systemd Services and logs

CONCEPTS

A unit is anything systemd manages: a service, timer, socket, target or mount. The .service suffix is optional on the command line.

Unit file /etc/systemd/system
↓ daemon-reload
Loaded known to systemd
↓ start
Active running
↓ enable
Boot target starts at boot

enable only controls start at boot; start only controls now. Use enable --now for both.

CONTROL SERVICES

systemctl start *<unit>*
Start a unit now

systemctl stop *<unit>*
Stop a unit

systemctl restart *<unit>*
Stop and start again

systemctl reload *<unit>*
Re-read config without restart

systemctl reset-failed *<unit>*
Clear the failed state

systemctl --user *<command>*
Manage your own user units

INSPECT

systemctl status *<unit>*
State, PID and last log lines

systemctl is-active *<unit>*
Print active state, exit code reflects it

systemctl is-enabled *<unit>*
Will it start at boot?

systemctl list-units --type=service
Loaded services

systemctl --failed
List failed units

systemctl list-unit-files
All installed unit files with state

INSPECT (cont.)

systemctl cat *<unit>*
Show unit file plus drop-ins

systemctl list-dependencies *<unit>*
Tree of required units

BOOT AND UNIT FILES

systemctl enable *<unit>*
Start at boot

systemctl disable *<unit>*
Do not start at boot

systemctl enable --now *<unit>*
Enable and start in one go

systemctl disable --now *<unit>*
Disable and stop in one go

systemctl mask *<unit>*
Block a unit from starting at all

systemctl daemon-reload
Re-read changed unit files

systemctl edit *<unit>*
Override settings in a drop-in

systemctl reboot
Restart the machine

LOGS (JOURNALCTL)

journalctl -u *<unit>*
Logs of one unit

journalctl -u *<unit>* **-f**
Follow new lines

journalctl -n 100
Last 100 entries

journalctl -b
Current boot only

journalctl -b -1
Previous boot

journalctl -p err
Errors and worse

journalctl --since "*<date>***" --until "***<date>***"**
Time window

journalctl -g *<regex>*
Filter messages by pattern

journalctl -k
Kernel messages

UNIT FILE BASICS

Minimal service

```
# /etc/systemd/system/
myapp.service
[Unit]
Description=My app
After=network-online.target
Wants=network-online.target

[Service]
ExecStart=/usr/local/bin/myapp --
serve
Restart=on-failure
User=myapp

[Install]
WantedBy=multi-user.target
```

Common directives

Directive	Meaning
`ExecStart=`	Command to run (absolute path)
`Restart=`	on-failure, always, no
`After= / Before=`	Ordering only
`Wants= / Requires=`	Pull in other units (weak / hard)
`WantedBy=`	Target that enables it
`Environment=`	Set VAR=value
`WorkingDirectory=`	Start directory
`Type=`	simple, exec, forking, oneshot, notify

systemd-analyze verify *<file>*
Lint a unit file

TIMERS AND ONE-OFFS

systemctl list-timers
Timers and next/last run

systemd-analyze calendar "*<expr>***"**
Check an OnCalendar expression

systemd-run --user --on-active=*<time>* *<cmd>*
Run a command later, once

Enable the .timer, not the .service. A timer starts the same-named service unless Unit= says otherwise.

BOOT ANALYSIS

systemd-analyze
Total boot time

systemd-analyze blame
Slowest units at boot