**Laboratórios Extras**

**Exercício Compactadores e Empacotadores 1**  
  
**1)Vamos entrar no diretório /tmp para mostrar como funciona:**   
  
# cd /tmp   
  
**2) Aqui, estou colocando um script para criar um arquivo grande:**   
  
# while true   
> do   
> cat /etc/passwd >> /tmp/arquivo.txt   
> done   
  
Deixem esse script executando por uns 30 segundos!   
Onde os ">" não precisam ser digitados, pois vai aparecer (> é o prompt secundário).   
Para parar basta dar CTRL+C para interromper a execução do script.   
  
Nesse script, estamos criando um loop infinito deixando o arquivo.txt sempre com o conteúdo do /etc/passwd (que é um arquivo do sistema que guarda informações sobre cada usuário no sistema, também será detalhada a sua função na aula de administração de usuários).   
  
**3) Então, depois de parar o loop (laço), vamos ver o tamanho do arquivo que o script gerou:**  
  
# cd /tmp   
# du -h arquivo.txt   
  
**4) E aproveita e faz uma cópia do mesmo:**   
  
# cp arquivo.txt arquivo2.txt   
  
Para testar os compactadores, vamos fazer assim: arquivo.txt - Para o gzip   
arquivo2.txt - Para o bzip2   
  
**5) Então vamos ao exemplo:**   
  
# cd /tmp   
# time gzip arquivo.txt   
# time bzip2 arquivo2.txt   
  
O comando time é apenas para marcar o tempo que o comando irá usar para terminar!   
  
Então depois que executei esse comando, e vocês irão ver pelo time que o bzip2 demora bem mais que o gzip!!!   
  
**6) Então vou ter dois arquivos no /tmp, um com a extensão gz e outro com bz2:**   
  
# ls arquivo\*   
arquivo.txt.gz   
arquivo2.txt.bz2   
  
**7) Agora comparem o tamanho dos dois arquivos:**  
  
#du -h arquivo\*   
68K arquivo.txt.gz   
24K arquivo2.txt.bz2  
  
Resumindo, verão que o gzip ganhou no tempo, mas perdeu em redução de tamanho!   
  
  
**Exercício Compactadores e Empacotadores 2**  
  
# cd /tmp   
# tar -cvf bkp\_etc.tar /etc   
  
Onde:   
-c - Para criar um backup   
-v - (verbose) Para mostrar detalhes para você na hora de criar   
-f - Para indicar o nome do arquivo.Essa opção sempre vem por último, pois é ela quem define o nome do arquivo.   
  
Então podemos fazer o teste de tamanho e verão que não reduziu tamanho:   
# du -h /etc   
13M /etc   
# du -h bkp\_etc.tar   
13M /etc   
  
**Exercício Compactadores e Empacotadores 3**  
  
Tar com Gzip:   
# cd /tmp # tar -cvzf bkp\_etc.tar.gz /etc   
  
Onde:   
c - Para criar um backup   
v - Para mostrar detalhes para você na hora de criar   
z - Para compactar com o GZIP   
f- Para indicar o nome do arquivo de backup   
  
Tar com Bzip2:   
# cd /tmp   
# tar -cvjf bkp\_etc.tar.bz2 /etc   
  
Onde:   
c - Para criar um backup   
v - Para mostrar detalhes para você na hora de criar   
j - Para compactar com o BZIP2   
f - para indicar o nome do arquivo de backup   
  
**Exercício Compactadores e Empacotadores 4**Mas se eu estou falando de backup, antes de restaurá-lo, é necessário ter certeza de que está tudo ok. Afinal o backup pode ter sido corrompido por exemplo.  
  
E como estou falando de BACKUP, tenho que ter certeza que esses arquivos estão realmente bons...   
  
**8) Então, tenho que testar a integridade desses arquivos, e, para isso usamos a opção t:**   
# tar -tvf bkp\_etc.tar   
# tar -tvzf bkp\_etc.tar.gz   
# tar -tvjf bkp\_etc.tar.bz2   
  
**9) No qual t é a opção que testa! Depois, para descompactar esses arquivos só mudamos para a opção para x que é de extract!**   
  
# tar -xvf bkp\_etc.tar   
# tar -xvzf bkp\_etc.tar.gz   
# tar -xvjf bkp\_etc.tar.bz2   
  
Lembrando que se fizer isso ele vai descompactar no /tmp mesmo!   
  
**10) Caso queira descompactar no / para substituir o /etc antigo no caso de um backup temos que especificar isso com o -C:**   
# tar -xvf bkp\_etc.tar -C /   
# tar -xvzf bkp\_etc.tar.gz -C /   
# tar -xvjf bkp\_etc.tar.bz2 -C /