

fedora 



 ubuntu

 Mandriva

# Curso de Formação LPIC-1

## Exame 101



# Curso Linux: formação

- › Criar, apagar, procurar e mostrar ficheiros
  - › Manipulação de directorias.
  - › Manipulação de ficheiros.
  - › Apagar e criar ficheiros.
  - › Links no sistema de ficheiros.
  - › Extensões de ficheiros e conteúdo.
  - › Procurar no sistema de ficheiros.
  - › Outros métodos de pesquisa.

# Gestão de ficheiros

## Tipos de ficheiros

- › Ficheiros regulares
- › Directorias
- › *Character device files*
- › *Block device files*
- › *Local Domain Sockets*
- › *Named Pipes (FIFOs)*
- › *Links Simbólicos*

# Gestão de ficheiros

## Tipos de ficheiros

Tipo de ficheiro	Simbolo	Criado por	Removido por
Ficheiro regular	-	Editores, cp, etc..	rm
Directoria	d	mkdir	rmdir, rm -r
<i>Character device file</i>	c	mknod	rm
<i>Block device file</i>	b	mknod	rm
<i>Local domain socket</i>	s	socket	rm
<i>Named Pipe</i>	p	mknod	rm
Link Simbólico	l	ln -s	rm

# Gestão de ficheiros

## Tipos de ficheiros: Ficheiros regulares

- › Apenas um aglomerado de bytes
- › Não é imposta qualquer estrutura no conteúdo
  - › Ficheiros de texto
  - › Executáveis
  - › Ficheiros de dados
  - › Bibliotecas partilhadas

# Gestão de ficheiros

## Tipos de ficheiros: Directorias

- › Contem referências para outros ficheiros
- › Criam-se com *mkdir* e removem-se com *rmdir*
- › O nome de um ficheiro é guardado dentro da directoria e não no ficheiro

# Gestão de ficheiros

## Tipos de ficheiros: *Character and Block devices*

- › *Device Files* permitem aos programas comunicar com os dispositivos do sistema (hardware e periféricos)
- › Módulos específicos são anexados a estes ficheiros
- › *Device drivers* encarregam-se dos detalhes de gestão do dispositivo
- › Caracterizados por dois números: *major* e *minor*
- › *Major* diz qual o ficheiro refere o driver
- › *Minor* especifica qual a unidade física
  - › */dev/lp0* – *major 6* e *minor 0*

# Gestão de ficheiros

## Tipos de ficheiros: *Local Domain Sockets*

- › Ligações entre processos
- › Permitem a comunicação entre eles
- › Locais ao computador
- › Só podem ser lidos e escritos por processos envolvidos na comunicação
- › Criados com *socket* e destruídos com *unlink* ou *rm*
  - › CUPS, X e Syslog



# Gestão de ficheiros

## Tipos de ficheiros: *Named Pipes*

- › Ligações entre processos
- › Permitem a comunicação entre eles
- › Locais ao computador
- › Conhecidos como FIFOs (*First in first out*)
- › Criados com *mknod* e removidos com *rm*

# Gestão de ficheiros

## Tipos de ficheiros: *Links Simbólicos*

- › Refere um ficheiro pelo seu nome
- › Segue o caminho referenciado no *link*
- › *Hard Link* é uma referência directa
- › *Soft Link* referência pelo nome
- › *Links simbólicos distinguem-se do ficheiro referenciado*

```
In -s Downloads/Torrents Torrents
ls -l /home/feiticeir0/Torrents
lrwxrwxrwx 1 feiticeir0 feiticeir0 18 2009-05-24 16:31 /home/feiticeir0/Torrents ->
Downloads/Torrents
```

# Gestão de ficheiros

## Determinar tipos de ficheiros

- › Ficheiros sem extensões: como descobrir o tipo?
- › Comando *file*
  - › *Stat*
  - › */usr/share/magic*
  - › *ASCII e EBCDIC*
  - › *Linguagem de programação*
- › Saída pode ser manipulada e filtrada
- › Usado também em directorias

```
file Imagens/  
Imagens/: directory
```

```
file vaoni_stock_1600.jpg  
vaoni_stock_1600.jpg: JPEG image data, JFIF standard 1.02
```

```
file acta_20090129  
acta_20090129: UTF-8 Unicode text
```

# Gestão de ficheiros

## Criar ficheiros

- › Comando *touch*
  - › Criar um ficheiro
  - › Criar um novo ficheiro de eventos (log)
  - › Actualizar um grupo de ficheiros para uma data comum

```
touch novo_ficheiro  
touch /home/feiticeir0/novo_ficheiro
```

- › Num ficheiro existente, actualiza os três tempos do ficheiro
  - › Acesso – ultima data de escrita/leitura
  - › Alteração – ultima data de alteração das permissões
  - › Modificação – Alteração do conteúdo do ficheiro

# Gestão de ficheiros

## Criar ficheiros

- › Comando dd
- › Útil para várias tarefas:
  - › Criar imagens de CD's
  - › Duplicar conteúdos de discos e/ou partições
- › if e of são as duas opções principais

```
dd if=/mnt/cdrom/images/boot.img of=/dev/fd0  
dd if=/dev/fd0 of=floppy.img
```

```
dd if=/dev/sda of=MBR.img count=1 bs=512
```

# Gestão de ficheiros

## Copiar ficheiros

- › Cria um novo ficheiro num *inode* separado
- › Replicar o ficheiro noutra localização

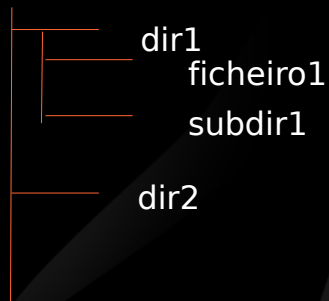
```
cp ficheiro1 ficheiro2
```

- › -d – copia os links
- › -f – não pergunta para substituir ficheiros existentes
- › -i – pergunta para substituir ficheiros existentes
- › -l – cria um *hard link* do ficheiro original
- › -r ou -R – recursivamente percorre directorias (copiando tudo)
- › -s – cria um *link simbólico* do ficheiro original
- › -u – apenas copia o ficheiro se o original for mais recente que o de destino ou este não existir
- › -x – não copia para outros sistemas de ficheiros

# Gestão de ficheiros

## Copiar ficheiros

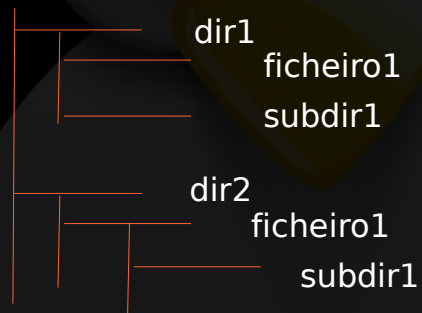
```
cp -R Imagens Imagens2  
tree
```



# Gestão de ficheiros

## Copiar ficheiros

```
cp -R Imagens/* Imagens2  
tree
```





# Gestão de ficheiros

## Mover objectos

- Apenas altera que directoria contém determinado ficheiro ou directoria
- Altera a entrada do ficheiro se este se mantém na mesma directoria

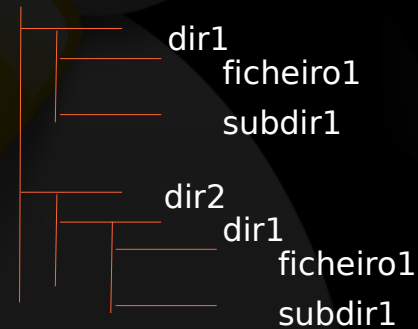
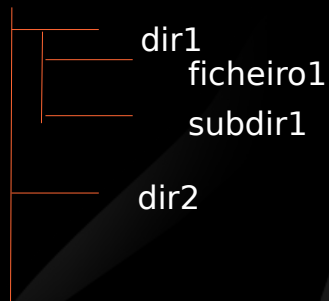
```
touch ficheiro1
File: `ficheiro1'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 806h/2054d  Inode: 34892     Links: 1
Access: (0644/-rw-r--r--) Uid: ( 1000/feiticeir0)  Gid: ( 1000/feiticeir0)
Access: 2009-05-24 17:45:43.000000000 +0100
Modify: 2009-05-24 17:45:43.000000000 +0100
Change: 2009-05-24 17:45:43.000000000 +0100
```

```
mv ficheiro1 ficheiro2
feiticeir0@nightcrawler ~ $ stat ficheiro2
File: `ficheiro2'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 806h/2054d  Inode: 34892     Links: 1
Access: (0644/-rw-r--r--) Uid: ( 1000/feiticeir0)  Gid: ( 1000/feiticeir0)
Access: 2009-05-24 17:45:43.000000000 +0100
Modify: 2009-05-24 17:45:43.000000000 +0100
Change: 2009-05-24 17:46:32.000000000 +0100
```

# Gestão de ficheiros

## Mover objectos

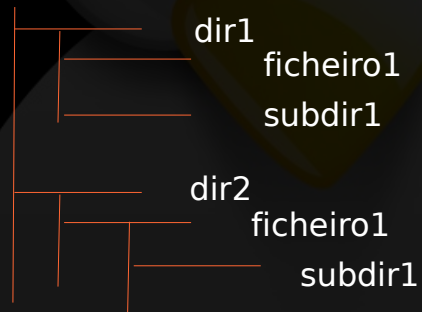
```
mv -f dir1 dir2
```



# Gestão de ficheiros

## Mover objectos

```
mv -f dir1/* dir2
```



# Gestão de ficheiros

## Criar e remover directorias

```
mkdir dir1
```

```
mkdir dir1/subdir1
```

- › Pensar sempre no ultimo segmento de qualquer caminho como o objecto a ser criado ou removido.
- › Restantes são objectos de suporte ou parentes
- › Comandos *mkdir* e *rmdir* são semelhantes

```
mkdir -p dir1/dir2/dir3/dir4
```

# Gestão de ficheiros

## Criar e remover directorias

- Rmdir – falta capacidade de remover directorias que não estejam vazias

```
rmdir -p /dir1/dir2/dir3/dir4
```

- Apenas remove dir4 se estiver vazia

# Gestão de ficheiros

## Remover objectos

- Comando *rm*

```
rm ficheiro  
rm -f ficheiro
```

```
rm -r dir1
```

# Gestão de ficheiros

Remover objectos

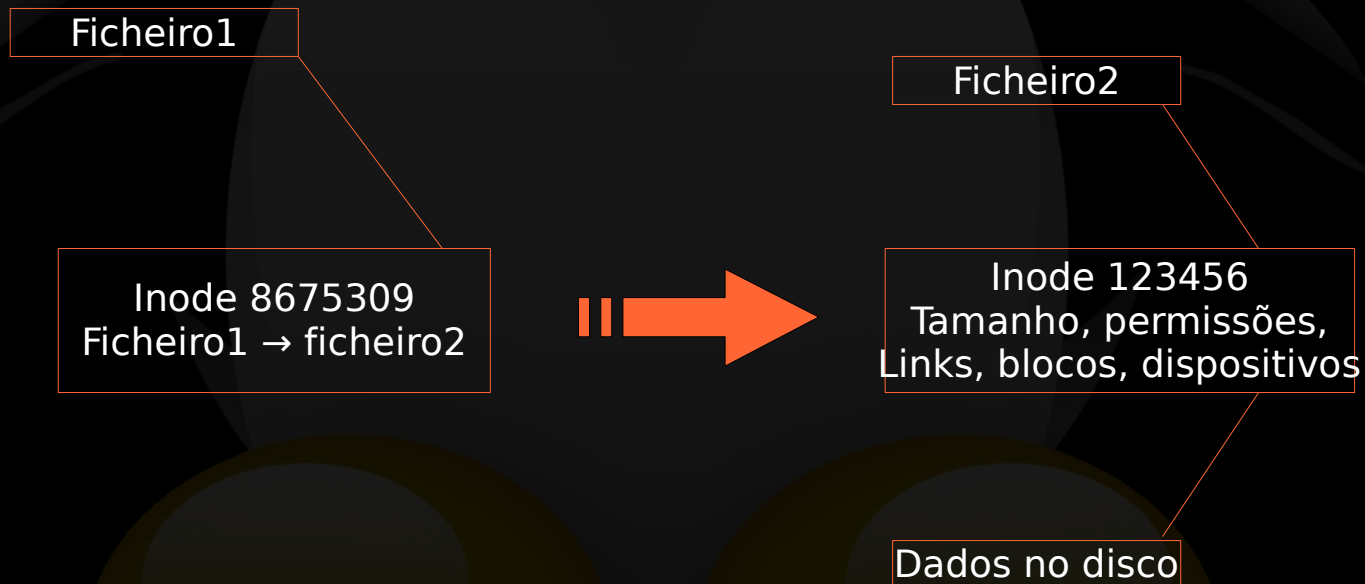


```
rm -rf /caminho
```

# Gestão de ficheiros

## Links no sistema: *Soft Links*

- › Útil ter-se acesso a um ficheiro em vários locais
- › Não se tem várias cópias, obtêm-se referências
- › Não ocupam o mesmo espaço que uma cópia de um ficheiro
- › *Soft Links*
  - › Apontador para outro ficheiro (atalho)
  - › Inode e dados próprios
  - › Pode apontar para outros sistemas de ficheiros
  - › Exista ou não – *broken* ou *staled*
  - › *ln -s*



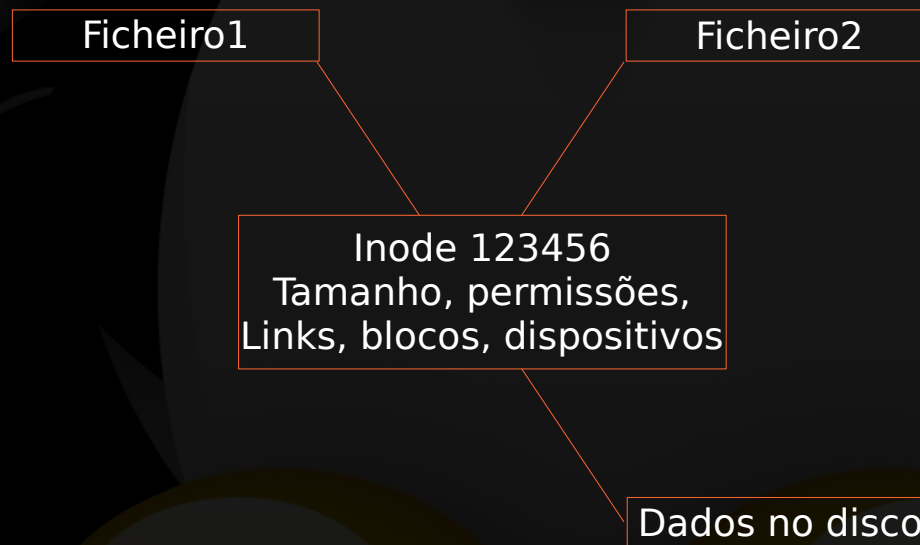


# Gestão de ficheiros

## Links no sistema: *Hard Links*

### › *Hard Links*

- › Nova entrada para um ficheiro existente
- › Nomes diferentes, dados iguais
- › Geralmente usado para criar uma cópia de segurança
- › Acesso leitura e escrita
- › Apenas ficheiros locais
- › Apaga-se apenas a entrada
- › In



# Gestão de ficheiros

## Que comando correr

- Às vezes é necessário saber que comando irá ser executado
- Conteúdo da variável *PATH*
- Comando *which*

```
which ls  
/bin/ls
```

```
which gawk  
/usr/bin/gawk
```

```
which -a gawk  
/usr/bin/gawk  
/bin/gawk
```

# Gestão de ficheiros

## Procurar por um comando

- Mais informações sobre comandos
- Três tipos de informação
  - O ficheiro binário
  - Página do manual
  - Ficheiros fonte (se existirem)

```
whereis ls
```

```
ls: /bin/ls /usr/share/man/man1/ls.1.bz2 /usr/share/man/man1p/ls.1p.bz2
```

# Gestão de ficheiros

## Procurar ficheiros

- Essencial ter-se um mecanismo de procura
- Duas formas:
  - Rápida e “suja”
    - locate
  - Lenta e metódica
    - find

# Gestão de ficheiros

## Procurar ficheiros: *Locate*

- › Forma mais rápida
- › Seguro
- › Orientado a uma base de dados
- › Jogar ao gato e ao rato com o conteúdo dos discos

```
locate readme  
/usr/lib64/openoffice/readmes  
/usr/lib64/openoffice/readmes/README_en-US
```

- › Devolve ficheiros e caminhos
- › Ideal para procurar ficheiros que se sabe existirem
- › Base de dados
  - › Actualizada manualmente
  - › Agendada no *crontab*

# Gestão de ficheiros

## Procurar ficheiros: *Locate*

- › Não ligando à capitalização

```
locate README  
/usr/share/doc/synaptic/README.Debian  
/usr/share/doc/synaptic/README.gz
```

```
locate -i readme  
/usr/lib/openoffice/readmes  
/usr/lib/openoffice/readmes/README_pt
```

- › Limite de resultados
  - › `locate -l X texto`

# Gestão de ficheiros

## Procurar ficheiros: *Locate*

- Actualização da base de dados

```
Updatedb  
slocate -u
```

- Ficheiro de configuração */etc/updatedb.conf*

```
# Filesystems that are pruned from updatedb database  
PRUNEFS="afs auto autofs cifs devfs devpts eventpollfs futexfs gfs hugetlbfs iso9660  
mqueue ncfs nfs NFS nfs4 nfsd nnpfs ocfs ocfs2  
pipefs proc ramfs rpc_pipefs selinuxfs sfs shfs smbfs sockfs spufs subfs supe  
rmount sysfs tmpfs udf usbfs vperfctrfs"  
  
# Paths which are pruned from updatedb database  
PRUNEPATHS="/tmp /var/tmp /root/.ccache"
```

# Gestão de ficheiros

## Procurar ficheiros: *find*

- › Forma mais lenta
- › Mais correcta
- › Várias (e confusas) secções
- › Comand poderoso

```
find caminho_inicial -opcoes argumentos
```

```
find / -iname *.txt
```



# Gestão de ficheiros

## Procurar ficheiros: *find*

- › Alguns operadores
  - › group – ficheiros pertencentes ao grupo especificado
  - › newer – ficheiros mais recentes que o especificado
  - › name – ficheiros com o nome especificado – importa capitalização
  - › iname – ficheiros com o nome especificado – não importa capitalização
  - › user – pertencentes ao utilizador especificado
  - › mtime – tempo de modificação do conteúdo
  - › atime – tempo do ultimo acesso de leitura/escrita
  - › ctime – tempo de alteração das permissões

# Gestão de ficheiros

## Procurar ficheiros: *find*

- › Execução de comandos nos resultados

```
find / *.txt -exec cp {} Ficheiros_texto \;
```

- › Usando o nome
  - › `find -name "Readme"`
- › Usando o nome sem ligar à capitalização
  - › `find -iname "Readme"`
- › Limitar a procura
  - › Maxdepth 2 e 3
    - › `find / -mindepth 3 -maxdepth 5 -name passwd`
- › Executar comandos
  - › `find / -iname *.mp3 -exec cp {} musicas_mp3 \;`
- › Inverter a procura
  - › `find . -not -iname *.txt`
- › Pelo inode
  - › `find -inum 149813`

# Gestão de ficheiros

## Procurar ficheiros: *find*

- › Execução de comandos nos resultados

```
find / *.txt -exec cp {} Ficheiros_texto \;
```

- › Permissões dos ficheiros
  - › `find . -perm -g=r -type f -exec ls -l {} \;`
  - › `find . -perm g=r -type f -exec ls -l {} \;`
- › Ficheiros vazios
  - › `find . -empty`
  - › `find . -empty -not -name ".*"`
- › Baseado no tipo de ficheiros
  - › `find . -type d`
  - › `find . -type f`
- › Tempo de modificação, acesso e alteração das permissões
  - › `find . -mtime|-atime|-ctime +X`
  - › `find . -newer ficheiro_referencia`
- › Por tamanho
  - › `find . -size -|+ xxx[k|M|G]`

# Curso Linux

## bibliografia

- › LPIC I, Exam Cram 2, Brunson - QUE Certification
- › LPI Linux Certification In a Nutshell, Pritchard, Pessanha, Langfeldt, Stranger & Dean – O REILLY
- › Linux Administration Handbook, Second edition, Nemeth Snyder Hein – Prentice Hall