

\$\$

script linux



En Bash, \$\$ est une variable qui contient le PID (Process ID) du processus en cours d'exécution. Cette variable peut être utilisée pour identifier de manière unique un processus ou pour contrôler les processus enfants créés à partir du script.

Par exemple, vous pouvez utiliser \$\$ pour créer un fichier journal unique pour chaque instance de votre script Bash en utilisant le PID comme partie du nom du fichier.

Notez cependant que le PID est une valeur dynamique qui change chaque fois que le processus est exécuté, donc \$\$ ne doit être utilisé que dans le contexte du processus en cours d'exécution.

Pour utiliser "\$\$" correctement dans le contexte du processus en cours d'exécution, vous pouvez le stocker dans une variable dès le début de votre script Bash. De cette façon, la variable contiendra le PID du processus en cours d'exécution tout au long du script, même si le PID change pendant l'exécution du script.

Voici un exemple de code Bash montrant comment stocker "\$\$" dans une variable :

```
#!/bin/bash

# stocker le PID dans une variable
current_pid=$$

# afficher le PID
echo "Le PID du processus en cours d'exécution est : $current_pid"

# faire quelque chose d'autre dans le script...
```

Ensuite, vous pouvez utiliser la variable current_pid tout au long de votre script pour référencer le PID du processus en cours d'exécution.

Notez que si votre script crée des processus enfants, le PID de ces processus enfants sera différent du PID du processus parent. Dans ce cas, vous pouvez utiliser \$\$ pour référencer le PID du processus parent, et utiliser un autre mécanisme pour référencer le PID des processus enfants, comme par exemple en stockant le PID de chaque processus enfant dans un tableau ou un fichier.

Pour connaître le PID du processus parent dans un script Bash, vous pouvez utiliser la variable spéciale \$PPID. Cette variable contient le PID du processus parent qui a créé le processus en cours d'exécution.

Voici un exemple de code Bash qui utilise \$PPID pour afficher le PID du processus parent :

```
#!/bin/bash
```

```
# stocker le PID du processus parent dans une variable
parent_pid=$PPID

# afficher le PID du processus parent
echo "Le PID du processus parent est : $parent_pid"

# faire quelque chose d'autre dans le script...
```

Notez que \$PPID ne fonctionne que si le processus en cours d'exécution a été créé par un processus parent dans le même système d'exploitation. Si le processus en cours d'exécution a été créé par un autre système d'exploitation ou un autre type de processus, \$PPID ne sera pas défini ou aura une valeur incorrecte.

From:

<https://www.abonnel.fr/> - **notes informatique & technologie**

Permanent link:

https://www.abonnel.fr/informatique/linux/commandes/double_dollar

Last update: **2023/02/18 17:32**

