

Variables d'environnement

Qu'est-ce qu'une variable d'environnement ?

Une variable d'environnement est une valeur dynamique chargée en mémoire, elle est accessible par plusieurs programmes du système d'exploitation.

Ces variables sont chargées d'une façon automatique lors de démarrage.

Trouver vos variables d'environnement :

Comment identifier les valeurs courantes de chacune de nos variables d'environnement ?

Une façon de le faire est d'utiliser la commande **env**, cette commande affiche toutes les variables d'environnement.

```
esprit@esprit-virtual-machine: $ env
SHELL=/bin/bash
SESSION_MANAGER=local/esprit-virtual-machine:@/tmp/.ICE-unix/1723,unix/esprit-virtual-machine:/tmp/.ICE-unix/1723
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/ubuntu:/etc/xdg
SSH_AGENT_LAUNCHER=gnome-keyring
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
LANGUAGE=en_US:
LC_ADDRESS=ar_TN.UTF-8
GNOME_SHELL_SESSION_MODE=ubuntu
LC_NAME=ar_TN.UTF-8
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
LC_MONETARY=ar_TN.UTF-8
GTK_MODULES=gail:atk-bridge
PWD=/home/esprit
XDG_SESSION_DESKTOP=ubuntu
LOGNAME=esprit
XDG_SESSION_TYPE=wayland
SYSTEMD_EXEC_PID=1753
XAUTHORITY=/run/user/1000/.mutter-Xwaylandauth.MT0BS2
HOME=/home/esprit
USERNAME=esprit
```

Les variables d'environnement les plus courantes :

- **PATH** : indique la liste des chemins où trouver les programmes à lancer.
- **SHELL** : indique l'interpréteur shell utilisé par défaut.
- **HOME** : contient le chemin absolu vers le répertoire personnel de l'utilisateur connecté.
- **LOGNAME** : pour le nom de l'utilisateur
- **HISTFILE** : pour le fichier historique.

- **HISTSIZE** : pour la limite de commandes historiques accessibles.

Afficher la valeur d'une variable :

Lorsque vous fournissez le nom d'une variable d'environnement à la commande **echo** et que vous le précédez d'un **\$** pour indiquer qu'il s'agit d'une variable, l'interpréteur de commandes affichera sa valeur.

```
esprit@esprit-virtual-machine:~$ echo $PATH  
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin  
esprit@esprit-virtual-machine:~$
```

Créer de nouvelles variables d'environnement :

Vous pouvez ajouter vos propres variables personnalisées à votre environnement. La méthode la plus simple consiste à utiliser le caractère **=**. La chaîne de caractères à gauche sera le nom de votre nouvelle variable et la chaîne de caractères à droite sera sa valeur.

```
esprit@esprit-virtual-machine:~$ myvar=esprit  
esprit@esprit-virtual-machine:~$ echo $myvar  
esprit
```

La variable **myvar** créée ne sera disponible que localement, dans la session actuelle du shell.

Si vous lancez un nouveau shell ou si vous fermez la session en utilisant **exit**, la variable **myvar** ne sera plus disponible.

C'est pourquoi il est nécessaire de déclarer la variable **myvar**. Voyons comment faire !

Note :

- Il n'est pas nécessaire de nommer les variables d'environnement en majuscules sous Linux, mais c'est une convention couramment adoptée.
- Ne mettez pas d'espace avant ou après l'opérateur **=**.

Déclarer les variables d'environnement :

Pour mettre une variable à la disposition des sous-processus, il faut la transformer d'une variable locale en une variable d'environnement.

Cela se fait par la commande **export**.

```
$ export myvar
```

On peut aussi créer et déclarer à la fois la variable d'environnement en utilisant la commande **export**.

```
$ export myvar=esprit
```

Note:

- Assurez-vous de ne pas utiliser \$ lorsque vous exécutez la commande export car vous voulez passer le nom de la variable au lieu de son contenu.
- Cette variable sera disponible uniquement dans le shell actuel et les processus enfants. Si vous voulez rendre la variable permanente, vous devrez l'ajouter à un fichier de configuration, comme .bashrc, .bash_profile, ou /etc/environment, selon vos besoins.

Supprimer des variables d'environnement :

Vous voulez savoir comment faire le ménage dans toutes ces variables éphémères que vous avez créées ?

Solution 1 : La première solution consiste tout simplement à fermer votre shell parent ou à redémarrer votre machine.

Solution 2 : On utilise la commande **unset**

```
$ unset myvar  
$ echo $myvar
```