Incus

System containers and VMs

CONCEPTS

Incus runs system containers (a full OS with init, like a light VM) and, with --vm, real QEMU virtual machines. Both use the same commands.

Remote images: image server
↓ incus launch
Image cached locally
↓ incus launch
Instance container or VM; profiles + devices
↓ incus publish

Profiles are reusable bundles of config and devices applied to instances. New instances get the default profile (root disk and NIC) unless you pass --profile.

images: is a remote name, debian/12 an image alias. Add other servers with incus remote add; incus remote list shows what is configured.

INSTANCES

incus launch *<image> <name>*
Create and start an instance

incus launch *<image> <name>* **--vm**
Launch as a virtual machine

incus init *<image> <name>*
Create without starting

incus list
List instances with state and IPs

incus start *<name>*
Start an instance

incus stop *<name>*
Stop an instance

incus restart *<name>*
Restart an instance

incus delete *<name>*
Delete a stopped instance

incus info *<name>*
Show status, network and snapshots

incus launch *<image> <name>* **-c limits.cpu=2 -c limits.memory=2GiB**
Launch with resource limits

ACCESS AND FILES

incus shell *<name>*
Login shell as root

incus exec *<name>* **--** *<cmd>*
Run a command inside

incus file push *<local> <name>/<path>*
Copy a file into an instance

incus file pull *<name>/<path> <local>*
Copy a file out

IMAGES

incus image list images: *<filter>*
Search a remote image server

incus publish *<name>* **--alias** *<alias>*
Turn an instance into an image

CONFIG AND PROFILES

incus config show *<name>*
Show instance config

incus config set *<name> <key>=<value>*
Set an option

incus config device add *<name> <dev>* **disk source=***<host>* **path=***<dir>*
Mount a host directory

incus config device add *<name> <dev>* **proxy listen=tcp:0.0.0.0:***<p>* **connect=tcp:127.0.0.1:***<p>*
Publish a port

SNAPSHOTS AND COPIES

incus snapshot create *<name> <snap>*
Create a snapshot

incus snapshot list *<name>*
List snapshots

incus snapshot restore *<name> <snap>*
Roll back to a snapshot

incus copy *<name> <new>*
Clone an instance