Namespaces

in Linux

Jens Låås, UU

Types of namespaces

- PID
  New process gets pid 1.
- NET
  Devices, protocol stacks, routing tables, firewall, sockets etc.
- Mount namespace
- UTC (hostname)
- IPC (System V IPC)

When?

- clone() systemcall.

cgroups

Control Groups provide a mechanism for aggregating/partitioning sets of
tasks, and all their future children, into hierarchical groups with specialized behaviour.

Resource control:

- CPU
- Memory
- IO

Together with namespaces makes containers.

CONFIG_CGROUPS
CONFIG_CGROUP_NS=y
CONFIG_CGROUP_FREEZER=y
CONFIG_CGROUP_DEVICE=y
CONFIG_CFQ_GROUP_IOSCHED=y
CONFIG_CPUSETS
CONFIG_UTS_NS=y
CONFIG_IPC_NS=y
CONFIG_USER_NS=y
CONFIG_PID_NS=y
CONFIG_NET_NS=y

CONFIG_VETH=m
CONFIG_SECURITY=y
CONFIG_SECURITY_NETWORK=y
CONFIG_SECURITY_FILE_CAPABILITIES=y

Usage

Container userspace toolshttp://lxc.sourceforge.net/mount -t cgroup

cgroup /dev/cgroup

modprobe veth

ip link add name veth0 type veth peer name veth1

ifconfig veth0 up

/opt/lxc/bin/lxc-destroy -n foo

/opt/lxc/bin/lxc-create -n foo -f /opt/lxc/etc/lxc/lxc-veth.conf

/opt/lxc/bin/lxc-start -n foo /bin/bash

/opt/lxc/bin/lxc-freeze -n foo

/opt/lxc/bin/lxc-unfreeze -n foo

/opt/lxc/bin/lxc-stop -n foo

Inside container

```
# IP-number for container:
ip a add 10.0.0.2/32 dev eth0
# Default route out of container:
ip r add 0/0 dev eth0
```

On host

```
# Route to container:
ip r add 10.0.0.2/32 dev veth0
```

lxc.conf (lxc 0.6.3)

```
lxc.utsname = beta
lxc.network.type = phys
lxc.network.flags = up
lxc.network.link = veth1
lxc.network.hwaddr = 4a:49:43:49:79:bf
lxc.network.ipv4 = 10.0.0.2/24
lxc.network.ipv6 = 2003:db8:1:0:214:1234:fe0b:3597
# max number of pts
lxc.pts = 256
# nr of ttys
lxc.tty = 6
lxc.cgroup.devices.deny = a
# allow pts devices (should be limited to the tty ones)
lxc.cgroup.devices.allow = c 136:* rw
# /dev/pts/ptmx
lxc.cgroup.devices.allow = c 5:2 rw
# /dev/null, zero, random, urandom
lxc.cgroup.devices.allow = c 1:3 rw
lxc.cgroup.devices.allow = c 1:5 rw
lxc.cgroup.devices.allow = c 1:8 rw
lxc.cgroup.devices.allow = c 1:9 rw
lxc.rootfs = /foo/bifrost-6.1-beta1
```

Why?

- Virtual router. Distribute ports or VLAN over multiple containers. Bifrost 6.1 supports booting in a (device-restricted) container.
- Application.
  Only files needed to run an application. Deploy in cluster.
- Application++
  Application files plus a minial OS-environment for admin from inside the container.
- OS.
  Complete separated OS-install with applications.
- Shared OS.
  Complete OS. Multiple containers use the same rootfs at the same time. Selected directories may be bind mounted for separation.

Problems

- lxc is currently a moving target.
- /sys/class/net is not namespace aware (yet).
- halt/reboot support missing kernel support.
- freeze so far only supports regular files.