

# LXT0206-Comandos de Navegação (Comando ls e cd)

📅 Data do post	@July 20, 2022 9:00 AM (GMT)
🔑 Palavras-chave	Uso básico da Linha de Comandos
☑ Publicado	<input type="checkbox"/>
📌 Pronto	Preparado
☑ IG Post	<input checked="" type="checkbox"/>
🌟 Status	Done

## Post-LinkedIn

Continuando a senda dos artigos, hoje mais uma vez vamos falar de Linux. O assunto agora é comando de navegação do Linux. Leia o artigo na íntegra

Siga-me e veja outros posts no meu Instagram <https://www.instagram.com/idalectiosilvatech/>

Deixe o like para medirmos a repercussão do conteúdo.

Guarde para rever noutro momento.

Compartilha o conteúdo com os amigos e não só

Comente o que acha do conteúdo e da iniciativa e deixe sugestões.

[#cybersecurity](#). [#technology](#) [#linux](#) [#CLI](#) [#Shell](#) [#changedirectory](#) [#listdirectory](#)

## Introdução

No Linux os arquivos estão organizados dentro de uma estrutura de directórios (pastas) e em cada directório podem existir outros directórios e arquivos. Os arquivos contêm dados, como texto legível por humanos, programas executáveis ou dados binários usados pelo computador. Os directórios são usados para organizar o sistema de arquivos. Eles podem conter arquivos e outros directórios.

Uma das habilidades fundamentais e, muito valorizada entre os usuários Linux, é saber aonde cada coisa está e entender como movimentar-se dentro dessa estrutura de directórios.

Para entender essa organização dos directórios e comandos, o comando `tree` dá um overview sobre o funcionamento dessa estrutura de directórios e arquivos.

```
├── Analise-de-clientes.txt
├── Auditoria
│   └── Analise-de-clientes.txt
1 directory, 2 files
```

Neste exemplo, `Documentos` é um directório que contém um arquivo (`Analise-de-Clientes.txt`) e um subdirectório (`Auditoria`). O directório `Auditoria`, no que lhe concerne, contém um arquivo chamado `Analise-de-clientes.txt`. O directório `Documentos` é considerado o pai do directório `Auditoria`.

## Comandos de Navegação

### Comando `pwd` (Print Working Directory — Imprimir directório actual)

Para quem leu o artigo anterior já sabe o que se referem cada uma das informações.

Analisando o prompt, podemos concluir que a nossa localização actual é o directório **Auditoria**.

O comando **pwd** serve para *imprimir o directório de trabalho ou a nossa localização actual*:

```
idaseg@debian:~$ pwd
/home/idaseg
```

A relação entre os directórios é feita por uma barra (/). Isso fica evidente, pois **Auditoria** é um subdirectório de **Documentos**, que no que lhe concerne é um subdirectório de **idaseg**, localizado em um directório chamado **home**. O pai de home se chama **root** e é representado pela primeira barra (/).

A saída do comando **pwd** difere um pouco do caminho mostrado no prompt de comando. Em vez de **/home/user**, o prompt de comando contém um til (~). O til é um caractere especial que representa o directório inicial do usuário. Trataremos disso em mais detalhes na próxima lição.

## Comando ls (List directory contents - Listar Conteúdo do Directório)

O comando **ls** é usado para listar arquivos ou directórios dentro de outros directórios. Os comandos **ls** é dos comandos Linux com mais parâmetros sendo que é quase impossível ter apenas um artigo falando sobre todos os parâmetros desses comandos. Portanto, apresentaremos aqueles que tendem a ser os mais usados no quotidiano dos usuários Linux.

```
idaseg@debian:~/Documentos/Auditoria$ ls
Analise-de-clientes.txt
```

OBS: uma forma de conhecer parâmetros do comando **ls** é usando o comando **man ls** ou podemos usar o comando **ls - -help** ou ainda **ls -h**, aonde aparece um mini manual.

Existem duas formas de chamar os parâmetros: reduzida e completa

```
idaseg@debian:~/Documentos/Auditoria$ ls -a Reduzida
.  ..  Analise-de-clientes.txt
idaseg@debian:~/Documentos/Auditoria$ ls --all Completa
.  ..  Analise-de-clientes.txt
```



**Nota:** O comando **ls** não fornece informações sobre o directório pai. Da mesma forma, por padrão, **ls** não exibe nenhuma informação sobre o conteúdo dos subdirectórios. O **ls** pode apenas “ver” o que está no directório actual.

Suas opções mais comuns são:

Aa Opção	Descrição
<b>ls /</b>	Lista os arquivos no directório <b>root</b>
<b>ls ..</b>	Lista os arquivos de um directório pai ou acima
<b>ls ../..</b>	Lista os arquivos de um directório que está dois níveis acima
<b>ls ~</b>	Lista os arquivos no directório home do usuário (/home/user)
<b>ls -d*/</b>	Lista apenas directórios
<b>ls *</b>	Lista arquivos com subdirectórios
<b>ls -R</b>	Lista o conteúdo do directório com seus subdirectórios
<b>ls -s</b>	Lista arquivos e seus tamanhos
<b>ls -l</b>	Lista o conteúdo do directório em formato tabela com colunas que incluem: permissões, links, proprietário, tamanho em bytes, data de modificação e nome ou directório do arquivo
<b>ls -a</b>	Lista arquivos ou directórios incluindo arquivos ou directórios ocultos e normalmente começam com (.)
<b>ls -t</b>	Lista os arquivos ou directórios e ordená-los pela última data e hora de modificação em ordem descendente (do maior para o menor).
<b>ls -A</b>	Lista quase todos os arquivos, menos as entradas “.” e “..”

Aa Opção	Descrição
<u>ls -b</u>	Lista os arquivos com escape, para caracteres não gráficos
<u>ls -c</u>	Lista e ordena pela data de modificação dos arquivos
<u>ls -C</u>	Lista por conteúdo por colunas
<u>ls -d</u>	lista somente os directórios
<u>ls -l</u>	Lista uma coluna somente com nomes dos arquivos
<u>ls -h</u>	Lista o tamanho dos arquivos em formato humano
<u>ls -m</u>	Lista os arquivos separados por vírgula
<u>ls -S</u>	Ordena por tamanho
<u>ls -x</u>	Lista em linhas ao invés de colunas
<u>ls- color=auto</u>	Diferencia os tipos de arquivos por cor

## Comando cd (change directory)

A navegação no Linux é feita principalmente com o comando `cd`. Esse comando *muda de directório*. Como vimos no comando `pwd`, sabemos que o directório actual é `/home/user/Documentos/Auditoria`. Esse caminho pode ser mudado inserindo um novo caminho:

```
idaseg@debian:~/Documentos$ cd /home/idaseg/Documentos/
idaseg@debian:~/Documentos$ pwd
/home/idaseg/Documentos
idaseg@debian:~/Documentos$ ls
Analise-de-clientes.txt Auditoria
```

Neste novo local, podemos “ver” `Analise-de-clientes.txt` e o subdirectório `Auditoria`, mas não o conteúdo de nosso subdirectório. Podemos navegar de volta para `Auditoria` assim:

```
idaseg@debian:~/Documentos$ cd Auditoria/
idaseg@debian:~/Documentos/Auditoria$ pwd
/home/idaseg/Documentos/Auditoria
idaseg@debian:~/Documentos/Auditoria$ ls
Analise-de-clientes.txt
```

## Caminhos absolutos e relativos

O comando `pwd` sempre mostra um *caminho absoluto*, ou seja, o caminho contém todas as etapas, desde o início do sistema de arquivos (`/`) até o final (`Auditoria`). Os caminhos absolutos sempre começam com um `/`.

```
/ └─ home
    └─ user
        └─ Documentos
            └─ Auditoria
```

O caminho absoluto contém todas as informações necessárias para se chegar a Auditoria a partir de qualquer lugar no sistema de arquivos. A desvantagem é que é tedioso digitá-lo inteiro.

O segundo exemplo (`cd Auditoria`) foi muito mais fácil de digitar. Esse é um exemplo de *caminho relativo*. Os caminhos relativos são mais curtos, mas só significam em relação à sua localização actual.

## Caminhos relativos especiais

O shell do Linux oferece maneiras de encurtar os caminhos durante a navegação. Para revelar os primeiros caminhos especiais, inserimos o comando `ls` com a flag `-a`. Essa flag modifica o comando `ls` para que *todos* os arquivos e directórios sejam listados, incluindo os arquivos e directórios ocultos:

```
idaseg@debian:~/Documentos/Auditoria$ ls -a
.  ..  Analise-de-clientes.txt
```

Esse comando revelou dois resultados adicionais: estes são caminhos especiais. Eles não representam novos arquivos ou diretórios, mas sim diretórios que você já conhece:

. Indica o *local atual* (neste caso, `Auditoria`).

.. Indica o *diretório pai* (neste caso, `Documentos`).

Geralmente, não é necessário usar o caminho relativo especial para o local atual. É mais fácil e compreensível digitar `Analise-de-clientes.txt` do que digitar `./analise-de-clientes.txt`. Desse modo, o caminho relativo para o diretório pai é:

```
idaseg@debian:~/Documentos/Auditoria$ cd ..
idaseg@debian:~/Documentos$ pwd
/home/idaseg/Documentos
```

O exemplo de `cd` é muito mais fácil quando se usa `..` em vez do caminho absoluto. Além disso, podemos combinar esse padrão para subir rapidamente pela árvore de arquivos.

```
idaseg@debian:~/Documentos$ cd ../../
idaseg@debian:/home$ pwd
/home
```

## Conclusão

Comandos de navegação importantes para usuários Linux, pois eles permitem que os mesmos acessem a diversas partes da hierarquia de diretórios e arquivos do Linux. Portanto, espero que esse artigo ajude o máximo de pessoas possível que desejam manusear melhor o Linux.

Siga-me e veja outros posts no meu Instagram <https://www.instagram.com/idaeciosilvatech/>

Deixe o like para medirmos a repercussão do conteúdo.

Guarde para rever noutro momento.

Compartilha o conteúdo com os amigos e não só

Comente o que acha do conteúdo e da iniciativa e deixe sugestões.

[#cybersecurity](#) [#technology](#) [#linux](#) [#CLI](#) [#Shell](#) [#changedirectory](#) [#listdirectory](#)