LXC Classic Linux containers

CONCEPTS

Every tool is a separate lxc-* program taking -n <name>; there is no single lxc command in the classic tools. The lxc client of LXD/Incus is a different thing: see the Incus sheet.

Template download, busybox
↓ `lxc-create`
Container dir config + rootfs
↓ `lxc-start`
RUNNING init + namespaces
↓ `lxc-stop`
STOPPED still defined
↓ `lxc-destroy`
Gone dir removed

Unprivileged containers (run as your user, uid-mapped) need /etc/subuid, /etc/subgid and an /etc/lxc/lxc-usernet quota; root-run containers are privileged by default.

Containers live in /var/lib/lxc/<name>/ (root) or ~/.local/share/lxc/ (unprivileged); -P <path> points any tool at another store.

CREATE AND DESTROY

`lxc-create -n <name> -t download`
Create from the image server (asks distro)

`lxc-create -n <name> -t download -- -d debian -r bookworm -a amd64`
Create non-interactively

`lxc-create -n <name> -t download -f <config>`
Create with a config file

`lxc-create -n <name> -t download -B btrfs`
Pick the backing store

`lxc-destroy -n <name>`
Delete a stopped container

`lxc-destroy -n <name> -f`
Stop and delete in one go

RUN AND STOP

`lxc-start -n <name>`
Start in the background

RUN AND STOP (cont.)

`lxc-start -n <name> -F`
Start in foreground, see boot output

`lxc-start -n <name> -l DEBUG -o <file>`
Log a failing start

`lxc-stop -n <name>`
Request clean shutdown

`lxc-stop -n <name> -k`
Kill instead of shutting down

`lxc-stop -n <name> -t <sec>`
Hard-stop after a timeout

`lxc-stop -n <name> -r`
Reboot the container

`lxc-freeze -n <name>`
Pause all processes

`lxc-unfreeze -n <name>`
Resume a frozen container

INSPECT

`lxc-ls -f`
List containers, state, IPs

`lxc-ls --running`
Only running containers

`lxc-ls --stopped`
Only stopped containers

`lxc-ls -F NAME,STATE,IPV4`
Pick the fancy columns

`lxc-info -n <name>`
Show state, PID, IPs, usage

`lxc-info -n <name> -iH`
Only the IP addresses, plain

`lxc-info -n <name> -s`
Only the state

`lxc-info -n <name> -S`
Usage stats (CPU, memory, I/O)

`lxc-top`
Live per-container usage

GET INSIDE

`lxc-attach -n <name>`
Shell inside the container

`lxc-attach -n <name> -- <cmd>`
Run one command inside

`lxc-attach -n <name> --clear-env -- <cmd>`
Run with a clean environment

GET INSIDE (cont.)

`lxc-console -n <name>`
Attach to the container console

CLONES AND SNAPSHOTS

`lxc-snapshot -n <name>`
Create next snapshot (snap0, snap1...)

`lxc-snapshot -n <name> -L`
List snapshots

`lxc-snapshot -n <name> -r <snap>`
Restore a snapshot

`lxc-snapshot -n <name> -d <snap>`
Delete a snapshot

`lxc-copy -n <name> -N <new>`
Clone a container

`lxc-copy -n <name> -N <new> -s`
Clone as copy-on-write snapshot

`lxc-copy -n <name> -e`
Start a throwaway ephemeral copy

`lxc-copy -n <name> -N <new> -R`
Rename a container

CONFIG AND LIMITS

`lxc.start.auto = 1`
Start at host boot

`lxc.net.0.type = veth`
Virtual NIC type

`lxc.net.0.link = <bridge>`
Bridge to attach to

`lxc.net.0.flags = up`
Bring the NIC up

`lxc.mount.entry = <src> <dst> none bind 0 0`
Bind-mount a host dir

`lxc.cgroup2.memory.max = <bytes>`
Memory limit (cgroup v2)

FILES AND PATHS

`/var/lib/lxc/<name>/config`
Per-container config (root)

`/var/lib/lxc/<name>/rootfs`
Container filesystem on the host