

LAB – ARQUIVAMENTO E COMPRESSÃO

7.1 Introdução

Este é o Lab 7: Arquivando e desarquivando arquivos. Ao realizar este laboratório, os alunos aprenderão como trabalhar com arquivos compactados.

Neste laboratório, você executará as seguintes tarefas:

- Criar arquivos compactados usando tar com e sem compressão
- Compactar e descompactar arquivos em um arquivo archive gzip
- Compactar e descompactar arquivos em um arquivo archive bzip2
- Usar o zip e o unzip para compactar e descompactar arquivos compactados

7.2 Comandos de arquivamento

Nesta tarefa, usaremos `gzip`, `bzip2`, `tar` e `zip/unzip` para arquivar e restaurar arquivos. Esses comandos foram projetados para mesclar vários arquivos em um único arquivo ou compactar um arquivo grande em um menor. Em alguns casos, os comandos executam as duas funções.

A tarefa de arquivar dados é importante por vários motivos, incluindo, entre outros, o seguinte:

- a) Arquivos grandes podem ser difíceis de transferir. A redução desses arquivos ajuda a tornar a transferência mais rápida.
- b) Transferir vários arquivos de um sistema para outro pode se tornar tedioso quando existem muitos arquivos. Mesclá-los em um único arquivo para transporte facilita esse processo.
- c) Os arquivos podem ocupar muito espaço rapidamente, especialmente em mídias removíveis menores, como pen drives. O arquivamento reduz esse problema.

Uma área potencial de confusão que um usuário iniciante do Linux pode ter origem da seguinte pergunta: por que existem tantos comandos de arquivamento diferentes? A resposta é que esses comandos têm recursos diferentes (por exemplo, alguns deles permitem proteger com senha o arquivo de arquivamento) e técnicas de compactação usadas.

O mais importante a saber por enquanto é como esses diferentes comandos funcionam. Com o tempo, você aprenderá a escolher a ferramenta de arquivamento correta para qualquer situação.

7.2.1 Passo 1

Use o seguinte comando `tar` para criar um arquivo do diretório `/etc/udev`. Salve o backup no diretório `~/mybackups`:

```
cd
mkdir mybackups
tar -cvf mybackups/udev.tar /etc/udev
ls mybackups
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ cd
sysadmin@localhost:~$ mkdir mybackups
sysadmin@localhost:~$ tar -cvf mybackups/udev.tar /etc/udev
tar: Removing leading `/' from member names
/etc/udev/
/etc/udev/rules.d/
/etc/udev/rules.d/70-persistent-cd.rules
/etc/udev/rules.d/README
/etc/udev/udev.conf
sysadmin@localhost:~$ ls mybackups/
udev.tar
sysadmin@localhost:~$
```

O comando `tar` é usado para mesclar vários arquivos em um único arquivo. Por padrão, ele não compacta os dados.

A opção `-c` diz ao comando `tar` para criar um arquivo `tar`. A opção `-v` significa "verbose", que instrui o comando `tar` para demonstrar o que está fazendo. A opção `-f` é usada para especificar o nome do arquivo `tar`.

FYI: `tar` significa Tape ARchive. Esse comando foi originalmente usado para criar backups em fita, mas hoje é mais comumente usado para criar arquivos compactados.

Importante: Você não é obrigado a usar a extensão `.tar` para o nome do arquivo, no entanto, é útil para determinar o tipo de arquivo. É considerado "bom estilo" ao enviar um arquivo para outra pessoa.

7.2.2 Passo 2

Exiba o conteúdo de um arquivo `tar` (`t` = list contents, `v` = verbose, `f` = filename):

```
tar -tvf mybackups/udev.tar
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ tar -tvf mybackups/udev.tar
drwxr-xr-x root/root          0 2015-01-28 16:32 etc/udev/
drwxr-xr-x root/root          0 2015-01-28 16:32 etc/udev/rules.d/
-rw-r--r-- root/root    306 2015-01-28 16:32 etc/udev/rules.d/70-pe
rsistent-cd.rules
-rw-r--r-- root/root    1157 2012-04-05 19:18 etc/udev/rules.d/README
E
-rw-r--r-- root/root      218 2012-04-05 19:18 etc/udev/udev.conf
sysadmin@localhost:~$
```

Observe que foi feito backup dos arquivos recursivamente usando nomes de caminho *relativos*. Isto é importante porque quando você extrai os arquivos, eles serão colocados no seu diretório atual, não sobrescrevendo os arquivos atuais.

7.2.3 Passo 3

Para criar um arquivo `tar` compactado, use a opção `-z`:

```
tar -zcvf mybackups/udev.tar.gz /etc/udev
ls -lh mybackups
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ tar -zcvf mybackups/udev.tar.gz /etc/udev
tar: Removing leading `/' from member names
/etc/udev/
/etc/udev/rules.d/
/etc/udev/rules.d/70-persistent-cd.rules
/etc/udev/rules.d/README
/etc/udev/udev.conf
sysadmin@localhost:~$ ls -lh mybackups/
total 16K
-rw-rw-r-- 1 sysadmin sysadmin 10K Jan 25 04:00 udev.tar
-rw-rw-r-- 1 sysadmin sysadmin 1.2K Jan 25 04:34 udev.tar.gz
sysadmin@localhost:~$
```

Observe a diferença no tamanho; O primeiro backup (10 Kbytes) é maior que o segundo backup (1,2 Kbytes).

A opção `-z` faz uso do utilitário `gzip` para realizar a compactação.

7.2.4 Passo 4

Extraia o conteúdo de um arquivo. Os dados são restaurados no diretório atual por padrão:

```
cd mybackups
tar -xvf udev.tar.gz
ls
ls etc
ls etc/udev
ls etc/udev/rules.d
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ cd mybackups
sysadmin@localhost:~/mybackups$ ls
udev.tar  udev.tar.gz
sysadmin@localhost:~/mybackups$ tar -xvf udev.tar.gz
etc/udev/
etc/udev/rules.d/
etc/udev/rules.d/70-persistent-cd.rules
etc/udev/rules.d/README
etc/udev/udev.conf
sysadmin@localhost:~/mybackups$ ls
etc  udev.tar  udev.tar.gz
sysadmin@localhost:~/mybackups$ ls etc
udev
sysadmin@localhost:~/mybackups$ ls etc/udev
rules.d  udev.conf
sysadmin@localhost:~/mybackups$ ls etc/udev/rules.d
70-persistent-cd.rules  README
sysadmin@localhost:~/mybackups$
```

Se você quiser que os arquivos "voltem" para o local original, primeiro você pode ir (comando `cd`) para o diretório `/` e executar o comando `tar`. No entanto, neste exemplo, isso exigiria que você fizesse login como administrador, pois a criação de arquivos no diretório `/etc` só pode ser feita pelo administrador.

7.2.5 Passo 5

Para adicionar um arquivo a um arquivo existente, use a opção `-r` para o comando `tar`. Execute os seguintes comandos para executar esta ação e verifique a existência do novo arquivo no arquivo `tar`:

```
tar -rvf udev.tar /etc/hosts
tar -tvf udev.tar
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ tar -rvf udev.tar /etc/hosts
tar: Removing leading `/' from member names
/etc/hosts
sysadmin@localhost:~/mybackups$ tar -tvf udev.tar
drwxr-xr-x root/root          0 2015-01-28 16:32 etc/udev/
drwxr-xr-x root/root          0 2015-01-28 16:32 etc/udev/rules.d/
-rw-r--r-- root/root        306 2015-01-28 16:32 etc/udev/rules.d/70-pe
rsistent-c
d.rules
-rw-r--r-- root/root       1157 2012-04-05 19:18 etc/udev/rules.d/README
E
-rw-r--r-- root/root        218 2012-04-05 19:18 etc/udev/udev.conf
-rw-r--r-- root/root        172 2017-08-21 23:28 etc/hosts
sysadmin@localhost:~/mybackups$
```

7.2.6 Passo 6

Nos exemplos a seguir, você usará `gzip` e `gunzip` para compactar e descompactar um arquivo. Execute os seguintes comandos para compactar uma cópia do arquivo `words`:

```
cp /var/log/syslog .  
ls -lh words  
gzip syslog  
ls -lh syslog.gz
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ cp /var/log/syslog .  
sysadmin@localhost:~/mybackups$ ls -lh syslog  
-rw-r--r-- 1 sysadmin sysadmin 6.1M Jan 25 07:39 syslog  
sysadmin@localhost:~/mybackups$ gzip syslog  
sysadmin@localhost:~/mybackups$ ls -lh syslog.gz  
-rw-r--r-- 1 sysadmin sysadmin 912k Jan 25 07:39 syslog.gz  
sysadmin@localhost:~/mybackups$
```

Observe que o tamanho do arquivo compactado (912k bytes no exemplo acima) é muito menor que o arquivo original (6.1M bytes no exemplo acima).

Muito importante: quando você usa o `gzip`, o arquivo original é substituído pelo arquivo compactado. No exemplo acima, o arquivo `syslog` foi substituído por `syslog.gz`.

Quando você descompactar o arquivo, o arquivo compactado será substituído pelo arquivo original.

Obs: se na tentativa de copiar o arquivo `syslog` aparecer uma mensagem de erro indicando que o usuário atual não tem permissão para leitura neste arquivo, utilize a seguinte sequência de comandos:

```
sudo cp /var/log/syslog .  
sudo chown aluno:aluno syslog
```


7.2.7 Passo 7

Execute os seguintes comandos para descompactar o arquivo `syslog.gz`:

```
ls -lh syslog.gz
gunzip syslog.gz
ls -lh syslog
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ ls -lh syslog.gz
-rw-r--r-- 1 sysadmin sysadmin 6.1M Jan 25 07:39 syslog.gz
sysadmin@localhost:~/mybackups$ gunzip syslog.gz
sysadmin@localhost:~/mybackups$ ls -lh syslog
-rw-r--r-- 1 sysadmin sysadmin 932k Jan 25 07:39 syslog
sysadmin@localhost:~/mybackups$
```

O Linux fornece um grande número de utilitários de compactação, além do `gzip/gunzip`. Cada um deles tem vantagens e desvantagens (compressão mais rápida, melhores taxas de compressão, mais flexíveis, mais portáteis, descompressão mais rápida, etc.).

Os comandos `gzip/gunzip` são muito populares no Linux, mas você deve estar ciente de que o `bzip2/bunzip2` também é popular em algumas distribuições Linux. É uma sorte que a maioria das funcionalidades (a maneira de executar os comandos) e as opções sejam as mesmas do `gzip/gunzip`.

7.2.8 Passo 8

Usar `bzip2` e `bunzip2` para compactar e descompactar um arquivo é muito semelhante ao uso de `gzip` e `gunzip`. O arquivo compactado é criado com uma extensão `.bz2`. A extensão é removida quando descompactada. Execute os seguintes comandos para compactar uma cópia do arquivo `words`:

```
ls -lh syslog
bzip2 syslog
ls -lh syslog.bz2
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ ls -lh syslog
-rw-r--r-- 1 sysadmin sysadmin 6.1M Jan 25 07:39 syslog
sysadmin@localhost:~/mybackups$ bzip2 syslog
sysadmin@localhost:~/mybackups$ ls -lh syslog.bz2
-rw-r--r-- 1 sysadmin sysadmin 474k Jan 25 07:39 syslog.bz2
sysadmin@localhost:~/mybackups$
```

Se você comparar o tamanho do arquivo `.bz2` resultante (474k) ao tamanho do arquivo `.gz` (932k) da Passo 7, perceberá que o `bzip2` fez um trabalho melhor ao compactar esse arquivo específico.

7.2.9 Passo 9

Execute os seguintes comandos para descompactar o arquivo `syslog.bz2`:

```
ls -lh syslog.bz2
bunzip2 syslog.bz2
ls -lh syslog

sysadmin@localhost:~/mybackups$ ls -lh words.bz2
-rw-r--r-- 1 sysadmin sysadmin 474k Jan 25 07:39 syslog.bz2

sysadmin@localhost:~/mybackups$ bunzip2 syslog.bz2

sysadmin@localhost:~/mybackups$ ls -lh syslog
-rw-r--r-- 1 sysadmin sysadmin 6.1M Jan 25 07:39 syslog
```

Embora os arquivos archive `gzip` e `bzip` sejam comumente usados no Linux, o tipo de arquivo `zip` é mais comumente usado por outros sistemas operacionais, como o Windows. Na verdade, o aplicativo Windows Explorer possui suporte interno para extrair arquivos `zip`.

Portanto, se você planeja compartilhar um arquivo com usuários do Windows, geralmente é preferível usar o tipo de arquivo `zip`. Ao contrário do `gzip` e do `bzip2`, quando um arquivo é compactado com o comando `zip`, uma cópia do arquivo original é compactada e o original permanece descompactado.

7.2.10 Passo 10

Use o comando `zip` para compactar o arquivo `syslog`:

```
zip syslog.zip syslog
ls -lh syslog.zip
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ zip syslog.zip syslog
  adding: words (deflated 73%)
sysadmin@localhost:~/mybackups$ ls -lh syslog.zip
-rw-rw-r-- 1 sysadmin sysadmin 912k Jan 25 21:25 syslog.zip
sysadmin@localhost:~/mybackups$
```

O primeiro argumento (`syslog.zip` no exemplo acima) do comando `zip` é o nome do arquivo que você deseja criar. Os argumentos restantes (`syslog` no exemplo acima) são os arquivos que você deseja colocar no arquivo compactado.

Importante: Você não é obrigado a usar a extensão `.zip` para o nome do arquivo compactado; no entanto, é útil para determinar o tipo de arquivo. Também é considerado "bom estilo" ao enviar um arquivo para outra pessoa.

7.2.11 Passo 11

Compacte o diretório `/etc/udev` e seu conteúdo com compactação `zip`:

```
zip -r udev.zip /etc/udev
ls -l udev.zip
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ zip -r udev.zip /etc/udev
  adding: etc/udev/ (stored 0%)
  adding: etc/udev/rules.d/ (stored 0%)
  adding: etc/udev/rules.d/70-persistent-cd.rules (deflated 29%)
  adding: etc/udev/rules.d/README (deflated 50%)
  adding: etc/udev/udev.conf (deflated 24%)
sysadmin@localhost:~/mybackups$ ls -l udev.zip
-rw-rw-r-- 1 sysadmin sysadmin 1840 Jan 25 21:33 udev.zip
sysadmin@localhost:~/mybackups$
```

O comando `tar` discutido anteriormente neste laboratório desce automaticamente através de qualquer subdiretório de um diretório especificado para ser arquivado. Com os comandos `bzip2`, `gzip` e `zip`, a opção `-r` deve ser especificada para executar a recursão em subdiretórios.

7.2.12 Passo 12

Para visualizar o conteúdo de um arquivo `zip`, use com a opção `-l` com o comando `unzip`:

```
unzip -l udev.zip
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ unzip -l udev.zip
Archive:  udev.zip
  Length      Date    Time    Name
-----
         0  2015-01-28  16:32    etc/udev/
         0  2015-01-28  16:32    etc/udev/rules.d/
       306  2015-01-28  16:32    etc/udev/rules.d/70-persistent-cd.rules
      1157  2012-04-05  19:18    etc/udev/rules.d/README
       218  2012-04-05  19:18    etc/udev/udev.conf
-----
      1681
                    5 files
sysadmin@localhost:~/mybackups$
```

7.2.13 Passo 13

Para extrair o arquivo `zip`, use o comando `unzip` sem nenhuma opção. Neste exemplo, primeiro precisamos excluir os arquivos que foram criados no exemplo anterior de `tar`:

```
rm -r etc
unzip udev.zip
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~/mybackups$ rm -r etc
sysadmin@localhost:~/mybackups$ unzip udev.zip
Archive:  udev.zip
  creating: etc/udev/
  creating: etc/udev/rules.d/
  inflating: etc/udev/rules.d/70-persistent-cd.rules
  inflating: etc/udev/rules.d/README
  inflating: etc/udev/udev.conf
sysadmin@localhost:~/mybackups$
```