

La commande tail

- Objet : La commande tail
- Niveau requis :
[débutant, avisé](#)
- Commentaires : permet d'afficher les dernières lignes d'un fichier
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#).
- Suivi :
[à-compléter](#)
 - Création par [MaTTuX_](#) le 17/06/2007 à 07:21
 - Testé par [milou](#) le 16/09/2015
- Commentaires sur le forum : [C'est ici](#) ¹⁾

Introduction

La commande `tail` permet de lire les dernières lignes d'un fichier.

Utilisation

```
tail /var/log/auth.log
```

[retour de la commande ci-dessus](#)

```
Jun 14 19:25:01 serveurdeb CRON[5760]: pam_unix(cron:session): session
closed for user root
Jun 14 19:25:01 serveurdeb CRON[5766]: pam_unix(cron:session): session
closed for user root
Jun 14 19:29:21 serveurdeb su[5791]: Successful su for root by mon_user
Jun 14 19:29:21 serveurdeb su[5791]: + pts/2 mon_user:root
Jun 14 19:29:21 serveurdeb su[5791]: pam_unix(su:session): session
opened for user root by admijoel(uid=1000)
Jun 14 19:29:54 serveurdeb su[5791]: pam_unix(su:session): session
closed for user root
Jun 14 19:30:01 serveurdeb CRON[5797]: pam_unix(cron:session): session
opened for user root by (uid=0)
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
opened for user root by (uid=0)
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
closed for user root
Jun 14 19:30:01 serveurdeb CRON[5797]: pam_unix(cron:session): session
closed for user root
```

Option -n

L'option -n <chiffre> détermine le nombre (<chiffre>) de lignes que l'on désire afficher.

Exemple :

Pour lire les trois dernières lignes du même fichier que précédemment, tapez :

```
tail -n 3 /var/log/auth.log
```

[retour de la commande ci-dessus](#)

```
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
opened for user root by (uid=0)
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
closed for user root
Jun 14 19:30:01 serveurdeb CRON[5797]: pam_unix(cron:session): session
closed for user root
```

Ou bien même on peut ignorer le n en le remplaçant par le chiffre ainsi :

```
tail -3 /var/log/auth.log
```

[retour de la commande ci-dessus](#)

```
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
opened for user root by (uid=0)
Jun 14 19:30:01 serveurdeb CRON[5798]: pam_unix(cron:session): session
closed for user root
Jun 14 19:30:01 serveurdeb CRON[5797]: pam_unix(cron:session): session
closed for user root
```

Merci à **francoise** sur le salon #debian-facile pour le retour de cette expérimentation !

Option -f

-f (f pour "follow", "suivre" en anglais). 😊

Ce paramètre permet de suivre la fin du fichier au fur et à mesure de son évolution.

C'est extrêmement utile pour suivre un fichier de log qui évolue souvent.

Exemple avec syslog :

```
tail -f /var/log/syslog
```

[retour de la commande ci-dessus](#)

```
Jun 14 19:55:01 serveurdeb /USR/SBIN/CRON[5886]: (root) CMD
(/root/scripts-root/backup/backup_mpeg.sh)
Jun 14 20:00:01 serveurdeb /USR/SBIN/CRON[5897]: (root) CMD
(/root/scripts-root/backup/rsync_avi.sh)
Jun 14 20:00:01 serveurdeb /USR/SBIN/CRON[5898]: (root) CMD
(/root/scripts-root/backup/backup_mpeg.sh)
```

Lien et remerciement :

- http://www.siteduzero.com/tutoriel-3-12771-manipuler-les-fichiers.html#ss_part_2

À voir :

- [CCZE](#) Une application de coloration syntaxique dédiée aux log.

Fort, non ? 😊

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:
<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:
<http://debian-facile.org/doc:systeme:tail>

Last update: **16/09/2015 19:27**

