Unprivilegierte Container: UID- und Groupmapping, SELinux & Co sind nicht nötig
- Weitere Sicherheitsmaßnahmen:
  - Namespaces: LXC verwendet Linux Namespaces um Prozesse zu kapseln Änderungen sind nur für Prozesse innerhalb eines Namespace sichtbar
  - CGroups: limitieren Ressourcen auf Prozess-Basis

- User Namesapces aktivieren

```
sysctl -w kernel.unprivileged_userns_clone=1
# oder
echo "kernel.unprivileged_userns_clone=1" > /etc/sysctl.d/80-lxc-userns.conf
```

- UID und GID Mapping

```
echo "root:1000000:65536" >  /etc/subuid
echo "root:1000000:65536" >  /etc/subgid
```

- LXC-Konfiguration - `/etc/lxc/default.conf`

```
lxc.idmap = u 0 1000000 65536
lxc.idmap = g 0 1000000 65536
```

```
DOWNLOAD_KEYSERVER="keyserver.ubuntu.com"  lxc-create -n mycontainer -t downloa
[...]
Distribution:
alpine
Release:
3.14
Architecture:
amd64

Downloading the image index
Downloading the rootfs
Downloading the metadata
The image cache is now ready
Unpacking the rootfs

---
You just created an Alpinelinux 3.14 x86_64 (20211026_13:00) container.
```

- Wird definiert in lxc.net.[i].type
  - empty: Nur Loopback-Interface
  - none: Teilt den Netzwerk-Namespace des Hosts. Nur in privilegierten Containern
  - veth: Es wird ein virtuelles Ethernet-Paar erstellt, wobei eine Seite dem Container und die andere Seite dem Host zugewiesen ist. Bridge und Router-Modus.
- Host-Bridge
  - Container erhält eine IP aus dem Host-Netzwerk
- NAT-Bridge
  - Container erhält eine IP aus einem separatem Netzwerk
  - Paket lxc-net (in Debian) kümmert sich um Erstellung der Bridge und NAT

- Unter /var/lib/lxc/<Container>/config

```
lxc.include = /usr/share/lxc/config/common.conf
lxc.include = /usr/share/lxc/config/userns.conf
lxc.arch = linux64

# Container specific configuration
lxc.idmap = u 0 1000000 65536
lxc.idmap = g 0 1000000 65536
lxc.rootfs.path = dir:/var/lib/lxc/mycontainer/rootfs
lxc.uts.name = mycontainer

# Network configuration
lxc.net.0.type = veth
lxc.net.0.flags = up
lxc.net.0.link = br0
```

Die wichtigsten LXC-Befehle

Container erstellen

- `lxc-create -n <container> -t download`

Starten und stoppen

- `lxc-start <Container>` Einen Container starten
- `lxc-stop <Container>` Einen Container stoppen

Mit einem Container verbinden

- `lxc-attach <Container> [command]` Eine Shell (default) oder einen Prozess in einem Container starten
- `lxc-console <Container>` Startet eine Login-Console (tty) in einem Container.

Informationen

- `lxc-ls -f` Übersicht über die vorhandenen Container
- `lxc-info <Container>` Info zu einem Container anzeigen

Container löschen

- `lxc-destroy <Container>` **VORSICHT - löscht auch das gesamte Dateisystem des Containers)**

LXD

- von Canonical
- alternatives Kommandozeilentool: `lxc`
- REST API
- Auch für VM verwendbar
- **kein** Paket für Debian verfügbar
- https://linuxcontainers.org/lxd/introduction/
  https://www.thomas-krenn.com/de/wiki/LXD_Grundbefehle

libvirt

- LXC container driver für libvirt
- Umgang mit LXC-Containern mit den bekannten libvirt-Werkzeugen (virsh, virt-manager ...)
- "deprecated" in RedHat
- https://libvirt.org/drvlxc.html

Proxmox VE

- auf Debian basierende Open-Source-Virtualisierungsplattform mit einem Web-Interface
- Eigenes Kommandozeilentool: `pct`
- https://www.proxmox.com/de/proxmox-ve
  https://pve.proxmox.com/pve-docs/pct.1.html

Links

- Internetserver im eigenen Heim Teil 1
  https://senioradmin.de/Inetserverathome.html
- Internetserver im eigenen Heim Teil 2
  https://senioradmin.de/Inetserverathome2.html

Diese Präsentation wurde mit Pandoc aus Markdown-Quelltext erstellt

```
pandoc -t beamer --from markdown --pdf-engine xelatex -o LXC.pdf LXC.md
```