

Capítulo 3 - Comandos Parte 2 e Editor de textos VI

3.1. Objetivos

Conhecer o editor de textos mais usado no mundo UNIX que é o VI (Visual Interface)

Redirecionamento de entrada e saída:

Esta parte é extremamente importante, pois se trabalha com isso sempre. Por default(padrão), a entrada do Shell é o teclado, a saída a tela, e os erros a tela também. Mas isso pode ser mudado com o uso de caracteres de redirecionamento, veja abaixo:

Para mudar saída padrão:



> Redireciona a saída em um arquivo apagando o conteúdo anterior(se existir);
>> Redireciona a saída no final de um arquivo, preservando-o;
2> Faz o mesmo que o > mas acrescenta os erros da saída;
2>> Faz o mesmo que o >> mas acrescenta os erros da saída;

Para mudar entrada padrão:



< Indica para o Shell que a entrada não será o teclado;
<< Serve para indicar o escopo de um programa (rótulo);

Comandos auxiliares:



| (pipe, pronuncia-se paípe) Serve para canalizar saída de dado para outro comando;
***tee** Serve para canalizar saída de dado para um arquivo;*
***&** Substitui o 2>>*



Atenção: Para seguir os exemplos abaixo depois, abra um shell e crie um diretório chamado "shell" em seu home e nele acrescente os arquivos script1,script2,script3.



Observação: O diretório papel/ inexistente.

Teste o seguinte comando:



```
$ ls shell/ papel/ > log_ls.txt
```

O comando `ls` listará os diretórios `shell/` e `papel/` e redirecionará a saída para o arquivo `log_ls.txt`.

Durante a execução do comando, será exibido a seguinte mensagem:



```
"ls: papel/: Arquivo ou diretório não encontrado"
```

(Saída de erro), como foi usado o `>` ao invés de `>>` (para erro), o que ficou no arquivo foi só a saída certa. Para ver o conteúdo do `log_ls.txt`:



```
$ cat log_ls.txt
```

Para acrescentar a saída com erros, mude o comando para:



```
$ ls papel/ 2>> log_ls.txt
```

Exemplo 2 (pipe e tee):



```
$ ls shell/ | sort | tee listagem.txt
```

Este comando lista o conteúdo do diretório `"shell"` canalizando sua saída para o comando `"sort"`, que ordena os arquivos por ordem alfabética, canalizando sua saída para o comando `"tee"` que canaliza toda a saída para o arquivo `"listagem.txt"`.

Exemplo de uso do `<<`

Este redirecionamento serve principalmente para marcar o fim de exibição de um bloco. Este é especialmente usado em conjunto com o comando `cat`, mas também tem outras aplicações.

Exemplo:



```
$ cat << final >> teste.txt  
estou  
realizando um  
teste  
final
```



Por que o shell apenas foi liberado depois que foi digitada a palavra final?

Porque este comando irá concatenar o conteúdo digitado e colocá-lo dentro de teste.txt até que a palavra final seja digitada.



```
$ cat teste.txt  
estou  
realizando um  
teste
```

Nesse caso poderia usar qualquer outra palavra além de final para que o comando cat parasse de concatenar o conteúdo digitado dentro de teste.txt.

Como eu poderia listar apenas os arquivos que começam com a letra s?



```
$ ls | grep ^s
```

E para listar todos arquivos terminados com a letra a?



```
$ ls | grep a$
```



*^ indica início de linha
\$ indica fim de linha*

Compreendendo melhor:

Faça mais um exemplo para entender os metacaracteres de início e final de linha:



```
$ cat << end >> teste2.txt
```

```
alex
```

```
bruno
```

```
camila
```

```
cintia
```

```
daniela
```

```
eduardo
```

```
ezequiel
```

```
fernanda
```

```
fabricio
```

```
end
```

Depois de colocar esses nomes, um em cada linha do arquivo teste2.txt vai verificar o conteúdo do mesmo:



```
$ cat teste2.txt
```

```
alex
```

```
bruno
```

```
camila
```

```
cintia
```

```
daniela
```

```
eduardo
```

```
ezequiel
```

```
fernanda
```

```
fabricio
```

Se estiver tudo ok filtre algumas linhas deste texto:



```
$ cat teste2.txt | grep ^c
```

camila

cintia



```
$ cat teste2.txt | grep a$
```

camila

cintia

daniela

fernanda

Como seria possível listar somente os diretórios usando o comando ls?



```
$ ls -l | grep ^d
```

drwxr-xr-x 2 linux linux 1024 2007-07-11 15:37 teste

Tendo em vista que a saída do comando `ls -l` sempre mostra primeiro as propriedades dos arquivos, pode saber se tratá-se de um arquivo ou diretório apenas observando o primeiro carácter de cada linha da saída do comando `ls -l`.

3.2. Editor de Texto VI (pronuncia-se “viai”)

Usa-se o VI para manipular (criar, alterar) arquivos no modo texto. É extremamente importante aprender a mexer em um editor de texto. O VI no começo parece ser um editor de textos alienígena, pois a maioria das pessoas está acostumada com editores WYSIWYG (What You See Is What You Get - O que vê é o que tem, por exemplo OpenOffice.org, Microsoft Office).



Nosso Curso escolheu o VI por ser um editor de texto padrão do GNU/Linux, ou seja, qualquer distro terá esse editor. Então não fica preso a distro. E ele também é um assunto que a LPI pede. LPI é a certificação para um profissional GNU/Linux.

- É pequeno, portanto carrega rapidamente e não ocupa muito espaço na memória;
- Pode ser usado em qualquer layout de teclado;
- Por ser 100% texto, sua dependência de configuração é mínima;
- Seus comandos aceitam expressões regulares, tornando-o muito poderoso e versátil;

Não existe apenas o VI como editor de texto no GNU/Linux. Quais outros editores de texto encontramos no Linux?

Tem o emacs, mcedit, pico (nano) etc. Ou seja editores para todos os gostos.

Para acessar um arquivo com o vi faça assim:



```
# vi /tmp/teste.txt
```

Ele irá abrir o arquivo, e, se o mesmo não existir no sistema, ele irá criar na hora. Caso esse arquivo esteja vazio, será mostrado apenas uma tela toda preta assim:



```
~  
~  
~  
~
```

Onde cada ~ representa uma linha!

3.2.1. O VI possui 3 modos de operação:



Modo de comando: Para dar ordens ao editor

Modo de linha de comando: Esse modo começa sempre com ":"

Modo de inserção: Para inserir texto

Sempre ao iniciar o VI ele estará no modo de comando.

Assim que entrar no VI se quiser começar a escrever no meu arquivo é necessário **teclar i**. Ou seja, teclo i para que ele entre em modo inserção. Quando eu teclar i vai aparecer no canto inferior esquerdo a palavra INSERT

Algumas versões não aparece o INSERT, mas mesmo assim ele entra em modo de inserção. Senão aparecer o INSERT quando teclar i, é que sua versão de Vi é um pouco menos didática, mas funciona do mesmo jeito.

Na verdade o VI é um "copyrighted", o que usa no Linux são clones dele, e o clone mais famoso é o VIM (VI Improved), que é praticamente 100% compatível com o VI. VIM é software livre, está sob licença GPL. Então, comesse a digitar o texto:

Bem Vindo ao Meu Sistema GNU/Linux!!!

Assim que terminar de digitar o texto, terá que sair do modo de inserção apertando o ESC. E depois que apertar o ESC, não poderá mais inserir texto nenhum. Então, se precisa salvar o arquivo que acabou de digitar, teclo " : "

Isso vai fazer com que apareça o : no canto inferior esquerdo da minha tela. Ou seja, ele está esperando teclar um comando para ele. Ou seja, quando o VI ficar com o : no canto inferior da tela ele está dizendo: **Pode digitar o comando para eu executar!**

Agora veja o comando que quer para o VI. Neste caso é preciso salvar o documento, então teclamos w . Isso faz com que ele salve o documento e o : vai sumir. Esse comando:



:w

Grava alterações em um arquivo que já foi aberto com um nome.

Agora, se apenas digitar:



```
# vi
```

Ele abre um arquivo em branco e sem nome, então é necessário salvar com nome:



```
:w nome_arquivo
```

Agora que já escreveu e salvou o arquivo, é necessário sair do vi. Então, novamente falo para o VI que é preciso executar um comando teclando:



```
:
```

E digite o comando para sair que é o q de quit! Vai ficar assim:



```
:q
```

E se for necessário fazer wq ao mesmo tempo, ou seja, se for preciso salvar e sair usando apenas um carácter ? Basta digitar:



```
:x
```

O x já faz os dois ao mesmo tempo (wq), ele salva e depois sai de uma vez só.
TREINANDO: Digite um texto no VI. Exemplo:

Todos os pacotes incluídos na distribuição oficial do Debian são livres de acordo com a Definição Debian de Software Livre. Isso assegura o uso livre e a redistribuição de pacotes com seu código fonte completo. A distribuição oficial Debian é a que está contida na seção main do repositório do Debian.

Agora substitua a palavra distribuição por distro, apenas na primeira ocorrência da palavra:



```
:s/distribuição/distro
```

Agora substitua a palavra Debian por DEBIAN, todas as vezes que essa palavra aparecer no texto:



```
:% s/Debian/DEBIAN/g
```

Onde g indica (todas as ocorrências)



Para pegar VI tem que praticar!!!



É importante falar que o VI tem mil maneiras de fazer a mesma coisa, então escolha uma em que melhor se adaptar.



Espero que todos pratiquem VI, mas é importante falar que não é obrigatório usá-lo. Se você já tem preferência para outro editor, sem problemas, mas não esqueça da certificação LPI.



Dica de estudo para o VIM:

<http://aurelio.net/doc/vim/>

Testando as opções do VIM:

<http://aurelio.net/vim/vim-avancado.txt>

<http://aurelio.net/vim/vim-basico.txt>

<http://aurelio.net/vim/vim-medio.txt>