

Lista de Exercícios 05

Entradas, Saídas e Redirecionamentos

Exercício 01) Descreva em detalhes o que cada um dos comandos a seguir está fazendo indicando claramente o efeito do redirecionamento.

a) `$ date > date`

b) `$ wc -l < /usr/share/dict/words`

c) `$ cat /etc/shadow 2>shadow`

d) `$ find /var -type f 1>/dev/null`

Exercício 02) O comando de copiar (`cp`), se executado sem parâmetros, exibe mensagem informando que faltam operandos:

```
$ cp
cp: missing file operand
Try 'cp --help' for more information.
```

a) Essa mensagem é exibida na saída padrão ou na de erro?

b) Mostre, com um comando (sem usar `strace` ou comandos similares de depuração), o porquê da resposta do item (a).

Exercício 03) Mostre passo a passo o que cada comando encadeado pelos *pipes* (|) está fazendo. Depois, mostre qual o **propósito** desse comando completo, ou seja, indique para que ele serve.

a) `$ cat /etc/passwd | cut -f7 -d':' | sort | uniq`

b) `$ ps aux | sort -k3 -n | tail -n 1`

c) `$ cat /etc/resolv.conf | grep nameserver | wc -l`

d) `$ ls -l /tmp | tail -n +2 | awk '{print $3}' \`
`| sort | uniq -c`

Exercício 05) Comandos que não aceitam dados da entrada padrão, somente por parâmetros, limitam sua aplicação com *pipes*. Existe um utilitário muito útil capaz de transformar a saída padrão em argumentos para outro programa chamado `xargs`. Reescreva os comandos a seguir usando `xargs` para contornar essa limitação.

a) Identificar o nome base do diretório atual.

```
$ pwd | basename
basename: missing operand
Try 'basename --help' for more information.
```

```
$ _____
```

b) Remover arquivos temporários.

```
$ find /tmp -type f -name '*.tmp' | rm
usage: rm [-f | -i] [-dPRrvW] file ...
        unlink file
```

```
$ _____
```