

## Sessió 6: GESTIÓ DE PROCESSOS

Els objectius d'aquesta sessió són conèixer amb més profunditat com gestionar processos des del shell Unix (`ps`, `kill`, `top` i `uptime`, i el directori `/proc`) i repassar shellscrips. Llegiu-vos això i repasseu la documentació sobre shellscrips.

### COMANDES

#### **ps**

La comanda `ps` mostra informació sobre els processos en execució a la màquina. La comanda admet gran quantitat d'opcions que es poden classificar en tres grups:

- **Selecció de processos a llistar:** per omisió només presenta els processos associats al terminal des del que s'ha iniciat la comanda, però podem indicar que volem veure tots els processos (`-e`), els d'un usuari (`-u` seguit del nom de l'usuari), els que s'executen a un terminal (`-t` seguit del nom del terminal), els que executen una comanda determinada (opció `-C`),...
- **Informació a mostrar:** per omisió només mostra el *pid*, el terminal associat (TTY), el temps de CPU consumit i el nom de l'executable, però podem fer que mostri més dades (opcions `-l` o `-f`) o indicar els camps a mostrar (opció `-o`).
- **Modificadors del resultat:** permeten especificar alguns detalls relatius a com es presenta la informació (nombre de columnes, presència de capçaleres,...).

Els tres tipus d'opcions es poden combinar. És possible que opcions presents a una versió de Unix no estiguin presents a una altra; en aquest cas caldrà buscar al manual alguna altra opció equivalent o pensar alguna altra forma d'aconseguir el mateix resultat.

#### **uptime**

`uptime` mostra informació variada sobre el sistema: l'hora actual, el temps transcorregut des de la posada en marxa del sistema (*uptime*), el nombre d'usuaris connectats i la càrrega mitjana (*load average*) al darrer minut, 5 minuts i 15 minuts.

La càrrega mitjana és el nombre mitjà de processos que estan als estats *Run* o *Ready*. A màquines amb un únic processador, valors de càrrega superiors a 3 poden provocar que el temps de resposta del sistema s'incrementi significativament. L'administrador del sistema hauria de comprovar periòdicament que aquest no superi un nivell raonable.

Per comprovar el funcionament de la comanda podeu crear una (o vàries) finestra(es) i executar la següent comanda<sup>1</sup> desde `bash` o `sh`: **`while (true) do date; done`** La comanda realitza un bucle infinit on a cada iteració imprimeix l'hora actual. Des d'una

---

<sup>1</sup> Aquest tipus de comandes seria aconsellable que les executéssiu únicament a una sistema on no hi hagi altres usuaris treballant; per tant no les executeu a albanta

altra finestra, comproveu com evolucionen les dades de càrrega mostrades per uptime.

## **top**

La comanda `top` ens mostra informació sobre els processos en execució al sistema ordenats per consum de CPU i, a diferència de `ps`, actualitza aquesta informació periòdicament. Per finalitzar l'execució de `top` cal prémer la tecla `q`.

Les primeres línies mostrades per `top` donen informació general sobre el sistema. La primera presenta la mateixa informació que la comanda `uptime`. La segona indica el nombre total de processos i a quants hi ha a cada estat (execució, bloquejat, aturat i *zombie*). La tercera mostra l'ús de la CPU (% en mode usuari, % en mode sistema i % *idle*, és a dir, sense codi a executar). Les dues últimes mostren informació sobre l'ocupació de la memòria física i sobre l'àrea de *swap*.

Podem fer que `top` mostri informació sobre els processos d'un usuari concret, que ordeni els processos per altres criteris, modificar la freqüència d'actualització,... Per veure totes les opcions disponibles podeu prémer la tecla `h`.

Normalment, no és conveninet utilitzar aquesta comanda dins de *shellscripts* ja que, per omissió, requereix que l'usuari premi una tecla per finalitzar la seva execució. Com `top` pot consumir molta CPU, és possible que no estigui disponible per a tots els usuaris.

## **kill i killall**

Són les comandes que permeten matar processos. La primera selecciona el procés a matar a partir del seu *pid*, la segona el selecciona a partir del nom de la comanda.

## **Explorant el directori /proc**

Moltes màquines Unix i Linux utilitzen un pseudo-sistema de fitxers per oferir informació variada sobre els sistema i sobre els processos que s'hi executen. Per omissió, aquest sistema de fitxers es monta a `/proc`. Si us situeu a aquest directori i executeu `ls` veureu un seguit de fitxers i de directoris.

Els fitxers ofereixen informació general sobre la màquina: `cpuinfo` (característiques de la CPU de la màquina), `meminfo` (utilització de la memòria), `version` (versió del sistema operatiu),...; podeu examinar aquests fitxers amb les comandes `more`, `cat`,...

El nom de la majoria de directoris és un nombre. Aquest nombre representa un identificador de procés (*pid*). Si us situeu dins del directori d'algun dels vostres processos podreu obtenir informació respecte el procés com ara la línia de comandes que l'ha creat (`cmdline`), les variables d'entorn (`environ`), identificadors variis associats al procés i informació d'estat (`status`),... Si executeu `man proc` podeu obtenir informació sobre el contingut de cadascun d'aquests fitxers.