

Comandos Basicos Linux

Comandos básicos

`cd diretorio`: abre um diretório. Por exemplo, para abrir a pasta /mnt, basta digitar `cd /mnt`. Para ir ao diretório raiz a partir de qualquer outro, digite apenas `cd`.

Para este comando existem abreviações, tais como:

- .** (ponto) => Diretório atual
- ..** (dois pontos) => Diretório anterior
- ~** (til) => Diretório HOME do utilizador
- /** (barra) => Diretório Raiz
- (hifen) => Último diretório

`ls [-al]`: listagem do diretório.

`cp [-ir]`: copiar arquivos.

`mv [-i]`: mover ou renomear arquivos.

`rm [--]`: deletar arquivos.

`mkdir/rmdir`: cria/deleta diretórios.

`ln -s path link`: cria links simbólicos (symlinks) para arquivos ou diretórios.

Outros comandos

`file`: determina o tipo do arquivo (/etc/magic).

`cat`: exibe o conteúdo do arquivo na tela.

`head / tail`: exibe linhas no início / fim do arquivo.

`less / more`: lista o conteúdo do arquivo.

`man filename`: manual online do programa.

`ctrl+alt+del/reboot`: reinicia o sistema.

`shutdown -h now/halt`: desliga o computador.

Combinações

CTRL+C: sai (kill) do programa.

CTRL+ALT+BackSpace: sai (kill) do servidor X.

CTRL+L: limpa a tela.

CTRL+A / E: move o cursor para o início / fim da linha.

CTRL+U / K: deleta da posição do cursor até o início / fim da linha.

CTRL+H: deleta palavra anterior ao cursor.

CTRL+R: busca comando digitado no `history` do bash.

CTRL+D: logout (para isto altere ou unset a var. `$IGNOREEOF`).

Mais no terminal

`stty -a`: lista configurações do terminal.

`reset`: reseta o terminal (volta ao normal).

(SHIFT)PGUP/PGDN: barra de rolagem do bash.

TAB: auto-completa os comandos digitados no terminal.

MOUSE2/3: cola o texto selecionado (gpm).

CTRL+S (Scroll Lock): desabilita o vt.

CTRL+Q (Scroll Lock): habilita o vt (tente isto caso o terminal trave).

ALT+Fx: muda de console. **CTRL+ALT+Fx**: muda de console em modo gráfico.

Usuários

`w`: informações gerais sobre usuários logados e seus processos.
`who`: informações dos usuários atuais (do utmp)
`last`: listagem do histórico de logins (/var/log/wtmp)
`lastlog`: retorna informações sobre últimos logins.

Processos

`CTRL+Z`: suspende o processo temporariamente.
`top`: os processos que consomem mais recursos do sistema.
`jobs`: lista as tarefas rodando em fore/background.
`bg/fg`: manda processo para o back/foreground.
`nice/renice`: altera prioridades.
`ps -auxw`: lista todos os processos do sistema:
 PID (process id), TTY (terminal ou ? caso seja um daemon), STAT (estado do processo), TIME (tempo de CPU consumido), COMMAND (o comando executado).
`pstree -p`: idem.
`time`: calcula o tempo decorrente do início ao término de um processo.

```
# time updatedb
real 1m42.233s
user 0m0.490s
sys 0m10.290s
```

Matando processos

`kill`: as opções mais comuns são (onde id é o mesmo que PID):
`kill -HUP id-do-processo`: reinicia processo.
`kill -9 id-do-processo`: mata processo.
`killall processo`: mata processo pelo nome.
`killall -HUP processo`: reinicia processo pelo nome.

Sistema

`df -h` (espaço livre e ocupado nos discos)
`du -sh(x)` (espaço ocupado pelo diretório e seus subdiretórios)
 Informações do sistema:
`free`: status da memória e swap.
`vmstat`: status da memória virtual (processos, cpu).
`lsdev, lspci`: listagem do hardware/dispositivos pci.
`pnpdump`: retorna configuração das placas ISA PnP.
`lsmod / rmmod`: lista/remove módulos na memória.
`procinfo`: cat /proc ;)
 Informações do ambiente X:
`xdpyinfo`: recursos do servidor X.
`showrgbq`: retorna a database de cores rgb.
`xlsfonts`: lista as fontes reconhecidas pelo X.
`xset m 5/2 1`: ajusta a velocidade e acel. do mouse.

Rede

Listando processos listening na porta 80 e seus PIDs
`lsof -n -i:80` (-i4: ipv4 e -n: sem resolver hostnames)
`fuser -v 80/tcp` (lista processos que escutam na porta tcp 80 em modo ps-like)

Pipes e Redirecionamentos...

Através de 'pipes', a saída de um comando (stdout [*file descriptor* 1]) é passada como a entrada (stdin) do outro. Em geral, a saída é o terminal e a entrada o teclado. Exemplos:

```
dmesg | less ; ls -l | more
echo "Broadcast Message" | wall
```

Através de '<' e '>' é possível definir qual será o stdin e o stdout. Exemplos:

```
dmesg > dmesg.txt ; more < dmesg.txt
ls -l /tmp >> list.txt (concatena)
```

O sinal de maior é equivalente ao comando `tee`.

E para alterar o stderr (mensagens de erro [*file descriptor* 2]):

```
ls /admin > list.txt 2>erros.txt (se o diretório não existir, o
stderr será o erros.txt e o stdout o list.txt)
ls /admin > list.txt 2>&l listagem_e_erros.txt (se houver, erros
serão repassados para o mesmo local que o stdout)
```

Operadores Lógicos

`&&`: 'e' (retorna *true* se todas as expressões forem verdadeiras)

`||`: 'ou' (retorna *true* se uma das expressões forem verdadeiras)

O sinal ';' executará ambas as expressões independente do retorno. Por exemplo:

```
make ; make install (os comandos serão executados em sequência)
make && make install (o segundo comando somente será executado se o
primeiro não retornar erro)
```

Permissões

As permissões dos arquivos são definidas através dos comandos `chmod`, `chown` e `chgrp`.

Estrutura do comando:

`chmod`

Ao listar as informações de um arquivo ou diretório, o formato é o seguinte:

`drwxrwxrwx.`

Respectivamente: diretório (d), permissão do dono (read/write/execute), do grupo (read/write/execute) e de outros (read/write/execute).

Por exemplo, para transformar um arquivo em executável:

- `chmod +x nome_do_arquivo` (executável para todos)
- `chmod g+x nome_do_arquivo` (executável para o grupo)

Para alterar o usuário e o grupo de um arquivo ou diretório:

- `chown root.root /sbin/firewall.sh (-R: recursivamente)`

Outros exemplos:

- `chmod 755` (executável): `-rwxr-xr-x`
- `chmod 4700` (suid) set user id para programas que precisam rodar com

permissão de root: `-rws-----`

Para calcular o valor numérico das permissões, basta considerar o valor do executável como 1, de escrita como 2 e de leitura como 4, que seria o equivalente decimal aos bits:

$rwx = 111$ (todos bits ligados) $= 2^{**2} + 2^{**1} + 2^{**0} = 7$

Dessa forma, uma permissão de leitura e escrita (4+2) para o owner, e de leitura apenas para os outros teria o valor 644. Para calcular a *umask*, que seria a máscara de permissão aplicada na criação de um novo arquivo, basta então subtrair 666 (ou 777 para diretórios) resultando em `umask 022`.

Como se encontrar no sistema

Localizar arquivo por nome:

```
find [path...] -name [nome_do_arquivo]
find . -name slackware.png
find / -name "*.png" -print (arquivos png do dir. atual)
find /home -size +5000k -print (arquivos com mais de 5Mb)
```

Local de um binário:

```
whereis (ou which) [nome_do_arquivo]
which gcc
gcc: /usr/bin/gcc
```

Criar um banco de dados com o local dos arquivos (para busca imediata)

```
updatedb
Para pesquisar: (s)locate [nome_do_arquivo]
```

Localizar texto em arquivo:

```
grep [param] [texto] [arquivo]
grep -ni man /var/log/packages/grep.tgz (-i : case insensitive, -n
: número da linha)
(use '(aspas simples) no [texto] para procurar palavra exata.)

ls -l | grep '^-..x' (lista executáveis)
ls -l | grep '^d' (lista diretórios - '^' indica a
primeira letra da linha)
```

Outros:

```
pwd: listar caminho atual
```

Verificando integridade de um arquivo:

```
sum: retorna checksum (16-bit) e número de blocos de cada arquivo.
$ sum arquivo.tar.gz
cksum: retorna CRC checksum e bytes de cada arquivo.
$ cksum *.tar.gz > cksums
md5sum: retorna o MD5 checksum (128-bit) do arquivo, e verifica a
```

consistência.
 \$ md5sum --check MD5SUMS

Operações com texto:

comm/diff: compara dois arquivos.
ispell: verificador ortográfico (-d br: dicionário em [português](#)).
sort: ordena em ordem crescente, alfabética, etc.
uniq: remove linhas duplicadas.
cut: retorna area delimitada (-c5: quinto caracter).
wc: conta linhas, palavras e bytes.
fold: ajusta o texto para a largura especificada.
nl: numera as linhas de um arquivo.
fmt: reformata as linhas de um arquivo.
expand/unexpand: converte tabs em espaços e vice-versa.
tr: remove e substitui caracteres (-d a-d para remover as letras entre a-d, tr a-d A-D para torná-las maiúsculas).

Alterar data do sistema:

```
$ date 0109143001 (09/01/2001 14:30)
$ date -s "mm/dd/aaaa hh:mm:ss"
```

Criando aliases

Adicionar arquivo de alias (ex: aliases.sh) em /etc/profile.d/

```
chmod 755 aliases.sh
```

```
# exemplo de script alias.sh
# System wide functions and aliases
alias cdrom.on="mount /dev/hdd /mnt/cdrom"
alias cdrom.off="umount /dev/hdd"
alias zipdisk="mount -t vfat /dev/hdb4 /mnt/zip"
alias rm="rm -i"
alias x="startx -- -nolisten tcp"
```

Criando arquivos compactados

tar.gz

É possível criar tarfile (tarball) da seguinte forma:

```
tar -c file1 file2 > foo.tar
```

Para 'zipar' o tar a mesma coisa:

```
gzip -c file1 file2 > foo.tar.gz
```

Ou então:

```
tar -cvzf diretorio.tar.gz diretorio/
```

tar.bz2

Troque o 'z' pelo 'j' para criar arquivos bzip2.

```
tar -jxvf arquivo.tar.bz2 (descompactando bz2)
```

zip

```
zip -r filename.zip files
```

O editor vi(m)

`a / i`: adiciona texto (append/insert).
`y / p`: copiar e colar (ycopy/paste).
`x`: deletar letra.
`/`: buscar palavra (n: next N: prev).
`u`: desfazer (undo).
`v`: selecionar texto (visual).
`:edit file`: cria novo buffer.
`:ls`: lista buffers.
`:buffer N`: edita buffer N.
`:bdelete`: deleta buffer.
`:so $VIMRUNTIME/syntax/2html.vim`: converte source em html.
`:ab SW slackware`: cria abreviação (abclear: remove abs.)
`map :qa!`: mapeia tecla (adicione em .vimrc)
 Para gravar digite ESC seguido por :wq (write and quit), :q! (sair sem gravar) ou :qa! (fechar todos).

Outros utilitários no console

`whatis/apropos`: descrição do programa.
`bc`: calculadora (ex: `echo "scale=2;1/10" | bc` //scale são as casas decimais).
`nano`: editor de texto simples (nano-editor.org).
`jed`: editor de texto para programadores.
`mc`: o midnight commander.

Magic SysRq!

Para situações onde não temos acesso ao sistema via ssh/telnet, a combinação ALT + SysRq/PrintScrn + X pode ser muito útil.

Ex: ALT + SysRq + ... **k**: mata o VC atual; **e**: sigterm (exceto ao init); **i**: sigkill (exceto ao init)

Para reiniciar o sistema de forma segura, digite na sequência:

Alt+SysRq **S** (sync), Alt+SysRq **U** (unmount), Alt+SysRq **B** (reboot)

No terminal, os processos serão descritos, portanto apenas siga para o próximo passo após o término do primeiro.

Links úteis:

http://www.devin.com.br/comandos_linux (Muito bom em Portugues)

<http://www.vivaolinux.com.br>

<http://www.dicas-l.com.br>

<http://oreilly.com/linux/command-directory> (ótimo mas em Inglês)