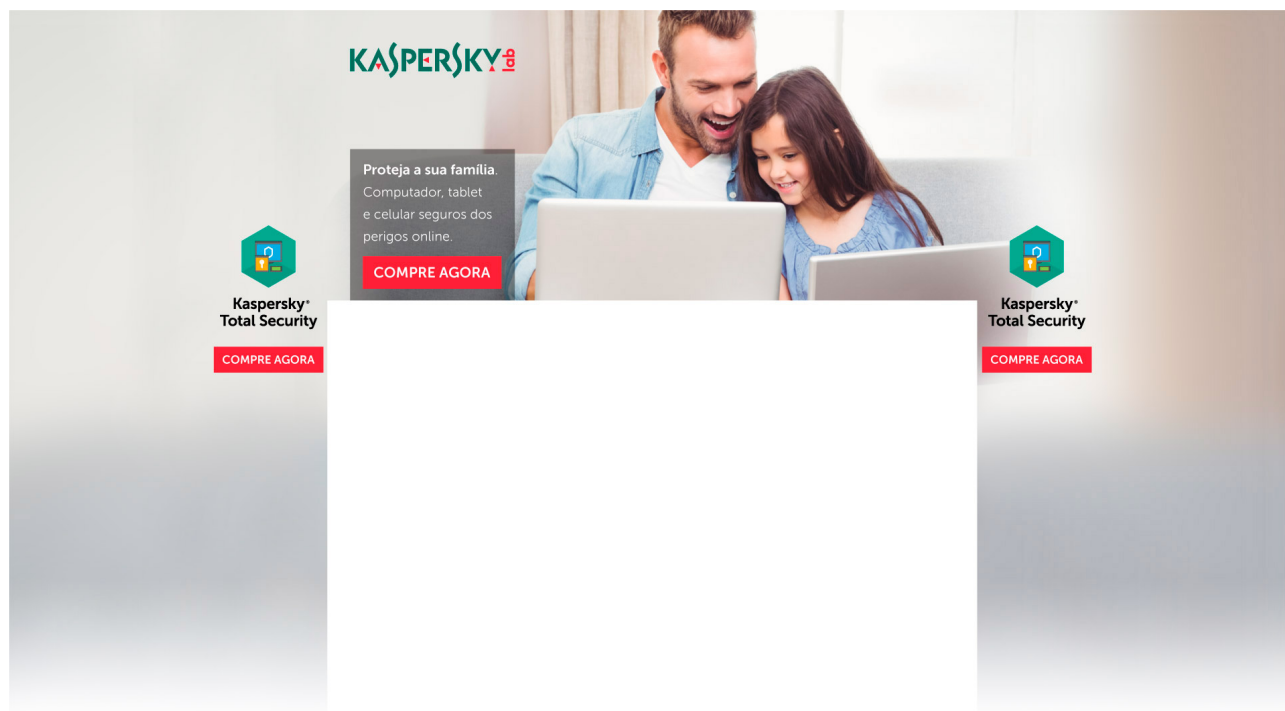


<https://www.tecmundo.com.br/como-fazer/27765-linux-como-recuperar-arquivos-deletados.htm>



Linux: como recuperar arquivos deletados

Por [Leonardo Müller](#) | [@leowmuller](#)

Por vezes, nos deparamos com situações em que vários arquivos de pendrives ou pastas do HD simplesmente desaparecem.

Normalmente, isso acontece por alguma alteração feita no disco quando você estava desatento. Sendo assim, só resta lamentar as perdas, certo?

Não definitivamente.

Saiba que existem formas que podem recuperar tudo o que você perdeu sem muita complicação.

Com alguma distribuição Linux instalada no seu PC, é possível reverter o processo e conseguir acessar tudo o que você perdeu ou pelo menos parte disso.

Para tal, você precisa instalar o software Foremost na sua distro.

Passo 1

Para adicionar o tal programa ao seu sistema, você pode fazer o download pela Central de Programas do Ubuntu ou baixar o arquivo [neste link](#).

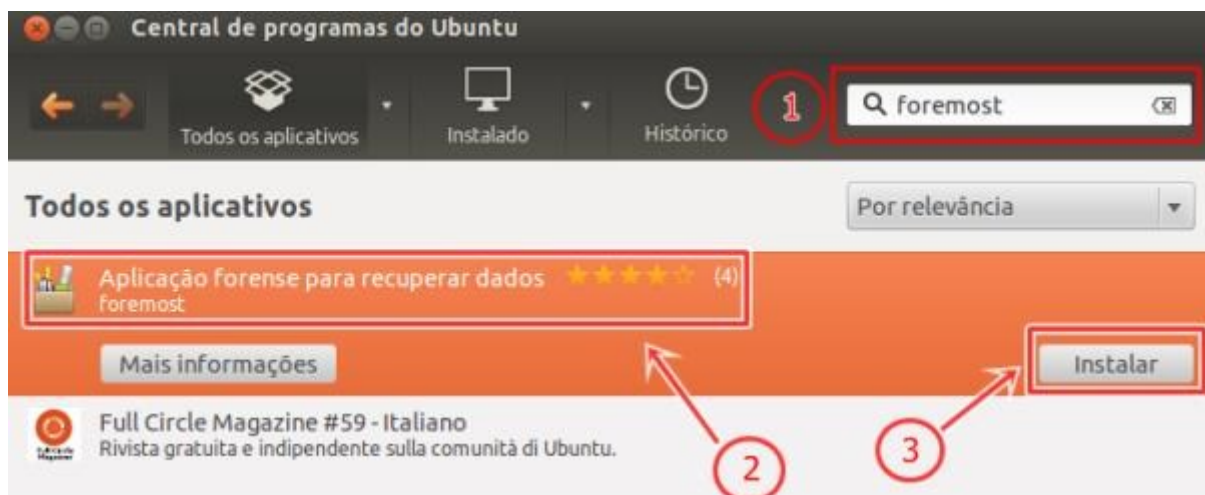
Como vamos utilizar uma versão do Ubuntu neste tutorial, faremos o processo a partir da Central de Programas.

Sendo assim, encontre a central no seu sistema, na barra de atalhos ou pesquisando através do menu de aplicativos do SO.

Passo 2

Com a Central de Programas aberta, pesquise o nome do Foremost na barra de buscas.

Assim que os resultados estiverem compilados, selecione o item “Aplicação forense para recuperar dados” e, em seguida, clique em “Instalar”



Passo 3

Agora, você precisa conectar o disco ou memória flash do qual deseja recuperar arquivos deletados.

No entanto, se o dispositivo em questão for um HD interno, será preciso reiniciar o Ubuntu para dar continuidade ao procedimento.

Passo 4

Com o PC novamente iniciado, é necessário acionar o terminal do sistema para restaurar os arquivos.

Sendo assim, vá ao menu de aplicativos do Ubuntu e pesquise “terminal”.

Em seguida, clique na alternativa correspondente entre os resultados.



Passo 5

Para continuar, você deverá estar logado no terminal, caso contrário, assim que a primeira linha de comando for digitada, a sua senha será requisitada.

Desse modo, digite o seguinte comando:

```
sudo fdisk -L
```

Feito isso, analise a tela exibida no terminal e identifique o disco do qual serão recuperados os arquivos observando o espaço total de cada um.

Em nosso caso, o pendrive que vamos utilizar ficou como “/dev/sdc”.

```
baixaki@PC000080: ~  
Partições lógicas fora da ordem do disco  
  
Disco /dev/sdb: 160.0 GB, 160000000000 bytes  
255 heads, 63 sectors/track, 19452 cylinders, total de 312500000 setores  
Unidades = setores de 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Identificador do disco: 0xf8000000  
  
Dispositivo Boot Início Fim Blocos Id Sistema  
/dev/sdb1 * 63 163840319 81920128+ 7 HPFS/NTFS/exFAT  
/dev/sdb2 163842048 312496127 74327040 7 HPFS/NTFS/exFAT  
  
Disco /dev/sdc: 4194 MB, 4194304000 bytes  
130 heads, 63 sectors/track, 1 cylinders, total de 8192000 setores  
Unidades = setores de 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Identificador do disco: 0x161d32dd  
  
Dispositivo Boot Início Fim Blocos Id Sistema  
/dev/sdc1 * 63 8177084 4088511 7 HPFS/NTFS/exFAT  
/dev/sdc2 8177085 8177147 31+ 21 Desconhecido  
baixaki@PC000080:~$
```

Passo 6

Neste momento, será necessário criar uma réplica raw do seu disco.

Dependendo do espaço do dispositivo, o procedimento pode ser um pouco demorado. Para tal, digite a linha:

```
sudo dd if=/dev/sdc/ of=pendrive.raw
```

Onde escrevemos “/dev/sdc/”, substitua pelo nome do seu disco. Feito isso, entre com a próxima linha de comando:

```
sudo foremost -t all -i pendrive.raw -o recuperados
```

Em nosso caso, utilizamos “all” para selecionar todos os itens para a recuperação. No entanto, se você procura apenas um documento ou imagem, substitua esta parte pela extensão do arquivo procurado.

Caso seja uma foto, coloque JPG, JPEG, GIF etc.

Passo 7

Com isso, os seus arquivos já devem estar recuperados.

Sendo assim, para acessá-los, digite o seguinte comando:

```
sudo nautilus recuperados
```

Desse modo, o navegador de arquivos exibirá um diretório com todos os arquivos recuperados separados por pastas que indicam a extensão dos documentos.

No entanto, os itens não retornam com os antigos nomes e será preciso procurar o que você deseja dentre as opções.

Nota: este processo de recuperação é válido apenas para os arquivos que foram deletados do disco, porém o local onde estavam não foi reutilizado.

Ou seja, nada foi gravado na mesma área onde eles tinham sido registrados anteriormente.

https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&prev=search&rurl=translate.google.com.br&sl=en&sp=nmt4&u=https://askubuntu.com/questions/60228/how-to-remove-all-files-from-a-directory&xid=17259,1500004,15700021,15700105,15700124,15700149,15700168,15700173,15700201&usg=ALkJrhgy-2h9SUAjBLc6hL45UgaVMwZJlg

Como remover todos os arquivos de um diretório?

O mais próximo que recebi é

```
# rm /path/to/directory/*.*
```

mas isso não funciona para arquivos que não possuem uma extensão ...

O Linux não usa extensões.

Cabe ao criador do arquivo decidir se ele deseja que o nome tenha uma extensão.

O Linux olha para o 1º byte para descobrir o tipo de arquivo com o qual está lidando.

Para remover tudo em um uso de diretório:

```
rm /path/to/directory/*
```

Você pode usar a opção `-r` para remover também os diretórios dentro do diretório em que você está removendo o conteúdo.

Caso contrário, ele mostrará um erro informando que não está removendo o diretório.

Se você também deseja excluir arquivos ocultos, execute o `shopt -s dotglob` antes de executar o `rm (...)` - [danjil 6 de setembro às 11:10](#)

O `*` significa all files ;) `*.*` Significa todos os arquivos contendo a `.` em algum lugar no nome - [Rinzwind 6 de setembro às 11:20](#)

@Rinzwind, com mais precisão, o asterisco significa "zero ou mais de qualquer caractere".

Então `*a*` significa zero ou mais caracteres, seguido por `a` seguido de zero ou mais caracteres.

Combinaria os nomes de arquivos `happy`, `apple`, `a` ou `la`. - [DisgruntledGoat 6 de setembro às 11:43](#)

@ user784637 você está muito facilmente impressionado - [barlop Sep 3 '14 às 16:10](#)Aqui está o comando que você deseja: `rm - *` - [anon58192932 10 de outubro de 14 às 21:04](#)

- Para remover a **pasta** com todos os seus conteúdos (**incluindo todas as pastas interiores**):

```
rm -rf /path/to/directory
```

- Para remover todo o **conteúdo da pasta** (**incluindo todas as pastas interiores**), mas não a própria pasta:

```
rm -rf /path/to/directory/*
```

- Para remover todos os "arquivos" de dentro de uma pasta (**não removendo pastas interiores**):

```
rm -f /path/to/directory/*
```

Onde:

- `rm` - significa "**remover**"
- `-f` - significa "**força**", que é útil quando você não quer ser solicitado / solicitado se quiser remover um arquivo, por exemplo.
- `-r` - significa "**recursivo**", o que significa que você deseja ir recursivamente em todas as pastas e remover tudo.

Achei que esta era a resposta mais abrangente e útil, além da resposta que foi marcada como Aceitada. - [inspirednz 20 de agosto de 16 às 1:58](#)

`rm -rf /path/to/directory/*` não remove um arquivo oculto na pasta, por exemplo, `.htaccess`. Talvez `rm -rf /path/to/directory/.` ? Não tentei. - [Mark Berry 29 de março às 17h33](#)

@MarkBerry `rm -rf /path/to/directory/*` - [Lilian A. Moraru Mar 29 '17 às 14:57](#)

@ LilianA.Moraru, fiz alguns testes hoje. `rm -rf /path/to/directory/*` apenas exclui o (s) arquivo (s) oculto (s) no diretório especificado.

Olhando para o comentário do @ danjil na resposta @ Rinzwind, para excluir arquivos ocultos, execute o `shopt -s dotglob` antes de rodar `rm -rf /path/to/directory/*`. - [Mark Berry 29 mar '17 às 22:59](#)

Para remover todos os arquivos no diretório (incluindo arquivos ocultos e subdiretórios), execute:

```
rm -rf /path/to/directory/{* ,.*}
```

Se você deseja excluir apenas arquivos no diretório / path / to / você pode fazer

```
find /path/to/directory -type f -print0 | xargs -0 rm
```

ou

```
find /path/to/directory -type f -exec rm '{}' \;
```

Você pode fazer cargas com `find`, a vantagem é que você pode listar o que é encontrado sem encaminhar para `rm` para que você possa ver o que será excluído antes de começar.

Para excluir todos os arquivos e diretórios (incluindo os **ocultos**) em um diretório, você pode tentar o seguinte:

- apague a pasta e recrie-a

```
rm -rf dir_name && mkdir dir_name
```
- use `find`

```
find dir_name -mindepth 1 -delete
```

Aqui especificamos `-mindepth 1` para excluir o nome do diretório `dir_name`. Dê uma olhada no seguinte link:

uma maneira simples de remover todos os arquivos do diretório

1. simplesmente vá para esse diretório na linha de comando
2. use o comando

```
rm `ls`
```

Isso não funcionará se você tiver nomes de vazio contendo, por exemplo, caracteres espaciais.

Não possui vantagens em relação a um simples `rm *`. - [Florian Diesch 27 de maio às 15:07](#)

```
rm -rf /path/to/directory/*
```

Isso eliminará todo o diretório. Se usarmos

```
rm -r /path/to/directory/*
```

ele só irá excluir sub-diretórios dentro do diretório e se usarmos

```
rm /path/to/directory/*
```

Ele apenas excluirá os arquivos presentes dentro do diretório.

E se um subdiretório estiver presente, ele lançará um erro informando que é um diretório.

-1 Isso está errado. `rm -r path/*` e `rm -rf path/*` ambos excluem *todas* as entradas (não ocultas) do `path` se são diretórios ou não. `path` si não é excluído por nenhum desses comandos. Por favor, veja [rm\(1\)](#). - [David Foerster nov 23 '14 às 9:09](#)

Tecmundo

São Paulo, SP, 30 Abril de 2018

Mkmouse