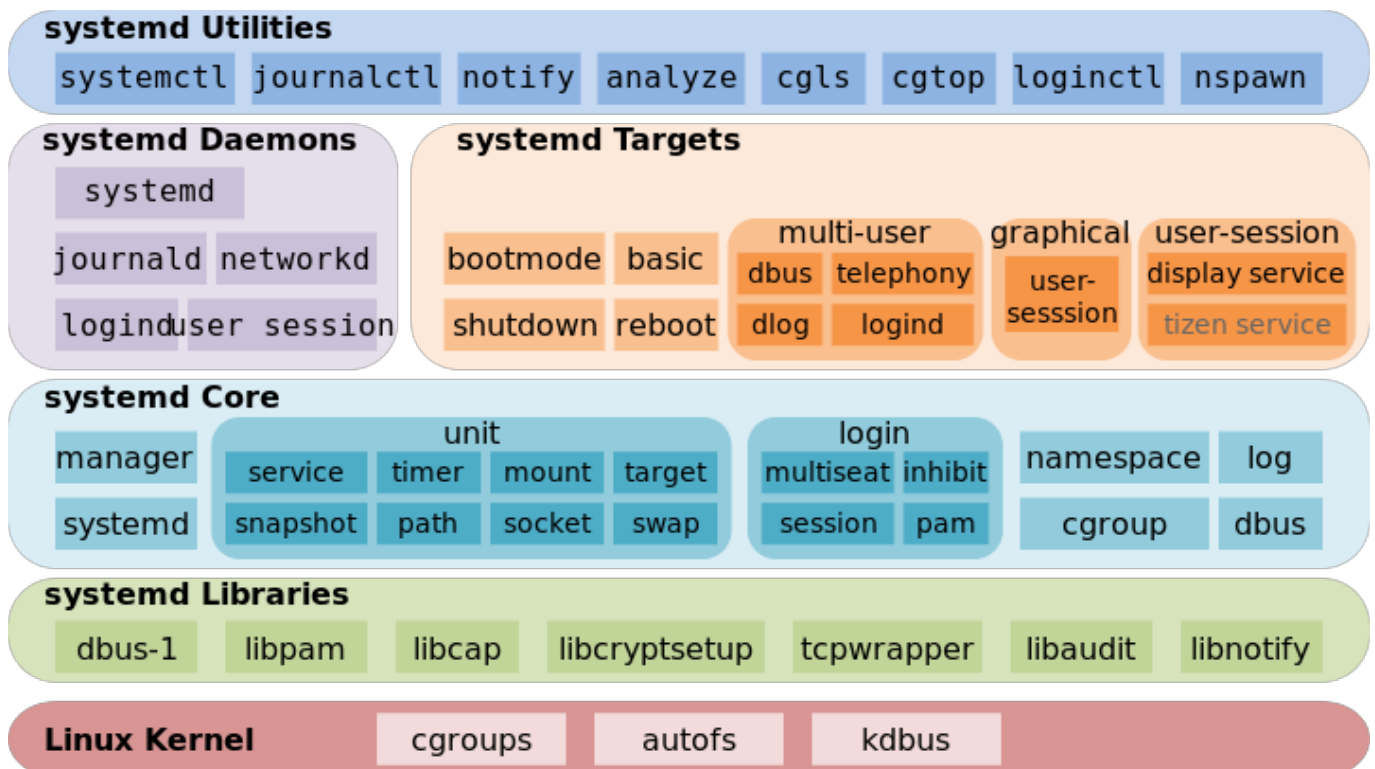


Systemd ispod haube

Systemd se navodi kao nasljednik SysVinita, odnosno kao naprednija zamjena za initd, prvi proces koji zatim podiže sve servise. Pogledajmo kako to systemd radi, i koje su razlike između njega i initd daemona. Radoznalost nas tjera da "podignemo poklopac motora" kako bismo vidjeli što se ispod krije. Već na prvom koraku iznenadit ćemo se shvativši da je systemd zapravo mnogo više od zamjene za SysVinit.

No krenimo redom. Naredna ilustracija pokazuje kako je strukturiran novi softver.



Na prvi pogled zapanjuje složenost novog softvera. Koliko tu ima slojeva! Četiri razine naslagane iznad jezgre koje komuniciraju s kernelom preko `cgroups` ([kontrolne grupe](#) [1]), `autofs` (automatsko montiranje prenosivih medija i mrežnih resursa) i `kdbus` (D-bus, komunikacija među procesima), zatim sedam biblioteka funkcija, `systemd` jezgra, na koju se nadovezuju daemoni, da bi na vrhu bili alati pomoću kojih možemo nadzirati i konfigurirati rad sustava. Odmah postaje očigledno da je `systemd` mnogo više od puke zamjene za `initd`. Novi softver sastoji se od 120.000 redaka koda, kada se izbače komentari. A još uvijek raste i razvija se!

Nećemo se odmah uhvatiti u koštac s cijelom novom strukturom, pokušati je obuhvatiti i razjasniti u jednom zahvatu. Radije ćemo ići korak po korak, pa će se cjelina slike vidjeti na kraju - baš kao da slažemo puzzle od malih komadića.

Počnimo od samog početka. Nakon uključivanja računala izvršava se program upisan u BIOS (Basic Input Output System) na matičnoj ploči. BIOS provjerava i inicijalizira hardver, a zatim pokreće bootloader (na pr. GRUB), koji potraži MBR (Master Boot Record) snimljen najčešće na tvrdom disku. Preko podataka koji su tamo upisani potraži i pokreće jezgru Linuxa. Kad se učita kernel, tradicionalno se pokreće `init` kao proces čiji je PID 1. `init` ima zadaću po zadanom redoslijedu pokretati daemone, pozadinske procese koji pružaju osnovne servise. Umjesto `init` daemona, proces broj jedan sada je `systemd` daemon.

Pogledajmo stablo procesa na tipičnom desktop računalu.

```
$ pstree
systemd??NetworkManager??dhclient
    ?           ??dnsmasq
    ?           ??{gdbus}
    ?           ??{gmain}
    ??accounts-daemon??{gdbus}
    ?           ??{gmain}
    ??acpid
    ??agetty
    ??avahi-daemon??avahi-daemon
    ??colord??{gdbus}
    ?           ??{gmain}
    ??cron
    ??cups-browsed??{gdbus}
    ?           ??{gmain}
    ??dbus-daemon
    ??fwupd??3*[ {GUsbEventThread} ]
    ?           ??{fwupd}
    ?           ??{gdbus}
    ?           ??{gmain}
    ??gnome-keyring-d??{gdbus}
    ?           ??{gmain}
    ?           ??{timer}
    ??irqbalance
    ??isdv4-serial-in
    ??lightdm??Xorg??{Xorg}
    ?           ??lightdm??upstart??at-spi-bus-laun??dbus-daemon
    ?           ?           ?           ?           ??{dconf worker}
    ?           ?           ?           ?           ??{gdbus}
    ?           ?           ?           ?           ??{gmain}
    ?           ?           ?           ??at-spi2-registr??{gdbus}
    ?           ?           ?           ?           ??{gmain}
    ?           ?           ?           ??bamfdaemon??{dconf worker}
    ?           ?           ?           ?           ??{gdbus}
    ?           ?           ?           ?           ??{gmain}
    ?           ?           ?           ?           ??{pool}
    ?           ?           ?           ??compiz??{dconf worker}
    ?           ?           ?           ?           ??{gdbus}
    ?           ?           ?           ?           ??{gmain}
    ?           ?           ?           ?           ??4*[ {pool} ]
    ?           ?           ?           ??dbus-daemon
...

```

Skratili smo ispis, jer je na radnoj stanici broj procesa mnogo veći nego na serveru. To je cijena ljubaznosti prema korisniku. Osim toga, čini se da ima nešto i u tvrdnjama tradicionalista da je systemd razvijen više radi poboljšanja performansi desktopa i sustizanja Windowsa i Maca nego što je potreban Linux serverima.

Radoznalost nas tjera da pogledamo pakete i koliko datoteka oni instaliraju:

```
$ dpkg -l | grep systemd
ii libpam-systemd:amd64      229-4ubuntu13      amd64      system and service m
anager - PAM module
ii libsystemd0:amd64        229-4ubuntu13      amd64      systemd utility l

```

```
library
ii libsystemd0:i386          229-4ubuntu13          i386          systemd utility library
ii python3-systemd         231-2build1            amd64         Python 3 bindings for systemd
ii systemd                 229-4ubuntu13          amd64         systemd service manager
ii systemd-sysv            229-4ubuntu13          amd64         systemd service manager - SysV links
```

```
$ dpkg -L systemd | wc -l
702
```

Dakle šest paketa, a jedan od njih, systemd, instalira čak 702 datoteke! Systemd je monstruozno velik, to je najmanje što se za početak o njemu može reći.

Zadnji paket, systemd-sysv, služi za unazadnu kompatibilnost sa SysVinit naslijeđem. Prije nego se time pozabavimo, prisjetimo se što su runlevels, sedam razina, od 0 do 6, u koje se sistem podiže odnosno spušta. Ovako ih definira Linux standard base:

0 Halt Gašenje OS-a.

1 Single-user mode Jednokorisnički način rada bez mreže i servisa, koriste ga administratori

2 Multi-user mode Višekorisnički način bez mreže i mrežnih servisa

3 Multi-user mode + networking Normalan način rada, mreža i mrežni servisi

4 Not used/user-definable Ne koristi se, za posebne namjene

5 Multi-user, network, +X Višekorisnički način s mrežom plus grafičko korisničko sučelje (runlevel 3 + display manager)

6 Reboot Gašenje i ponovno pokretanje OS-a

Iako je ovo standard, nisu ga se držale sve distribucije Linuxa, a niti sve verzije Unixa. Koga to zanima, može pogledati [članak](#) [2] na Wikipediji.

Runleveli se pri bootanju ne izvršavaju sekvencijalno, svi po redu, nego je u `/etc/inittab` bilo navedeno koji je runlevel zadan. Tako se serveri dižu u runlevel 3, evo kako je to zadano u `/etc/inittab`:

```
id:3:initdefault:
```

Korisnička računala odmah kreću u runlevel 5. Naravno, neki sistemci i na serveru vole X-e, pa su i tamo dizali grafičko sučelje. Admin je mogao prebacivati sustav u druge runlevele. Kad bi na serveru nešto trebalo hitno i na miru obaviti, admin bi prebacio sustav u jednokorisnički način rada, pogasivši time sve mrežne servise:

```
# init 1
```

Runlevel 1 zovu i *rescue mode*, a često ga se pozivalo i slovom s (single).

```
# init s
```

Na isti je način mogao s konzole ugasiti (`init 0`) ili restartati računalo (`init 6`). Systemd donosi naredbu `telinit`, s kojom sistemci stare škole mogu raditi kao dosad.

```
# telinit 1
```

Nemojte se prerano radovati, systemd smatra runlevele zastarjelim, iako čuva kompatibilnost radi

jednostavnijeg prijelaza na nove standarde. Uбудuće ćemo umjesto `init/telinit` koristiti naredbu `systemctl`, a umjesto `runlevela` pozivat ćemo "targete". Evo kako ćemo mijenjati `runlevel` u razinu 5:

```
# systemctl isolate runlevel5.target
```

Ili još bolje, zaboravimo `runlevele`:

```
# systemctl isolate graphical.target
```

Ove dvije naredbe rade istu stvar, prebacuju OS u grafički način rada s mrežom i mrežnim servisima. Mladi kolege mogu odmah učiti novu koncepciju targeta, a mi starci ćemo kad nam "stane mozak" iz zaborava iščupati `runlevele`.

Kako se po novome prebacujemo u jednokorisnički način rada?

```
# systemctl isolate rescue.target
```

Ako želimo saznati u kojem `runlevelu` radi računalo, možemo to učiniti na stari način:

```
# runlevel
N 5
```

ili novi način (sad pitamo za target):

```
# systemctl get-default
```

```
graphical.target
```

Stari SysVinit koristio je `/etc/rcx.d` direktorije (umjesto `x` upišite broj `runlevela`) u kojima su bile smještene skripte za podizanje servisa (počinju slovom `S`, `start`) i njihovo spuštanje (počinju slovom `K`, od `kill`). Systemd koristi `/etc/systemd` direktorij, ali zadržava i stare `rc.d`. Koliko će taj suživot starog i novog trajati, kad će i da li će sva ta "starudija jednog dana nestati, to valjda znaju samo u RedHatu. A mi ćemo se u nastavku posvetiti usvajanju novih znanja, nećemo se previše vraćati u prošlost.

Kategorije: [Operacijski sustavi](#) [3]

Source URL: <https://sysportal.carnet.hr./node/1748>

Links

[1] <https://en.wikipedia.org/wiki/Cgroups>

[2] <https://en.wikipedia.org/wiki/Runlevel>

[3] <https://sysportal.carnet.hr./taxonomy/term/26>