

Współczesne systemy komputerowe

Start systemu, zarządzanie usługami

Start systemu

- System operacyjny Linux startuje w następujących krokach:
 - **UEFI** (*Unified Extensible Firmware Interface*) lub dawniej **BIOS** (*Basic Input/Output System*), często używany w środowisku wirtualnym,
 - **Boot Manager** (GRUB2, rEFInd, systemd-boot),
 - **Jądro systemu** (*kernel*),
 - **Startowy ramdysk** (*Initial RAM File System, Initial RAM Disk*),
 - **Menadżer startu systemu i usług** (*systemd*, dawniej *init*),
 - **Kolejne procesy i usługi systemu** (uruchamia je demon *systemd*).

GRUB

- Jest to program, którego część rezyduje w **MBR** dysku, początkowych sektorach dysku i w katalogu **/boot** (w przypadku **BIOS**) lub partycji **ESP** (*EFI System Partition*), zamontowanej do **/boot** (w przypadku **UEFI**).
- GRUB potrafi załadować jądro systemu, plik **/boot/vmlinuz-#-#-#-#**.
- GRUB potrafi uruchamiać też inne systemy operacyjne, m. in. z rodziny Windows.
- Plik konfiguracyjny GRUB-a **/boot/grub/grub.cfg**, jest generowany automatycznie na podstawie szablonu z katalogu **/etc/grub.d** i ustawień z **/etc/default/grub**.
- Wyświetli 10 pierwszych linii pliku konfiguracyjnego loadera.

```
root@debian:~# head /boot/grub/grub.cfg
#
# DO NOT EDIT THIS FILE
#
# It is automatically generated by grub-mkconfig using templates
# from /etc/grub.d and settings from /etc/default/grub
#

### BEGIN /etc/grub.d/00_header ###
if [ -s $prefix/grubenv ]; then
    set have_grubenv=true
```

- Sprawdź w pliku konfiguracyjnym wartość zmiennej **timeout**.

```
root@debian:~# cat /boot/grub/grub.cfg | grep timeout
set timeout=30
if [ x$feature_timeout_style = xy ] ; then
    set timeout_style=menu
    set timeout=5
# Fallback normal timeout code in case the timeout_style feature is
set timeout=5
```

- Zmień czas oczekiwania (na 10 sekund, użyj edytora **nano**) na wybór systemu operacyjnego w GRUB-ie, zmiany wprowadza się w **/etc/default/grub**.

```
root@debian:~# cat /etc/default/grub | grep TIMEOUT
GRUB_TIMEOUT=5
```

```
root@debian:~# nano /etc/default/grub
```

- Po wprowadzeniu zmian wygeneruj nowe pliki konfiguracyjne.

```
root@debian:~# update-grub
Generating grub configuration file ...
Found background image: /usr/share/images/desktop-base/desktop-grub.png
Found linux image: /boot/vmlinuz-6.12.57+deb13-amd64
Found initrd image: /boot/initrd.img-6.12.57+deb13-amd64
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
```

```
root@debian:~# cat /boot/grub/grub.cfg | grep timeout
set timeout=30
if [ x$feature_timeout_style = xy ] ; then
    set timeout_style=menu
    set timeout=10
# Fallback normal timeout code in case the timeout_style feature is
set timeout=10
```

- • Zrestartuj system, zwróć uwagę na czas oczekiwania na wybór systemu.

```
root@debian:~# reboot
```

lub

```
root@debian:~# systemctl reboot
```

- Zaznacz **Debian GNU/Linux** i przejdź do trybu edycji, wciśnij [E].
- Na końcu linii zaczynającej się od **linux /boot/vmlinuz-#.#.#...** dopisz **init=/bin/bash** a następnie wciśnij klawisze [CTRL+X] (start systemu).
- W taki sposób można uruchomić powłokę **bash** z prawami użytkownika **root**, można zmienić w ten sposób zapomniane hasło użytkownika administracyjnego, należy jednak wcześniej zamontować system plików z możliwością zapisu.

```
root@(none):/# whoami
root
```

```
root@(none):/# mount -o remount,rw,sync /
```

```
root@(none):/# passwd
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

- Zamontuj system plików tylko do odczytu i wyłącz komputer (jeśli wirtualny, zamknij okno systemu gościa, nie zapisuj stanu maszyny).

```
root@(none):/# mount -o remount,ro /
```

- Dlatego w systemach produkcyjnych zależy zabezpieczać loader systemu przed dostępem osób niepowołanych.

systemd

- Menadżerem startu systemu i usług jest demon **systemd**.
- W stosunku do skryptów **SysVinit** lub **BSDinit** ma większe możliwości:
 - możliwość równoległego uruchamiania usług,
 - uruchamianie serwisów za pomocą aktywacji gniazda lub na żądanie,
 - śledzenie działania procesów przy pomocy grup kontrolnych **cgroups**,
 - wsparcie dla zachowywania i przywracania stanu usług w systemie,
 - utrzymanie punktów montowania i automatycznego montowania w systemie.
- **systemd** posługuje się tzw. unitami, wyświetl typy unitów.

```
root@debian:~# systemctl -t help
Available unit types:
service
mount
swap
socket
target
device
automount
timer
path
slice
scope
```

- Podstawowe typy unitów.

*.service	serwisy i usługi systemowe
*.timer	aktywacja zadań w oparciu o czas
*.socket	aktywacja usług w oparciu o port (systemd nasłuchuje port)
*.target	podobne do klasycznych poziomów pracy, grupowanie unitów w logiczną funkcjonalność
*.path	monitorowanie ścieżki w celu aktywacji zadania
*.mount	montowanie systemów plików (np. /boot, /tmp)
*.automount	automontowanie systemów plików (np. /proc, /sys)
*.device	pliku urządzeń

- Wyświetl wszystkie dostępne unity związane z usługami (zakończenia działania polecenia wymaga wciśnięcia klawisza [Q]).

```
root@debian:~# systemctl list-unit-files --type=service
UNIT FILE                                STATE      PRESET
```

accounts-daemon.service	enabled	enabled
alsa-restore.service	static	-
alsa-state.service	static	-
alsa-utils.service	masked	enabled
anacron.service	enabled	enabled
apparmor.service	enabled	enabled
apt-daily-upgrade.service	static	-
apt-daily.service	static	-
apt-listchanges.service	static	-
autovt@.service	alias	-
avahi-daemon.service	enabled	enabled
bluetooth.service	enabled	enabled
bolt.service	static	-
capsule@.service	static	-
colord.service	static	-
...		

- Każdy unit może być w jednym ze stanów.

enabled	uruchamiany po starcie systemu
disabled	nieuruchamiany po starcie systemu, ale może być załadowany jako zależność
static	może być uruchomiony przez inny unit, nie może być uruchomiony ręcznie
masked	mocniejsze niż disabled , nie może być uruchomiony jako zależność, może być uruchomiony po odmaskowaniu
generated	wygenerowany automatycznie na podstawie skryptów /etc/init.d

Poziomy pracy

- Stan systemu operacyjnego definiują tzw. **poziomy pracy**.

poweroff.target	zatrzymywanie systemu: (poweroff , shutdown -h)
rescue.target	tryb awaryjny, minimalny zestaw usług
multi-user.target	wielu użytkowników, bez interfejsu graficznego
graphical.target	wielu użytkowników, z interfejsem graficznym
reboot.target	restart systemu (reboot , shutdown -r)

- Sprawdź domyślny poziom pracy systemu.

```
root@debian:~# systemctl get-default
graphical.target
```

- Zmień poziom pracy systemu.

```
root@debian:~# systemctl isolate multi-user.target
```

- Zaloguj się w konsoli tekstowej jako **root**.
- Sprawdź poziom pracy systemu, zmień poziom pracy na graficzny.

```
root@debian:~# systemctl isolate graphical.target
```

- Kombinacją klawiszy [CTRL+ALT+F2] przełącz się na konsolę tekstową, zaloguj się i sprawdź działanie skrótów [ALT+F3] do [ALT+F6].
- Kombinacją [ALT+F1] wróć do środowiska graficznego.
- Zmień domyślny poziom pracy systemu, zrestartuj maszynę.

```
root@debian:~# systemctl set-default multi-user.target
Created symlink '/etc/systemd/system/default.target' → '/usr/lib/systemd/system/multi-user.target'.
```

```
root@debian:~# systemctl get-default
multi-user.target
```

```
root@debian:~# reboot
```

- Zaloguj się, ponownie zmień domyślny poziom pracy systemu i zrestartuj maszynę.

```
root@debian:~# systemctl set-default graphical.target
Created symlink '/etc/systemd/system/default.target' → '/usr/lib/systemd/system/graphical.target'.
```

```
root@debian:~# systemctl get-default
graphical.target
```

```
root@debian:~# reboot
```

Zarządzanie usługami

- Wyświetl załadowane usługi systemu

```
root@debian:~# systemctl list-units --type=service
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION    >
accounts-daemon.service            loaded active running Accounts Servi>
alsa-restore.service               loaded active exited Save/Restore S>
apparmor.service                   loaded active exited Load AppArmor >
avahi-daemon.service               loaded active running Avahi mDNS/DNS>
colord.service                     loaded active running Manage, Instal>
console-setup.service              loaded active exited Set console fo>
cron.service                       loaded active running Regular backgr>
cups-browsed.service               loaded active running Make remote CU>
cups.service                       loaded active running CUPS Scheduler
dbus.service                       loaded active running D-Bus System M>
fwupd.service                      loaded active running Firmware updat>
gdm.service                        loaded active running GNOME Display >
geoclue.service                   loaded active running Location Looku>
ifupdown-pre.service               loaded active exited Helper to sync>
keyboard-setup.service             loaded active exited Set the consol>
...
```

- Status usługi, kolumna **ACTIVE**:
 - **active** - usługa działa,
 - **inactive** - usługa nie działa, czasem z powodu błędu.
- Kolumna **SUB**:

- **running** - usługa jest uruchomiona,
- **exited** - usługa zakończona sukcesem,
- **waiting** - usługa czeka na zdarzenie.
- Sprawdź status usługi **cron**.

```
root@debian:~# systemctl status cron
● cron.service - Regular background program processing daemon
   Loaded: loaded (/usr/lib/systemd/system/cron.service; enabled; preset: ena>
   Active: active (running) since Wed 2025-12-10 14:40:04 EST; 13min ago
  Invocation: b3d39dac80124a21b9e193e497674eec
     Docs: man:cron(8)
  Main PID: 777 (cron)
    Tasks: 1 (limit: 2296)
   Memory: 444K (peak: 1.8M)
      CPU: 8ms
   CGroup: /system.slice/cron.service
           └─777 /usr/sbin/cron -f

Dec 10 14:40:04 debian systemd[1]: Started cron.service - Regular background pr>
Dec 10 14:40:04 debian cron[777]: (CRON) INFO (pidfile fd = 3)
Dec 10 14:40:04 debian cron[777]: (CRON) INFO (Running @reboot jobs)
```

- Wyłącz usługę i sprawdź jej status.

```
root@debian:~# systemctl disable cron
Synchronizing state of cron.service with SysV service script with
/usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install disable cron
Removed '/etc/systemd/system/multi-user.target.wants/cron.service'.
```

```
root@debian:~# systemctl list-unit-files --type=service | grep cron
anacron.service                                enabled          enabled
cron.service                                   disabled        enabled
```

```
root@debian:~# systemctl status cron
● cron.service - Regular background program processing daemon
   Loaded: loaded (/usr/lib/systemd/system/cron.service; disabled; preset: en>
   Active: active (running) since Thu 2025-12-11 01:52:43 EST; 2min 47s ago
  Invocation: ef56e486f23243e28820b8d14708d42b
     Docs: man:cron(8)
  Main PID: 763 (cron)
    Tasks: 1 (limit: 2296)
   Memory: 444K (peak: 1.5M)
      CPU: 7ms
   CGroup: /system.slice/cron.service
           └─763 /usr/sbin/cron -f

Dec 11 01:52:43 debian systemd[1]: Started cron.service - Regular background pr>
Dec 11 01:52:43 debian cron[763]: (CRON) INFO (pidfile fd = 3)
Dec 11 01:52:43 debian cron[763]: (CRON) INFO (Running @reboot jobs)
```

- Przywróć autostart usługi.

```
root@debian:~# systemctl enable cron
```

```
Synchronizing state of cron.service with SysV service script with
/usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable cron
Created symlink '/etc/systemd/system/multi-user.target.wants/cron.service' →
'/usr/lib/systemd/system/cron.service'.
```