

## Comandos básicos do terminal

1. `pwd` (print working directory): Retorna o diretório atual em que você está.
2. `ls`: Vai mostrar todos os arquivos e pastas existentes dentro do diretório atual.
3. `cd` (change directory): Ao digitar `cd nome_do_diretorio` você vai entrar dentro dele.
4. `cd . .` (cd dois pontos): Volta um diretório.
5. `cd ~`: Te leva para o diretório inicial.
6. `rm -f`: Você vai ter que confirmar a exclusão do arquivo.
7. `rm -rf`: Exclui o arquivo ou diretório sem perguntar para você se deseja confirmar a exclusão. (**CUIDADO!!**)
8. `mkdir`: Criando diretórios: `mkdir nome_dir`
9. `touch`: Criando arquivos: `touch index.js`
10. `mv`: movendo arquivos e pastas: `mv file1 file2 diretório_final`
11. `mv` **PARA RENOMEAR ARQUIVO OU DIRETÓRIO** - Não pode ser o nome de outro diretório ou arquivo:
  - `mv nome_atual novo_nome`
12. `find`: Procurando arquivos: `find diretório -iname 'nome_do_arquivo'`
  - Se você só lembra uma parte do nome do arquivo, você pode usar assim:  
`find diretório -iname '*partequedvocelembra*'`
13. `grep`: Procura por trechos de texto dentro dos arquivos ou diretórios e retorna onde eles foram encontrados.
  - Procurando um texto: `grep "trecho de texto" *`
  - Procurando dentro de diretório e sub-diretórios: `grep -R "trecho" /var/www/`
  - `grep -i "trecho"` (independente de LOWER CASE OU UPPER)
14. `cat`: Permite que você crie, una e exiba arquivos no formato padrão de tela ou em outro arquivo, entre outras coisas.
  - Sintaxe básica:
    - `cat [OPÇÃO] [ARQUIVO]`
  - Criar novo arquivo:
    - `cat > arquivo.txt`
  - Visualizar conteúdo de um Arquivo:
    - `cat arquivo.txt`
  - Redirecionar conteúdo:
    - `cat fonte.txt > destino.txt`

## Comandos essenciais e eficientes

1. `awk`: Muito conhecido pela eficácia de criar filtros de conteúdos de arquivos.
  - Supondo que tenhamos um `coluns.txt`, dessa maneira:  

```
188 :2106102121 192.168.0.89 267 :2206211021 192.168.0.88 438
:2306211021 192.168.0.87
```
  - `cat coluns.txt | awk '{print $1}'`, vai te retornar:

```
188
267
438
```

- Contar as linhas do resultado acima: `cat coluns.txt | '{print $1}' | wc -1`
- Dentro do nosso arquivo temos um trecho que começa com ":" que se refere à uma data. Precisamos exibir, os 4 primeiros caracteres após o ":".
- `cat coluns.txt | awf -F ":" '{print substr($2,1,4)}'`
- `awk -F ":" ->` Ignora o caractere ":" da segunda coluna;
- `{print substr($2,1,4)}'` -> Exibe a segunda coluna do primeiro ao quarto caractere. ]

2. lsof: O comando basicamente lista os arquivos abertos pelos processos em execução no OS. (no linux tudo é arquivo.) - O comando lsof deve ser analisado com calma, pois existem diversas colunas em sua saída e cada uma contém um significado. (<http://bacana.one/como-usar-o-comando-lsof-do-linux>)

3. curl: Na maioria dos sistemas baseado em Unix, é usado como abreviação para Client URL, verifica conectividade com URL e também é uma ferramenta de transferência de dados.

- Sintaxe básica: `curl [OPTIONS] [URL]`
- Uso mais simples, mostrar o conteúdo de uma página:
  - `curl sitefake.com` (renderiza o código fonte da página)
- Fazendo downloads de arquivos de um local remoto:
  - `curl -O http://sitefake.com/testefile.tar.gz`
  - -O: vai salvar o arquivo no diretório atual.
  - -o: Permite especificar um nome para o arquivo ou local
- Requisições HTTP:
  - POST: `curl -data "text=Hello" https://myDomain.com/firstPage.jsp`
  - GET: `curl http://mydomain.com`

4. wget: Serve para recuperar conteúdos e arquivos da internet, (World Wide Web Get)

- `wget https://wordpress.org/latest.zip`
- Download de multiplos arquivos:
  - Crie um arquivo txt com o link de seus downloads. (nano downloads.txt)
  - E use: `wget -i downloads.txt`
- Limitando a velocidade de download:
  - `wget --limit-rate=500k https://wordpress.org/latest.zip`
- Download em segundo plano:
  - `wget -b http://example.com/beefy-file.tar.gz`

5. tail: Comando utilizado para exibir os últimos registros de um arquivo de texto.

- Últimas 20 linhas do arquivo teste.txt
  - `tails -n 20 teste.txt`
- Examinando um arquivo que está sendo atualizado constantemente como arquivos de logs:
  - `tail -f /var/log/syslog`

6. head: Função inversa do tail.

7. less: Permite a paginação de arquivos

- Você pode usar:
  - less Artigo.txt (para pagnar normalmente)
- Ou utilizar:
  - less -N Artigo.txt (para pagnar e numerar as linhas)

8. ssh: SSH Client é um programa para acessar uma máquina remota e executar comandos nela.

- Sintaxe básica:
  - ssh [opções] [usuário@]hostname [comando]
- Opções de comando:
  - -4: força ssh usar apenas endereços IPv4
  - -6: força ssh usar apenas endereços IPv6
  - -p: indica a porta para conexão na máquina remota.
  - -q: modo "quiet". Suprime a maior parte das mensagens de aviso e diagnóstico.
  - -V: apenas exibe o número da versão do ssh e sai do ssh.
  - -v: Modo verboso. O ssh exibe na tela mensagens de debug sobre o seu progresso.
- Exemplo completo:
  - ssh -p 22 [root@10.1.2.3](#)

9. kill: Serve para "matar" um processo, ele permite que você mate através de um pid:

- kill pid
- Se você conhece o nome do processo, você pode usar uma vertente do kill, o pkill.
- pkill chrome