



Certified Tech Developer

The Ultimate Degree

Infraestrutura I

Introdução

No exercício a seguir, vamos criar um shell script para realizar backup de arquivos no nosso linux.

Objetivos

- Backup de arquivos e pastas na VM linux de Infra I.

Exercício 2

Crie um script em shell para realizar um backup. O arquivo de backup gerado deverá conter o seguinte formato: backup-data_de_criação.tar.bz2. Junto a esse arquivo deverá ser gerado um arquivo de log chamado log BACKUP onde deverá conter as seguintes informações:

```
Execução do Backup - dd/mm/aaaa  
Horário de início - hh:mm:ss  
Arquivos compactados:  
Caminho completo do arquivo1  
Caminho completo do arquivo2  
Caminho completo do arquivo3...  
Diretório de backup: /$HOME/username/backup/  
Horário da Finalização do backup - 20:26:08
```



O diretório onde ficará armazenado o backup deverá ser o "\$HOME/backup/", assim como o arquivo de log. O diretório a ser compactado é de sua escolha. Deverá ter uma condição (0 ou 1) para informar se deseja (1) ou não (0) gerar o log. O becape deverá ser feito de forma recursiva, preservando as permissões dos arquivos.

Exercício 2

1. Para iniciar vamos criar um arquivo texto para o script, por exemplo, backup_files.sh.

```
touch backup_files.sh
```

2. Vamos editar o arquivo para iniciar o script. Use o editor que estiver disponível no seu linux, como por exemplo, vi ou nano;

```
nano backup_files.sh
```

3. Vamos inserir as informações do nosso script, começando pelas variáveis:
 - a. **DATA:** Data Formatada
 - b. **DIR:** Diretório onde ficará nosso backup
 - c. **DIR_ORIGEM:** pasta de origem, de onde estão nossos arquivos
 - d. **ARQ_LOG:** Arquivo contendo o log do arquivo
 - e. **LOG:** Flag para controlar se queremos ter log ou não
 - f. **ARQ:** Nome do arquivo de backup final

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



```
DATA=$(date +%d_%m_%y)

DIR="$HOME/backup/"

DIR_ORIGEM("$HOME/scripts/")

ARQ_LOG="$DIR/logBKP"

LOG="1" #1 -sim, 0 - não

ARQ="backup-$DATA.tar.bz2"
```

4. Agora iremos adicionar o resto da lógica contendo nosso condicional de log e o processo de backup

```
mkdir -p "$DIR" 2>/dev/null

#Confere se é para gerar histórico do backup

if [ $LOG -eq 1 ]; then

    #Insere linhas no histórico de backup com informações sobre
    data de início

    echo "Execução do backup - $(date +%d/%m/%Y)" >> "$ARQ_LOG"
    echo "Horário de início - $(date +%k:%M:%S)" >> "$ARQ_LOG"
    echo >> "$ARQ_LOG"

    echo "Arquivos compactados:" >> "$ARQ_LOG"

#Compacta os arquivos e insere no histórico de backup os arquivos
copiados

    tar -cvpjf "$DIR$ARQ" "$DIR_ORIGEM" >> "$ARQ_LOG"
```



```
#Inclui informações finais no histórico de backup

echo >> "$ARQ_LOG"

echo "Diretório de backup: ${DIR}" >> "$ARQ_LOG";

echo >> "$ARQ_LOG";

echo "Horário da Finalização do backup - $(date +%k:%M:%S)" >>
"$ARQ_LOG"

echo "===== " >>
"$ARQ_LOG"

echo >> "$ARQ_LOG"

else

#Caso o histórico do backup não seja ativado apenas compacta os
arquivos e salva no local determinado.

tar -cvpjf "$DIR$ARQ" "$DIR_ORIGEM"

fi
```

5. Vamos criar alguns arquivos de teste na pasta scripts

```
digitalhouse@linux-infra:~$ echo "aaaaaaaaaaaaa" >>
$HOME/scripts/file1.txt

digitalhouse@linux-infra:~$ echo "bbbbbbbbbbbbbbb" >>
$HOME/scripts/file2.txt
```



6. Agora vamos testar

```
digitalhouse@linux-infra:~$ ./backup_files.sh  
digitalhouse@linux-infra:~$ ls -l $HOME/backup  
  
qual é a saída???
```

Conclusão

Este exercício, demonstra a utilização de um shell script que pode ajudar o usuário a fazer uma cópia (backup) dos seus arquivos de forma automática e simples.