

Podrška li systemd stvarno runlevele?



Systemd se diči kompatibilnošću sa dobrim starim SysVinitom. Prije svega to bi značilo da su zadržani runleveli, a onda i skripte koje podižu/spuštaju servise, smještene u /etc/rc[0-6].d direktorijima koje slijede logiku runlevela. No hajdemo pogledati je li to baš tako, da li je ispoštovana tradicija, ili nas samo žele umiriti kako se ne bi previše bunili, Jer, zna se, ljudi ne vole promjene, pogotovo ako su im nametnute.

Podsjetimo se kako izgleda veza runlevela i systemd targeta.

```
Runlevel Target
0 poweroff.target
1 rescue.target
2,3,4 multi-user.target
5 graphical.target
6 reboot.target
```

Vidimo da su runleveli 2 (multiuser), 3 (multiuser + networking + services) i runlevel 4 (uglavnom se nije koristio) skupljeni u jednoj jedinici, multi-user.targetu. Ako na to dodamo još X-e, dobijemo runlevel 5 iliti po novome graphical.target.

Jednostavno i razumljivo objašnjenje ovog odnosa posudit ćemo iz man stranice za naredbu runlevel:

"Runlevels su zastarjeli način pokretanja i zaustavljanja grupa servisa koji se koristio u SysV initu. Systemd pruža sloj koji omogućuje kompatibilnost, odnosno mapira runlevele s ciljevima (targetima) i naredbama poput runlevel. U SysVinitu samo je jedan runlevel mogao biti aktivan u datom trenutku, dok systemd može aktivirati višestruke targete istovremeno, pa je mapiranje targeta u runlevele samo opribližno i zapravo zbunjuje. Runlevele ne treba koristiti u novom kodu, a najkorisniji su samo kao jednostavan način komuniciranja s kernelom, odnosno njegovim boot parametrima."

Da je tome tako pokazat ćemo na primjeru graphical.targeta. Pogledajmo koji su sve targeti aktivni na toj razini koja nalikuje runlevelu 5. Provjerimo najprije trenutni runlevel, po starinski:

```
$ runlevel
N 5
```

Aktivan je samo jedan runlevel, onaj peti. A sad pogledajmo kako bi to izgledalo po novome, koji su targeti aktivni:

```
$ systemctl list-units --type=target
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
basic.target                        loaded active active Basic System
cryptsetup.target                   loaded active active Encrypted Volumes
getty.target                         loaded active active Login Prompts
graphical.target                     loaded active active Graphical Interface
local-fs-pre.target                  loaded active active Local File Systems (Pre)
local-fs.target                      loaded active active Local File Systems
```

```
multi-user.target    loaded active active Multi-User System
network-online.target loaded active active Network is Online
network.target       loaded active active Network
nss-user-lookup.target loaded active active User and Group Name Lookups
paths.target         loaded active active Paths
remote-fs-pre.target loaded active active Remote File Systems (Pre)
remote-fs.target     loaded active active Remote File Systems
slices.target        loaded active active Slices
sockets.target       loaded active active Sockets
sound.target         loaded active active Sound Card
swap.target          loaded active active Swap
sysinit.target       loaded active active System Initialization
time-sync.target     loaded active active System Time Synchronized
timers.target        loaded active active Timers
```

LOAD = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB = The low-level unit activation state, values depend on unit type.

20 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.

Ako ste očekivali jednostavan i jezgrovit odgovor, poput onog koji daje naredba runlevel, prevarili ste se. Istovremeno je (na Ubuntu desktopu) aktivno 20 jedinica tipa target. Neke od njih nemaju veze sa runlevelima, naprosto su podrška za zvuk, na primjer. Na nekadašnjim U*X serverima time se ne bi ni zamerali, zar ne? Ali na današnjim desktopima, koji se moraju nadmetati u ljubaznosti prema korisnicima, ovakvo postavljanje ima smisla.

Zadnji redak ispisa sugerira da možemo izlistati sve jedinice, uključivo i one jedinice koje nisu aktivne.

```
$ systemctl list-units --type=target --all
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
basic.target                        loaded active active Basic System
cryptsetup.target                  loaded active active Encrypted Volumes
emergency.target                   loaded inactive dead   Emergency Mode
failsafe-graphical.target          loaded inactive dead   Graphical failsafe fallback
final.target                       loaded inactive dead   Final Step
getty.target                       loaded active active Login Prompts
graphical.target                   loaded active active Graphical Interface
halt.target                        loaded inactive dead   Halt
local-fs-pre.target                loaded active active Local File Systems (Pre)
local-fs.target                    loaded active active Local File Systems
multi-user.target                  loaded active active Multi-User System
network-online.target              loaded active active Network is Online
network-pre.target                 loaded inactive dead   Network (Pre)
network.target                     loaded active active Network
nss-user-lookup.target             loaded active active User and Group Name Lookups
paths.target                       loaded active active Paths
reboot.target                      loaded inactive dead   Reboot
remote-fs-pre.target               loaded active active Remote File Systems (Pre)
remote-fs.target                   loaded active active Remote File Systems
rescue.target                      loaded inactive dead   Rescue Mode
shutdown.target                   loaded inactive dead   Shutdown
slices.target                      loaded active active Slices
sockets.target                     loaded active active Sockets
sound.target                       loaded active active Sound Card
swap.target                        loaded active active Swap
```

```
sysinit.target      loaded active    active System Initialization
time-sync.target    loaded active    active System Time Synchronized
timers.target       loaded active    active Timers
umount.target       loaded inactive  dead   Unmount All Filesystems
```

LOAD = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB = The low-level unit activation state, values depend on unit type.

29 loaded units listed.

To show all installed unit files use 'systemctl list-unit-files'.

Ispada da su neke jedinice učitane, ali nisu aktivne, na primjer rescue.target koji je zadužen za vraćanje u jednokorisnički način rada, da bi admin mogao na miru pronaći i riješiti probleme.

U SysVinit sustavima u tu svrhu napisali bi, s admin ovlastima:

```
# init 1
```

ili

```
# init single
```

Analogno, sada bi to, na prvi pogled, trebalo napraviti ovako nekako:

```
# systemctl start rescue.target
```

Ali to bi bilo pogrešno. Kako su nekadašnji runleveli ovdje izmješani, ispravna sintaksa naredbe jest:

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

Izolacija ovdje znači da treba eliminirati sve unite koji su suvišni za jednokorisnički način rada bez mreže i mrežnih servisa. Dobro je imati na umu zašto prizvati parametar isolate. Kad jednom steknemo samopouzdanje, moći ćemo naredbu pisati i ovako:

```
$ systemctl rescue
```

Tada će se pokrenuti naredba wall koja će svim korisnicima ispisati obavijest "The system is going down to rescue mode NOW!"

Ako je raspad sustava toliki da se ne može ući u jednokorisnički spasilački način rada, systemctl nudi još jednu razinu niže:

```
$ systemctl emergency
```

Time ćemo montirati root filesystem samo za čitanje, ali neće se montirati ni jedan drugi lokalni datotečni sustav i, naravno, radit ćemo bez mreže. Ako bismo to pokušali prevesti u logiku runlevela, to bi bio runlevel 0,5. ;)

Targeti su, rekli smo, skup unita različitih vrsta, grupiranih na smislen način. Vidjeli smo na našem primjeru da je učitano 29 targeta, od kojih je 20 aktivno. Ako želimo vidjeti sve unite koji sačinjavaju te targete, naredba bi bila:

```
$ systemctl list-unit-files
```

Nećemo ovdje prenijeti ispis svih unita. Bit će jasno zašto ako samo zbrojimo unite:

```
$ systemctl list-unit-files | wc -l  
295
```

Ako želimo upoznati sve unite, bit će to ogroman posao. Zanimat će nas i njihova međuovisnost, koju možemo vizualizirati ovako:

```
$ systemctl list-dependencies cron.service  
cron.service  
? ??system.slice  
? ??sysinit.target  
? ??apparmor.service  
? ??brltty.service  
? ??console-setup.service  
? ??dev-hugepages.mount  
? ??dev-mqueue.mount  
? ??friendly-recovery.service  
? ??keyboard-setup.service  
? ??kmod-static-nodes.service  
? ??plymouth-read-write.service  
? ??plymouth-start.service  
? ??proc-sys-fs-binfmt_misc.automount  
? ??resolvconf.service  
? ??setvtrgb.service  
? ??sys-fs-fuse-connections.mount  
? ??sys-kernel-config.mount  
? ??sys-kernel-debug.mount  
? ??systemd-ask-password-console.path  
? ??systemd-binfmt.service  
? ??systemd-hwdb-update.service  
? ??systemd-journal-flush.service  
? ??systemd-journald.service  
? ??systemd-machine-id-commit.service  
? ??systemd-modules-load.service  
? ??systemd-random-seed.service  
? ??systemd-sysctl.service  
? ??systemd-timesyncd.service  
? ??systemd-tmpfiles-setup-dev.service  
? ??systemd-tmpfiles-setup.service  
? ??systemd-udev-trigger.service  
? ??systemd-udev.service  
? ??systemd-update-utmp.service  
? ??cryptsetup.target  
? ??local-fs.target  
? ? ??.mount  
? ? ??systemd-fsck-root.service  
? ? ??systemd-remount-fs.service  
? ??swap.target  
? ??dev-disk-by\x2duuid-6f04dafa\x2da58c\x2d40ef\x2da96e\x2d6aa202741932.swap
```

Zabavno, zar ne?

Na kraju, ostaje nejasno čemu zapravo služe skripte u /etc/rcX.d? Nalik su na slijepo crijevo, organ koji je nekad davno vjerojatno imao funkciju, ali je sada nepotreban. Ako idete editirati te skripte,

ništa nećete postići. Doduše, postoji način da se promjene u rc. skriptama aktiviraju, ali o tome možda drugom prilikom.

Sve u svemu, najbolje je zaboraviti runlevele, kao da ih nikada nije bilo, a prihvatiti novotarije kao da su tu oduvijek. A priče o runlevelima neka ostanu za nas stare sistemce, kad se uz kavu budemo prisjećali "dobrih starih vremena".

sub, 2017-10-07 17:47 - Aco Dmitrović **Kuharice:** [Linux](#) [1]

Kategorije: [Operacijski sustavi](#) [2]

Vote: 5

Vaša ocjena: Nema Average: 5 (1 vote)

story_tag: [Linux](#) [3]

[systemd](#) [4]

[runlevels](#) [5]

Source URL: <https://sysportal.carnet.hr./node/1767>

Links

[1] <https://sysportal.carnet.hr./taxonomy/term/17>

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

[3] <https://sysportal.carnet.hr./taxonomy/term/119>

[4] <https://sysportal.carnet.hr./taxonomy/term/120>

[5] <https://sysportal.carnet.hr./taxonomy/term/151>