

Nous allons d'abord voir comment afficher le contenu d'un fichier. Il y a en gros deux commandes basiques sous Linux qui permettent de faire cela :

- `cat` ;
- `less`.

Aucune de ces commandes ne permet d'éditer un fichier, elles permettent juste de le **voir**. Nous étudierons l'édition plus tard, ça mérite au moins un chapitre entier.

Mais... pourquoi deux commandes pour afficher un fichier ? Une seule n'aurait pas suffi ?

En fait, chacune a ses spécificités ! Nous allons les voir dans le détail.

Pour nos exemples, nous allons travailler sur un fichier qui existe déjà : `syslog`. Il se trouve dans le dossier `/var/log`. Commencez par vous y rendre :

```
mateo21@mateo21-desktop:~$ cd /var/log
```

Ce dossier contient plusieurs fichiers de **log**, c'est-à-dire des fichiers qui gardent une trace de l'activité de votre ordinateur. Vous pouvez en faire la liste si vous le voulez, en tapant `ls` :

```
mateo21@mateo21-desktop:/var/log$ ls
acpid                daemon.log.0      kern.log.0         scrollkeeper.log.2
acpid.1.gz           daemon.log.1.gz   kern.log.1.gz     syslog
acpid.2.gz           daemon.log.2.gz   kern.log.2.gz     syslog.0
acpid.3.gz           daemon.log.3.gz   kern.log.3.gz     syslog.1.gz
acpid.4.gz           debug             lastlog           syslog.2.gz
apparmor             debug.0           lpr.log           syslog.3.gz
apport.log           debug.1.gz        mail.err          syslog.4.gz
apport.log.1         debug.2.gz        mail.info         syslog.5.gz
apport.log.2.gz      debug.3.gz        mail.log          syslog.6.gz
apport.log.3.gz      dist-upgrade      mail.warn         udev
apport.log.4.gz      dmesg             messages          unattended-upgrades
apport.log.5.gz      dmesg.0           messages.0        user.log
apt                  dmesg.1.gz        messages.1.gz     user.log.0
auth.log             dmesg.2.gz        messages.2.gz     user.log.1.gz
auth.log.0           dmesg.3.gz        messages.3.gz     user.log.2.gz
auth.log.1.gz        dmesg.4.gz        news              user.log.3.gz
auth.log.2.gz        dpkg.log          popularity-contest uucp.log
auth.log.3.gz        dpkg.log.1        popularity-contest.0 wtmp
bittorrent           dpkg.log.2.gz     popularity-contest.1.gz wtmp.1
boot                 faillog           popularity-contest.2.gz wvdialconf.log
bootstrap.log        fontconfig.log    popularity-contest.3.gz Xorg.0.log
bttmp                fsck              pycentral.log     Xorg.0.log.old
bttmp.1              gdm               samba
cups                  installer         scrollkeeper.log
daemon.log           kern.log          scrollkeeper.log.1
```

Le fichier sur lequel nous allons travailler, `syslog`, contient des informations de log de ce qui s'est passé récemment sur l'ensemble de votre ordinateur.

Vous noterez qu'il est fréquent de voir des fichiers sans extension sous Linux. Notre fichier s'appelle `syslog` tout court, et non pas `syslog.txt` ou `syslog.log` comme on pourrait avoir l'habitude de le voir sous Windows.

Un fichier sans extension peut être ouvert et lu sans aucun problème comme n'importe quel autre fichier.

cat : afficher tout le fichier

La commande `cat` permet d'afficher tout le contenu d'un fichier dans la console d'un coup.

Il vous suffit d'indiquer en paramètre le nom du fichier que vous voulez afficher, en l'occurrence `syslog` :

```
mateo21@mateo21-desktop:/var/log$ cat syslog
Nov 14 00:44:23 mateo21-desktop syslogd 1.4.1#21ubuntu3: restart.
Nov 14 00:44:23 mateo21-desktop anacron[6725]: Job `cron.daily' terminated
Nov 14 00:44:23 mateo21-desktop anacron[6725]: Normal exit (1 job run)
Nov 14 00:44:25 mateo21-desktop NetworkManager: <info> eth1: link timed out.
Nov 14 00:44:51 mateo21-desktop NetworkManager: <info> eth1: link timed out.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <debug> [1194997508.332093] nm_device_802_11_wireless_get_acti
vation_ap(): Forcing AP 'WIFI'
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> User Switch: /org/freedesktop/
NetworkManager/Devices/eth1 / WIFI
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> Deactivating device eth1.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1): cancelling...
```

```

Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1) cancellation handler scheduled...
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1): waiting for device to cancel activa
tion.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1) cancellation handled.
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> Activation (eth1): cancelled.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> SUP: sending command 'DISABLE_NETWORK 0'
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> SUP: response was 'OK'
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> SUP: sending command 'AP_SCAN 0'
Nov 14 00:45:16 mateo21-
desktop NetworkManager: nm_act_request_get_ap: assertion `req != NULL' failed
Nov 14 00:45:16 mateo21-
desktop NetworkManager: nm_act_request_get_stage: assertion `req != NULL' failed
Nov 14 00:45:16 mateo21-desktop NetworkManager: <info> eth1: link timed out.
Nov 14 00:45:21 mateo21-
desktop NetworkManager: nm_act_request_get_ap: assertion `req != NULL' failed
Nov 14 00:45:21 mateo21-desktop NetworkManager: ap_is_auth_required: assertion `ap !
= NULL' failed
Nov 14 00:45:21 mateo21-desktop NetworkManager: <info> Activation (eth1/
wireless): association took too long (>120s), asking for new key.
Nov 14 00:45:21 mateo21-
desktop NetworkManager: nm_dbus_get_user_key_for_network assertion `req !
= NULL' failed
Nov 14 00:47:45 mateo21-desktop init: tty4 main process (4517) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty5 main process (4518) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty2 main process (4520) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty3 main process (4522) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty1 main process (4524) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty6 main process (4525) killed by TERM signal
Nov 14 00:47:46 mateo21-desktop avahi-daemon[5390]: Got SIGTERM, quitting.
Nov 14 00:47:48 mateo21-desktop exiting on signal 15
Nov 14 00:48:42 mateo21-desktop syslogd 1.4.1#21ubuntu3: restart.

```

...

Comme le fichier est très gros (il fait plusieurs centaines de lignes), je n'ai pas copié tout ce qui s'est affiché dans ma console. Ne vous étonnez pas si vous voyez tout s'afficher d'un coup : c'est normal, c'est le but. La commande `cat` vous envoie tout le fichier à la figure. Elle est plus adaptée lorsque l'on travaille sur de petits fichiers que sur des gros, car dans un cas comme celui-là, on n'a pas le temps de lire tout ce qui s'affiche à l'écran.

Il y a peu de paramètres vraiment intéressants à utiliser avec la commande `cat`, car c'est une commande somme toute très basique. On notera quand même le paramètre `-n` qui permet d'afficher les numéros de ligne :

```

mateo21@mateo21-desktop:/var/log$ cat -n syslog
 1      Nov 14 00:44:23 mateo21-desktop syslogd 1.4.1#21ubuntu3: restart.
 2      Nov 14 00:44:23 mateo21-desktop anacron
[6725]: Job `cron.daily' terminated
 3      Nov 14 00:44:23 mateo21-desktop anacron
[6725]:      Normal exit (1 job run)
 4      Nov 14 00:44:25 mateo21-
desktop NetworkManager: <info> eth1: link timed out.
 5      Nov 14 00:44:51 mateo21-
desktop NetworkManager: <info> eth1: link timed out.
 6      Nov 14 00:45:08 mateo21-
desktop NetworkManager: <debug> [1194997508.332093]

```

...

less : afficher le fichier page par page

La commande `cat` est rapide. Trop rapide. Tout le fichier est lu et affiché d'un coup dans la console, ce qui fait que l'on n'a pas le temps de le lire s'il est très gros.

C'est là qu'une autre commande comme `less` devient vraiment indispensable. La grosse différence entre `less` et `cat`, c'est que `less` affiche progressivement le contenu du fichier, page par page. Ça vous laisse le temps de le lire dans la console. :-)

Notez qu'il existe aussi une commande très proche : `more`. Pour faire simple, la différence entre `more` et `less` c'est que `more` est vieux et possède peu de fonctionnalités, tandis que `less` est beaucoup plus puissant et rapide. Bref, utilisez `less`, mais si

vous voyez un jour quelqu'un utiliser `more`, ne soyez pas surpris.

Comment ça marche ? Eh bien la commande est très simple : `less nomdudossier`.

```
mateo21@mateo21-desktop:/var/log$ less syslog
Nov 14 00:44:23 mateo21-desktop syslogd 1.4.1#21ubuntu3: restart.
Nov 14 00:44:23 mateo21-desktop anacron[6725]: Job `cron.daily' terminated
Nov 14 00:44:23 mateo21-desktop anacron[6725]: Normal exit (1 job run)
Nov 14 00:44:25 mateo21-desktop NetworkManager: <info> eth1: link timed out.
Nov 14 00:44:51 mateo21-desktop NetworkManager: <info> eth1: link timed out.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <debug> [1194997508.332093] nm_device_802_11_wireless_get_acti
vation_ap(): Forcing AP 'WIFI'
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> User Switch: /org/freedesktop/
NetworkManager/Devices/eth1 / WIFI
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> Deactivating device eth1.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1): cancelling...
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1) cancellation handler scheduled...
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1): waiting for device to cancel activa
tion.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> Activation (eth1) cancellation handled.
Nov 14 00:45:08 mateo21-desktop NetworkManager: <info> Activation (eth1): cancelled.
Nov 14 00:45:08 mateo21-
desktop NetworkManager: <info> SUP: sending command 'DISABLE_NETWORK 0'
```

Ce qui est intéressant pour nous ici, c'est que la commande `less` a arrêté la lecture du fichier au bout de quelques lignes (la taille d'un écran de console). Cela vous laisse le temps de lire le début du fichier.

On n'a lu pour le moment que les toutes premières lignes du fichier.

Et comment lire la suite ?

Il y a quelques raccourcis clavier à connaître. ;-))

Les raccourcis basiques indispensables

Commençons par les quelques raccourcis clavier les plus indispensables, à connaître absolument.

- **Espace** : affiche la suite du fichier. La touche **Espace** fait défiler le fichier vers le bas d'un « écran » de console. C'est celle que j'utilise le plus souvent.
- **Entrée** : affiche la ligne suivante. Cela permet donc de faire défiler le fichier vers le bas ligne par ligne.
- Vous pouvez aussi utiliser la touche **Flèche vers le bas**.
- **d** : affiche les onze lignes suivantes (soit une moitié d'écran). C'est un peu l'intermédiaire entre **Espace** (tout un écran) et **Entrée** (une seule ligne).
- **b** : retourne en arrière d'un écran.
- Vous pouvez aussi appuyer sur la touche **Page Up**.
- **y** : retourne d'une ligne en arrière.
- Vous pouvez aussi appuyer sur la touche **Flèche vers le haut**.
- **u** : retourne en arrière d'une moitié d'écran (onze lignes).
- **q** : arrête la lecture du fichier. Cela met fin à la commande `less`.

La casse des caractères est importante. Ainsi, si je vous dis qu'il faut appuyer sur la touche « d », ce n'est pas un « D » majuscule (si vous essayez, vous verrez que ça ne fonctionne pas). Sous Linux, on fait souvent la différence entre majuscules et minuscules : souvenez-vous-en !

Si on tape **Espace**, on avance donc d'un écran dans le fichier :

```
Nov 14 00:47:45 mateo21-desktop init: tty4 main process (4517) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty5 main process (4518) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty2 main process (4520) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty3 main process (4522) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty1 main process (4524) killed by TERM signal
Nov 14 00:47:45 mateo21-desktop init: tty6 main process (4525) killed by TERM signal
Nov 14 00:47:46 mateo21-desktop avahi-daemon[5390]: Got SIGTERM, quitting.
Nov 14 00:47:48 mateo21-desktop exiting on signal 15
```

```
Nov 14 00:48:42 mateo21-desktop syslogd 1.4.1#21ubuntu3: restart.
Nov 14 00:48:42 mateo21-desktop kernel: Inspecting /boot/System.map-2.6.22-14-generic
Nov 14 00:48:42 mateo21-desktop kernel: Loaded 25445 symbols from /boot/
System.map-2.6.22-14-generic.
Nov 14 00:48:42 mateo21-desktop kernel: Symbols match kernel version 2.6.22.
Nov 14 00:48:42 mateo21-desktop kernel: No module symbols loaded -
kernel modules not enabled.
Nov 14 00:48:42 mateo21-desktop kernel: [ 0.000000] Linux version 2.6.22-14-
generic (buildd@palmer) (gcc version 4.1.3 20070929 (prerelease))
```

:

Quelques raccourcis plus avancés

Ce ne sont pas des raccourcis que l'on utilise tous les jours, mais ça vaut le coup de savoir qu'ils existent. :-)

- = : indique où vous en êtes dans le fichier (numéro des lignes affichées et pourcentage).
- h : affiche l'aide (toutes les commandes que je vous apprends ici, je les tire de là). Tapez q pour sortir de l'aide.
- / : tapez / suivi du texte que vous recherchez pour lancer le mode recherche. Faites Entrée pour valider. Pour ceux qui savent s'en servir, sachez que les expressions régulières sont acceptées. Je ne vais pas vous faire un cours sur les expressions régulières ici, ce serait trop long, mais il y en a un dans mon cours sur le PHP [Concevez votre site Web avec PHP et MySQL](#) dans la même collection.
- n : après avoir fait une recherche avec /, la touche n vous permet d'aller à la prochaine occurrence de votre recherche. C'est un peu comme si vous cliquiez sur le bouton « Résultat suivant ».
- N : pareil que n, mais pour revenir en arrière.

Comme vous le voyez, la commande less est très riche. On peut utiliser beaucoup de touches différentes pour se déplacer dans le fichier.

Prenez le temps de vous familiariser avec : c'est un peu perturbant au début, mais lorsque vous aurez appris à vous en servir, vous aurez déjà fait un grand pas en avant... et puis ça vous sera très utile plus tard, croyez-moi. :-)