

# RIFERIMENTO RAPIDO SYSTEMD

Gestione dei servizi, unità, timer e journalctl

## Gestione dei servizi

### Comandi base per i servizi

```
systemctl start nginx
systemctl stop nginx
systemctl restart nginx
systemctl reload nginx # reload config
systemctl status nginx
```

### Abilitazione e disabilitazione

```
systemctl enable nginx # start at boot
systemctl disable nginx # remove from boot
systemctl enable --now nginx # enable + start
systemctl is-enabled nginx
```

### Stati del servizio

**active (running)** Il servizio è in esecuzione normalmente  
**active (exited)** Eseguito una volta e terminato correttamente  
**inactive (dead)** Il servizio è fermo  
**failed** Il servizio è andato in crash o terminato con errore

**activating** Il servizio si sta avviando

## File unit

### Posizione dei file unit

**/etc/systemd/system/** Unità create dall'amministratore (massima priorità)  
**/run/systemd/system/** Unità generate a runtime  
**/usr/lib/systemd/system/** Unità installate dai pacchetti  
**~/.config/systemd/user/** Unità a livello utente

### Unità di servizio base

```
[Unit]
Description=My Application
After=network.target
[Service]
ExecStart=/usr/bin/myapp --config /etc/myapp.conf
Restart=on-failure
User=appuser
[Install]
WantedBy=multi-user.target
```

### Applicazione delle modifiche

```
systemctl daemon-reload # reload unit files
systemctl restart myapp # apply changes
```

## Timer

### Unità timer

```
[Unit]
Description=Run backup daily
[Timer]
OnCalendar=*-*-* 02:00:00
Persistent=true
[Install]
WantedBy=timers.target
```

### Sintassi di OnCalendar

**\*-\*-\* 02:00:00** Ogni giorno alle 2:00  
**Mon \*-\*-\* 09:00:00** Ogni lunedì alle 9:00  
**\*-\*-\* 01 00:00:00** Primo giorno di ogni mese  
**hourly / daily / weekly** Pianificazioni abbreviate

### Gestione dei timer

```
systemctl list-timers --all
systemctl start backup.timer
systemctl enable backup.timer
systemd-analyze calendar "daily"
```

## Target

### Target comuni

**multi-user.target** Avvio normale, multi-utente, senza GUI  
**graphical.target** Desktop grafico completo  
**rescue.target** Modalità ripristino mono-utente  
**emergency.target** Shell minimale, solo root  
**network-online.target** Rete completamente configurata  
**timers.target** Tutte le unità timer pronte

### Comandi per i target

```
systemctl get-default
systemctl set-default multi-user.target
systemctl isolate rescue.target
systemctl list-dependencies graphical.target
```

## Journalctl

### Visualizzazione dei log

```
journalctl -u nginx # logs for unit
journalctl -u nginx -f # follow (tail)
journalctl -u nginx --no-pager
journalctl -b # current boot only
```

### Filtraggio dei log

```
journalctl --since "2026-03-01"
journalctl --since "1 hour ago"
journalctl -p err # errors and above
journalctl _PID=1234
```

### Livelli di priorità

**emerg (0)** Il sistema non è utilizzabile  
**alert (1)** Azione immediata richiesta  
**crit (2)** Condizione critica  
**err (3)** Condizione di errore  
**warning (4)** Condizione di avviso  
**info (6)** Informativo  
**debug (7)** Messaggi di debug

### Manutenzione dei log

```
journalctl --disk-usage
journalctl --vacuum-size=500M
journalctl --vacuum-time=30d
```

## Rete

### networkctl

```
networkctl list
networkctl status eth0
networkctl up eth0
networkctl down eth0
```

### systemd-resolve

```
resolvectl status
resolvectl query example.com
resolvectl flush-caches
resolvectl statistics
```

### Attesa di rete

```
# In unit file [Unit] section:
After=network-online.target
Wants=network-online.target
```

## Mount

### Unità mount

```
[Unit]
Description=Mount data volume
[Mount]
What=/dev/sdb1
Where=/mnt/data
Type=ext4
Options=defaults,noatime
[Install]
WantedBy=multi-user.target
```

### Unità automount

```
[Unit]
Description=Automount data on access
[Automount]
Where=/mnt/data
TimeoutIdleSec=300
[Install]
WantedBy=multi-user.target
```

### Convenzione di denominazione

**/mnt/data** File unit: `mnt-data.mount`

**/var/lib/app** File unit: `var-lib-app.mount`

Il percorso di mount con `/` sostituito da ``, trattino iniziale rimosso

## Variabili d'ambiente

### Impostazione delle variabili d'ambiente

```
[Service]
Environment=APP_ENV=production
Environment=PORT=8080
EnvironmentFile=/etc/myapp/env
```

### Formato del file ambiente

```
# /etc/myapp/env
APP_ENV=production
DATABASE_URL=postgres://localhost/db
SECRET_KEY=changeme
```

### Hardening del servizio

**ProtectSystem=strict** File system in sola lettura  
tranne i percorsi consentiti  
**ProtectHome=true** Nasconde /home, /root, /run/  
user  
**NoNewPrivileges=true** Impedisce l'escalation dei  
privilegi  
**PrivateTmp=true** /tmp isolata per il servizio  
**ReadWritePaths=/var/lib/myapp** Consente le scritture in  
percorsi specifici

## Dipendenze

### Direttive di ordinamento e requisiti

**After=b.service** Avvia dopo b (solo ordinamento)  
**Before=b.service** Avvia prima di b (solo ordinamento)  
**Requires=b.service** Dipendenza rigida; fallisce se b fallisce  
**Wants=b.service** Dipendenza morbida; non fallisce se b  
fallisce  
**BindsTo=b.service** Si ferma quando b si ferma  
**Conflicts=b.service** Non può essere eseguito  
contemporaneamente a b

### Ispezione delle dipendenze

```
systemctl list-dependencies nginx
systemctl list-dependencies --reverse nginx
systemd-analyze dot nginx.service | dot -Tsvg > deps.svg
```

## Pattern comuni

### Policy di riavvio

**Restart=no** Non riavviare mai (default)  
**Restart=on-failure** Riavvia in caso di uscita non zero  
**Restart=always** Riavvia sempre (per i daemon)  
**RestartSec=5** Attendi 5 secondi prima di riavviare  
**StartLimitBurst=3** Numero massimo di riavvii  
nell'intervallo  
**StartLimitIntervalSec=60** Intervallo per il conteggio dei burst

### Override senza modifica

```
systemctl edit nginx # creates drop-in
# /etc/systemd/system/nginx.service.d/override.conf
systemctl cat nginx # show effective config
systemctl revert nginx # remove overrides
```

### Analisi del sistema

```
systemd-analyze # boot time
systemd-analyze blame # per-unit time
systemd-analyze critical-chain
systemctl list-units --failed
```