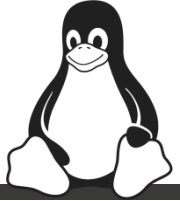




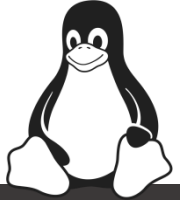
Prof. Vitor Luiz

Simulado LPIC-1



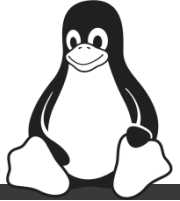
PARAMETROS DE REDIRECIONAMENTO

- Também chamados de stdin e stdout.
- **Pipe (|)**
 - > Redireciona a saída de um comando a entrada de outro, ou seja, concatena.
- **Write (>)**
 - > Redireciona a saída para outro local (um arquivo, por exemplo).
 - > Cria o arquivo, caso não exista.
- **Read (<)**
 - > Recebe os parametros de um arquivo como entrada de um comando.
- **Append (>>)**
 - > Redireciona um parametro para a última linha de outro arquivo.



EDITORES DE TEXTO

- Entre os vários editores de texto existentes para o Unix, o **vi** e o **emacs** são universais.
- Os editores de textos podem ser utilizados para criar arquivos de texto, shell, scripts e modificar arquivos de configuração do sistema operacional.
- Além do emacs e vi, outros editores de textos mais simples ~são:
 - > **pico**
 - > **nano**
 - > **mcedit**



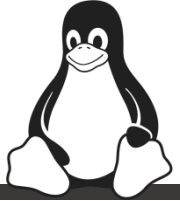
EDITORES DE TEXTO

■ nano

- > Editor de texto nativo no Debian
- > Pode ser utilizado para criar arquivos de texto, shell, scripts e modificar arquivos de configuração do sistema operacional

■ Sintaxe: `$nano` ou `$nano arquivo`

■ As teclas de atalho do nano estão descritas no rodapé do editor e utilizam a tecla Ctrl como atalho.



EDITOR DE TEXTO

■ vi

- > Editor de texto fantástico para o modo de comando do Linux.
- > Possui dois modos de edição.

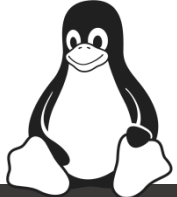
- Modo Escrita

- <i> ou <insert>

- Modo Comando

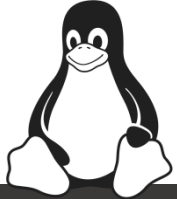
- <esc>

■ Sintaxe: `$vi` ou `$vi arquivo`



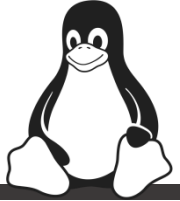
EDITORES DE TEXTO

| Comando | Ação |
|------------------|--|
| : + q | Sai do vi sem salvar (pode solicitar gravação) |
| : + q! | Sai do vi forçadamente sem salvar |
| : + w | Salva o arquivo atual |
| : + wq | Sai do vi salvando o arquivo atual |
| : + w<nome arq.> | Salva o arquivo atual com o nome do arquivo |
| : + e<nome arq.> | Abre o arquivo informado |
| : + r<nome arq.> | Insere o arquivo informado no ponto onde está o cursor |
| : + u | Desfazer a última ação |



EDITORES DE TEXTO

| Comando | Ação |
|---------------------------|---|
| cc (c 2x) | Recorta a linha onde o cursor está posicionado |
| dd (d 2x) | Apaga a linha atual onde o cursor está posicionado |
| y | Copia a linha atual para o bufer |
| p | Cola a última entrada colocada no buffer |
| v | Modo seleção de texto |
| /padrão | Localiza por um padrão dentro do texto |
| n | Vai para a próxima entrada do padrão informado |
| :%s/palavra/novaentrada/g | Substitui todas as ocorrências de uma palavra, por outra. |



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

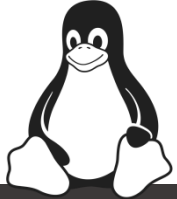
■ **find**

- > Procura por arquivos e diretórios no sistema operacional
- > Permite busca por nome, tamanho, data de criação, data de acesso, por tipo, etc.

■ Sintaxe: `$find <caminho> <expressão>`

■ Expressões:

- name <arquivo>: Procura pelo nome do arquivo informado
- iname <arquivo>: Procura pelo arquivo independente se está escrito em maiúsculo ou minúsculo
- not: Nega uma expressão
- type: Determinados tipos de arquivos.
- mtime <período>: Procura por arquivos modificados nos últimos n períodos.
- size <tamanho>: Procura por arquivos com determinado tamanho
- empty: Procura por arquivos ou diretórios vazios.
- atime <período>: Procura por arquivos acessados no último n períodos.
- ctime <período>: Procura por arquivos criados no último n período.



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **find**

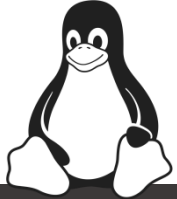
- Exemplos:

- > Procura pelo arquivo exemplo.xls a partir da raiz (/), do tipo arquivo (f)

```
root@debian:~# find / -name exemplo.xls -type f  
/root/exemplo.xls
```

- > Procura por todos os arquivos com a extensão .conf a partir do diretório atual /etc (.)

```
root@debian:/etc# find . -name "*.conf"  
./apt/listchanges.conf  
./insserv.conf  
./systemd/timesyncd.conf  
./systemd/bootchart.conf  
./systemd/resolved.conf
```



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

■ find

■ Exemplos:

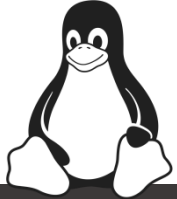
- > Procura por arquivos que **NÃO** terminam com *.conf a partir do diretório atual

```
root@debian:/etc# find . -type f -not -name "*.conf"
./aliases
./apt/apt.conf.d/70debconf
./apt/apt.conf.d/01autoremove-kernels
```

- > Procura os arquivos modificados nos últimos 7 dias

```
root@debian:/etc# find . -mtime -7 -type f
./ld.so.cache
./passwd-
./subuid-
./mysql/debian.cnf
```

- O sinal de menos (-) indica o número de dias passados.



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **find**

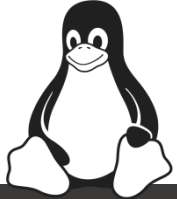
- Exemplos:

- > Procura por arquivos SUID

```
root@debian:/etc# find / -perm /u=s  
/sbin/mount.nfs  
/usr/sbin/exim4
```

- > Procura arquivos criados (ctime) nas últimas 6 horas.

```
root@debian:/etc# find . -ctime -0,25 -type f  
./ld.so.cache  
./init/syslog-ng.conf  
./logcheck/violations.ignore.d/syslog-ng  
./logcheck/ignore.d.server/syslog-ng
```



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **find**

- Exemplos:

- > Procura por acessados (amin) as últimas 24 horas

- ```
$find . -type f -atime -1
```

- > Procura por arquivos maiores (+) que 10MB

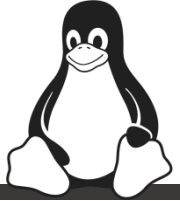
- ```
$find . -type f -size +10M
```

- > Procura por diretórios vazios a partir da pasta /root

- ```
$find /root -type d -empty
```

- > Procura por nome dentro do arquivo

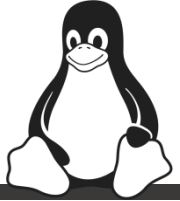
- ```
$find /root -name "*. *" -exec grep -Hin "seguranca" {} \;
```



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

■ Expressões regulares

- > Método formal de se especificar um padrão de texto a ser procurado em um ou mais arquivos.
- > É composta de caracteres com funções especiais (**metacaracteres**) que agrupados com caracteres (de A a Z) formam uma expressão que o shell e editores podem entender e buscar.
- > As expressões regulares são úteis para buscar ou validar textos variáveis como:
 - Endereço IP
 - Endereço de e-mail
 - URL's
 - Números de CNPJ, RG, CPF, etc...



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

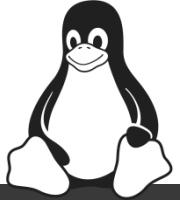
■ **grep**

- > Filtra linhas de um determinado arquivo procurando por uma expressão regular.
- > Pode ler um ou mais arquivos que são passados como argumentos.
 - Indica o nome do arquivo seguido de dois pontos e a linha encontrada, caso passado caractere coringa ou mais de um arquivo.

■ Sintaxe: `$grep <opções> <expressão regular> <arquivos>`

■ Opções:

- c: Conta a quantidade de ocorrências do padrão nos arquivos
- n: Mostra além do texto procurado, o número da linha dentro do arquivo.



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **egrep e fgrep**

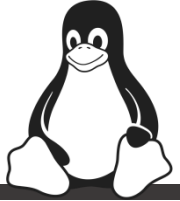
- > Possuem a mesma funcionalidade do grep, porém aceita um número maior de metacaracteres e expressões regulares.

- Sintaxe: `$egrep <opções> <expressão regular> <arquivos>`

- Exemplo:

- > Utilizando o egrep para remover comentários num arquivo.

- `# egrep -v ^#|^$ squid.conf.bkp > squid.conf`



COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **grep**

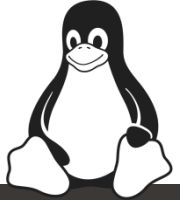
- Exemplos:

- > Procura pela linha que inicia com a palavra root no arquivo /etc/passwd

```
root@kali:/etc# grep ^root /etc/passwd  
root:x:0:0:root:/root:/bin/bash
```

- > Procura pelo termo nologin no arquivo /etc/passwd

```
root@debian:~# grep nologin /etc/passwd  
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

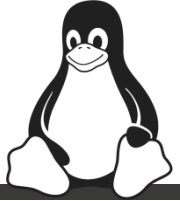
COMANDOS PARA LOCALIZAÇÃO DE ARQUIVOS

- **grep**

- Exemplos:

- > Procura pelos registros gerados no dia 15 de maio (May 15) no arquivo /var/log/syslog

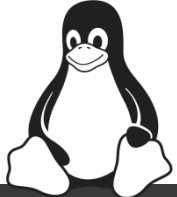
```
root@debian:/var/log# grep "^May 15" messages |more
May 15 21:03:56 debian rsyslogd: [origin software="rsyslogd" swVersion="8.4.2" x
-pid="466" x-info="http://www.rsyslog.com"] start
May 15 21:03:56 debian kernel: [    0.000000] Initializing cgroup subsys cpuset
May 15 21:03:56 debian kernel: [    0.000000] Initializing cgroup subsys cpu
May 15 21:03:56 debian kernel: [    0.000000] Initializing cgroup subsys cpuacct
May 15 21:03:56 debian kernel: [    0.000000] Linux version 3.16.0-4-686-pae (de
```



COMPACTADOR DE ARQUIVOS

■ gzip e gunzip – bzip2 e bunzip2

- > Principais comandos compactadores e descompactar arquivos para o ambiente Linux.
 - > O algoritmo de compactação do gzip é mais veloz, enquanto o do bzip chega a maior taxa de compressão.
 - > Compactam arquivos empacotados com o tar.
 - > Arquivos são gerados com a extensões .gz e .bz2 , respectivamente
- Sintaxe para compactar: `#gzip <arquivo>` e `bzip2 <arquivo>`
 - Sintaxe para descompactar: `#gunzip <arquivo.gz>` e `bunzip2 <arquivo.bz2>`



COMPACTADOR DE ARQUIVOS

■ gzip e gunzip

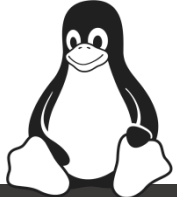
■ Exemplo:

> Compactando um arquivo

- Antes `-rw-r--r-- 1 bilhetagem bilhetagem 1,9M Dez 7 08:08 images.sql`
- Depois `root@agualimpa:/usr/local/zabbix-3.2.2/database/mysql# gzip images.sql`
`-rw-r--r-- 1 bilhetagem bilhetagem 1,1M Dez 7 08:08 images.sql.gz`

> Descompactando um arquivo

```
root@agualimpa:/usr/local/zabbix-3.2.2/database/mysql# gunzip images.sql.gz
```



COMPACTADOR DE ARQUIVOS

■ bzip2 e bunzip2

■ Exemplo:

> Compactando um arquivo

- Antes

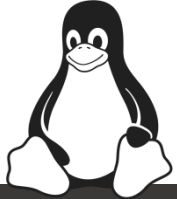
```
-rw-r--r-- 1 bilhetagem bilhetagem 1,9M Dez 7 08:08 images.sql  
root@agualimpa:/usr/local/zabbix-3.2.2/database/mysql# bzip2 images.sql
```

- Depois

```
-rw-r--r-- 1 bilhetagem bilhetagem 957K Dez 7 08:08 images.sql.bz2
```

> Descompactando um arquivo

```
root@agualimpa:/usr/local/zabbix-3.2.2/database/mysql# bunzip2 images.sql.bz2
```



EMPACOTAMENTO DE ARQUIVOS

- **tar**

- > Aglutina arquivos em outros arquivos ou fitas e podem extrair e atualizar estes arquivos

- Sintaxe: `#tar <opções> arquivo.tar <arquivos>`

- Opções:

- c: Criar um arquivo;

- v: Mostrar cada arquivo que é incluído.

- f: Especifica em seguida o caminho para o arquivo a ser criado.

- x: Extrai os arquivos empacotados

- j: Empacota ou Desempacota arquivos com o algoritmo bzip2.

- z: Empacota ou Desempacota arquivo com o algoritmo gzip.



EMPACOTAMENTO DE ARQUIVOS

- **tar**

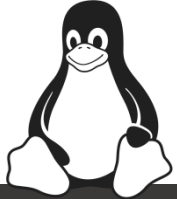
- **Exemplo:**

- > Empacotando dois arquivos (zabbix-server e zabbix-agent) em um único arquivo (zabbix.tar)

```
root@agualimpa:/var# tar cvf zabbix.tar zabbix-server zabbix-agent
zabbix-server
zabbix-agent
```

- > Extrai os arquivos empacotados dentro de zabbix.tar para o diretório corrente.

```
root@agualimpa:/var# tar xvf zabbix.tar
zabbix-server
zabbix-agent
```



EMPACOTAMENTO DE ARQUIVOS

■ tar.gz e tar.bz2

■ Exemplos:

- > Empacotando de arquivos com os algoritmos gzip e bzip2.

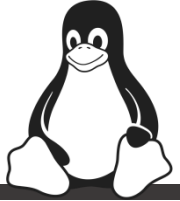
```
root@agualimpa:/usr/local/zabbix-3.2.2# tar zcvf database.tar.gz database/  
root@agualimpa:/usr/local/zabbix-3.2.2# tar jcvf database.tar.bz2 database
```

- > RESULTADO:

```
4,0M Mai 18 12:31 database.tar.bz2  
4,5M Mai 18 12:44 database.tar.gz
```

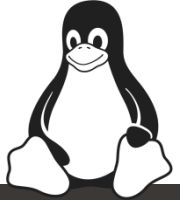
- > Desempacotando arquivos com o algoritmo gzip e bzip2.

```
root@agualimpa:/usr/local/zabbix-3.2.2# tar zxvf database.tar.gz  
root@agualimpa:/usr/local/zabbix-3.2.2# tar jxvf database.tar.bz2
```



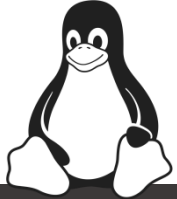
CONECTIVIDADE TCP/IP

- Fundamental para o funcionamento da rede é que a interface de rede esteja configurada.
 - > As interfaces podem ser configuradas automaticamente ou manualmente.
- No processo de configuração da conectividade com a rede, alguns comandos e arquivos são de extrema importância
 - > `#ifconfig`
 - > `#route`
 - > `/etc/network/interfaces`
 - > `/etc/resolv.conf`



CONECTIVIDADE TCP/IP

- No Linux há várias formas de identificação das interfaces de rede.
 - > eth<n>: Nome geralmente dado a interface de rede cabeada
 - > eth<n>:<n>: Nomeclatura utilizada para identificar interfaces virtuais, por exemplo: eth1:1.
 - > wlan<n>: Nome geralmente dado a interface de rede WiFi
 - > ppp<n>: Nome geralmente dado a interface que utiliza o protocolo PPP para acesso a internet
 - > lo: Nomeclatura que identifica a interface de loopback (acesso local)



CONECTIVIDADE TCP/IP

■ ifconfig

> Visualiza e configura manualmente a interface de rede.

■ Sintaxe: `#ifconfig <interface> <parâmetros>`

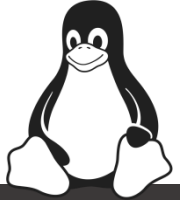
■ Parâmetros:

netmask: Informa a máscara da subrede em que a interface esta.

gw: Informa o gateway padrão utilizado pela interface.

down: Desativa a interface de rede

up: Ativa a interface de rede

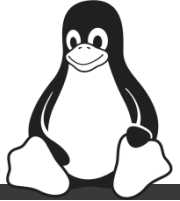


CONECTIVIDADE TCP/IP

■ ifconfig

■ Exemplo:

```
root@agualimpa:/# ifconfig
eth0      Link encap:Ethernet  Endereço de HW 00:50:56:b1:7c:9e
          inet end.: 172.16.1.139  Bcast:172.16.1.255  Masc:255.255.255.0
          endereço inet6: fe80::250:56ff:feb1:7c9e/64  Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:221778336 errors:0 dropped:53462 overruns:0 frame:0
          TX packets:161027405 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:1000
          RX bytes:27167087523 (25.3 GiB)  TX bytes:16831945851 (15.6 GiB)
```



CONECTIVIDADE TCP/IP

■ ifconfig

■ Exemplos:

> À interface eth0 é ativada e atribuído o endereço IP 192.168.1.2.

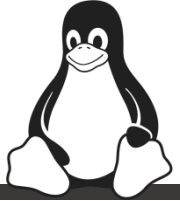
```
#ifconfig eth0 192.168.1.2 up
```

> Desativando a interface eth1

```
#ifconfig eth1 down
```

> Criando a interface virtual eth0:1 + atribuindo o endereço 192.168.1.5, com máscara de 24 bits e utilizando o gateway 192.168.1.1.

```
#ifconfig eth0:1 192.168.1.5 netmask 255.255.255.0 gw 192.168.1.1
```



CONECTIVIDADE TCP/IP

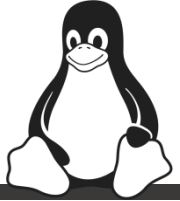
■ route

- > Manipula a tabela de roteamento IP do kernel
- > Seu principal uso é configurar rotas estáticas para hosts ou redes especificadas através de uma interface

■ Sintaxe: `#route <opções> <rede> netmask <mascara> gw <gateway> dev <interface>`

■ Opções:

- > -n: Mostra endereços numéricos, sem tentar resolver o nomes
- > -net: Utilizar quando o alvo é o endereço de uma rede
- > -host: Utilizar quando o alvo é o endereço de máquina
- > add: Adiciona uma rota
- > del: Apaga uma rota



CONECTIVIDADE TCP/IP

■ route

■ Exemplos:

> Adiciona uma rota para a rede 192.56.76.x através da interface eth0

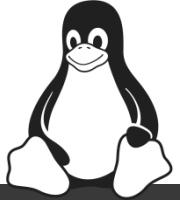
```
#route add -net 192.56.76.0 netmask 255.255.255.0 dev eth0
```

> Adiciona uma rota default (que será usada se nenhuma outra rota for encontrada)

```
#route add default gw 192.168.0.1
```

> Apaga a rota padrão

```
#route del default
```



CONECTIVIDADE TCP/IP

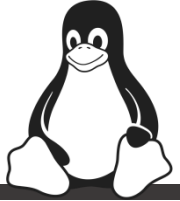
■ /etc/network/interfaces

> Arquivo de configuração das interfaces de rede em distribuições Debian like.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
allow-hotplug eth0
#iface eth0 inet dhcp
iface eth0 inet static
    address 172.16.1.139
    netmask 255.255.255.0
    gateway 172.16.1.1
    dns-nameservers 172.16.1.71 172.16.1.63
    dns-search asinf.gna.saneago
```

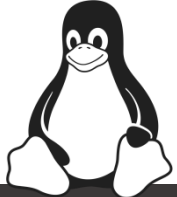


CONECTIVIDADE TCP/IP

- **/etc/resolv.conf**

- > Arquivo para resolução de nomes DNS.

```
domain asinf.gna.saneago  
search asinf.gna.saneago  
nameserver 172.16.1.71
```

COMANDOS DE VISUALIZAÇÃO

■ du

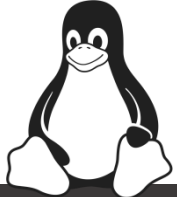
- > Utilizado para identificar o espaço utilizado em disco, por pastas ou arquivos.
- > Sem nenhum argumento, du relata o espaço de disco usado pelo diretório atual

■ Sintaxe: \$du <opções> arquivo

■ Opções:

- a: exibe informações de arquivos e diretórios
- h: exibe informações em modo humano (KB, MB e GB)
- s: exibe informações apenas do diretório informado.

```
root@agualimpa:/home# du -sh /home/vitorluiz/  
18M      /home/vitorluiz/
```



COMANDOS DE VISUALIZAÇÃO

■ df

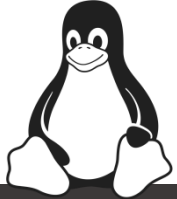
- > Informa o espaço de disco usado pelo sistema de arquivo
- > Sem argumentos, informa o espaço usado e disponível em todos sistemas de arquivos

■ Sintaxe: \$df <opções> arquivo

■ Opções:

- P: exibe informações com 6 colunas de informações diversas
- a: exibe todas informações de todos os sistemas de arquivos
- h: exibe informações em modo humano (KB, MB e GB).

```
root@agualimpa:/home# df -h
Sist. Arq.                               Tam  Usad Dispon.  Uso%
Montado em
rootfs                                   14G   7,4G    5,7G    57%
/
```



COMANDOS DE VISUALIZAÇÃO

■ free

- > Informa o quanto de memória RAM está livre ou em uso no sistema
- > Apresenta informações sobre a memória física e swap.

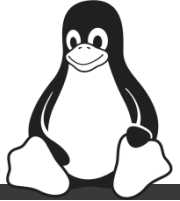
■ Sintaxe: `$free <opções>`

■ Opções:

- m: exibe informações em Megabytes (MB)
- h: exibe informações em modo humano (KB, MB e GB).

```
m135143@agualimpa:~$ free -m
```

| | total | used | free | shared | buffers | cached |
|--------------------|-------|------|------|--------|---------|--------|
| Mem: | 3968 | 965 | 3003 | 0 | 294 | 265 |
| -/+ buffers/cache: | | 405 | 3563 | | | |
| Swap: | 4766 | 0 | 4766 | | | |



COMANDOS DE VISUALIZAÇÃO

■ **uname**

> Exibe informações sobre o sistema

■ **Sintaxe: \$uname <opções>**

■ **Opções:**

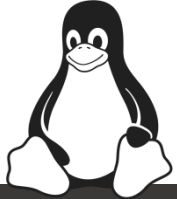
-a: exibe todas as informações sobre o sistema instalado

-s: exibe informações do Kernel

-m: arquitetura do sistema x86 ou x64

-o: informações do sistema operacional

```
root@agualimpa:/home/vitorluiz# uname -o  
GNU/Linux
```



COMANDOS DE VISUALIZAÇÃO

■ sort

> Organiza linhas dentro do arquivo de texto.

■ Sintaxe: `$sort <opções> arquivo`

■ Opções:

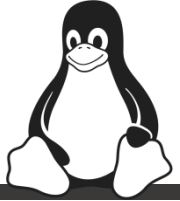
-h: modo humano

-M: organiza por entradas de mês dentro do arquivo (JAN – FEV – MAR)

-r: classifica os dados decrescente

-n: classifica dados numéricos

```
root@agualimpa:/home/vitorluiz# sort /etc/passwd
backup:x:34:34:backup:/var/backups:/bin/sh
bilhetagem:x:1000:1000:bilhetagem,,,:/home/bilhetagem:/bin/bash
bin:x:2:2:bin:/bin:/bin/sh
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
```



COMANDOS DE VISUALIZAÇÃO

■ **cut**

> Usado para mostrar apenas seções específicas de um arquivo de texto ou da saída de outros comandos.

■ Sintaxe: `$cut <opções> arquivo`

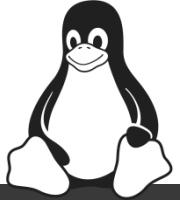
■ Opções:

-f <campo>: mostra a lista de campos, mas necessita do parâmetro -d.

-d <delimitador>: determina o caractere delimitador, na omissão do campo o valor delimitador é o TAB.

-b <bytes>: mostra a lista de bytes do arquivo.

-c <núm. Caracteres> :: Mostra o número de caracteres do arquivo



COMANDOS DE VISUALIZAÇÃO

■ awk

- > Permite a manipulação de textos a partir de uma sequência de padrões.
- > É uma linguagem interpretada (não há necessidade de compilação)

■ Exemplos:

- > Apresenta as duas primeiras colunas do resultado do comando `df -h`.
`$ df -h | awk '{print $1 " " $2}'`
- > Apresenta apenas o conteúdo da segunda linha do comando `free -mh`
`$ free -mh | awk 'NR==2'`

```
root@agualimpa:/home/vitorluiz# free -mh | awk 'NR==2'
Mem:          3,9G          966M          2,9G          0B          294M          266M
```