

# history


- Objet : history
- Niveau requis : [débutant, avisé](#)
- Commentaires : *Visualiser jusqu'à l'ensemble des 500 dernières commandes que vous avez saisies.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
  - Création par [MaTTuX\\_ Mattuxalhem...](#)
  - Testé par [smolski](#) *pareil* 😊
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#)<sup>1)</sup>

## Introduction

La commande history permet de visualiser l'ensemble des 500 dernières commandes que vous avez saisies dans votre console.

- History dans le terminal <user> relève les commandes de l'user.
- History dans le terminal <root> relève les commandes de root.

## Utilisation

La commande s'utilise habituellement dans un terminal en tapant sur  pour remonter le cours des commandes précédentes.

Nous pouvons également l'invoquer textuellement en tapant :

```
history
```

et obtenir la liste numérotée des 500 dernières commandes lancées

[résultat de la commande précédente](#)

```
4 cat base.tex
5 ls -al
6 cd Desktop/
7 cd Work/
8 pwd
9 history
```

## History - Reverse-Search

Terminal ouvert, en tapant : **Ctrl+R** nous passons en :

```
(reverse-i-search)`reco':
```

Et en tapant maintenant les premières lettres de la commande recherchée, celle-ci s'autocomplete intégralement.

Pour remonter davantage dans l'historique sur le thème des premières lettres, il suffit de taper de nouveau sur **Ctrl+R** pour lister la suite des commandes similaires.

Astuce également valable dans un shell zsh. (*dixit le **captfab**...*) 😊

## !! La dernière commande

La commande **!!** permet de rappeler la dernière commande passée, tout comme **↑**

```
!!
```

[résultat de la commande précédente](#)

```
4 cat base.tex
5 ls -al
6 cd Desktop/
7 cd Work/
8 pwd
9 history
```

Normal, la dernière commande passée était bien history 😊

## ! + Numéro

La commande **!numéro** permet d'atteindre la commande à droite du numéro.

Essayez en tapant **!8**, j'obtiens ici **pwd** qui est la huitième ligne de commande en mémoire dans l'**history**.

```
!8
```

[résultat de la commande précédente](#)

```
pwd
/home/user
```

## ! + lettres

La commande **!lettres\_début\_commande\_recherchées** permet d'atteindre la commande à droite du numéro et de l'exécuter Essayez ! en tapant :

```
!ca
```

j'obtiens ici **cat base.tex** :

```
!ca
```

[résultat de la commande précédente](#)

```
cat base.tex  
/home/cobex4
```



Si on veut juste visualiser la commande sans l'exécuter, on ajoute :p après la ou les première lettre

```
!ca:p
```

```
cat base.tex
```

## Effacer history

Pour effacer l'historique, on utilise l'option -c.

```
history -c
```

```
history
```

[résultat de la commande précédente](#)

```
4 history  
cobex4@pc:~>
```

## Sauvegarder history

Pour sauvegarder votre historique dans un fichier .txt, faites simplement :

```
history > history.txt
```

Pour [lire en console](#) le fichier obtenu :

```
more history.txt
```

résultat de la commande précédente

```
1  ssh mattux@chubaka
2  ssh mattux@r2d2
3  su
4  cd backup/
5  mv .mozilla-thunderbird/ /home/mattux/
6  mv .ssh/ /home/mattux/
7  ls -la /home/mattux/
8  cd ../.ssh/
9  ls -l
10 cp -r ../backup/.ssh/ /home/mattux/
11 ls -l
```

Voilà votre fichier peut être très long tout dépend du nombre de commandes que vous avez fait. 😊

[MaTTuX\\_](#) 2009/01/14 16:30

## Importer une sauvegarde dans history

Par exemple, pour importer votre fichier **history.txt** précédemment

```
history -r history.txt
```



Cela va ajouter le contenu de **history.txt** dans le fichier **.bash\_history** en cours, pas le remplacer. Ppir cela, il faudrait effacer l'historique avant.

## Réutiliser les arguments de la commande précédente

Il est possible de réutiliser les arguments de la commande précédente. Cela peut être utile en cas de faute de frappe par exemple.

Ainsi, on rappelle tous les arguments avec !\*, le premier avec !^, le dernier avec !\$, le *nième* avec !!:n et des *nième* au *mième* avec !!:n-m.

Par exemple, je fais une faute de frappe dans une commande

```
eco Salut les fous facile
```

```
bash: eco : commande introuvable
Pour corriger
```

```
<code user>echo !*
```

```
echo Salut les fous facile  
Salut les fous facile
```

Autre exemple, je me trompe d'option. Je voulais lister tous les fichiers installés par un paquet avec `dpkg`, mais j'utilise la mauvaise option

```
dpkg -S libstdc++-dev
```

J'aurais dû utiliser l'option **-L**. Pas besoin de tout retaper

```
dpkg -L !$
```

Il est même possible de rappeler ces arguments en les modifiant. Imaginons que vous souhaitiez afficher le contenu de `monFichierAvec1NomSuperLong.txt`

```
cat monFichierAvec1NomSuperLong.txt
```

Ventrebleu, ça n'est pas le bon fichier, vous vouliez en fait le contenu de `monFichierAvec2NomSuperLong.txt` :-/ Pas besoin de tout retaper, vous pouvez réaliser l'opération avec

```
cat !!:1:s/1/2/
```

Un poil d'explications :-p !!:1 rappelle l'argument 1 de la dernière commande, donc `monFichierAvec1NomSuperLong.txt`. On demande ensuite de substituer (**s**) la première occurrence de **1** pour la remplacer par **2**

Pour aller plus loin : [la fonction de substitution avec sed](#)

## ASTUCES

### Affichage alphabétique de l'historique des commandes

*Astuce un peu sérieuse... Pour debianeux averti donc !*

Offert par **phlinux** dans le forum, là :

- <http://debian-facile.org/forum/viewtopic.php?pid=10709#p10709>

*Qu'il est mignon de nous donner cela... Merci **phlinux** !*


Habituellement, nous pouvons **remonter l'historique** de nos commandes dans le terminal avec `↑`... Simple.

Pourquoi ne pas remonter en utilisant les lettres alphabétiques débutant ces commandes ? Yop !

## Exemple

Par exemple tu écris :

```
g
```

puis tu utilises la touche  et s'affichent à la suite toutes les commandes qui commencent par "g" et qui se trouvent dans ton ~/.bash\_history..

Pareil, bien sûr, si tu tapes "ge", s'affiche > gedit /tra/la/la...touche...> .geany /super/coptère

Pour ce faire, éditer avec nano :

1- à la fin de ~/.bashrc

```
# appel alphabétique commandes
shopt -s histappend
PROMPT_COMMAND='history -a'
```

2 - dans ~/.inputrc (à créer au lieu d'utiliser /etc/inputrc)


```
"\e[A": history-search-backward
"\e[B": history-search-forward
```

Bien sûr, utilisable pour root (à condition de lui cloquer la même chose)

## Recherche par thème

Pour rechercher une commande par thème avec  vous écrivez dans ~/.inputrc (à créer au lieu d'utiliser /etc/inputrc) ceci :

```
"\e[A": history-search-backward
"\e[B": history-search-forward
set show-all-if-ambiguous on
set completion-ignore-case on
```

Il vous suffit alors d'autocompléter par la touche  les 3 premières lettres de la commande recherchée et il ne sera proposé par history que toutes les commandes commençant uniquement par ces lettres à chaque appui.

Exemple :

```
cp dijon_130419/dijon...
cp site...
cp vignettes/
...
```

Redémarrer le terminal pour que l'effet se passe... ;)

## Inhiber ou valider l'historique des commandes dans une session shell

Vous ne souhaitez pas qu'une commande ou une liste de commandes se retrouve dans l'historique des commandes.

Pour faire cela il suffit de taper la commande suivante :

```
set +o history
```

A partir de là plus aucune commande ne sera enregistré dans l'historique.

C'est pratique par exemple si vous tapez en clair un mot de passe.

Pour valider à nouveau l'enregistrement de l'historique des commandes, il suffit de taper la commande :

```
set -o history
```

## Remerciements

Source proposée conjointement par DosXX et phreg :

- [L'ami korben !](#)
- [Une astuce avec history.](#)

Voir aussi la commande **Reverse-Search** en haut du tuto pour faire la même chose au clavier sans fichier à ajouter !

## Lister history

Pour obtenir la liste historique du thème recherché, tapez :

```
history | grep themerecherche
```

Oui c'est du pur geek façon l'patron **MaTTuX\_** ! 😊

## Rechercher dans l'historique Bash avec Ctrl+R

Exemple : Dans votre terminal si vous tapez Ctrl + r, vous obtiendrez :

[romain@arch\_machine ~]\$ (reverse-i-search)`': Ensuite tapez un mot clé correspondant à ce que vous pourriez avoir envie de rechercher dans l'historique :

Ces derniers jours, j'ai utilisé des lignes de commandes qui comportées « gcc », si je tape « gcc » il m'affiche :

[romain@arch\_machine ~]\$ (reverse-i-search)`: gcc -g test.c Ci-dessus, on peut voir qu'il m'affiche une ligne commande entière (avec les arguments) qui contient l'expression « gcc ». Cette ligne de commande correspond à la dernière ligne de commande que j'ai tapé et qui contenait le mot clé gcc.

En appuyant une deuxième fois sur Ctrl + R, je remonte dans l'historique Bash enregistré. Il m'affiche :

[romain@arch\_machine ~]\$ (reverse-i-search)`: gcc test.c Ci-dessus, il m'affiche une autre ligne de commande que j'ai tapé dernièrement et qui contient également l'expression « gcc ».

En appuyant successivement sur CTRL + r vous pouvez remonter dans l'historique des commandes.

Un autre exemple, si je tape Ctrl +r vfath, il m'affiche :

[romain@arch\_machine ~]\$ (reverse-i-search)`: cd /mnt/vfath Il a trouvé la dernière commande que j'avais tapé et qui comprenait le mot clé vfath.

Si vous souhaitez sélectionner la ligne de commande qu'il vous propose, rien de plus simple, soit vous appuyez sur la touche entrée pour exécuter la commande, soit vous l'éditez directement. Pour l'éditer, déplacer le curseur avec les flèches droite et gauche. Pour annuler appuyer sur Ctrl +c.

Remerciements au site: <http://www.commandeslinux.fr/rechercher-dans-lhistorique-bash-avec-ctrl/>

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:programmation:bash:history>

Last update: **24/10/2022 22:28**

