

Sistemas Operacionais

Introdução

Linux Shell Script

Um Shell Script é um arquivo que contém comandos que poderíamos escrever e executar individualmente no prompt de comando de um terminal Linux.

Porém, ao colocarmos esses comandos num único arquivo (o shell script), torna-se possível executar apenas esse script, o que fará com que os comandos armazenados no script sejam executados.

1. Primeira linha do Shell Script

Costuma-se informar na primeira linha do shell script, a localização do interpretador dos comandos do shell. O mais utilizado atualmente é o bash, de forma que a 1ª linha do script fica assim:

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

2. Comentários

Quando quiser acrescentar comentários ao script, coloque-os precedidos por um #
Por exemplo:

```
# Isto é um comentário
```

3. Comandos

Qualquer comando que costumamos escrever no prompt de comando do terminal pode ser colocado no shell script
Por exemplo:

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

```
echo "Meu primeiro shell script"
```

```
echo "O diretório corrente é"  
pwd
```

```
echo "O conteúdo do diretório corrente é"
```

```
ls
```

4. Execução do arquivo shell script
 - a. Coloque a extensão .sh no nome do arquivo shell script
 - b. Ele não será, por padrão, um arquivo executável
 - c. Altere a sua permissão para que fique executável

```
chmod +x <nome-do-arquivo>
```

- d. Digite no terminal, no diretório em que o shell script se encontra

```
./nome-do-arquivo.sh
```

5. Verificando se o java está instalado:
Comando which java:
 - retorna /usr/bin/java - se o java estiver instalado
 - não retorna nada se o java não estiver instalado

6. Comando grep
Utilizado para pesquisar uma string em um arquivo texto
Exemplo:

```
grep prog lista  
(procura as ocorrências de prog no arquivo lista)
```

Opção -q do grep: não exibe nada na tela e devolve um código de retorno igual a zero, se encontrou a ocorrência ou um código de retorno diferente de zero.

Opção -c do grep: exibe em quantas linhas encontrou ocorrências da string procurada

7. Direcionador | (pipe)
Direciona a saída de um comando para a entrada de outro comando
Exemplo:

```
which java | grep /usr/bin/java  
(procura as ocorrências de usr/bin/java na saída do comando which java)
```

8. \$?
Muito usado em Shell Script, o \$? Representa o código de retorno do último comando executado.

9. If
Dentro de um arquivo Shell Script, é possível utilizar a instrução if, que deve ser finalizada por um **fi** (if contrário)
Exemplo (ATENÇÃO: PRESICA TER O ESPAÇO APÓS O [E ANTES DO]):

```
if [ $? = 0 ]  
Then echo "OK"  
fi
```

(se código de retorno do último comando executado for igual a zero, então exibe OK)

```

if [ $? = 0 ]
then echo "OK"
else echo "Não OK"
fi

```

(se código de retorno do último comando executado for igual a zero, então exibe OK, senão exibe não OK)

No comando if, os operadores de comparação são:

| Comparação de Strings | | Comparação Numérica | |
|-----------------------|-------------|---------------------|------------------|
| = | É igual | -lt | É menor que |
| != | É diferente | -gt | É maior que |
| -n | É não nula | -le | É menor ou igual |
| -z | É nula | -ge | É maior ou igual |
| | | -eq | É igual |
| | | -ne | É diferente |

10. Exemplo de shell script que verifica se o java está instalado

```

#!/bin/bash

which java | grep -q /usr/bin/java

if [ $? = 0 ]
then echo "OK"
else echo "Não OK"
fi

```

11. README.TXT

É interessante vocês criarem um arquivo com esse nome, ou LEIAME.TXT, para ser distribuído juntamente com o script de instalação e o arquivo .jar. Nesse arquivo, coloque as instruções para instalação, como por exemplo, que o script precisa ser executado por um usuário sudoer.