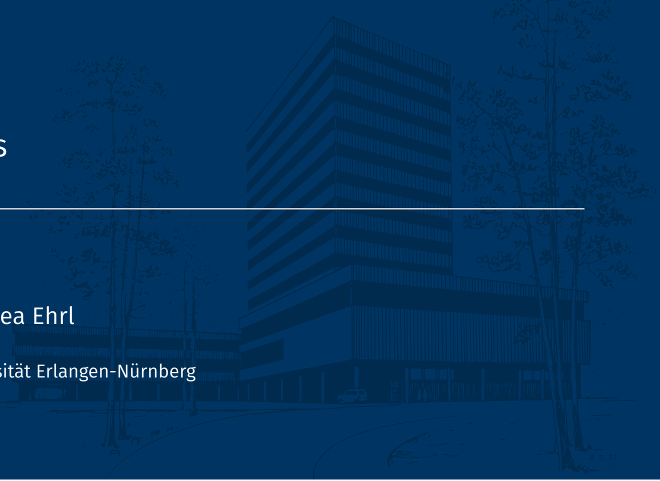


LXC

Linux Containers

11. November 2019

Vanessa Hack, Dorothea Ehrl

Friedrich-Alexander-Universität Erlangen-Nürnberg

Lehrstuhl für Verteilte Systeme
und Betriebssysteme

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Inhalt

1. Was sind Container?

2. Funktionsweise

3. Organisation

4. Ausblick

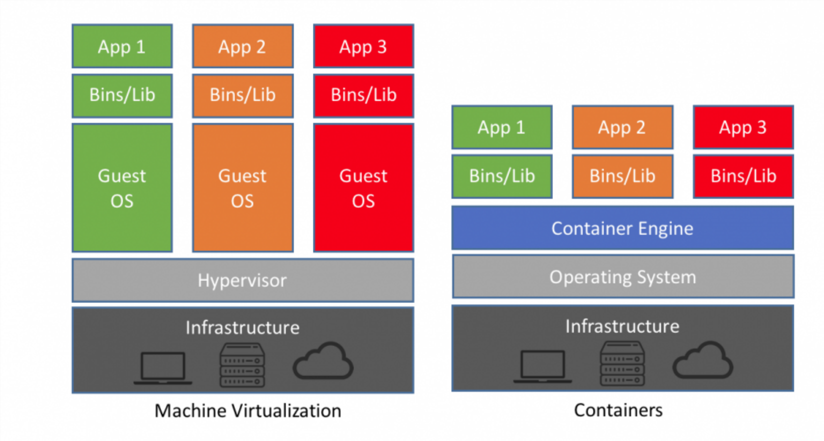
Was sind Container?

Überblick

Container

- keine Hardwareemulation
- Host Kernel wird geteilt
- Vom Host-System isolierte Prozessumgebungen
- Kleiner und schneller als VMs
- inkludierte Abhängigkeiten ermöglichen Portierbarkeit und Fokus auf Anwendungsentwicklung

Container vs. VMs

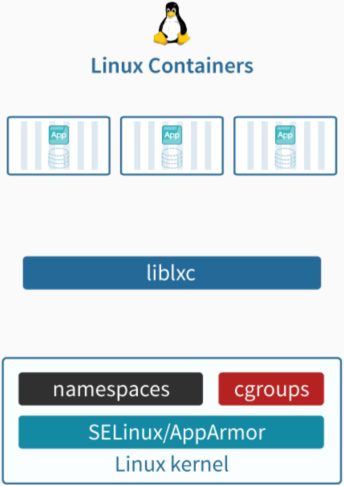


Funktionsweise

LXC

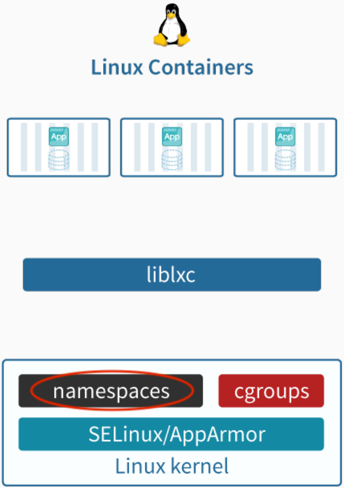
- Entwicklungsstart 2008
- erstes stabiles Release 2014
- Ziel: Umgebung entwickeln, die so nah wie möglich an einer Standard Linux Installation ist, jedoch ohne einen separaten Kernel zu benötigen
- Komponenten:
  - liblxc library
  - APIs für verschiedene Programmiersprachen (Python, Lua, ruby, Haskell...)
  - Containervorlagen und Tools zur Kontrolle der Container

Aufbau



Linux Containers
App
App
App
liblxc
namespaces
cgroups
SELinux/AppArmor
Linux kernel

Aufbau

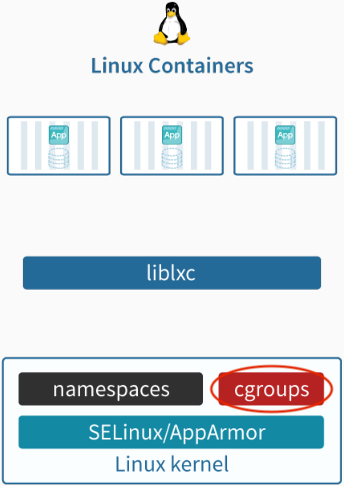


namespaces:

- Abstraktion, die Prozessen innerhalb eines namespace eigene isolierte Instanz einer globalen Variable „vorspielt“
- Änderungen sind nur für Prozesse innerhalb eines namespace sichtbar
- eindeutige Zuordnung von Namen innerhalb eines namespace
- außerhalb Name erneut verwendbar

LXC: ipc, uts, mount, pid, network, user

Aufbau



cgroups:

- Prozesse werden in hierarchische Gruppen eingeteilt
- ausgewählte Ressourcenzuweisung an definierte Gruppen
- ermöglicht Limitierung, Priorisierung und Isolation von Ressourcen

Weiterhin wichtig

chroots

- setzt neues Rootverzeichnis für einen Prozess und dessen Kinder
- verhindert damit, dass auf Dateien außerhalb der neuen Wurzel zugegriffen wird

SELinux/AppArmor

- Zugangslimiterung von Dateien nur für berechtigte Anwendungen

Kochrezept Linux Container

Ubuntu laut Website als eine der wenigen Distributionen mit allen Abhängigkeiten per default gegeben

1. **LXC Paket installieren**

   ```
   sudo apt-get install lxc
   ```

2. **Rechte setzen zur Erstellung eines Virtual Ethernet Device (/etc/lxc/lxc-usernet)**

   ```
   username veth lxcbr0 10
   ```

3. **LXC config file erstellen**

   ```
   mkdir ~/.config/lxc directory
   cp /etc/lxc/default.conf ~/.config/lxc/default.conf
   ```

   In dieser Datei ergänzen:

   ```
   lxc.idmap = u 0 100000 65536
   lxc.idmap = g 0 100000 65536
   ```

Kochrezept Linux Container

4. **Container erstellen**

   ```
   lxc-create -t download -n my-container
   ```

5. **Distribution, Version und Architektur auswählen**
6. **Container starten (im Hintergrund)**

   ```
   lxc-start -n my-container -d
   ```

7. **In eine shell gelangen**

   ```
   lxc-attach -n my-container
   ```

8. **Container beenden**

   ```
   lxc-stop -n my-container
   ```

9. **Status anzeigen**

   ```
   lxc-info -n my-container
   lxc-ls -f
   ```

Kochrezept Linux Container

4. **Container erstellen**

   ```
   lxc-create -t download -n my-container
   ```

5. **Distribution, Version und Architektur auswählen**
6. **Container starten (im Hintergrund)**

   ```
   lxc-start -n my-container -d
   ```

7. **In eine shell gelangen**

   ```
   lxc-attach -n my-container
   ```

8. **Container beenden**

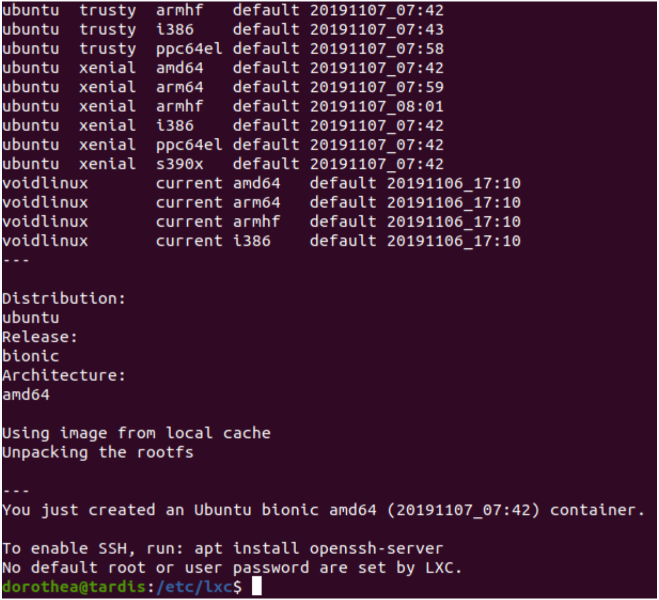
   ```
   lxc-stop -n my-container
   ```

9. **Status anzeigen**

   ```
   lxc-info -n my-container
   lxc-ls -f
   ```

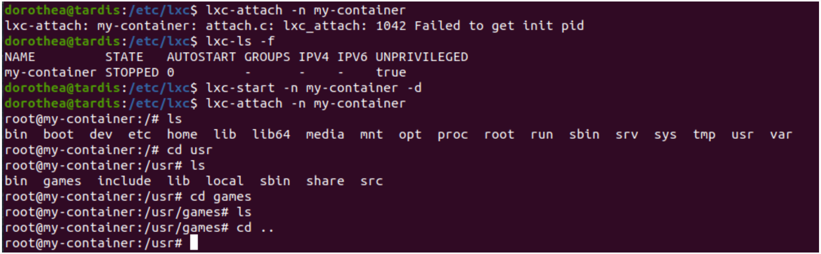
schnell und schmerzlos: https://linuxcontainers.org/lxd/try-it/

Ergebnis



```
ubuntu  trusty  armhf   default 20191107_07:42
ubuntu  trusty  i386    default 20191107_07:43
ubuntu  trusty  ppc64el default 20191107_07:58
ubuntu  xenial  amd64   default 20191107_07:42
ubuntu  xenial  arm64   default 20191107_07:59
ubuntu  xenial  armhf   default 20191107_08:01
ubuntu  xenial  i386    default 20191107_07:42
ubuntu  xenial  ppc64el default 20191107_07:42
ubuntu  xenial  s390x   default 20191107_07:42
voidlinux       current amd64   default 20191106_17:10
voidlinux       current arm64   default 20191106_17:10
voidlinux       current armhf   default 20191106_17:10
voidlinux       current i386    default 20191106_17:10
---

Distribution:
ubuntu
Release:
bionic
Architecture:
amd64

Using image from local cache
Unpacking the rootfs

---
You just created an Ubuntu bionic amd64 (20191107_07:42) container.

To enable SSH, run: apt install openssh-server
No default root or user password are set by LXC.
dorothea@tardis:/etc/lxc$ 
```

Der Container läuft



```
dorothea@tardis:/etc/lxc$ lxc-attach -n my-container
lxc-attach: my-container: attach.c: lxc_attach: 1042 Failed to get init pid
dorothea@tardis:/etc/lxc$ lxc-ls -f
NAME         STATE   AUTOSTART GROUPS IPV4 IPV6 UNPRIVILEGED
my-container STOPPED 0         -      -    -    true
dorothea@tardis:/etc/lxc$ lxc-start -n my-container -d
dorothea@tardis:/etc/lxc$ lxc-attach -n my-container
root@my-container:/# ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@my-container:/# cd usr
root@my-container:/usr# ls
bin  games  include  lib  local  sbin  share  src
root@my-container:/usr# cd games
root@my-container:/usr/games# ls
root@my-container:/usr/games# cd ..
root@my-container:/usr# 
```

Organisation

Struktur

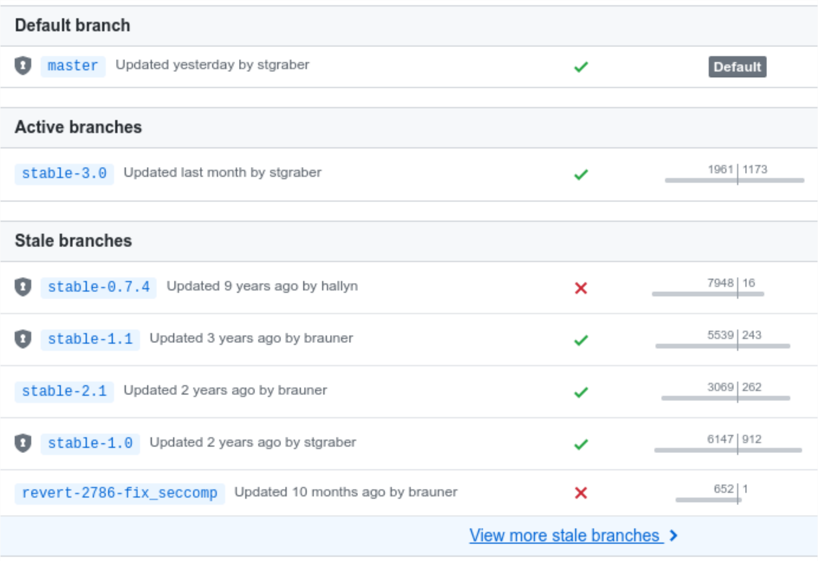
- Canonical
  - „The Company Behind Ubuntu“
  - wurde neben Ubuntu gegründet
  - startete das Projekt LXC
- Maintainer
  - Stéphane Graber
  - Serge Hallyn
  - Christian Brauner

Releases

Aktuell unterstützte Releases:

Version	Release Datum
LXC 2.0 LTS	April 2016
LXC 3.0 LTS	März 2018
LXC 3.2 (feature release)	Juli 2019

Branches



Default branch

🛡	master	Updated yesterday by stgraber	✓	Default

Active branches

	stable-3.0	Updated last month by stgraber	✓	1961 | 1173

Stale branches

🛡	stable-0.7.4	Updated 9 years ago by hallyn	✗	7948 | 16
🛡	stable-1.1	Updated 3 years ago by brauner	✓	5539 | 243
	stable-2.1	Updated 2 years ago by brauner	✓	3069 | 262
🛡	stable-1.0	Updated 2 years ago by stgraber	✓	6147 | 912
	revert-2786-fix_seccomp	Updated 10 months ago by brauner	✗	652 | 1

View more stale branches ›

Beitragen

Voraussetzungen

1. **Coding Style Regeln befolgen**
    - maximal 80 Zeichen in einer Zeile
    - Deklarationen von Variablen am Anfang eines Blocks
    - goto benutzen
    - unbenutzte return Werte zu void casten:

    ```
    (void)chowmod(fullpath, destuid, 0, 0664);
    ```

2. **Format muss GitHub pull request sein**
3. **es muss unterschrieben werden**

Kommunikation

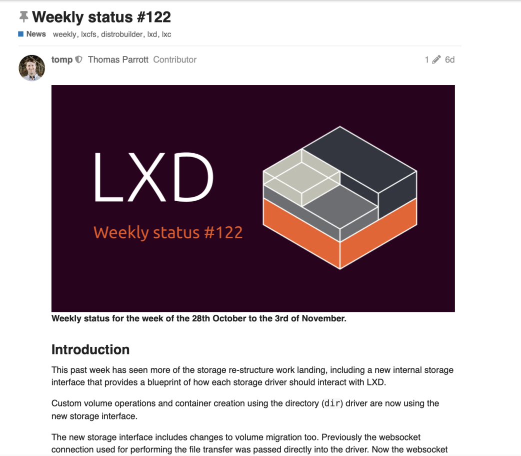
Support und Informationen

- Issues unter GitHub
- IRC Channel: #lxcontainers
- Mailing Liste
- Forum: https://discuss.linuxcontainers.org/
- Blog von Stéphane Graber: https://stgraber.org/2013/12/20/lxc-1-0-blog-post-series/

Entwicklung

- GitHub
- IRC Channel: #lxc-dev
- Mailing Liste

wöchentliche Neuigkeiten



Weekly status #122

News weekly, lxcfs, distrobuilder, lxd, lxc

tomp Thomas Parrott Contributor 1 6d

Weekly status for the week of the 28th October to the 3rd of November.

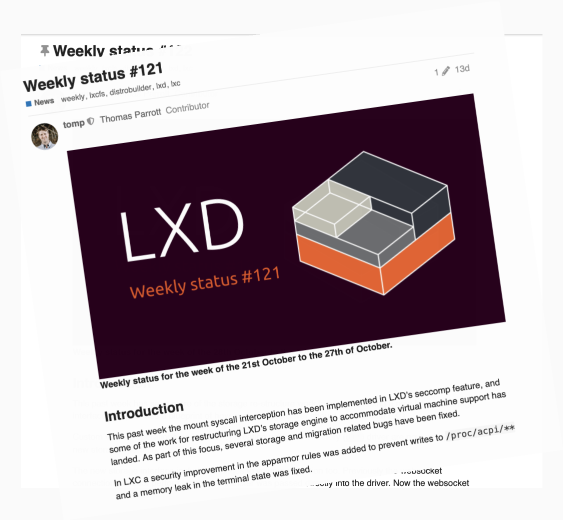
Introduction

This past week has seen more of the storage re-structure work landing, including a new internal storage interface that provides a blueprint of how each storage driver should interact with LXD.

Custom volume operations and container creation using the directory (`dir`) driver are now using the new storage interface.

The new storage interface includes changes to volume migration too. Previously the websocket connection used for performing the file transfer was passed directly into the driver. Now the websocket

wöchentliche Neuigkeiten



Weekly statu

Weekly status #121

News weekly, lxcfs, distrobuilder, lxd, lxc

1 13d

tomp Thomas Parrott Contributor

Weekly status for the week of the 21st October to the 27th of October.

Introduction

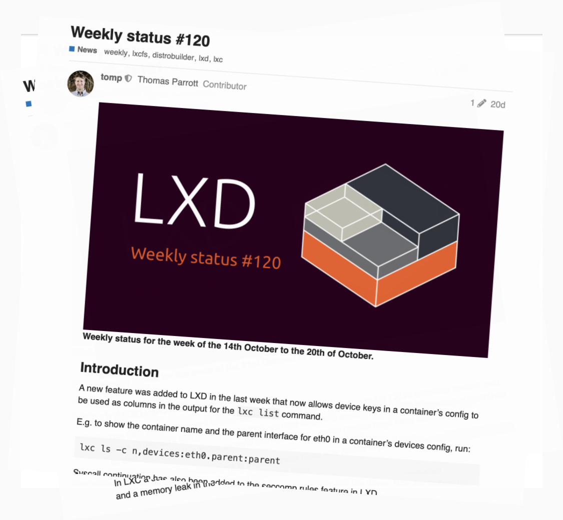
This past week the mount syscall interception has been implemented in LXD's seccomp feature, and some of the work for restructuring LXD's storage engine to accommodate virtual machine support has landed. As part of this focus, several storage and migration related bugs have been fixed.

In LXC a security improvement in the apparmor rules was added to prevent writes to `/proc/acpi/**` and a memory leak in the terminal state was fixed.

websocket

irectly into the driver. Now the websocket

wöchentliche Neuigkeiten



Weekly status #120

News weekly, lxcfs, distrobuilder, lxd, lxc

tomp Thomas Parrott Contributor

1 20d

Weekly status for the week of the 14th October to the 20th of October.

Introduction

A new feature was added to LXD in the last week that now allows device keys in a container's config to be used as columns in the output for the `lxc list` command.

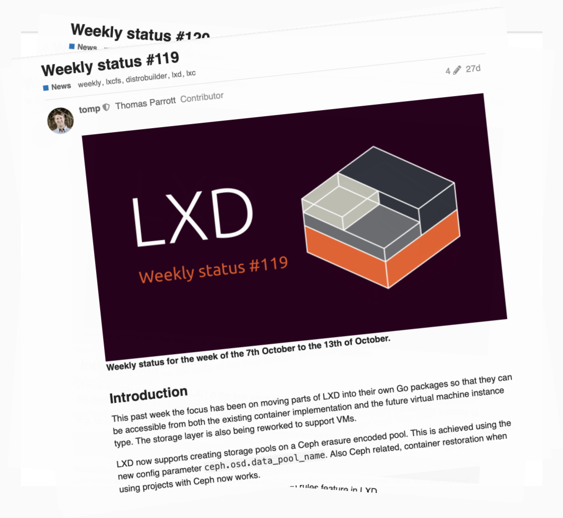
E.g. to show the container name and the parent interface for eth0 in a container's devices config, run:

```
lxc ls -c n,devices:eth0.parent:parent
```

Syscall continuation has also been added to the seccomp rules feature in LXD.

In LXC a
and a memory leak in the

wöchentliche Neuigkeiten



Weekly status #120

News

Weekly status #119

News weekly, lxcfs, distrobuilder, lxd, lxc

4 27d

tomp Thomas Parrott Contributor

Weekly status for the week of the 7th October to the 13th of October.

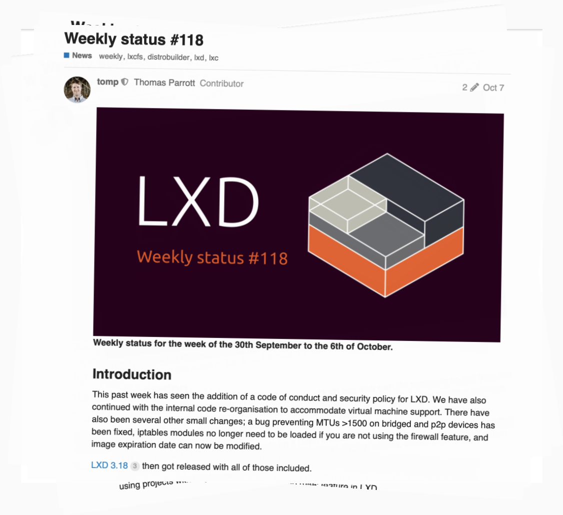
Introduction

This past week the focus has been on moving parts of LXD into their own Go packages so that they can be accessible from both the existing container implementation and the future virtual machine instance type. The storage layer is also being reworked to support VMs.

LXD now supports creating storage pools on a Ceph erasure encoded pool. This is achieved using the new config parameter `ceph.osd.data_pool_name`. Also Ceph related, container restoration when using projects with Ceph now works.

rules feature in LXD

wöchentliche Neuigkeiten



Weekly status #118

News weekly, lxcfs, distrobuilder, lxd, lxc

tomp Thomas Parrott Contributor 2 Oct 7

Weekly status for the week of the 30th September to the 6th of October.

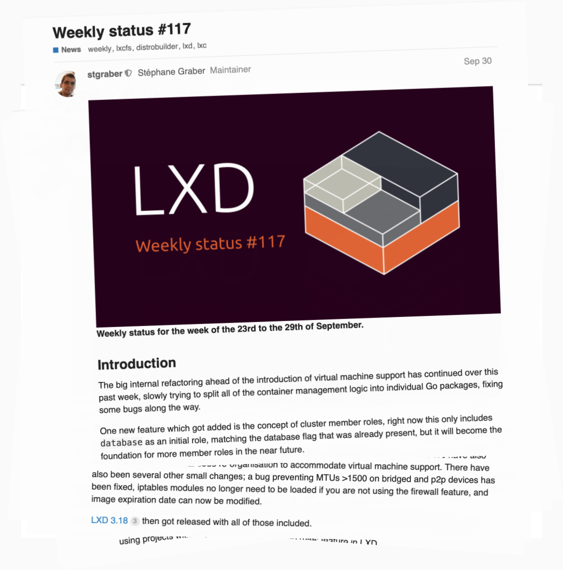
Introduction

This past week has seen the addition of a code of conduct and security policy for LXD. We have also continued with the internal code re-organisation to accommodate virtual machine support. There have also been several other small changes; a bug preventing MTUs >1500 on bridged and p2p devices has been fixed, iptables modules no longer need to be loaded if you are not using the firewall feature, and image expiration date can now be modified.

LXD 3.18 3 then got released with all of those included.

using projects wi… …feature in LXD

wöchentliche Neuigkeiten



Weekly status #117

News weekly, lxcfs, distrobuilder, lxd, lxc

stgraber Stéphane Graber Maintainer Sep 30

Weekly status for the week of the 23rd to the 29th of September.

Introduction

The big internal refactoring ahead of the introduction of virtual machine support has continued over this past week, slowly trying to split all of the container management logic into individual Go packages, fixing some bugs along the way.

One new feature which got added is the concept of cluster member roles, right now this only includes `database` as an initial role, matching the database flag that was already present, but it will become the foundation for more member roles in the near future.

… have also … organisation to accommodate virtual machine support. There have also been several other small changes; a bug preventing MTUs >1500 on bridged and p2p devices has been fixed, iptables modules no longer need to be loaded if you are not using the firewall feature, and image expiration date can now be modified.

LXD 3.18 3 then got released with all of those included.

using projects … feature in LXD

Ausblick

Docker

Docker Container

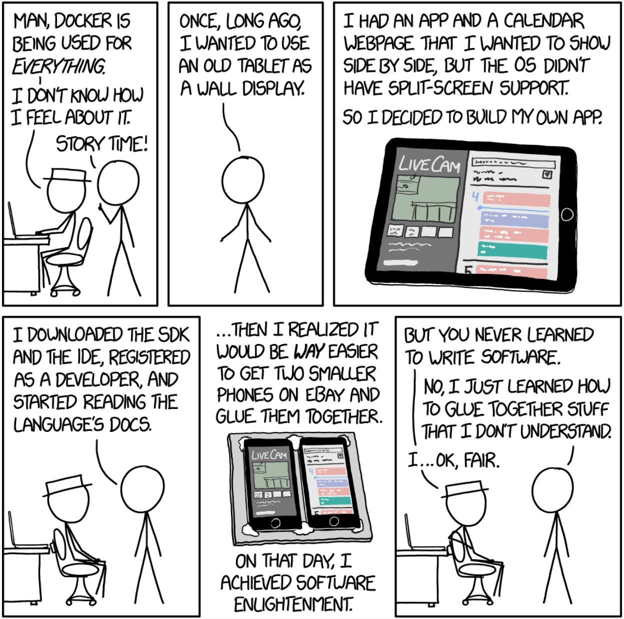
- Anwendungscontainer
- ursprünglich mit LXC implementiert, später durch libcontainers ersetzt
- höheres Sicherheitsrisiko als LXC, da Anwendungen mit root Rechten ausgeführt werden

LXD

LXD

- System Container der nächsten Generation
- benutzt intern LXC
- sichereres Design (unprivilegierte Container)
- kann über ein Netzwerk kontrolliert werden

Software Erleuchtung mit Containern


MAN, DOCKER IS BEING USED FOR *EVERYTHING*.
I DON'T KNOW HOW I FEEL ABOUT IT.
STORY TIME!
ONCE, LONG AGO, I WANTED TO USE AN OLD TABLET AS A WALL DISPLAY.
I HAD AN APP AND A CALENDAR WEBPAGE THAT I WANTED TO SHOW SIDE BY SIDE, BUT THE OS DIDN'T HAVE SPLIT-SCREEN SUPPORT.
SO I DECIDED TO BUILD MY OWN APP.
LIVE CAM
4
5
I DOWNLOADED THE SDK AND THE IDE, REGISTERED AS A DEVELOPER, AND STARTED READING THE LANGUAGE'S DOCS.
...THEN I REALIZED IT WOULD BE *WAY* EASIER TO GET TWO SMALLER PHONES ON EBAY AND GLUE THEM TOGETHER.
LIVE CAM
4
ON THAT DAY, I ACHIEVED SOFTWARE ENLIGHTENMENT.
BUT YOU NEVER LEARNED TO WRITE SOFTWARE.
NO, I JUST LEARNED HOW TO GLUE TOGETHER STUFF THAT I DON'T UNDERSTAND.
I... OK, FAIR.

Fragen?

Quellen (1)

https://www.kernel.org

https://www.net.in.tum.de/fileadmin/TUM/NET/NET-2016-07-1/NET-2016-07-1_01.pdf

https://itnext.io/chroot-cgroups-and-namespaces-an-overview-37124d995e3d

https://entwickler.de/online/besuch-im-docker-maschinenraum-126456.html

linux man pages

https://wiki.ubuntuusers.de/LXC/

https://help.ubuntu.com/lts/serverguide/lxc.html

https://linuxacademy.com/blog/containers/history-of-container-technology/

https://stackshare.io/stackups/lxc-vs-lxd

https://github.com/lxc/lxc

https://stgraber.org/2013/12/20/lxc-1-0-blog-post-series/

Quellen (2)

https://ubuntu.com/blog/tag/lxc
https://www.linux.com/tutorials/condensing-your-infrastructure-system-containers/
https://xkcd.com/1988/