Multipass Ubuntu VMs on demand

START HERE

`multipass launch lts --name <name>`
Create a VM

`multipass transfer <file> <name>:.`
Copy a file in

`multipass list`
See your VMs

`multipass stop <name>`
Shut it down

`multipass shell <name>`
Log in

`multipass delete <name>`
Remove it

`multipass exec <name> -- <cmd>`
Run one command

`multipass purge`
Free the disk space

Image Ubuntu release → `launch` **Running** instance → `stop` **Stopped** snapshots here → `delete` **Deleted** recoverable → `purge` **Gone** disk freed

CONCEPTS

Multipass runs Ubuntu VMs on your host. Launch, use and throw them away in seconds.

Deleted instances are only hidden until you purge them; recover brings them back.

The instance named primary is the default for shell, exec and stop when no name is given.

CREATE AND INSPECT

`multipass launch [<image>] --name <name>`
Create and start an instance

`multipass launch --cpus <n> --memory <size> --disk <size>`
Set the VM resources
Defaults are 1 CPU, 1G memory and 5G disk.

`multipass images`
List available images

`multipass list`
List instances and their IPs

`multipass info <name>`
Show details of an instance

`multipass list --format json`
Machine-readable list

`multipass info --all`
Details of every instance

`multipass launch --cloud-init <file>`
Configure at first boot

`multipass launch --mount <host-dir>:<vm-dir>`
Mount a folder at launch

`multipass launch --network <spec>`
Add a network interface

`multipass launch --bridged`
Add a bridged network

`multipass launch --timeout <seconds>`
Allow a slow start

`multipass networks`
List host networks

`multipass clone <name> --name <new>`
Copy a stopped instance

ACCESS

`multipass shell [<name>]`
Open a shell in the instance

`multipass exec <name> -- <cmd>`
Run a command in the instance

`multipass exec <name> --working-directory <dir> -- <cmd>`
Run in a given directory

`multipass exec <name> -- <cmd> | grep <text>`
Pipe the output to the host

`multipass alias <name>:<cmd> [<alias>]`
Run a VM command from the host

`multipass aliases`
List defined aliases

`multipass unalias <alias>`
Remove an alias

LIFECYCLE

`multipass stop <name>`
Stop an instance

`multipass start <name>`
Start a stopped instance

`multipass restart <name>`
Restart an instance

`multipass suspend <name>`
Suspend, keeping memory state

`multipass stop --all`
Stop every instance

`multipass stop --force <name>`
Force off immediately
Can corrupt the instance; needs Multipass 1.14 or newer.

`multipass delete <name>`
Delete (recoverable)

`multipass recover <name>`
Bring back a deleted instance

`multipass purge`
Permanently remove deleted instances

`multipass delete --purge <name>`
Delete and purge at once

FILES

`multipass transfer <file> <name>:<path>`
Copy host file to instance

FILES (cont.)

`multipass transfer <name>:<path> <dest>`
Copy instance file to host

`multipass transfer --recursive <src> <name>:<path>`
Copy a directory tree

`multipass transfer --parents <file> <name>:<path>`
Create missing target dirs

`multipass mount <host-dir> <name>:<vm-dir>`
Share a host folder

`multipass mount --type native <host-dir> <name>:<vm-dir>`
Use a native mount

`multipass unmount <name>[:<path>]`
Remove a mount

SNAPSHOTS

`multipass snapshot <name> --name <snap>`
Snapshot a stopped instance

`multipass snapshot <name> --comment "<text>"`
Add a note to the snapshot

`multipass snapshots`
List snapshots

`multipass restore <name>.<snap>`
Roll back to a snapshot

`multipass restore --destructive <name>.<snap>`
Roll back without saving state

`multipass info --snapshots <name>`
Snapshot details

`multipass delete <name>.<snap>`
Delete a snapshot for good

SETTINGS

`multipass get --keys`
List all setting keys

`multipass get <key>`
Read a setting

`multipass set client.primary-name=<name>`
Rename the primary instance

SETTINGS (cont.)

`multipass set local.driver=<driver>`
Choose the hypervisor

`multipass set local.<name>.cpus=<n>`
Resize a stopped instance's CPUs

`multipass set local.<name>.memory=<size>`
Resize its memory

`multipass set local.<name>.disk=<size>`
Grow its disk

`multipass set local.privileged-mounts=true`
Allow mounts as host root

`multipass version`
Show client and daemon versions

`multipass help <command>`
Help for one command

RECIPES

Throwaway test box

```
multipass launch 24.04 --name scratch
multipass exec scratch -- sudo apt-get install -y <pkg>
multipass delete --purge scratch
```

Share project, work inside

```
multipass launch --name dev --cpus 2 --memory 4G
multipass mount ~/<project> dev:/home/ubuntu/<project>
multipass shell dev
```

Checkpoint before risky change

```
multipass stop dev
multipass snapshot dev --name before-upgrade
multipass start dev
# if it goes wrong
multipass stop dev && multipass restore dev.before-upgrade
```

Provision with cloud-init

```
# cloud.yaml starts with #cloud-config
multipass launch --name web --cloud-init cloud.yaml
multipass exec web -- cloud-init status --wait
```