

## Øvelse 8 – Processer og services i Linux

### Information

Formålet med følgende øvelser, er at gøre dig i stand til at udføre grundlæggende håndtering af kørende processer i Linux.

Som nævnt i dagens forberedelse er en af egenskaberne ved et operativsystem, at der kan eksekveres mange applikationer på samme tid.

I Linux refererer man til kørende applikationer som *processer*. Dette kigger vi på i de følgende øvelser.

**Fortsæt med at notere i dit Linux cheat sheet**

---

### Links til beskrivelse af kommandoerne i øvelsen

- [beskrivelse af ps kommandoen](#)
- 

### Instruktioner

I det første trin skal du se, hvilke processer den nuværende bruger har kørende.

---

#### 1. Visning af kørende processer

- Eksekver kommandoen `ps`.

Resultatet bør ligne nedstående billede:

```
martin@L02485:~$ ps
  PID TTY          TIME CMD
   10 pts/0    00:00:00 bash
   21 pts/0    00:00:00 ps
```

I venstre side af outputtet ses Process Id (PID). Hver process i Linux har et unikt ID. Herefter kommer *TeleTypewriter* (TTY), som viser hvilken terminal der eksekverede en kommando.

CPU tid (TIME) viser det samlede CPU-tidsforbrug.

Til sidst viser Command (CMD), hvilken kommando der eksekverede processen.

- Eksekver kommandoen `ps -aux`

```

martin@L02485:~$ ps -aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0    916    528 ?        S1    12:41    0:00 /init
root         8  0.0  0.0   1272    356 ?        Ss    12:41    0:00 /init
root         9  0.0  0.0   1272    364 ?        S     12:41    0:00 /init
martin     10  0.0  0.0   6072   5056 pts/0    Ss    12:41    0:00 -bash
martin     24  0.0  0.0   7788   3300 pts/0    R+   13:08    0:00 ps -aux

```

Denne gang vises processerne for alle brugere.

Der er et par nye kolonner i outputtet; de mest interessante er %CPU, %MEM og STAT. %CPU og %MEM viser hver især det procentvise CPU- og hukommelsesforbrug. STAT viser processorens tilstand: S=Sleep but interruptible, R=Running, og T=Terminated.

## 2▢ Installation og observation af en service

I de næste tre trin installerer vi en service i form af Apache webserver.

Typisk bruger man ordet "service", når en applikation tilbyder en "service" til andre. I dette tilfælde en API via netværk (også i slang kaldet en "Web API"), som andre systemer kan benytte sig af (f.eks. en browser).

En service er blot en applikation, som eksekveres på operativsystemet, og den har derfor også en proces.

1. Eksekver kommandoen `sudo apt install apache2 && sudo systemctl unmask apache2`.
2. Udskriv alle processer til konsollen og `grep` på ordet "apache2".

Bemærk tilstanden (STAT) er S. Altså, processen kører ikke, men venter på en hændelse. I de næste trin prøver vi at vække den.

## 3▢ Aktivering af service via hændelse

1. Eksekver kommandoen `sudo apt install curl`.  
*Curl er et værktøj, som kan lave HTTP-forespørgsler til f.eks. Apache2 webserveren.*
2. Eksekver kommandoen `curl 127.0.0.1`.  
*Nu sendes der et HTTP-request til Apache2 webserveren.*
3. Gennemse request svar og bekræft, at det er HTML fra Apache-serveren.

Det betyder, at selv om en proces ikke kører (S), kan den ved behov vækkes af operativsystemet i tilfælde af en specifik hændelse.

## 4 Genstart og afslutning af processer

I de sidste trin fokuseres der på at genstarte en proces og at "slå den ihjel".

1. Eksekver kommandoen `nano &`.  
*Nano er en teksteditor, som vi med ampersand-symbolet (&) kører i baggrunden.*
2. Udskriv alle processer til konsollen og grep på ordet `nano`, men undgå at inkludere resultatet fra `grep`-kommandoen selv.  
*Der kommer formodentlig to resultater; den ene er grep-kommandoen, som lige er blevet eksekveret.*
3. I kolonne 2 kan du se process ID (PID); eksekver følgende kommando med process-ID'et `kill -1 <Skriv PID her>`  
*(Dette sender SIGHUP-signal og beder processen om at genindlæse sin konfiguration).*
4. Udskriv alle processer til konsollen og grep på ordet `nano`.  
Outputtet skulle gerne ligne outputtet fra tidligere. Det er fordi processen er genstartet, men ikke slukket; starttid og PID forbliver det samme.
5. Eksekver nu følgende kommando `kill -9 <Skriv PID her>`  
*(Dette sender SIGKILL-signal og tvinger processen til at stoppe med det samme).*
6. Nu skulle processen være slukket og ikke synlig, hvis du udskriver processerne.

---

## Links

- [Basic Linux commands](#)
- [Basic Linux commands videos](#)

---

Last update: 2026-03-20 13:58:28