

LAB – LISTANDO ARQUIVOS E DIRETÓRIOS

6.1 Introdução

Este é o Lab 6: listando arquivos e diretórios. Ao realizar este laboratório, os alunos aprenderão como navegar e gerenciar arquivos e diretórios.

Neste laboratório, você executará as seguintes tarefas:

- Listar arquivos e diretórios
- Copiar, mover e apagar arquivos e diretórios

6.2 Arquivos e Diretórios

Nesta tarefa, você explorará os conceitos de arquivos e diretórios.

Em um sistema operacional Linux, os dados são armazenados em *arquivos* e os arquivos são armazenados em *diretórios*. Você pode estar acostumado com o termo *pastas* para descrever diretórios.

Diretórios também são arquivos; os dados que eles contêm são os nomes dos arquivos que foram inseridos neles, junto com o número inode (um número identificador exclusivo atribuído a cada arquivo) para onde os dados desse arquivo existem no disco.

Como um usuário Linux, você desejará saber como manipular esses arquivos e diretórios, incluindo como listar arquivos em um diretório, copiar, excluir e mover arquivos.

Aviso: Os nomes de arquivos e diretórios no Linux fazem distinção entre maiúsculas e minúsculas. Isso significa que um arquivo chamado `ABC` não é o mesmo que um arquivo chamado `abc`.

6.2.1 Passo 1

Digite o seguinte comando para imprimir o diretório de trabalho:

```
pwd
```

```
sysadmin@localhost:~$ pwd
/home/sysadmin
sysadmin@localhost:~$
```

O *diretório de trabalho* é o diretório em que sua janela de terminal está atualmente. Isso também é chamado de *diretório atual*. Isso será importante para quando você estiver executando comandos futuros, pois eles se comportarão de maneira diferente com base no diretório em que você está atualmente.

A saída do comando `pwd` (`/home/sysadmin` no exemplo acima) é chamada de caminho. A primeira barra representa o diretório raiz, o nível superior da estrutura de diretórios.

Na saída acima, `home` é um diretório sob o diretório raiz e `sysadmin` é um diretório no diretório `home`.

Quando você abre pela primeira vez uma janela de terminal, você será colocado em seu diretório `home`. Este é um diretório onde você tem acesso total e outros usuários normalmente não têm acesso por padrão. Para ver o caminho para seu diretório `home`, você pode executar o seguinte comando para visualizar o valor da variável `HOME`:

```
echo $HOME
```

```
sysadmin@localhost:~$ echo $HOME
/home/sysadmin
sysadmin@localhost:~$
```

6.2.2 Passo 2

Você pode usar o comando `cd` com um *caminho* para um diretório para alterar seu diretório atual. Digite o seguinte comando para tornar o diretório raiz seu diretório de trabalho atual e verifique com o comando `pwd`:

```
cd /
```

```
pwd
```

```
sysadmin@localhost:~$ cd /
```

```
sysadmin@localhost:/$ pwd
```

```
/
```

```
sysadmin@localhost:/$
```

6.2.3 Passo 3

Para voltar ao seu diretório home, o comando `cd` pode ser executado sem um caminho. Volte para o seu diretório home e verifique digitando os seguintes comandos:

```
cd
```

```
pwd
```

```
sysadmin@localhost:/$ cd
```

```
sysadmin@localhost:~$ pwd
```

```
/home/sysadmin
```

```
sysadmin@localhost:~$
```

Observe a mudança no *prompt*. O caractere til `~` representa seu diretório *home*. Esta parte do prompt informará em qual diretório você está atualmente.

6.2.4 Passo 4

O comando `cd` pode ser inserido com um caminho para um diretório especificado como um *argumento*. Execute o comando `cd` com o diretório `/home` como um argumento, digitando o seguinte:

```
cd /home  
pwd  
  
sysadmin@localhost:~$ cd /home  
sysadmin@localhost:/home$ pwd  
/home  
sysadmin@localhost:/home$
```

Quando o caminho fornecido como argumento para o comando `cd` começa com a barra `/`, esse caminho é referido como um “caminho absoluto”. Caminhos absolutos são sempre caminhos completos do diretório raiz para um subdiretório ou arquivo.

6.2.5 Passo 5

Mude de volta para o diretório inicial, usando o comando `cd` com o til `~` como um argumento:

```
cd ~  
pwd  
  
sysadmin@localhost:/home$ cd ~  
sysadmin@localhost:~$ pwd  
/home/sysadmin  
sysadmin@localhost:~$
```

Quando o caminho fornecido como um argumento para o comando `cd` começa com um caractere til `~`, o terminal expandirá o caractere para o diretório *home* de um usuário com uma conta no sistema.

Se nenhum outro caractere ou uma barra seguir o til, ele será expandido para o diretório inicial do **usuário atualmente ativo** no shell.

Se um nome de usuário seguir imediatamente o caractere til, o shell expandirá o til e o nome de usuário para o diretório *home* desse nome de usuário. Por exemplo, `~bob` seria expandido para `/home/bob`.

Os caminhos que começam com um til são considerados caminhos absolutos, porque depois que o shell expande o caminho do til, um caminho absoluto é formado.

6.2.6 Passo 6

Use o comando `echo` abaixo para exibir alguns outros exemplos de uso do til como parte do caminho:

```
echo ~ ~sysadmin ~root ~mail ~nobody
```

```
sysadmin@localhost:~$ echo ~ ~sysadmin ~root ~mail ~nobody  
/home/sysadmin /home/sysadmin /root /var/mail /nonexistent  
sysadmin@localhost:~$
```


6.2.7 Passo 7

Tente mudar para o diretório home do usuário root digitando o seguinte comando:

```
cd ~root
```

```
sysadmin@localhost:~$ cd ~root
```

```
-bash: cd: /root: Permission denied
```

```
sysadmin@localhost:~$
```

Observe a mensagem de erro; isso indica que o shell tentou executar `cd` com `/root` como um argumento e ele falhou devido à permissão ser negada. Você aprenderá mais sobre permissões de arquivos e diretórios em um laboratório posterior.

6.2.8 Passo 8

Usando um caminho absoluto, mude para o diretório `/usr/bin` e exiba o diretório de trabalho usando os seguintes comandos:

```
cd /usr/bin
```

```
pwd
```

```
sysadmin@localhost:~$ cd /usr/bin
```

```
sysadmin@localhost:/usr/bin$ pwd
```

```
/usr/bin
```

```
sysadmin@localhost:/usr/bin$
```

6.2.9 Passo 9

Use um caminho absoluto para mudar para o diretório `/usr` e exibir o diretório de trabalho, emitindo os seguintes comandos:

```
cd /usr
```

```
pwd
```

```
sysadmin@localhost:/usr/bin$ cd /usr
```

```
sysadmin@localhost:/usr$ pwd
```

```
/usr
```

```
sysadmin@localhost:/usr$
```

6.2.10 Passo 10

Use um caminho absoluto para mudar para o diretório `/usr/share/doc` e exibir o diretório atual, emitindo os seguintes comandos:

```
cd /usr/share/doc
pwd
sysadmin@localhost:/usr$ cd /usr/share/doc
sysadmin@localhost:/usr/share/doc$ pwd
/usr/share/doc
sysadmin@localhost:/usr/share/doc$
```

Caminhos Absolutos vs. Relativos

Suponha que você esteja no diretório `/usr/share/doc` e deseje ir para o diretório `/usr/share/doc/bash`. Digitar o comando `cd /usr/share/doc/bash` resulta em uma boa quantidade de digitação. Em casos como esse, você pode usar nomes de caminho relativos.

Com nomes de caminho relativos, você fornece "direções" de onde deseja ir do diretório atual. Os exemplos a seguir ilustrarão o uso de caminhos relativos.

6.2.11 Passo 11

Usando um caminho relativo, mude para o diretório `/usr/share/doc/bash` e exiba o diretório de trabalho, emitindo os seguintes comandos:

```
cd bash
```

```
pwd
```

```
sysadmin@localhost:/usr/share/doc$ cd bash
```

```
sysadmin@localhost:/usr/share/doc/bash$ pwd
```

```
/usr/share/doc/bash
```

```
sysadmin@localhost:/usr/share/doc/bash$
```

Nota: Se não houver um diretório `bash` no diretório atual, o comando anterior falhará.

6.2.12 Passo 12

Use um caminho relativo para mudar para o diretório acima do diretório atual:

```
cd ..
```

```
pwd
```

```
sysadmin@localhost:/usr/share/doc/bash$ cd ..
```

```
sysadmin@localhost:/usr/share/doc$ pwd
```

```
/usr/share/doc
```

```
sysadmin@localhost:/usr/share/doc$
```

O `..` representa um nível acima da localização atual do seu diretório.

6.2.13 Passo 13

Use um caminho relativo para mudar um nível acima do diretório atual e, em seguida, no diretório `dict`:

```
cd ../dict
```

```
pwd
```

```
sysadmin@localhost:/usr/share/doc$ cd ../dict
```

```
sysadmin@localhost:/usr/share/dict$ pwd
```

```
/usr/share/dict
```

```
sysadmin@localhost:/usr/share/dict$
```

6.3 Listando Arquivos e Diretórios

Nesta tarefa, você explorará a maneira de listar arquivos e diretórios.

6.3.1 Passo 1

Para listar o conteúdo do diretório atual, use o comando `ls`:

```
cd
ls
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:/usr/share/dict$ cd
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$
```

Na saída do comando `ls` anterior, os nomes dos arquivos foram colocados em uma cor azul clara. Esse é um recurso que muitas distribuições do Linux fornecem automaticamente por meio de um recurso chamado de *alias* (mais sobre esse recurso em um laboratório posterior).

A cor indica o tipo do item. A tabela a seguir descreve algumas das cores mais comuns:

Cor	Tipo de Arquivo
Preto ou Branco	Arquivo regular
Azul	Arquivo de directório
Ciano	Arquivo de link simbólico (um arquivo que aponta para outro arquivo)
Verde	Arquivo executável (um programa)

6.3.2 Passo 2

Nem todos os arquivos são exibidos por padrão. Existem arquivos, chamados de arquivos ocultos, que não são exibidos por padrão. Para exibir todos os arquivos, incluindo arquivos ocultos, use a opção `-a` para o comando `ls`:

```
ls -a
```

```
sysadmin@localhost:~$ ls -a
```

```
.          .bashrc    .selected_editor  Downloads  Public
..         .cache     Desktop           Music      Templates
.bash_logout .profile  Documents         Pictures   Videos
sysadmin@localhost:~$
```

Arquivos ocultos começam com um período (um caractere de ponto). Normalmente, esses arquivos e geralmente diretórios ficam ocultos porque não são arquivos que você normalmente deseja ver.

Por exemplo, o arquivo `.bashrc` mostrado no exemplo acima contém informações de configuração para o shell `bash`. Este é um arquivo que você normalmente não precisa ver em uma base regular.

Dois importantes "arquivos de pontos" existem em todos os diretórios: `.` (que representa o diretório atual) e `..` (que representa o diretório acima do diretório atual).

6.3.3 Passo 3

Por si só, o comando `ls` apenas forneceu os nomes dos arquivos e diretórios dentro do diretório especificado (ou atual). Execute o seguinte comando para ver como a opção `-l` fornece mais informações sobre um arquivo:

```
ls -l /etc/hosts
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls -l /etc/hosts
-rw-r--r-- 1 root root 150 Jan 22 15:18 /etc/hosts
sysadmin@localhost:~$
```

Então, o que significa toda essa saída extra? A tabela a seguir fornece uma breve descrição do que cada parte da saída de `ls -l` significa:

-	O primeiro caractere, um - no exemplo anterior, indica que tipo de "arquivo" é esse. Um caractere - é para arquivo simples, enquanto um caractere d seria para um diretório.
rw-r--r--	Isso representa as permissões do arquivo. As permissões são discutidas em um laboratório posterior.
1	Isso representa algo chamado de contagem de link físico (discutido mais adiante).
root	O usuário proprietário do arquivo.
root	O grupo proprietário do arquivo.
150	O tamanho do arquivo em bytes.
Jan 22 15:18	A data/hora quando o arquivo foi modificado pela última vez.

6.3.4 Passo 4

Às vezes, você quer ver não apenas o conteúdo de um diretório, mas também o conteúdo dos subdiretórios. Você pode usar a opção `-R` para realizar isso:

```
ls -R /etc/udev

sysadmin@localhost:~$ ls -R /etc/udev
/etc/udev:
rules.d  udev.conf

/etc/udev/rules.d:
70-persistent-cd.rules  README

sysadmin@localhost:~$
```

A opção `-R` significa "recursiva". Todos os arquivos no diretório `/etc/udev` serão exibidos, assim como todos os arquivos em cada subdiretório, neste caso o subdiretório `rules.d`.

Tenha cuidado com a opção `-R`. Alguns diretórios são muito grandes!

6.3.5 Passo 5

Você pode usar caracteres globbing (curingas) para limitar os arquivos ou diretórios que você vê. Por exemplo, o caractere `*` pode corresponder a "zero ou mais caracteres" em um nome de arquivo. Execute o seguinte comando para exibir apenas os arquivos que começam com a letra `s` no diretório `/etc`:

```
ls -d /etc/s*
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls -d /etc/s*
/etc/securetty  /etc/sgml      /etc/shells   /etc/ssl      /etc/sysctl
.conf
/etc/security   /etc/shadow    /etc/skel     /etc/sudoers   /etc/sysctl
.d
/etc/services   /etc/shadow-   /etc/ssh      /etc/sudoers.d /etc/system
d
sysadmin@localhost:~$
```

Observe que a opção `-d` impede que os arquivos dos subdiretórios sejam exibidos. Ele sempre deve ser usado com o comando `ls` quando você estiver usando globbing de arquivo.

6.3.6 Passo 6

O `?` caractere pode ser usado para corresponder exatamente 1 caractere em um nome de arquivo. Execute o seguinte comando para exibir todos os arquivos no diretório `/etc` com exatamente quatro caracteres:

```
ls -d /etc/????
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls -d /etc/????
/etc/bind   /etc/init  /etc/motd  /etc/perl  /etc/skel
/etc/dpkg   /etc/ldap  /etc/mtab  /etc/sgml  /etc/udev
sysadmin@localhost:~$
```

6.3.7 Passo 7

Usando colchetes [] você pode especificar um único caractere para combinar com um conjunto de caracteres. Execute o seguinte comando para exibir todos os arquivos no diretório /etc que começam com as letras a, b, c ou d:

```
ls -d /etc/[abcd]*
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls -d /etc/[abcd]*
/etc/adduser.conf          /etc/blkid.conf          /etc/cron.weekly
/etc/adjtime               /etc/blkid.tab           /etc/crontab
/etc/alternatives          /etc/ca-certificates     /etc/dbus-1
/etc/apparmor.d            /etc/ca-certificates.conf /etc/debconf.c
onf                        /etc/calendar            /etc/debian_ve
rsion                     /etc/cron.d              /etc/default
/etc/bash.bashrc           /etc/cron.daily          /etc/deluser.c
onf                       /etc/cron.hourly         /etc/depmod.d
/etc/bind                  /etc/cron.monthly        /etc/dpkg
/etc/bindresvport.blacklist
sysadmin@localhost:~$
```

6.4 Copiando, Movendo e Renomeando Arquivos e Diretórios

Nesta tarefa, você copiará, moverá e removerá arquivos e diretórios.

6.4.1 Passo 1

Faça uma cópia do arquivo `/etc/hosts` e coloque-o no diretório atual. Em seguida, liste o conteúdo do diretório atual antes e depois da cópia:

```
ls
cp /etc/hosts hosts
ls
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ cp /etc/hosts hosts
sysadmin@localhost:~$ ls
Desktop  Downloads  Pictures  Templates  hosts
Documents  Music      Public    Videos
sysadmin@localhost:~$
```

Observe como o segundo comando `ls` exibe uma cópia do arquivo `hosts`.

6.4.2 Passo 2

Em seguida, você removerá o arquivo e o copiará novamente, mas o sistema informará o que está sendo feito. Isso pode ser conseguido usando a opção `-v` ou `--verbose`. Digite os seguintes comandos:

```
rm hosts
ls
cp -v /etc/hosts hosts
ls
```

Observe que o comando `rm` é usado para excluir um arquivo. Mais informações sobre esse comando serão fornecidas posteriormente neste laboratório.

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ rm hosts
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ cp -v /etc/hosts hosts
`/etc/hosts' -> `hosts'
sysadmin@localhost:~$ ls
Desktop  Downloads  Pictures  Templates  hosts
Documents  Music      Public    Videos
sysadmin@localhost:~$
```

Observe que a opção `-v` exibe a origem e o destino quando o comando `cp` é executado.

6.4.3 Passo 3

Digite os seguintes comandos para copiar o arquivo `/etc/hosts`, usando o caractere ponto `.` para indicar o diretório atual como o destino:

```
rm hosts
ls
cp -v /etc/hosts .
ls
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ rm hosts
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ cp -v /etc/hosts .
`/etc/hosts' -> `hosts'
sysadmin@localhost:~$ ls
Desktop  Downloads  Pictures  Templates  hosts
Documents  Music      Public    Videos
sysadmin@localhost:~$
```

O caractere `.` é uma maneira prática de dizer "o diretório atual". Pode ser usado com todos os comandos do Linux, não apenas com o comando `cp`.

6.4.4 Passo 4

Digite os seguintes comandos para copiar do diretório de origem e preservar os atributos de arquivo usando a opção `-p`:

```
rm hosts
ls
cd /etc
ls -l hosts
cp -p hosts /home/sysadmin
cd
ls -l hosts
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ rm hosts
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ cd /etc
sysadmin@localhost:/etc$ ls -l hosts
-rw-r--r-- 1 root root 150 Jan 22 15:18 hosts
sysadmin@localhost:/etc$ cp -p hosts /home/sysadmin
sysadmin@localhost:/etc$ cd
sysadmin@localhost:~$ ls -l hosts
-rw-r--r-- 1 sysadmin sysadmin 150 Jan 22 15:18 hosts
sysadmin@localhost:~$
```

Observe que os modos de data e permissão foram preservados. Observe que o registro de data e hora na saída acima é o mesmo para o original e a cópia (22 de janeiro, 15:18) no exemplo fornecido acima. Sua saída pode variar.

6.4.5 Passo 5

Digite os seguintes comandos para copiar usando um nome de destino diferente:

```
rm hosts
cp -p /etc/hosts ~
cp hosts newname
ls -l hosts newname
rm hosts newname
```

```
sysadmin@localhost:~$ rm hosts
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$ cp -p /etc/hosts ~
sysadmin@localhost:~$ cp hosts newname
sysadmin@localhost:~$ ls -l hosts newname
-rw-r--r-- 1 sysadmin sysadmin 150 Jan 22 15:18 hosts
-rw-r--r-- 1 sysadmin sysadmin 150 Jan 22 16:29 newname
sysadmin@localhost:~$ rm hosts newname
sysadmin@localhost:~$
```

A primeira cópia com a opção `-p` preservou o timestamp original. Lembre-se de que o til `~` representa seu diretório home (`/home/sysadmin`).

A segunda cópia especificou um nome de arquivo diferente (`newname`) como o destino. Como foi emitido sem a opção `-p`, o sistema usou a data e a hora atuais para o destino, portanto, não preservou o registro de data e hora original encontrado no arquivo de origem `/etc/hosts`.

Por fim, observe que você pode remover mais de um arquivo de cada vez, conforme mostrado no último comando `rm`.

6.4.6 Passo 6

Para copiar todos os arquivos em um diretório, use a opção `-R`. Para esta tarefa, você copiará o diretório `/etc/udev` e exibirá o conteúdo do diretório copiado:

```
mkdir Myetc
cp -R /etc/udev Myetc
ls -l Myetc
ls -lR Myetc
```

```
sysadmin@localhost:~$ mkdir Myetc
sysadmin@localhost:~$ cp -R /etc/udev Myetc
sysadmin@localhost:~$ ls -l Myetc
total 0
drwxr-xr-x 1 sysadmin sysadmin 32 Jan 22 16:35 udev
sysadmin@localhost:~$ ls -lR Myetc
Myetc:
total 0
drwxr-xr-x 1 sysadmin sysadmin 32 Jan 22 16:35 udev

Myetc/udev:
total 4
drwxr-xr-x 1 sysadmin sysadmin 56 Jan 22 16:35 rules.d
-rw-r--r-- 1 sysadmin sysadmin 218 Jan 22 16:35 udev.conf

Myetc/udev/rules.d:
total 8
-rw-r--r-- 1 sysadmin sysadmin 306 Jan 22 16:35 70-persistent-cd.rules
-rw-r--r-- 1 sysadmin sysadmin 1157 Jan 22 16:35 README
sysadmin@localhost:~$
```

6.4.7 Passo 7

Para remover um diretório, use a opção `-r` para o comando `rm`:

```
ls
rm -r Myetc
ls
```

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ ls
Desktop  Downloads  Myetc      Public    Videos
Documents Music      Pictures   Templates
sysadmin@localhost:~$ rm -r Myetc
sysadmin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
sysadmin@localhost:~$
```

Observe que o comando `rmdir` também pode ser usado para excluir diretórios, mas apenas se o diretório estiver vazio (se não contiver arquivos).

Observe também a opção `-r`. Esta opção remove diretórios e seus conteúdos de forma recursiva.

6.4.8 Passo 8

Mover um arquivo é análogo a um "recortar e colar". O arquivo é "cortado" (removido) do local original e "colado" no destino especificado. Mova um arquivo no diretório local executando os seguintes comandos:

```
touch premove
ls
mv premove postmove
ls
rm postmove
```

Comando Linux	Descrição
<code>touch premove</code>	Cria um arquivo vazio chamado <code>premove</code>
<code>mv premove postmove</code>	Este comando "recorta" o arquivo <code>premove</code> e o "cola" num arquivo chamado <code>postmove</code>
<code>rm postmove</code>	Apaga o arquivo <code>postmove</code>

Sua saída deve ser semelhante à seguinte:

```
sysadmin@localhost:~$ touch premove
sysadmin@localhost:~$ ls
Desktop  Downloads  Pictures  Templates  premove
Documents Music      Public   Videos
sysadmin@localhost:~$ mv premove postmove
sysadmin@localhost:~$ ls
Desktop  Downloads  Pictures  Templates  postmove
Documents Music      Public   Videos
sysadmin@localhost:~$
```