pactl Sound server control

START HERE

`pactl info`
Server and default devices

`pactl list short sinks`
Output devices

`pactl list short sources`
Inputs and monitors

`pactl set-default-sink <sink>`
Pick the output

`pactl set-sink-volume @DEFAULT_SINK@ 50%`
Set volume

`pactl set-sink-mute @DEFAULT_SINK@ toggle`
Mute / unmute

`pactl list short sink-inputs`
Playing streams

`pactl move-sink-input <id> <sink>`
Move stream to a device

Card hardware, profile → **has** → **Sink** output, port → **fed by** → **Sink input** app playback stream → **monitor** → **Source** input or sink monitor

CONCEPTS

pactl talks to a PulseAudio server. On PipeWire the pipewire-pulse service answers the same protocol, so these commands work unchanged.

Names or numeric indexes both work for devices; indexes change after restarts, names do not. @DEFAULT_SINK@ and @DEFAULT_SOURCE@ avoid looking names up.

Add -f json (or --format=json) for machine-readable output, or use list short for tab-separated columns.

INSPECT

`pactl info`
Server, version, default sink/source

`pactl list short sinks`
Output devices, one per line

`pactl list short sources`
Input devices and monitors

`pactl list short cards`
Sound cards

`pactl list short sink-inputs`
Playback streams

INSPECT (cont.)

`pactl list short source-outputs`
Recording streams

`pactl list sinks`
Full detail: ports, volume, state

`pactl list short modules`
Loaded modules

`pactl list short clients`
Connected clients

`pactl -f json list sinks`
Machine-readable output

DEFAULT DEVICE

`pactl get-default-sink`
Print default output

`pactl set-default-sink <sink>`
Change default output

`pactl get-default-source`
Print default input

`pactl set-default-source <source>`
Change default input

`pactl set-default-sink @NONE@`
Back to priority-based choice

`pactl set-sink-port <sink> <port>`
Switch speakers / headphones

`pactl set-source-port <source> <port>`
Switch mic port

VOLUME AND MUTE

`pactl get-sink-volume <sink>`
Show volume

`pactl set-sink-volume <sink> 50%`
Absolute volume

`pactl set-sink-volume @DEFAULT_SINK@ +5%`
Raise by 5 points

`pactl set-sink-volume @DEFAULT_SINK@ -5%`
Lower by 5 points

`pactl set-sink-volume <sink> 0dB`
Volume in decibels

`pactl set-sink-mute <sink> toggle`
Toggle mute

`pactl set-sink-mute <sink> 1`
Mute (0 = unmute)

`pactl get-sink-mute <sink>`
Show mute state

`pactl set-source-volume <source> 80%`
Mic level

`pactl set-source-mute <source> toggle`
Mute the mic

`pactl get-source-volume <source>`
Show mic level

VOLUME AND MUTE (cont.)

`pactl set-sink-volume <sink> 40% 60%`
Per channel (left, right)

Give as many values as the sink has channels.

Values above 100% amplify and may distort; relative +/- steps apply to the current volume.

STREAMS

`pactl move-sink-input <id> <sink>`
Send a stream to another output

`pactl set-sink-input-volume <id> 30%`
Per-app volume

`pactl set-sink-input-mute <id> toggle`
Mute one app

`pactl move-source-output <id> <source>`
Record from another input

`pactl set-source-output-volume <id> 50%`
Per-recording volume

CARDS AND DEVICES

`pactl set-card-profile <card> <profile>`
Change profile (e.g. HDMI, A2DP)

CARDS AND DEVICES (cont.)

`pactl suspend-sink <sink> true`
Suspend output (false resumes)

MODULES

`pactl load-module <name> [args]`
Load; prints module index

`pactl unload-module <id or name>`
Unload by index or name

`pactl load-module module-null-sink sink_name=<name>`
Virtual output

`pactl load-module module-loopback source=<src> sink=<sink>`
Route input to output

`pactl list short modules | grep <name>`
Find a module's index

EVENTS AND SAMPLES

`pactl subscribe`
Stream server events, runs until stopped

`pactl play-sample <name> [sink]`
Play cached sample

OPTIONS

`pactl -s <server> <command>`
Use another server

`pactl -f json <command>`
JSON output (text is default)

`pactl --version`
Print client version

RECIPES

Volume keys script

```
pactl set-sink-volume
@DEFAULT_SINK@ +5%
pactl set-sink-mute
@DEFAULT_SINK@ toggle
pactl get-sink-volume
@DEFAULT_SINK@
```

Move every stream to one sink

```
for i in $(pactl list short
sink-inputs | cut -f1); do
  pactl move-sink-input "$i"
<sink>
done
```

Loopback, then remove it

```
id=$(pactl load-module module-
loopback latency_msec=50)
# listen to the mic through the
speakers
pactl unload-module "$id"
```

Virtual sink for capturing audio

```
pactl load-module module-null-
sink sink_name=capture
# play to "capture", record
from capture.monitor
pactl list short sources | grep
capture
```