Scripting Shell

Module 02 – Écriture des scripts



1



Objectifs

- Créer un fichier de script Bash
- Le rendre exécutable
- Commenter son code



- Les scripts Shell sont des fichiers texte dans lesquels sont stockées des suites de commandes. Ces fichiers peuvent être créés et modifiés par l'éditeur de texte vim.
- Lors de l'interprétation du script par le Shell, les commandes sont lues et interprétées séquentiellement.
- Une ligne de commentaire commence par le caractère #.
 Ce même caractère peut être placé derrière le dernier élément d'une ligne à condition qu'il soit précédé d'au moins un espace.



3

Scripting Shell **Écriture des scripts**

Le shebang

Afin d'éviter des erreurs, il est conseillé de spécifier le Shell à utiliser grâce au shebang.



- Il doit être écrit sur la première ligne du script.
- Il est matérialisé au moyen de la chaîne #! suivie du chemin du Shell à utiliser (exemple : #!/bin/bash).
- Il est possible d'indiquer le SH, le KSH, le Python, le Perl, ou tout autre langage interprété.
- S'assurer du chemin spécifié, en utilisant par exemple la commande which.



4

Scripting Shell **Écriture des scripts**

Commenter

- Il est important de commenter ses scripts, afin de :
 - Rendre les modifications a posteriori plus simples.
 - Les rendre plus compréhensibles pour d'autres personnes.
- Tout texte inséré après le signe dièse (#) ne sera pas interprété.

```
$ vi monscript.sh

#!/bin/bash

# ce script affiche la phrase « Hello World ! »

echo Hello world ! # on peut commenter après une instruction
```



5

Scripting Shell **Écriture des scripts**

Commenter

• L'affectation d'un entête de script complet fait partie des bonnes pratiques.

```
#!/bin/bash

# FILE: ~toris.sh

# USAGE: ./~toris.sh

# DESCRIPTION: une cli pour la gestion des flux

# OPTIONS: ---

# REQUIREMENTS: ---

# BUGS: ---

# NOTES: ---

# AUTHOR: Penthium2, contact@viperr.org

COMPANY: viperr

# CREATED: 05.03.2015

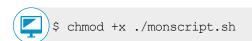
# REVISION: 02.08.2019
```



Scripting Shell **Écriture des scripts**

Fichier exécutable

Il faudra affecter la permission execute pour rendre le script exécutable.







7

Scripting Shell **Écriture des scripts**

Débuguer

Les erreurs dans les scripts ne sont pas toujours faciles à repérer. Plusieurs méthodes peuvent être utilisées afin de les identifier.

• L'exécution du script en mode *trace* permet d'exécuter une à une les commandes de celui-ci tout en les affichant précédées du signal plus (+).





Scripting Shell **Écriture des scripts**

Débuguer

• L'intégration dans le script de commandes de débogue se traduit par l'ajout temporaire de commandes permettant d'analyser une situation.

```
echo "tapez votre nom"
read nom

## echo de debug avec pause :
echo "valeur de la variable nom : $nom"
read # pause avant de poursuivre le déroulement du script
## fin du debug
echo "Bienvenue $nom sur la machine $HOSTNAME"
```

• Cela permet de contrôler le résultat d'exécution de certaines parties du script et d'identifier des sources potentielles de dysfonctionnement.



9

Scripting Shell **Écriture des scripts**

Débuguer

• Les commandes de débogue doivent être commentées avant la mise en production :

```
echo "tapez votre nom"
read nom

## echo de debug avec pause :
#echo "valeur de la variable nom : $nom"

#read

## fin du debug
echo "Bienvenue $nom sur la machine $HOSTNAME"
```



