

Lektion 6 – Laboration 2

1: Hitta dubblett-binärer

Använd "ls", "sort" och "uniq" för att hitta alla kommandon i "/bin" och "/usr/bin" med samma namn.

2: Testa processer med "kill"

För att "kill" ska "lyckas", måste den ha rättigheter att skicka signaler till angiven process, och processen måste finnas. Om båda villkoren är sanna, lyckas "kill", och sätter avslutskod till noll.

Om man anger signalen 0 (noll) till "kill", så verifierar den båda villkoren, men skickar ingen signal. Det kan man använda för att ta reda på om en process med ett visst ID finns, om man har rätt rättigheter.

Exempel:

```
kill -0 1234 && echo "it's alive!"
```

Det kan vara mycket användbart när man skriptar med processer.

OBS att du ska använda PID, och inte jobb-ID nedan. Använd "jobs -l" för att få ut PID för varje jobb!

- Starta en lång process, exempelvis "sleep 200"
- Tryck [ctrl+z] för att pausa processen
- Använd nu "kill -0" enligt ovan, kedjat med ett "echo" – anrop som säger att processen lever
 - (använd "&&" för att kedja ihop)

Jämför ditt resultat med andra. Är det rimligt?

- Slå ihjäl den pausade processen med "kill"
- Kör ditt "kill -0" – test igen

Vad får du för resultat? Jämför med de andra.

- Ta nu fram processen i förgrunden

Vad händer?

- Kör ditt "kill -0" – test åter igen
- Bli av med den tråkiga utskriften ("No such process") på något sätt!
- Kedja på ytterligare ett "echo" – anrop som säger att processen är död (om den nu är det)

Upprepa gärna och testa med olika varianter så du får en känsla för hur det hänger ihop.

3: Källkods-statistik

Linux-kärnan är skriven i språket "C", där man använder header-filer flitigt. Namnet på header-filer slutar med ".h".

En del av källkoden till Linux-kärnan ligger i `"/usr/src/linux-headers-4.15.0-136/include"` och underkataloger.

För att lista alla header-filer där, och slippa skriva så mycket, kan man göra

```
ls /usr/src/*/include/*/*.h
```

- Skapa en pipe som lägger ihop alla filerna ovan och letar rätt på alla rader som innehåller `"#include"`
 - Tips: börja med att byta ut `"ls"` mot `"cat"`
- Utöka pipen med hjälp av `"sort"` och `"uniq"` för att ta reda på vilka de fem mest inkluderade headerfilerna i Linux-kärnan är. Spara resultatet i en fil - `"topincludes.txt"`.
 - I mitt test hade den mest populära filen inkluderats 703 gånger.