

Processos

Els processos, com veureu en aquesta secció, són un altre dels elements més importants del sistema operatiu GNU/Linux. Es parla de processos que arrenquen i s'aturen, que s'executen, que moren, que ocupen CPU i memòria, saturant el sistema. És important que enteneu que els processos són programes que, mentre s'executen, resideixen en memòria i ocupen temps de la CPU. Quan el programa arriba a la última instrucció, el procés finalitza, acaba o mor. Molt sovint, aquests programes no arriben al final fins que no es produeix un determinat estat, i mentre, no deixen d'executar-se.

Multitasca

Respecte dels processos un sistema operatiu donat pot ser multitasca o pot ser monotasca. Usualment, un sistema no pot ser multiusuari sense ser multitasca. Estrictament parlant, un sistema multitasca només ho pot ser si té més d'un xip central, d'una unitat de procés central. Així, els PCs normals, que només en tenen una, no poden ser multitasca real, però hi ha tècniques per emular-la. GNU/Linux empra una d'aquestes tècniques per emular-la, per la qual cosa se'n considera un sistema operatiu multitasca. Així podeu considerar que en un mateix moment hi ha diferents processos (en alguns casos, podríeu parlar de milers) executant-se alhora.

Relacions i comunicacions entre processos

Tots aquests processos es relacionen entre ells creant-ne de nous. De fet, així com el sistema de fitxers té una arrel i tot hi està connectat en forma d'arbre, els processos tenen una estructura molt similar. En arrencar el nucli, quan l'ordinador s'encén i el carrega en memòria, el sistema inicia el primer procés de tots, anomenat init i que té l'identificador 1. Si aquest procés mor, el sistema s'atura immediatament i sense remei. Aquest primer procés inicia, segons té configurat, una sèrie de processos que permeten al sistema oferir els serveis que ha de donar, primer establint les comunicacions amb els dispositius, i després oferint les interfícies de cara a l'usuari. Tots els processos que arranca tenen com a identificador de procés pare, l'1, però no els processos que arrenquen els que l'init ha arrencat. Així es genera un arbre.

Quan un usuari inicia una sessió, l'interpret d'ordres té un identificador de procés, que podria ser 1000. Si l'usuari executa un programa, com Thunderbird, aquest podria tenir l'identificador 1001 (si no s'ha executat cap altre programa al sistema entre l'inici de sessió i l'arrencada de l'aplicació), però l'identificador de procés pare seria el 1000.

Entre processos poden existir comunicacions, o bé per la xarxa o bé pel que s'anomenen canonades. Les canonades són eines que els programadors fan servir per passar informació d'un procés a un altre i així fer-los dependre. De canonades hi ha de molts tipus i no són objecte d'estudi en aquest curs, per això recomanem que només tingueu present que quan empreu el | a la línia de comandes, esteu emprant un tipus de canonada.


Gestió de processos

Tot i que un dels trets més reconeguts de GNU/Linux sigui l'estabilitat, això no treu que els programes de tant en tant topin amb algun problema i es pengin, o que l'usuari executi algun programa per error i el vulgui aturar. Hi ha diferents formes de gestionar els processos, depenent de en quin estat ens trobem i de com s'hagi executat el procés.

Tots els usuaris poden gestionar els seus propis processos des de la mateixa sessió en la que es troben, tot i que no sempre és senzill. També poden gestionar els seus processos d'una sessió des d'una altra, però només l'usuari root pot gestionar tots els processos, siguin seus, d'altres usuaris o del sistema.

Visualització de processos

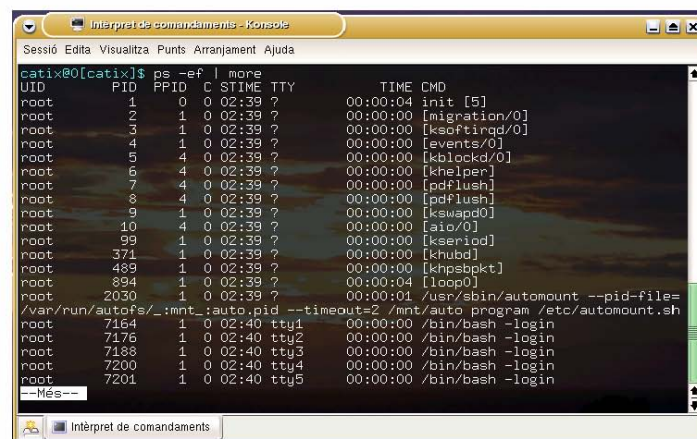
Per tal de visualitzar els processos, podeu emprar, principalment, la comanda `ps`. Aquesta comanda, sense paràmetres, mostra l'identificador de cada procés d'aquesta sessió, l'identificador de terminal en la que es troba la sessió (TTY), el temps d'execució i la línia de comandes que representa.



```
catix@0[catix]$ ps
  PID TTY          TIME CMD
 12140 pts/0    00:00:00 bash
 12165 pts/0    00:00:00 ps
catix@0[catix]$
```

Imatge 2-4-7. Visualització de processos (1).

La comanda `ps` té moltes opcions, la combinació d'algunes us poden permetre veure tots els processos del sistema, com `-ef`.



```
catix@0[catix]$ ps -ef | more
UID          PID    PPID  C  STIME TTY          TIME CMD
root           1      0  0  02:39 ?        00:00:04 init [s]
root           2      1  0  02:39 ?        00:00:00 [migration/0]
root           3      1  0  02:39 ?        00:00:00 [ksoftirqd/0]
root           4      1  0  02:39 ?        00:00:00 [events/0]
root           5      4  0  02:39 ?        00:00:00 [kblockd/0]
root           6      4  0  02:39 ?        00:00:00 [khelper]
root           7      4  0  02:39 ?        00:00:00 [pdflush]
root           8      4  0  02:39 ?        00:00:00 [pdflush]
root           9      1  0  02:39 ?        00:00:00 [kswapd0]
root          10      4  0  02:39 ?        00:00:00 [aio/0]
root          99      1  0  02:39 ?        00:00:00 [kseriod]
root         371      1  0  02:39 ?        00:00:00 [khubd]
root         489      1  0  02:39 ?        00:00:00 [khpsbpkt]
root         894      1  0  02:39 ?        00:00:04 [loop0]
root        2030      1  0  02:39 ?        00:00:01 /usr/sbin/automount --pid-file=
/var/run/autofs/_mnt:auto.pid --timeout=2 /mnt/auto program /etc/automount.sh
root        7164      1  0  02:40 tty1      00:00:00 /bin/bash -login
root        7176      1  0  02:40 tty2      00:00:00 /bin/bash -login
root        7188      1  0  02:40 tty3      00:00:00 /bin/bash -login
root        7200      1  0  02:40 tty4      00:00:00 /bin/bash -login
root        7201      1  0  02:40 tty5      00:00:00 /bin/bash -login
--Mes--
```

Imatge 2-4-8. Visualització de processos (2).

Tot i que `ps` és molt útil, només mostra l'estat en un moment determinat. Si es vol tenir una llista refrescada dels processos que s'estan executant, es pot emprar la comanda `top`:

```
top - 02:49:34 up 10 min, 0 users, load average: 0.13, 2.09, 2.09
Tasks: 45 total, 2 running, 43 sleeping, 0 stopped, 0 zombie
Cpu(s): 2.3% us, 3.6% sy, 0.0% ni, 94.0% id, 0.0% wa, 0.0% hi, 0.0% si
Mem: 283972k total, 241172k used, 42800k free, 32344k buffers
Swap: 0k total, 0k used, 0k free, 131976k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
11251 root        16   0 36724  16m  20m  S   2.7   6.1   1:13.46 XFree86
12118 catix       15   0 34836  18m  32m  S   2.0   6.5   0:15.42 kdeinit
12171 catix       16   0 2088  1016  1876  R   0.7   0.4   0:00.10 top
11999 catix       15   0 27356  12m  24m  S   0.3   4.6   0:02.09 kdeinit
12107 catix       15   0 30452  14m  28m  S   0.3   5.1   0:02.30 kdeinit
12130 catix       16   0 30048  14m  27m  R   0.3   5.3   0:02.35 kdeinit
1 root        16   0   76   76   56  S   0.0   0.0   0:04.56 init
2 root        RT   0   0   0   0   0  S   0.0   0.0   0:00.00 migration/0
3 root        34  19   0   0   0  S   0.0   0.0   0:00.00 ksoftirqd/0
4 root         5 -10   0   0   0  S   0.0   0.0   0:00.93 events/0
5 root         5 -10   0   0   0  S   0.0   0.0   0:00.14 kblockd/0
6 root         5 -10   0   0   0  S   0.0   0.0   0:00.00 khelper
7 root        20   0   0   0   0  S   0.0   0.0   0:00.00 pdflush
8 root        15   0   0   0   0  S   0.0   0.0   0:00.07 pdflush
9 root        25   0   0   0   0  S   0.0   0.0   0:00.00 kswapd0
10 root        6 -10   0   0   0  S   0.0   0.0   0:00.00 aio/0
99 root        25   0   0   0   0  S   0.0   0.0   0:00.00 kseriod
```

Imatge 2-4-9. Comanda TOP.

Per sortir-ne hi ha prou amb prémer q.

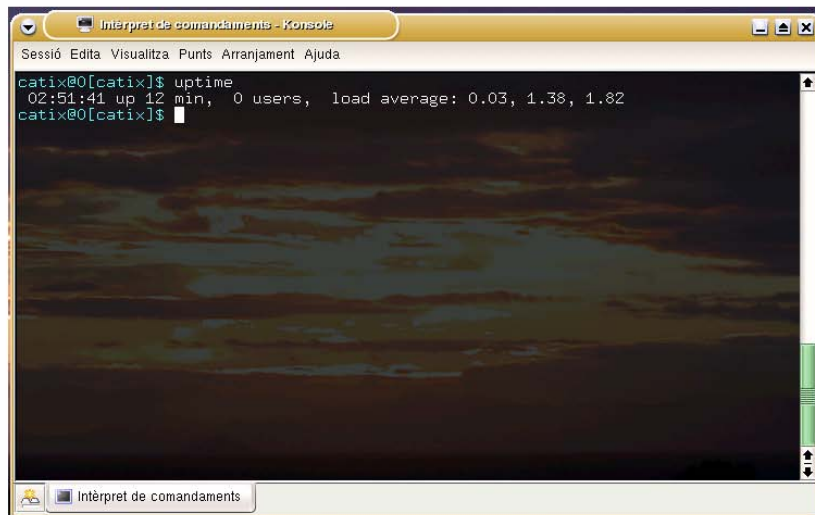
Càrrega de la CPU i de la memòria

Com ja hem comentat, cada procés ocupa una quantitat de memòria i un percentatge de temps de CPU. Si ens interessa veure quines proporcions s'estan ocupant, podem emprar o bé el top o bé el ps amb l'opció o i els paràmetres %cpu i %mem.

```
catix@catix$ ps -o %cpu,%mem
%CPU %MEM
0.3 0.8
0.0 0.2
catix@catix$ ps -o %cpu,%mem,args
%CPU %MEM COMMAND
0.3 0.8 /bin/bash
0.0 0.2 ps -o %cpu,%mem,args
catix@catix$
```

Imatge 2-4-10. Carrega del sistema (1).

Tot i que si volem veure un estat general del sistema podem emprar les comandes uptime. La comanda uptime dóna el temps que porta el sistema arrancat, el nombre d'usuaris que tenen sessions iniciades i els valors mitjos de càrrega del sistema pels últims 1, 5 i 15 minuts.



Imatge 2-4-11. Carrega del sistema (2).

Enviament de senyals als processos

Per a forçar a processos a canviar el seu estat, podeu emprar la comanda kill, que permet enviar diferents senyals a un procés determinat. Els senyals més habituals són el SIGHUP (-1), el SIGTERM (-15) i el SIGINT (-1). El primer reinicia el procés, sense aturar-lo. El SIGTERM indica al procés que ha de finalitzar i permet que finalitzi les tasques més immediates que ja no poden anul·lar-se. El SIGINT és la senyal de tancament immediat, que no convé que empreu sinó és perquè el procés no respon.

Prioritat dels processos

Tots els processos del sistema s'executen amb una prioritat determinada. Per visualitzar la prioritat dels processos, podeu emprar l'opció -l de la comanda ps. En general la prioritat és sempre igual, però es pot modificar abans d'iniciar el procés i un cop iniciat per tal que puguin finalitzar abans. Les comandes a emprar són nice, per ajustar la prioritat abans de l'execució, i renice, per ajustar-la durant l'execució.

Execució especial de programes

El processos, com s'ha dit abans, són programes que s'executen. En el moment d'executar-los ho podeu fer de diferents formes, des del punt de vista del procés que es genera. Una d'aquestes formes és, senzillament, invocar-lo amb les opcions que necessiteu, que és el mètode típic.

Si el procés no requereix interacció amb l'usuari, podeu emprar la comanda nohup i/o el símbol & al final de l'ordre. La comanda nohup fa que el programa s'executi en una espècie d'entorn separat, immune a talls i amb la sortida redirigida cap a un fitxer. El símbol &, que és equivalent l'ús de la comanda bg, permet que el procés s'executi en segon pla i que l'usuari pugui executar altres processos. Si es vol reprendre un procés que es troba en execució en segon pla, es pot emprar la comanda fg.

Xarxes i comunicacions

Tot sistema operatiu modern que vulgui tenir certa significació té la capacitat de connectar-se en xarxa i compartir informació amb els ordinadors que hi ha connectats. GNU/Linux és un sistema operatiu modern amb una certa importància, per tant, es pot connectar en xarxa. Generalment, quan es parla de connectar-se en xarxes es fa referència a xarxes amb protocols TCP/IP, sigui en els medis que sigui (cable de coure, fibra òptica, wireless, etc.). GNU/Linux, com d'altres sistemes operatius, pot comunicar-se en d'altres protocols, però només parlarem de xarxes TCP/IP.

Conceptes i termes

Un dels elements indispensables per la comunicació en xarxa és disposar d'una interfície de xarxa. Aquesta interfície, des del punt de vista del sistema, pot ser física, si representa la interfície real del maquinari, o virtual, si representa una extensió basada en algun estàndard específic. El nom de la interfície pot variar, segons el dispositiu, així si és una tarja Ethernet, el dispositiu s'anomenarà eth seguit d'un nombre, inicialment 0, depenent de la quantitat de targes Ethernet del maquinari.

Les interfícies poden tenir assignada una o més adreces IP. Una adreça IP és un dígit que identifica cada interfície de les que estan connectades en xarxa. Aquest dígit té una representació que consisteix en quatre enters de 0 a 255 separats per punts: 192.168.0.2, 195.77.124.56, etc. Aquestes adreces s'agrupen segons la xarxa i tenen associada una màscara, que indica la mida de la xarxa en la què es troben. Quan un equip vol enviar una informació a un altre ha de conèixer la seva adreça IP. Un cop la coneix, envia la informació a aquesta adreça IP.

Per enviar-ho, primer de tot calcula si l'adreça del destí es troba en la mateixa xarxa que la seva pròpia, i sinó s'hi troba, buscarà el que s'anomena porta d'enllaç per defecte, que no és res més que un equip de la mateixa xarxa que està connectat amb d'altres xarxes o sap com arribar-hi. Així a cada interfície se li assigna una adreça IP i la màscara associada i a l'equip se li assigna una porta d'enllaç per defecte. Si necessita arribar a alguns llocs per més d'un lloc, es disposa d'una taula de rutes que indica per quina porta d'enllaç arribarà a quines xarxes.

Sobre les comunicacions a nivell d'adreça IP, en el que només es poden identificar i comunicar a nivell d'equips, cal discernir com es transporta la informació. Per això, entre altres coses, hi ha el que s'anomena protocols. Els protocols, en general, són especificacions que descriuen quina informació s'intercanvia i com. Per discernir-ne el transport hi ha protocols com TCP, UDP o ICMP (aquest és una mica especial). Generalment, s'empraran els altres dos. Així hi ha tota una sèrie de protocols que funcionen sobre TCP i una altra sobre UDP. Aquests protocols es separen en més nivells, però això ja no és temari d'aquest curs.

Els serveis de xarxa, com la web, el correu, etcètera, depenent que a l'equip servidor hi hagi un programari executant-se. Aquest programari s'anomena servidor i dona un servei. Generalment, aquest servei es dona mitjançant un protocol i el protocol s'estableix sobre un canal d'informació i mitjançant un dels protocols de transport indicats al paràgraf anterior. Aquest canal s'anomena port i sol passar que el número de port identifica el servei que es dona. Així un servidor web com l'Apache treballarà al port 80 TCP del servidor, que és on el navegador s'hi connectarà.

Configuració bàsica

Com hem vist en el punt anterior, el primer de tot és disposar d'una adreça IP, una màscara i una porta d'enllaç. Existeixen tècniques per fer que això ho obtingui l'equip automàticament, com el DHCP. Aquestes tècniques s'estan estenent molt últimament degut a què fa que connectar i desplegar grans quantitats d'equips és senzill i que no es perd capacitat de controlar el que hi ha instal·lat.

Per tal de configurar la interfície de xarxa amb DHCP, en mode text, cal que utilitzeu la comanda `dhcpclient`. Per tal que quedi fixat per cada reinici, podeu emprar l'entrada adient a `/etc/network/interfaces`.

Si heu d'assignar l'adreça ho podeu fer editant el fitxer `/etc/network/interfaces` i escrivint el contingut adient, especificant l'adreça IP, la màscara i la porta d'enllaç per defecte.

En element que directament no entra en la comunicació, però que és important és la resolució de noms. Per això cal que configureu els fitxers `/etc/resolv.conf` i `/etc/nsswitch.conf`. El primer descriu quins són els servidors DNS i com els consultarà i el segon indica en quin ordre cal consultar les bases de dades locals (a `/etc/hosts`), els DNS, o les NIS (que és un sistema que no s'empra gaire).

Eines bàsiques

Hi ha un seguit de comandes bàsiques que s'empren per comprovar l'estat de la xarxa. La primera que comentarem és el ping. La comanda ping serveix per veure si un equip respon i quan triga, com ho fa, en definitiva, es comprova la connectivitat bàsica, però no assegura que hi hagi cap servei actiu. El ping té un problema molt important i és que molts cops la manca de resposta s'associa amb què la màquina no estigui arrencada o connectada, però pot succeir que, per un o altre motiu, no pugui respondre perquè aquestes connexions estiguin denegades en algun tallafocs.

Un comanda molt útil és traceroute, que permet veure per quins salts (portes d'enllaç) passa una determinada connexió. Serveix per comprovar quin és l'últim salt abans d'arribar a un punt mort, si el hi ha. Malgrat tot, té el mateix inconvenient que el ping, els tallafocs no el deixen passar.

Una altra comanda útil és el dig, que permet llençar consultes contra el DNS i obtenir les adreces IP que corresponen a un determinat nom de dom