

Virtualization with LXC

February 03, 2012

SUSE®

LXC is a lightweight “virtualization” method to run multiple virtual units (containers, akin to “chroot”) simultaneously on a single control host. Containers are sufficiently isolated with cgroups to guarantee the required security.

LXC provides an operating system-level virtualization where the *Kernel* controls the isolated containers. In contrast to full virtualization solutions like Xen, KVM, or `libvirt` the *processor* simulates a complete hardware environment and controls its virtual machines.

Terminology

cgroups

Kernel Control Groups (commonly referred to as just “cgroups”) are a Kernel feature that allows aggregating or partitioning tasks (processes) and all their children into hierarchical organized groups to isolate resources.

Container

A “virtual machine” on the host server that can run any Linux system, for example openSUSE, SUSE Linux Enterprise Desktop, or SUSE Linux Enterprise Server.

Container Name

A name that refers to a container. The name is used by the `lxc` commands.

Kernel Namespaces

A Kernel feature to isolate some resources like network, users, and others for a group of processes.

LXC Host Server

The system that contains the LXC system and provides the containers and management control capabilities through cgroups.

Overview

Conceptually, LXC can be seen as an improved *chroot* technique. The difference is that a chroot environment separates only the file system, whereas LXC goes further and provides resource management and control via cgroups.

Benefits of LXC

- Isolating applications and operating systems through containers.
- Providing nearly native performance as LXC manages allocation of resources in real-time.
- Controlling network interfaces and applying resources inside containers through cgroups.

Limitations of LXC

- All LXC containers are running inside host system's Kernel and not with a different Kernel.
- Only allows Linux “guest” operating systems.
- LXC is not a full virtualization stack like Xen, KVM, or `libvirt`.
- Security depends on the host system. LXC is not secure. If you need a secure system, use KVM.

Setting up an LXC Host

The LXC host provides the cgroups and controls all containers.

Procedure 1. Preparing an LXC Host

1. Install the following packages:
 - `lxc`
 - `bridge-utils`
2. If you want to access the virtual container's ethernet interface, create a network bridge. A network bridge allows to share the network link on the physical interface of the host (`eth0`):
 - a. Open YaST and go to *Network Device* → *Network Settings*.
 - b. Click *Add*.
 - c. Select *Bridge* as device type. Proceed with *Next*.
 - d. Activate *Dynamic Address* and select *DHCP*.
 - e. Choose your bridged device(s), usually `eth0`. Proceed with *Next*. Optionally check your devices with the `ifconfig` command. Close the *Network Settings* module.
3. If you have created a network bridge, assign its interface zone:
 - a. Start YaST and go to *Security & Users* → *Firewall*.
 - b. Open the *Interfaces* tab.
 - c. Select your bridge device (usually `br0`).
 - d. Click *Change...* and select *External Zone*. Proceed with *Ok*.
 - e. Finish with *Next*.

The LXC host is now prepared for setting up containers.

Setting up LXC Containers

A container is a “virtual machine” that can be started, stopped, frozen, or cloned (to name but a few tasks). To set up an LXC container, proceed as follows:

Procedure 2. Creating a Container

1. Create a configuration file with the container name in it (`/etc/lxc/CONTAINER/config`) and edit it according to the following example:

```
lxc.utsname = vps0
lxc.network.type = veth
lxc.network.flags = up
lxc.network.link = br0
lxc.network.hwaddr = 00:30:6E:08:EC:80
lxc.network.ipv4 = 192.168.1.10
lxc.network.name = eth0
```

- ❶ Container name, should also be used as part in your configuration file
- ❷ Type of network virtualization to be used for the container. The option `veth` defines a peer network device. It is created with one side assigned to the container and the other side is attached to a bridge by the `lxc.network.link` option.
- ❸ Network actions. The value `up` in this case activates the network.
- ❹ Host network interface to be used for the container.
- ❺ Allocated MAC address of the virtual interface. This MAC address needs to be unique in your network and different from the host MAC address.
- ❻ IPv4 address assigned to the virtualized interface. Use the address `0.0.0.0` to make use of DHCP. Use `lxc.network.ipv6` if you need IP6 support.

- ⑦ Dynamically allocated interface name. This option will rename the interface in the container.

More example files can be found in `/usr/share/doc/packages/lxc/examples/`. Find details about all options in the `lxc.conf` man page.

2. Create a container by using the configuration file from Step 1 (page 2):

- For an openSUSE 11.4 “guest”:

```
lxc-create -t opensuse -f lxc.conf -n CONTAINER
```

- For a SUSE Linux Enterprise Server 11 SP2 “guest”:

```
lxc-create -t sles -f lxc.conf -n CONTAINER
```

Downloading and installing the base packages for openSUSE or SUSE Linux Enterprise Server will take some time. The container will be created in `/var/lib/lxc/CONTAINER`.

3. Finalize the configuration of the container:

- a. Change the root path to the installed LXC container with the `chroot` command:

```
chroot /var/lib/lxc/CONTAINER_NAME/rootfs/
```

- b. Change the password for user `root` with `passwd root`.

- c. Create an `operator` user without `root` privileges:

```
useradd -m operator
```

- d. Change the operator's password:

```
passwd operator
```

- e. Leave the `chroot` environment with `exit`.

Procedure 3. Starting, Accessing, and Stopping Your Container

1. Start the container:

```
lxc-start -n CONTAINER_NAME
```

2. Connect to the container and log in:

```
lxc-console -n CONTAINER_NAME
```

3. Stop and remove your container *always* with the two steps:

```
lxc-stop -n CONTAINER_NAME  
lxc-destroy -n CONTAINER_NAME
```

Starting Containers at Boot Time

LXC containers can be started at boot time. However, you need to follow certain conventions. Every container has a subdirectory with its name in `/etc/lxc/`, for example, `/etc/lxc/my-sles`. This directory needs to be created once. There you place your configuration file (named `config`).

To set up the automatical start of LXC containers, proceed as follows:

1. Activate the cgroup service with `insserv boot.cgroup`. This has to be done only once to enable this service at boot time. The command will populate the `/sys/fs/cgroup` directory.
2. Create a directory `/etc/lxc/CONTAINER`.
3. Copy your configuration file to `/etc/lxc/CONTAINER/config`.
4. Run `/etc/init.d/boot.cgroup start` to set up cgroups properly.
5. Run `/etc/init.d/lxc start` to start your containers.
6. Wait a few seconds and run `/etc/init.d/lxc list` to print the state of all your containers.

After this procedure, your LXC containers are correctly configured and will start automatically the next time you boot your computer.

For More Information

LXC Home Page

<http://lxc.sourceforge.net>

Kernel Control Groups (cgroups)

http://www.suse.com/documentation/sles11/book_sle_tuning/data/cha_tuning_cgroups.html

Managing Virtual Machines with libvirt

http://www.suse.com/documentation/sles11/book_sles_kvm/data/part_managing_virtual.html

LXC Container Driver

<http://libvirt.org/drvlxc.html>

Legal Notice

Copyright© 2006–2012 Novell, Inc. and contributors. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or (at your option) version 1.3; with the Invariant Section being this copyright notice and license. A copy of the license version 1.2 is included in the section entitled “GNU Free Documentation License”.

For Novell trademarks, see the Novell Trademark and Service Mark list <http://www.novell.com/company/legal/trademarks/tmlist.html>. All other third party trademarks are the property of their respective owners. A trademark symbol (®, ™ etc.) denotes a Novell trademark; an asterisk (*) denotes a third party trademark.

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither Novell, Inc., SUSE LINUX Products GmbH, the authors, nor the translators shall be held liable for possible errors or the consequences thereof.