

systemd

erste init-Skripte

Durch Konfigurationsdateien ersetzt. Siehe <http://0pointer.de/blog/projects/the-new-configuration-files> und <http://fedoraproject.org/wiki/Systemd>

Units

3 Ebenen der Konfiguration von Units:

1. `/etc/systemd/system/...` ← (höchste Priorität) Hier dürfen sich Admins austoben
2. `/run/systemd/...` ← dynamisch aus systemd
3. `{/usr}/lib/systemd/system/...`: aus der Paketverwaltung, nicht anfassen.
4. `/usr/lib/systemd/system-preset`: aus der Paketverwaltung, nicht anfassen.

systemd ist aber auch weiterhin kompatibel zu den aus [sysVinit](#) bekannten Startskripten in

- `/etc/init.d/`

Systemd Architektur

- `systemd-hostnamed` -> `hostnamedctl`
- `systemd-machined` -> `machinedctl`
- `systemd-journald` -> `journaldctl`
- `systemd-localed` -> `localedctl`
- `systemd-logind` -> `logindctl`
- `systemd-shutdown` -> `shutdown`
- `systemd-timedated` -> `timedatectl`
- Systemd Unit Arten
 - `service`: von Systemd gestartete Anwendungen
 - `socket`: Socket Aktivierung
 - `scopes`: ausserhalb von systemd gestartete Anwendungen
 - `slice`: Ressourcen für Prozessgruppen
 - `path`: Pfad Aktivierung
 - `mount`: Mountpoints (teilweise aus `/etc/fstab`)
 - `automount`: automatische Mountpoints
 - `target`: Gruppen von Units

Standard Target

Das Standard-Boot-Target ändern

```
systemctl set-default graphical.target
```

Das Standard-Boot-Target anzeigen

```
systemctl get-default
```

oder

```
readlink /etc/systemd/system/default.target
```

Dienste starten und eintragen

Dienst sofort, einmalig starten

systemd nativ

```
systemctl start dienst.service
```

sysVinit kompatibel

```
service dienst start
```

Dienst immer beim Booten starten

systemd nativ

```
systemctl enable dienst.service  
systemctl reenable dienst.service
```

sysVinit kompatibel

```
chkconfig dienst on
```

Dienst sofort, einmalig beenden

systemd nativ

```
systemctl stop dienst.service
```

sysVinit kompatibel

```
service dienst stop
```

Dienst restart

systemd nativ

```
systemctl restart dienst.service
```

Nur starten, wenn der Dienst bereits aktiv ist

```
systemctl try-restart dienst.service
```

sysVinit kompatibel

```
service dienst restart
```

Dienst neu laden

systemd nativ

```
systemctl reload dienst.service  
systemctl reload-or-restart dienst.service  
systemctl reload-or-try-restart dienst.service
```

sysVinit kompatibel

```
service dienst reload
```

Dienst beim Booten nicht starten

systemd nativ

```
systemctl disable dienst.service
```

sysVinit kompatibel

```
chkconfig dienst off
```

Dienst maskieren

dann kann er überhaupt nicht mehr gestartet werden

```
systemctl mask dienst.service
systemctl unmask dienst.service
```

Dienste anzeigen

systemd nativ

```
systemctl list-units -t service
systemctl list-units --type service --state=failed --all
systemctl list-unit-files -t service
systemctl is-enable service
systemctl is-active service
```

sysVinit kompatibel

```
service --status-all
```

Targets

auflisten

```
systemctl --type=target
```

isolieren

alle anderen Targets beenden

```
systemctl isolate rescue.target
systemctl rescue
systemctl emergency
systemctl single
systemctl halt
systemctl poweroff
systemctl reboot
systemctl suspend
systemctl hibernate
systemctl hybrid-sleep
```

Startoptionen eines Dienstes ändern

```
cp -a /lib/systemd/system/dienst.service /etc/systemd/system/  
vim /etc/systemd/system/dienst.service
```

Änderungen anzeigen:

```
systemd-delta
```

Änderungen aktivieren:

```
systemctl daemon-reload
```

Dienst neu starten:

```
systemctl restart dienst
```

Journald

- [journald](#)

Logind

systemd version 230: Now you can no longer expect a long running background processes to continue after logging out. I believe this breaks the expectations of many users. For example, you can no longer start a [screen](#) or [tmux](#) session, log out, and expect to come back to it. compiling systemd with `-without-kill-user-processes` or by setting

[/etc/systemd/logind.conf](#)

```
KillUserProcesses=no
```

Weitere Befehle und Subsysteme

- [Analyse](#)
- [Snapshots](#)
- [Systemd-Prozesse](#)
- [Systemd-Ressourcenkontrolle](#)

Powersave verhindern

```
systemd-inhibit <cmd>
```

Virtualisierung erkennen

systemd-detect-virt

*.ctl

- [loginctl](#)
- [localectl](#)
- [hostnamectl](#)
- [timedatectl](#)

Beispiele

- [Systemd Service Unit Files](#)

Doku

- [CRE Interview mit Lennart](#)

From:

<https://wiki.lab.linuxhotel.de/> - **Linuxhotel Wiki**

Permanent link:

<https://wiki.lab.linuxhotel.de/doku.php/lpi1:systemd?rev=1487178029>

Last update: **2017/05/10 06:38**

