

systemd è un gestore di sistema e di servizi per Linux, compatibile con gli initscript SysV e LSB. systemd fornisce una notevole capacità di parallelizzazione, usa socket e D-Bus per l'avvio dei demoni, offre un avvio su richiesta dei demoni, tiene traccia dei processi con l'utilizzo del control groups di Linux, supporta lo snapshotting e il restore dello stato del sistema, mantiene i punti di mount e di automount e implementa un elaborato servizio di controllo logico basato sulle relazioni delle dipendenze.

I file delle unità disponibili possono essere visti in /usr/lib/systemd/system/ e /etc/systemd/system/ (questi ultimi hanno la precedenza sui primi).

Nella posizione /lib/systemd/system troviamo i servizi installati dai gestori dei pacchetti. Meglio non toccare questi files.

Si possono vedere le unità installate con: `systemctl list-unit-files`

Le unità possono essere, per esempio,

- servizi (.service)
- punti di montaggio (.mount)
- dispositivi (.device)
- oppure i sockets (.socket).

elencare i servizi disponibili

```
# systemctl list-unit-files --type=service
```

esempio : `systemctl list-unit-files --type=service`

elencare i socket disponibili

```
# systemctl list-unit-files --type=socket
```

```
systemctl list-unit-files --type=socket
```

elencare i servizi attivi

```
# systemctl
```

esempio : `systemctl`

elencare i servizi in errore in fase di avvio

```
# systemctl -failed
```

esempio : `systemctl -failed`

Attivare immediatamente una unità:

```
# systemctl start <unit>
```

esempio : `systemctl start sshd.service`

Fermare immediatamente una unità:

```
# systemctl stop <unit>
```

esempio : `systemctl stop sshd.service`

Far ripartire una unità:

```
# systemctl restart <unit>
```

esempio `systemctl restart sshd.service`

Chiedere ad una unità di ricaricare la sua configurazione:

```
# systemctl reload <unit>
esempio : systemctl reload sshd.service
```

Mostrare lo stato di una unità, compreso se sta funzionando o no:

```
# systemctl status <unit>
esempio : systemctl status sshd.service
```

Uccidere un servizio

```
# systemctl kill nome.servizio
esempio : systemct kill wpa_supplicant.service
```

Killare con un diverso segnale (letteralmente il processo non può sfuggire nemmeno con double fork o fork bombing):

```
# systemctl -kill -s SIGKILL nome.servizio
esempio : systemctl kill -s SIGKILL wpa_supplicant
```

Controllare se una unità è già attivata o no:

```
# systemctl is-enabled <unit>
esempio : systemctl is-enabled sshd.service
```

Attivare l'avvio automatico al boot:

```
# systemctl enable <unit>
esempio : systemctl enable sshd.service
```

disattivare l'avvio automatico al boot:

```
# systemctl disable <unit>
esempio : systemctl disable sshd.service
```

ricaricare le configurazioni di systemctl

```
# systemctl daemon-reload
esempio : systemct daemon-reload
```