



Título do Curso:

Administração de Sistemas
GNU/Linux



Conteúdo do curso

ADMINISTRAÇÃO DE SISTEMAS GNU/LINUX

- ✓ Tópico 1: Introdução ao sistema operacional GNU/Linux.
- ✓ Tópico 2: Introdução ao Shell e comandos básicos.
- ✓ Tópico 3: Manipulação de conteúdos com comandos no Shell.
- **Tópico 4: Comandos para gerenciamento do sistema e do Hardware.**
- Tópico 5: Editor de Texto VI.
- Tópico 6: Administração de usuários e grupos.
- Tópico 7: Gerenciamento de permissões.
- Tópico 8: Gerenciamento de processos.
- Tópico 9: Sistemas de arquivos e particionamento.
- Tópico 10: Expressões regulares.
- Tópico 11: Introdução ao Shell Script.
- Tópico 12: Gerenciamento de Pacotes.
- Tópico 13: Agendamento de tarefas (cron) e Backup.





DGP

Tecnologia da Informação

Tópico 4

Comandos para gerenciamento do
Sistema e do Hardware

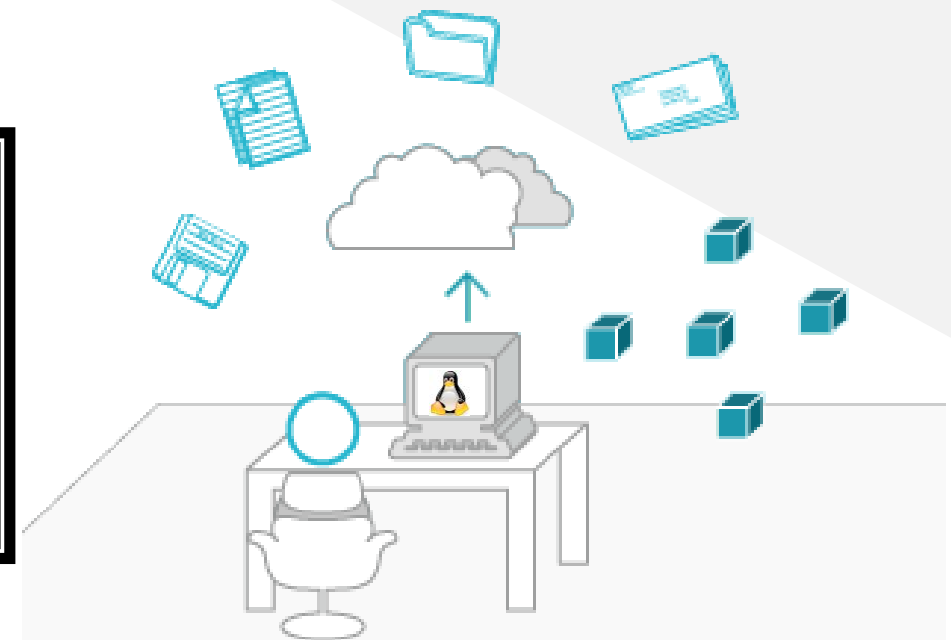


Comandos para gerenciamento

do Sistema e do Hardware

- Neste slide teremos comandos que nos auxiliam na administração do sistema e dispositivos de Hardware.

```
Initializing USB Mass Storage driver...  
scsi1 : SCSI emulation for USB Mass Storage devices  
usbcore: registered new driver usb-storage  
USB Mass Storage support registered.  
usb-storage: device found at 3  
usb-storage: waiting for device to settle before scanning  
Vendor: Kingston Model: DataTraveler 2.0 Rev: PMAP  
Type: Direct-Access ANSI SCSI revision: 00  
usb-storage: device scan complete  
SCSI device sda: 7823360 512-byte hdwr sectors (4006 MB)  
sda: Write Protect is off
```



Comandos para coletar informações do sistema

Tópico 4: Comandos para
gerenciamento do sistema e
do Hardware.



Informações do sistema

Comando “uname”

- `uname` → Exibe informações sobre o sistema instalado, incluindo a versão do Kernel:
 - Ex.: `uname -a`
 - O parâmetro “-a” (all) exibe todas as informações disponíveis.
 - Entretanto, o “uname” não exibe qual distribuição e sua versão (na maioria dos casos). Para identificar a distribuição, por padrão há um arquivo dentro de “/etc”. Seguem alguns exemplos:

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 4.18.0-147.el8.x86_64 #1 SMP Wed Dec 4 21:51:45 UTC 2019 x86_64 x86_64 x
86_64 GNU/Linux
[root@localhost ~]#
[root@localhost ~]# cat /etc/redhat-release
CentOS Linux release 8.1.1911 (Core)
[root@localhost ~]#
```

```
root@debian:~# uname -a
Linux debian 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1 (2020-01-26) x86_64 GNU/Linux
root@debian:~#
root@debian:~# cat /etc/debian_version
10.3
root@debian:~# _
```



Informações do sistema

Comando “uptime”

- uptime → Exibe um resumo de informações sobre o sistema como:
 - Hora atual;
 - Tempo que o sistema está em execução (“up”, “no ar”);
 - Quantidade de usuários logados;
 - “Load Average”, que mostra quantos processos em média estão aguardando (na fila) para serem executados, sendo que as separações por “vírgula” representam os intervalos de tempo de 1, 5 e 15 minutos.

```
[root@mail ~]# uptime
08:46:31 up 1 min,  1 user,  load average: 0.60, 0.18, 0.06
```

- O exemplo a seguir, mostra um servidor que está ligado a 198 dias, 22 horas e 27 minutos:

```
root@bridge:~# uptime
17:03:23 up 198 days, 22:27,  1 user,  load average: 0.00, 0.00, 0.00
```



Informações de memória

Comando “free”

- free → Exibe informações sobre a utilização da memória RAM e SWAP.
 - Ex.: `free -m`
 - O parâmetro “-m” exibe a utilização em MB, da mesma forma que “-g” ou “-k” podem ser utilizados para exibição em GB e KB respectivamente.

```
[root@mail ~]# free -m
```

	total	used	free	shared	buffers	cached
Mem:	503	356	146	0	28	279
-/+ buffers/cache:		47	455			
Swap:	996	0	996			

- OBS.: A coluna “shared” não deve ser considerada.

```
shared memory column should be ignored; it is obsolete.
```

- Podemos executar o “`free -s 10`”, para atualizar o status do consumo de memória a cada 10 segundos.
- Para obter mais informações sobre o uso da memória, podemos acessar o arquivo “`/proc/meminfo`”.
Ex.: `cat /proc/meminfo`



Informações sobre discos

Comandos “df” e “du”

- df → Exibe informações sobre o espaço livre/utilizado em disco:
 - Ex.: `df -h`
 - O parâmetro “-h” exibe de forma “inteligível” (*humam readable*).

```
root@darkstar:~# df
Filesystem            1K-blocks      Used Available Use% Mounted on
/dev/hda2              16276168    3350884   12925284   21% /
```

```
root@darkstar:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda2       16G   3.2G   13G   21% /
```

- du → Exibe o tamanho ocupado em disco de arquivos ou diretórios:
 - Ex.: `du -hs /etc`
 - O parâmetro “-s” exibe o tamanho total ocupado pelo diretório “/etc”;
 - O parâmetro “-h” exibe de forma “inteligível” (*humam readable*).

```
root@darkstar:~# du -hs /lib
100M    /lib
```

```
root@darkstar:~# du -hs /usr/share/doc/Linux-HOWTOs/
32M     /usr/share/doc/Linux-HOWTOs/
```



Informações sobre arquivos

Comando “file”

- file → Exibe o tipo de um determinado arquivo (se o mesmo é texto, imagem, arquivo compactado, entre outros).
 - Ex.: `file <nome_do_arquivo>`
 - O exemplo a seguir nos mostra o comando “file” verificando um pacote RPM:

```
[root@mail CentOS]# file httpd-2.2.3-31.el5.centos.i386.rpm
httpd-2.2.3-31.el5.centos.i386.rpm: RPM v3 bin i386 httpd-2.2.3-31.el5.centos
```

- Comando “file” verificando um arquivo compactado com o compressor GZIP:

```
[root@mail CentOS]# file /root/backup.tar.gz
/root/backup.tar.gz: gzip compressed data, from Unix, last modified: Wed Mar 21
09:47:46 2012
```

- Comando “file” verificando um arquivo de formato “texto puro”:

```
[root@mail CentOS]# file /etc/passwd
/etc/passwd: ASCII text
```



Informações do sistema

Acessos (Logons)

- `w` → Exibe a saída do comando “uptime” e informações sobre os usuários conectados, como tempo ocioso e processo que este usuário está executando.

- Ex.:

```
[root@mail ~]# w
10:01:45 up 3:07, 2 users, load average: 0,01, 0,01, 0,00
USER      TTY      FROM          LOGIN@      IDLE        JCPU        PCPU       WHAT
root      tty1     -             06:56       0.00s       0.35s       0.00s      w
root      tty2     -             07:07       2:47m       0.02s       0.01s      vi /var/log/mes
```

- `who` → Exibe quais usuários estão logados no sistema, qual o “terminal” este usuário está conectado, data e hora do Logon, e por fim, o IP de origem desta conexão (caso seja uma conexão remota).

- Ex.:

```
[root@mail ~]# who
root      tty1      2012-03-21 06:56
root      tty2      2012-03-21 07:07

root@bridge:~# who
root      pts/0      Nov 23 16:59 (201.58.123.25)
```

- `whoami` → Exibe qual o nome do usuário logado no terminal atual:

- Ex.:

```
[root@mail ~]# whoami
root
```

Configurações de rede

Conceitos



- No GNU/Linux, as configurações de rede podem ser realizadas de 3 formas:
 - Configurar IP estático/fixo, nos arquivos de configurações de rede.
 - Esta opção é mais comum em Servidores, em que o IP fixo é fundamental para o pleno funcionamento.
 - Definir um IP manualmente, que permanecerá até o próximo reboot.
 - Esta opção é muito usada para realizar testes e/ou manutenções no ambiente (por ser temporário).
 - Solicitar um IP através do cliente DHCP (caso tenha um servidor DHCP na rede).
 - Geralmente a mais utilizada em Desktops (assim como em sistemas Windows), devido a flexibilidade e mobilidade.
- Para solicitar um IP via DHCP, temos os comandos “**ifdown**” + “**ifup**” ou o “**dhclient**”:
 - Ex.: **ifdown** <interface> (para **desabilitar**) + **ifup** <interface> (para **habilitar**).
 - Ex.: **dhclient** ou **dhclient** <interface>
 - Para identificar as suas interfaces de rede, execute “**ifconfig**” ou “**ip addr**” conforme slides a seguir.



Informações de rede

Comando “ifconfig”

- ifconfig → Permite verificar o IP atual ou configurar um IP para um determinado adaptador de rede:
 - Ex.: `ifconfig <interface>`
 - O comando acima apenas verifica o IP de uma determinada interface;

```
[root@localhost ~]# ifconfig
eth0      Link encap:Ethernet  Endereço de HW 08:00:27:38:70:95
          inet end.: 10.0.2.15  Bcast:10.0.2.255  Masc:255.255.255.0
          endereço inet6: fe80::a00:27ff:fe38:7095/64  Escopo:Link
```

- Ex.: `ifconfig <interface> [X.X.X.X] netmask [Y.Y.Y.Y]`
 - O comando acima atribui um IP a uma determinada interface;

```
debian:~# ifconfig eth3 172.16.0.10 netmask 255.255.255.0
```

```
debian:~# ifconfig eth3
eth3      Encapsulamento do Link: Ethernet  Endereço de HW 08:00:27:42:42:02
          inet end.: 172.16.0.10  Bcast:172.16.0.255  Masc:255.255.255.0
```



Informações de rede

Comando “route”

- route → Permite visualizar ou modificar rotas ou o “Default Gateway”:
 - Ex.: `route add default gw [X.X.X.X]`
 - Define o “Default Gateway”;

```
root@darkstar:~# route add default gw 192.168.1.1
root@darkstar:~#
```

- Ex.: `route -n`
 - Apenas exibe as rotas existentes;

```
root@darkstar:~# route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
192.168.1.0      0.0.0.0         255.255.255.0   U        0      0        0 eth0
127.0.0.0        0.0.0.0         255.0.0.0       U        0      0        0 lo
0.0.0.0          192.168.1.1    0.0.0.0         UG       0      0        0 eth0
root@darkstar:~# _
```

- OBS.: As configurações realizadas através dos comandos “`ifconfig`” e “`route`” são voláteis, ou seja, serão aplicados até o computador ser reiniciado.



Informações de rede

Comando “ip”

- O comando “ip” possui diversas opções (chamadas de objetos), que permite ver e alterar configurações de rede, roteamento e tunelamento.
- Apenas para se ter uma ideia da quantidade de opções, segue um print do manual do comando “ip” (resultado do “man ip”) com os objetos e opções disponíveis:

```
IP(8)                                     Linux                                     IP(8)

NAME
    ip - show / manipulate routing, network devices, interfaces and tunnels

SYNOPSIS
    ip [ OPTIONS ] OBJECT { COMMAND | help }

    ip [ -force ] -batch filename

OBJECT := { link | address | addrlabel | route | rule | neigh | ntable | tunnel | tuntap |
            maddress | mroute | mrule | monitor | xfrm | netns | l2tp | tcp_metrics | token |
            macsec }

OPTIONS := { -U[ersion] | -h[uman-readable] | -s[tatistics] | -d[etails] | -r[esolve] |
            -iec | -f[amily] { inet | inet6 | ipx | dnet | link } | -4 | -6 | -I | -D | -B |
            -0 | -l[oops] { maximum-addr-flush-attempts } | -o[neline] | -rc[vbuf] [size] |
            -t[imestamp] | -ts[hort] | -n[etns] name | -a[ll] | -c[olor] | -br[ief] | -j[son]
            | -p[retty] }
```

OBS.: Este comando não possui a opção “--help”.

As opções são:
[ip address help](#)
[man ip](#)



Informações de rede

Comando “ip address”

- A principal utilidade do comando é definir um endereço IP. Segue exemplo:
 - Listando as interfaces. Ex.: `ip address` ou `ip addr list` ou `ip addr`

```
[root@localhost ~]# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:f8:ef:33 brd ff:ff:ff:ff:ff:ff
```

- Definindo endereço IP na interface “enp0s3”. Ex.: `ip address add x.x.x.x/mask dev <interface>`

```
[root@localhost ~]# ip address add 10.0.0.23/24 dev enp0s3
```

- Listando somente a interface “enp0s3” e seu IP. Ex.: `ip address list enp0s3` ou `ip addr list enp0s3`

```
[root@localhost ~]# ip addr list enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:f8:ef:33 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.23/24 scope global enp0s3
        valid_lft forever preferred_lft forever
```




Informações de rede

Comando “ip address”

- A remoção de um endereço IP possui sintaxe similar, tendo duas possibilidades:
 - Remover todos os endereços de uma interface (caso tenha mais de um IP):
 - Ex.: `ip address flush dev enp0s3`
 - Remover apenas um endereço da interface “enp0s3”:
 - Ex.: `ip address del x.x.x.x/mask dev <interface>`



Informações de rede

Comando “ip route”

- route → Permite visualizar ou modificar rotas ou o “Default Gateway”:
 - Ex.: `route add default via [X.X.X.X] dev <interface>`
 - Define o “Default Gateway”;

```
[root@localhost ~]# ip route add default via 10.0.0.1 dev enp0s3
```

- Ex.: `ip route`
 - Apenas exibe as rotas existentes;

```
[root@localhost ~]# ip route
default via 10.0.0.1 dev enp0s3
10.0.0.0/24 dev enp0s3 proto kernel scope link src 10.0.0.23
```

- OBS.: As configurações realizadas através dos comandos “`ip address`” e “`ip route`” são voláteis, ou seja, serão aplicados até o computador ser reiniciado.

Comandos para coletar e definir configurações de hardware

Tópico 4: Comandos para gerenciamento do sistema e do Hardware.



Informações sobre Hardware

Comando “dmesg”

- dmesg → exibe todo o Hardware reconhecido/carregado pelo kernel durante a inicialização. Ex.:

```
[root@mail ~]# dmesg
Linux version 2.6.18-164.el5 (mockbuild@builder16.centos.org) (gcc version 4.1.2
20080704 (Red Hat 4.1.2-46)) #1 SMP Thu Sep 3 03:33:56 EDT 2009
BIOS-provided physical RAM map:
 BIOS-e820: 00000000000010000 - 0000000000009fc00 (usable)
 BIOS-e820: 0000000000009fc00 - 000000000000a0000 (reserved)
 BIOS-e820: 000000000000f0000 - 00000000000100000 (reserved)
 BIOS-e820: 00000000000100000 - 0000000001ffff0000 (usable)
 BIOS-e820: 0000000001ffff0000 - 00000000020000000 (ACPI data)
 BIOS-e820: 00000000ffffc0000 - 00000000100000000 (reserved)
0MB HIGHMEM available.
511MB LOWMEM available.
Memory for crash kernel (0x0 to 0x0) notwithin permissible range
disabling kdump
Using x86 segment limits to approximate NX protection
On node 0 totalpages: 131056
  DMA zone: 4096 pages, LIFO batch:0
  Normal zone: 126960 pages, LIFO batch:31
DMI 2.5 present.
Using APIC driver default
```

- Comando muito útil, pois possibilita uma análise passo a passo da identificação do Hardware durante a inicialização do kernel, auxiliando a resolução de problemas (*troubleshooting*).
- Nas distribuições antigas, as mensagens eram armazenadas no arquivo “/var/log/dmesg”.



Informações sobre Hardware

Listando dispositivos

- lspci → exibe informações do chipset e dispositivos PCI:
 - Ex.: `lspci`

```
root@darkstar:~# lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:02.0 VGA compatible controller: InnoTek Systemberatung GmbH VirtualBox Graphics Adapter
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
00:04.0 System peripheral: InnoTek Systemberatung GmbH VirtualBox Guest Service
00:05.0 Multimedia audio controller: Intel Corporation 82801AA AC'97 Audio Controller (rev 01)
00:06.0 USB Controller: Apple Computer Inc. KeyLargo/Intrepid USB
00:07.0 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:0d.0 SATA controller: Intel Corporation 82801HBM/HEM (ICH8M/ICH8M-E) SATA AHCI Controller (
```

- lsusb → exibe informações sobre dispositivos USB conectados:
 - Ex.: `lsusb`

```
debian:~# lsusb
Bus 001 Device 003: ID 13fe:1f00
Bus 001 Device 002: ID 80ee:0021
Bus 001 Device 001: ID 0000:0000
```



Informações sobre Hardware

Exemplo prático

- **Exemplo prático:** Ao inserir um Pen Drive, o Hardware será identificado automaticamente, porém, para obter mais detalhes, podemos verificar:

1 – Quais os dispositivos USB já conectados antes de inserir o Pen Drive:

```
root@darkstar:~# lsusb  
Bus 1 Device 2: ID 80ee:0021  
Bus 1 Device 1: ID 0000:0000
```

2 – Após conectar, qual número foi atribuído para identificar o dispositivo:

```
root@darkstar:~# lsusb  
Bus 1 Device 3: ID 13fe:1f00  
Bus 1 Device 2: ID 80ee:0021  
Bus 1 Device 1: ID 0000:0000
```

3 – Verificar com o “dmesg”, o local para acessar o dispositivo (neste caso, “/dev/sda”):

```
Initializing USB Mass Storage driver...  
scsi1 : SCSI emulation for USB Mass Storage devices  
usbcore: registered new driver usb-storage  
USB Mass Storage support registered.  
usb-storage: device found at 3  
usb-storage: waiting for device to settle before scanning  
Vendor: Kingston Model: DataTraveler 2.0 Rev: PMAP  
Type: Direct-Access ANSI SCSI revision: 00  
usb-storage: device scan complete  
SCSI device sda: 7823360 512-byte hdwr sectors (4006 MB)  
sda: Write Protect is off
```

Informações sobre Hardware

Módulos/Drivers



- Para instalar/desinstalar dispositivos que o sistema não reconheça automaticamente, devemos realizar o download do módulo (driver) do dispositivo a ser instalado e utilizar os comandos abaixo:
- `lsmod` → exibe os módulos (drivers) carregados no sistema:
 - Ex.: `lsmod`
- `insmod` → Instala/carrega um novo módulo no Kernel.
 - Ex.: `insmod [arquivo] <opções>`
- `rmmod` → Remove um módulo (devemos ter cautela na realização do mesmo, tendo em vista que ao remover um módulo, “desativamos” o hardware associado ao módulo);
 - Ex.: `rmmod <nome_do_modulo>`

Informações sobre Hardware

Diretório “/proc”



- O diretório “/proc” é um diretório virtual do sistema Linux com algumas características importantes:
 - É um diretório utilizado exclusivamente pelo *kernel* para gerenciamento do sistema e seus recursos;
 - Existe apenas enquanto o computador está ligado;
 - Possui diversos arquivos com o tamanho de 0 bytes, porém, podemos encontrar conteúdos nestes arquivos;
 - O maior arquivo deste diretório se chama “kcore”, que possui tamanho próximo ao disponível na memória RAM;
 - Não podemos gravar ou criar arquivos neste diretório.

```
root@darkstar:~# mkdir /proc/teste
mkdir: cannot create directory '/proc/teste': No such file or directory
root@darkstar:~# touch /proc/arquivo
touch: cannot touch '/proc/arquivo': No such file or directory
root@darkstar:~# ls /
bin/   dev/   home/  media/  opt/   root/  srv/   tmp/   var/
boot/  etc/   lib/   mnt/    proc/ sbin/  sys/   usr/
```




Informações sobre Hardware

Diretório “/proc”

- Através do diretório “/proc”, podemos:
 - Obter informações do processador → Ex.: `cat /proc/cpuinfo`

```
[root@localhost ~]# cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 158
model name     : Intel(R) Core(TM) i7-9700K CPU @ 3.60GHz
stepping       : 13
cpu MHz        : 3600.006
cache size     : 12288 KB
physical id    : 0
siblings       : 1
core id        : 0
cpu cores      : 1
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 22
wp             : yes
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge
```



Informações sobre Hardware

Diretório “/proc”

- Através do diretório “/proc”, podemos:
 - Listar os dispositivos IDE conectados → Ex.: `ls -l /proc/ide`

```
debian:~# ls -l /proc/ide/
total 2
-r--r--r-- 1 root root 0 2012-03-21 15:03 drivers
lrwxrwxrwx 1 root root 8 2012-03-21 15:03 hda -> ide0/hda
lrwxrwxrwx 1 root root 8 2012-03-21 15:03 hdc -> ide1/hdc
dr-xr-xr-x 3 root root 0 2012-03-21 15:03 ide0
dr-xr-xr-x 3 root root 0 2012-03-21 15:03 ide1
```

- Informações sobre dispositivos SCSI, SATA ou SAS → Ex.: `cat /proc/scsi/scsi`

```
[root@localhost ~]# cat /proc/scsi/scsi
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 00
  Vendor: ATA      Model: VBOX HARDDISK   Rev: 1.0
  Type:   Direct-Access      ANSI SCSI revision: 05
```



Informações sobre Hardware

Armazenamento

- Além das opções descritas no slide anterior, temos um comando específico para listar todos os dispositivos de armazenamento conectados em nosso computador (veremos mais detalhes sobre este comando no slide sobre particionamento):
 - Ex.: `fdisk -l`

```
debian:~# fdisk -l

Disk /dev/hda: 10.4 GB, 10485760000 bytes
255 heads, 63 sectors/track, 1274 cylinders
Units = cilindros of 16065 * 512 = 8225280 bytes

Dispositivo Boot      Start      End      Blocks    Id  System
/dev/hda1              1         79      634536    82  Linux swap / Solaris
/dev/hda2  *          80      1274    9598837+   83  Linux

Disk /dev/sda: 4005 MB, 4005560320 bytes
255 heads, 63 sectors/track, 486 cylinders
Units = cilindros of 16065 * 512 = 8225280 bytes

Dispositivo Boot      Start      End      Blocks    Id  System
/dev/sda1  *           1        487    3911648+   b  W95 FAT32
```



No próximo slide...

- Tópico 5: Editor de Texto VI.



Referências

- BONAN, Adilson Rodrigues. **LINUX – Fundamentos, Prática & Certificação LPI**. Editora: Alta Books. RJ. 2010;
- PEREIRA, Guilherme Rodrigues. **Slides para aula expositiva**. Centro Universitário UNA.
- SILVA, Gleydson Mazioli. **Guia Foca GNU/Linux**. Disponível em: <https://guiafoca.org/>



DGP

Tecnologia da Informação

Obrigado!



Guilherme Rodrigues



contato@dgpti.com.br



www.dgpti.com.br