

# COMANDES BÀSIQUES EN GNU/LINUX

Les comandes GNU/Linux són paraules reservades del sistema que corresponen a instruccions que indiquen al sistema operatiu una tasca a realitzar, per executar aquestes accions usarem una terminal o línia de comandes.

Una **terminal** és un programa sense interfície gràfica on es pot executar les ordres en format text.

/ → directori arrel  
.  
→ directori actual

Podeu obtenir una descripció i modificadors de cada comanda usant **man** o **--help**  
ex. **man pwd** o **pwd --help**

Podeu utilitzar el tabulador per autocompletar.

## COMANDES RELACIONADES AMB DIRECTORIS

Comanda	Descripció i exemples
<b>pwd</b>	Ens mostra en pantalla l'actual directori de treball en el qual ens trobem a mode de camí absolut ('path' des del directori arrel).
<b>cd</b>	Canvi de directori. <b>cd</b> o <b>cd ~</b> : canvia directament al directori 'home' de l'usuari. «~» correspon al directori de l'usuari <b>cd directori</b> : canvia al subdirectori especificat <b>cd ..</b> : pugem un nivell (cd ../../ en pujarem dos) <b>cd /</b> : anem al directori arrel <b>cd -</b> : tornar al directori anterior
<b>ls</b>	Llista els arxius i directoris de l'actual directori de treball <b>ls -l</b> : llistat d'arxius i directoris no ocults amb informació . <b>ls -a (ls -la)</b> : fitxers ocults de sistema (els ocults s'identifiquen perquè tenen un «.» davant). La sortida de la comanda ls -la té l'aspecte següent: drwxr-xr-x 5 root root 1024 Sep 23 11:37 GNUstep -rw-r--r-- 1 root root 247 Mar 11 23:10 Xrootenv.0 El primer grup (d'estructura per exemple; drwxr-xr--) l'explicarem més endavant amb la comanda 'chmod'. La segona dada és un número que correspon al nº d'enllaços que té. Després vénen dues columnes que informen del propietari i grup al qual pertany l'arxiu o directori (en l'exemple anterior "root root"). La següent columna és la mida en bytes. Seguim amb la data ("mes dia"), hora ("xx:xx") i finalment el nom. <b>ls -R</b> : llista els arxius i directoris de manera recursiva entrant en tots els subdirectoris. <b>ls -lad info*</b> : llista els directoris (d) que comencen per 'info' (*) <b>ls -l concret</b> : mostrarà només els arxius que tinguin per nom 'concret'
<b>mkdir</b>	<b>mkdir directori</b> : es crea un directori anomenat 'directori' que penjarà del directori de treball actual <b>mkdir /cami/directori</b> : crea un directori anomenat 'directori' que penjarà de <i>camí</i> si aquest existeix <b>mkdir -p /noucami/directori</b> : crea un <i>directori</i> a la vegada que crea una nova ruta de directoris i subdirectoris en cas que no existeixin
<b>rmdir</b>	<b>rmdir directori</b> : esborra el directori especificat en cas que estigui buit Si el directori no està buit, haurem d'usar <b>rm</b> , que ara veurem <b>rmdir -p /cami/directori</b> : esborra el <i>directori</i> i tot el <i>camí</i> especificat en cas que els directoris i subdirectoris estiguin buits.

## COMANDES RELACIONADES AMB FITXERS

Comanda	Descripció i exemples
<b>rm</b>	Esborra fitxers. <b>rm <i>fitxer</i></b> : esborra el fitxer especificat. Pots esborrar des de qualsevol directori sempre que tinguis drets sobre aquell arxiu (Ex; <b>rm /usr/local/arxiu</b> ) <b>rm -i <i>fitxer</i></b> : esborra interactivament demanant confirmació <b>rm -rf <i>fitxer</i></b> : esborra forçadament (cas contrari al paràmetre -i )
<b>cat</b>	Mostra el contingut d'un fitxer. <b>cat <i>fitxer</i></b> : mostra en pantalla el contingut del <i>fitxer</i> especificat
<b>more</b>	Pagina el contingut d'un arxiu llarg o el contingut d'un directori que conté un elevat nombre d'arxius i subdirectoris. Té subcomandes associades a tecles com la barra espaiadora que avança pàgina a pàgina, la lletra b que retrocedeix pàgina, la tecla 'enter' que avança línia a línia o la lletra q que et fa sortir al prompt. <b>more <i>arxiu</i></b> : pagina el contingut de l'arxiu <i>arxiu</i>
<b>cp</b>	Copia un arxiu o directori a una altra ubicació. Si el nou arxiu té el mateix nom l'hauràs de copiar fora del directori actual. Si li canvies el nom la còpia pot ser en el mateix directori o fora d'ell. <b>cp <i>arxiu nouarxiu</i></b> : Exemple en el mateix directori <b>cp /etc/passwd /etc/passwd2</b> <b>cp <i>arxiu nouarxiu</i></b> : En directoris diferents <b>cp /alt/drivers /opt/linux/drivers</b> <b>cp <i>arxiu</i> /usr</b> : copia l'arxiu al directori /usr <b>cp /etc/hosts.allow <i>permesos</i></b> : copia el fitxer hosts.allow al directori de treball actual anomenant-lo ' <i>permesos</i> '.
<b>mv</b>	Mou la ubicació d'un fitxer o directori fent que desaparegui l'original. <b>mv /usr/local/squid/etc/squid.conf .</b> : mou el squid.conf al directori actual ( . ) <b>mv <i>arxiu1 arxiu2</i></b> : converteix l' <i>arxiu1</i> en <i>arxiu2</i> (al mateix directori) → el reanomena <b>mv *.zip ../perl</b> : mou tots els arxius .zip al directori perl (que es troba pujant dos nivells i baixant un fins a 'perl')
<b>cmp</b>	Compara dos arxius informant de totes les seves diferències. <b>cmp <i>arxiu1 arxiu2</i></b> : compara l' <i>arxiu1</i> amb l' <i>arxiu2</i> (pots indicar els 'path' també)
<b>diff</b>	Compara dos arxius informant de totes les seves diferències. És semblant a la comanda cmp però amb format de sortida diferent <b>diff <i>arxiu1 arxiu2</i></b> : compara l' <i>arxiu1</i> amb l' <i>arxiu2</i> (pots indicar els 'path' també)
<b>find</b>	Cercador de fitxers a partir del directori que indiquem. <b>find / -name access.log</b> : busca l'arxiu access.log a partir del directori arrel (/) <b>find . -size 35 -exec ls -l {} \;</b> : cerca al directori actual els arxius de 35 bytes i els mostra Aquesta comanda també admet metacaràcters (find / -name "*iss*" -print) <b>find . -type d</b> : cerca tots els directoris del directori actual <b>find . -user root</b> : cerca tots fitxer que el seu propietari sigui root del directori actual
<b>echo</b>	Permet escriure un text. <b>echo "hola"</b> : mostra per pantalla 'hola' <b>echo "hola" &gt;&gt; <i>hola.txt</i></b> : escriu 'hola' al fitxer <i>hola.txt</i>
<b>grep</b>	Busca una cadena dins d'un fitxer. <b>grep "jbot" /etc/passwd</b> : cerca la cadena 'jbot' al fitxer /etc/passwd imprimint la línia <b>grep -n 'mestres' /home/base_dades</b> : busca la cadena mestres dins del fitxer de nom <i>base_dades</i> indicant a la sortida el nom de l'arxiu i el n° de línia on es troba <b>grep -v "root" /etc/shadow &gt; \$HOME/usuarios_pelats</b> : busca totes les línies del fitxer /etc/shadow que <u>no tinguin</u> (-v) la cadena 'root' i ho redirecciona (poses dins) del fitxer <i>usuarios_pelats</i> que es troba dins de /home/usuari (\$HOME) <b>ls -la   grep ".txt"</b> : busca i llista tots els fitxers '.txt' del directori actual de treball
<b>ln</b>	Enllaça un arxiu existent en un altre (al mateix o una altre directori) Per defecte són hard link (si s'esborra l'original la informació no es perd) <b>ln -s <i>arxiu_vell arxiu_nou</i></b> : enllaça (link) l'arxiu que vulguem en un altre que haurem creat. L'opció -s és per crear un soft link (similar a un accés directe) i pugui ser visible a la sortida de la comanda 'ls' com apuntem a continuació: <b>lrwxr--r-- 1 root 243 Apr 23 16:00 arxiu_nou → arxiu_vell</b> <b>ln -s /etc/rc.d/init.d/httpd . S35web</b> : fa un enllaç simbòlic del fitxer httpd (que és un script d'arrencada) en el directori actual ( . ) amb el nom S35web

touch	touch arxiu: crea l'arxiu <i>arxiu</i> Si l'arxiu ja existeix actualitza la data de modificació																										
chmod	<div><div>drwxrwxrwx</div><div><div>d = Directory</div><div>r = Read</div><div>w = Write</div><div>x = Execute</div></div><div><div>chmod 777</div><div><div></div><div></div><div></div></div><div><div>rwx   rwx   rwx</div><div>Owner   Group   Others</div></div></div><table><tr><td>7</td><td>rwx</td><td>111</td></tr><tr><td>6</td><td>rw-</td><td>110</td></tr><tr><td>5</td><td>r-x</td><td>101</td></tr><tr><td>4</td><td>r--</td><td>100</td></tr><tr><td>3</td><td>-wx</td><td>011</td></tr><tr><td>2</td><td>-w-</td><td>010</td></tr><tr><td>1</td><td>--x</td><td>001</td></tr><tr><td>0</td><td>---</td><td>000</td></tr></table></div>	7	rwx	111	6	rw-	110	5	r-x	101	4	r--	100	3	-wx	011	2	-w-	010	1	--x	001	0	---	000	<p>Assigna els permisos d'un directori o fitxer/s (com a root o en els que tu tinguis drets)</p> <p><b>chmod 764 sendmail:</b> assigna a l'arxiu <i>sendmail</i> els permisos de lectura, escriptura i execució (7) per al seu usuari, de lectura i escriptura (6) per als usuaris del seu grup i de només lectura (4) per a la resta [ -rwxrw-r-- ]</p> <p>Els 10 caràcters d'aquesta primera columna que apareix en fer un 'ls -l' es desglosa en:</p> <p><u>primer caràcter</u>; ens indica si és de tipus arxiu (-), directori (d), enllaç (l),...</p> <p><u>primer grup de 3 caràcters</u>; són els permisos del propi usuari sobre l'arxiu o directori. Com que corresponen les tres posicions en binari, si una posició està activada es correspondrà amb un 1 i si està desactivada correspondrà a un 0. Així per exemple si trobem '--x' correspondrà a 001 per tant a 1, si trobem 'rw-' serà un 100 que en decimal correspon al nº 4, si fós 'rw-' seria el 110 que correspon al 6, el 'rwx' és el 7.</p> <p><u>segon grup de 3 caràcters</u>; fa referència als permisos que tenen els usuaris que pertanyen a aquell grup. La filosofia dels números és la mateixa que en la tríada anterior</p> <p><u>tercer grup</u>; són els permisos per a la resta de la gent</p> <p>Existeix una altra manera d'assignar permisos:</p> <p><b>chmod ugo+x fitxer:</b> a <i>fitxer</i> li donem permís d'execució (x) al propi usuari (u), al grup (g) i als altres (o -other-)</p>	
	7	rwx	111																								
6	rw-	110																									
5	r-x	101																									
4	r--	100																									
3	-wx	011																									
2	-w-	010																									
1	--x	001																									
0	---	000																									
chown	<p>El root o l'usuari sobre on tingui drets poden canviar o assignar un propietari a un fitxer o directori.</p> <p><b>chown web /home/httpd/html:</b> web serà el propietari del directori <i>html</i></p> <p><b>chown -R nobody /cache:</b> nobody serà el propietari (de manera recursiva '-R' a tots els fitxers i subdirectoris) del directori <i>/cache</i></p> <p><b>chown root:other index.html:</b> canviem directament el propietari i el grup a <i>index.html</i></p>																										
chgrp	<p>Canvia el grup d'un fitxer o directori. L'exemple anterior també serveix per a 'chgrp'</p> <p><b>chgrp nougrup fitxer:</b> canvia el grup que li pertany a <i>fitxer</i> per un nou grup (nougrup)</p>																										
umask	<p>Defineix els permisos a mode de màscara que donarem als fitxers que creem a partir d'aquest moment. Per la seva característica s'entén que el valor que posarem en aquesta comanda sortirà de restar a 777 els permisos que haguem decidit.</p> <p><b>umask 017:</b> defineix que a partir d'ara els permisos dels fitxers seran 760 (777-760=017)</p>																										
head	<p>Mostra les primeres línies (normalment les 10 primeres) d'un fitxer</p> <p><b>head fitxer:</b> mostra les primeres línies del <i>fitxer</i></p> <p><b>head -50 .././routers:</b> mostra les 50 primeres línies del fitxer <i>routers</i> (via 'path' relatiu)</p>																										
tail	<p>Mostra les últimes línies d'un fitxer.</p> <p><b>tail logfile:</b> mostra les últimes (normalment 10) línies del fitxer <i>logfile</i></p> <p><b>tail -f /usr/local/squid/logs/access.log:</b> mostra de forma dinàmica (-f) les últimes línies del fitxer <i>access.log</i> (especificat amb 'path' absolut)</p> <p><b>tail +10 fitxer:</b> mostra a partir de la línia 10 del <i>fitxer</i></p>																										
sort	<p>Ordena/Classifica els registres d'un/s fitxer/s</p> <p><b>sort arxiu:</b> ordena els registres de l'<i>arxiu</i></p>																										
wc	<p>Mostra en pantalla el número de caràcters, paraules i línies de l'arxiu especificat</p> <p><b>wc alumnes :</b> mostra les característiques anteriors del fitxer <i>alumnes</i></p> <p><b>wc -c log.2 :</b> ens mostra la mida en bytes de l'arxiu <i>log.2</i></p>																										
zip	<p>Comprimeix arxius i directoris en format '.zip'</p> <p><b>zip arxiuzip arxius_a_comprimir :</b> comprimirà en un arxiuzip els arxius_a_comprimir (i directoris) que especifiquem</p> <p><b>unzip arxiu.zip :</b> descomprimirà el fitxer <i>arxiu.zip</i></p>																										

> , >>	<p>Redirigim la sortida des de la terminal a un fitxer.</p> <p><b>history &gt; comandes.txt:</b> Creem l'arxiu <i>comandes.txt</i> i hi guardem el resultat de la comanda <i>history</i></p> <p><b>history &gt;&gt; comandes.txt:</b> Afegim al fitxer <i>comandes.txt</i> el resultat de la comanda <i>history</i> (si no existeix el crearà)</p>
(pipe)	<p>Permet encadenar la sortida d'una instrucció amb l'entrada d'una altra.</p> <p><b>history   grep «rm»:</b> Ens mostra l'historial de comandes on hem usat «rm»</p>
./	<p>Executa un fitxer que ha de tenir característiques prèvies d'executable (x)</p> <p><b>./httpd stop</b> : executa l'arxiu (script en aquest cas) 'httpd' si estem en el mateix directori de treball on es troba ell</p> <p>Per executar programes (arxius executables) com l'httpd anterior hi ha 3 maneres:</p> <p><b>httpd</b> : indicaria que es troba en el path (camí)</p> <p><b>./httpd</b> : si està en el mateix directori de treball</p> <p><b>/etc/rc.d/init.d/httpd</b> : indicant el camí absolut també el podrem executar</p> <p><b>/opt/netscape/netscape &amp;</b> : executarem (obrirem) el Netscape mantenint (&amp;)també activa la consola virtual des d'on l'haguem 'llençat'</p> <pre>f.sh #!/bin/bash # ./f.sh pwd</pre>

## BIBLIOGRAFIA I WEBGRAFIA

«Eugeni Alonso». [http://personal.biada.org/~ealonso/matin/linux/comandes\\_basiques.html](http://personal.biada.org/~ealonso/matin/linux/comandes_basiques.html)



Autor: Xavier Baubés Parramon  
Aquest document es llicència sota Creative Commons versió 4.0.  
Es permet compartir i adaptar el material però reconeixent-ne l'autor original.