

Iniciando com Linux

CURSO LINUX UBUNTU



SOBRE OS DIREITOS AUTORAIS

Todos os direitos são reservados. Nenhuma parte deste livro pode ser reproduzida ou transmitida livremente de alguma forma, sem o consentimento prévio dos seus autores.

Se por ventura tiver interessado na publicação de uma análise do livro por favor entre em contato:

Por e-mail: pedrodelfino@e-tinet.com

Pelo Twitter: [@apedrodelfino](https://twitter.com/apedrodelfino)

Por Facebook: <https://www.facebook.com/facedelfino>

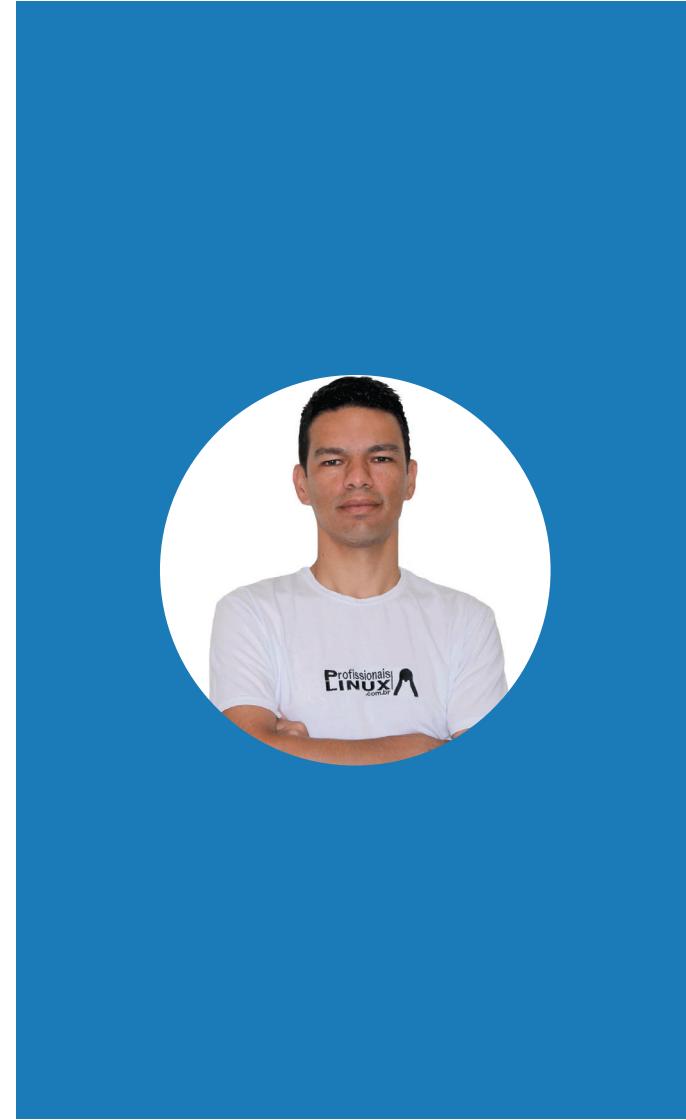
Atualizações do livro visitem sempre: <http://e-tinet.com/curso-linux-ubuntu/>

UTILIZAÇÃO DO LIVRO

O autor deste livro não se responsabiliza direta ou indiretamente pela utilização de qualquer um dos exercícios ou dicas nele contidos. Estes exercícios e dicas são baseados em experiências pessoais e profissionais de Pedro Delfino, bem como da experiência adquirida através de testemunhos de outros usuários. O objetivo deste livro é ensinar todos como usar o Ubuntu Linux, principalmente no seu modo texto. O livro é indicado para usuários iniciantes do Linux em modo texto, com o principal objetivo de se tornar futuros administradores de sistemas Linux.

SOBRE PEDRO DELFINO

Pedro Delfino é o fundador e autor da E-tinet, um blog sobre tecnologia da informação. Atualmente Pedro Delfino gerencia o setor de TI de uma média empresa, onde trabalha com Linux e diversas soluções open source. Já foi professor de Linux, e é formado em Análise de Sistemas.



AGRADECIMENTOS

Fazer um Curso de Linux é o sonho que tenho desde que iniciei a trabalhar nesta área. A internet esta cheia de informações sobre Linux e Ubuntu, hoje eu sou mais um a fornecer conteúdo e quero aproveitar meu conhecimento como um diferencial. Aproveito a ocasião para agradecer a todos que de alguma forma me ajudam a manter o E-tinet. Agradeço a todos que me enviaram e-mails solicitando que este curso estivesse em forma de Livro.

SUMÁRIO

Começando com Ubuntu Linux.....	8
Comparando Ubuntu Linux com outras distribuições Linux.....	17
Ferramentas para instalar softwares adicionais no Ubuntu Linux.....	24
Como manipular repositórios do APT?.....	34
APT Gerenciando pacotes (softwares).....	40
APT - Como adicionar repositório de terceiros?.....	48
Como instalar pacotes (softwares) no Ubuntu via APT?.....	59
Como atualizar pacotes (softwares)com o APT?.....	67
Como remover pacotes (softwares) no Ubuntu via APT?.....	72
Encontrando pacotes (softwares) no Ubuntu com APT.....	82
Como limpar os pacotes baixados com o APT no Ubuntu.....	88
Como instalar pacotes (softwares) com o dpkg no Ubuntu Linux	93

Como remover pacotes (softwares) com dpkg no Ubuntu?	99
Usando o dpkg para obter informações sobre pacotes .deb no Ubuntu.....	104
Gerenciando pacotes(software) no Ubuntu com aptitude.....	115
Como pesquisar pacotes (softwares) com o aptitude no Ubuntu Linux?.....	119
Instalando pacotes/softwares com aptitude no Ubuntu.....	124
Como remover pacotes (softwares) com aptitude no Ubuntu e liberar espaço do seu HD?.....	128
Usando o Shell do Ubuntu Linux.....	135
Como funciona o ambiente Shell do Ubuntu Linux?.....	141
Shell do Linux, dicas para linha de comando e redirecionadores.....	147
Como usar ALIAS no Shell do Ubuntu Linux?.....	157
Como usar os poderes do usuário root no Ubuntu Linux?.....	164
Como permitir que um usuário execute um comando determinado com poderes do root?.....	170
Trabalhando com Arquivos no Linux.....	175



COMEÇANDO COM
UBUNTU LINUX

COMEÇANDO COM UBUNTU LINUX:

Quer utilizar o Ubuntu Linux no trabalho todos os dias? Pois bem, estou iniciando uma série de artigos, para demonstrar como o Ubuntu Linux pode ajudar no seu currículo, e valorizar o seu perfil profissional.

Este curso de linux, é destinado principalmente aos usuários e administradores de sistemas. Ele irá lhe ajudar a melhorar seu rendimento e mostrará rapidamente como encontrar e instalar softwares no Ubuntu, assim como atualizar, manter e controlar a segurança do seu sistema. Vamos mostrar a maneira mais eficiente de utilização de Ubuntu no trabalho, com algumas das ferramentas poderosas que estão ao seu alcance.



O objetivo deste curso de linux, é descrever de forma fácil e rápida conteúdos como:

Comandos:

Comandos Linux, trabalhar direto no console na linha de comando mesmo, demonstrando exemplos inteligentes de forma úteis para navegar pelas assustadoras linhas de comando do Linux.

Trabalhar com repositórios de Software:

Métodos para fazer download e instalação de software personalizado, que são utilizados por sistema Ubuntu / Debian.

Recursos adicionais:

Mostrar como encontrar informações úteis sobre Ubuntu, tais como: listas que você pode inscrever-se, e outros recursos.

Como encontrar documentação local no Linux?

Vou mostrar como usar as ferramentas, e ajudar nos comandos e recursos.

Enfim, vamos ver como administrar um sistema linux, administrando serviços e servidores tudo via linha de comando.

Como a ideia é aprender a administrar servidores linux, neste curso de linux não haverá um grande número de screenshots. O que você vai aprender é um caminho mais rápido para usar o seu sistema Ubuntu Linux, Debian Linux, assim como o Linux em geral.

Ubuntu, Debian e Linux:

Ubuntu é um sistema operacional baseado no Debian GNU / Linux (www.debian.org/). Debian início na década de 1990, devido à sua maturidade, é considerado uma distribuição Linux líder em termos de estabilidade e segurança. O Debian também é conhecido por sua forte adesão ao software livre (www.debian.org/intro/free). E o Ubuntu foi formado com esta base.

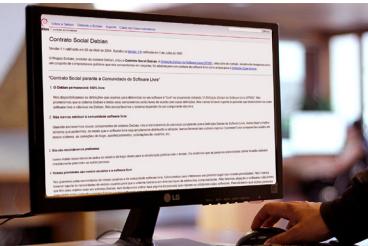


Debian deu origem não só para Ubuntu, mas muitas outras distribuições Linux, as listas completas de distribuição podem ser vistas em (www.debian.org/misc/children-distros):

- Xubuntu – Utiliza o desktop Xfce, sistema baseado no Ubuntu.
- Kubuntu – Utiliza o desktop KDE, sistema baseado no Ubuntu.
- Edubuntu – Um derivado Ubuntu para escolas.
- Knoppix – Com desktop KDE, é um live CD baseado no Debian.
- Kanotix – É um Debian live CD.
- Damn Small Linux – Pequeno (50MB) live CD baseado no Knoppix.
- Mepis – Live CD baseada no Debian e Ubuntu.

Xubuntu, Kubuntu, Edubuntu tem a mesma base da distribuição Ubuntu, a única diferença é o padrão dos desktops, por exemplo, o Kubuntu utiliza o KDE Desktop e Adept Package Manager, que não estão instalados no Ubuntu, mas se você quiser, tem como instalar é claro. Edubuntu é voltada para aplicações educacionais. Muitas aplicações não são instaladas por padrão no ubuntu, nem precisa estar.

Cronograma de lançamento de versão do Ubuntu Linux, [veja sempre aqui](#).



Como o Debian e Ubuntu são sistemas de código aberto, com muitos softwares construídos sobre a Licença GNU GPL (www.gnu.org/copyleft/gpl.html), qualquer pessoa pode visualizar seu código fonte, modificar e distribuir livremente essas alterações. Geralmente, a única exigência é que se respeitem os termos da GPL, que basicamente determina:

Qualquer mudança que você faça, deve ser disponibilizada para que outras pessoas possam utilizar da mesma forma (ver www.debian.org/social_contract).

Você não tem apenas sistema livre que o mundo inteiro apoia, mas também tem um produto que está em constante evolução, que é conduzido por pessoas apaixonadas pelo que fazem. Muitas outras distribuições Linux oferecem as mesmas vantagens, mas o Ubuntu tem certamente mais popularidade.



Observação: Você já se perguntou de onde é o nome Ubuntu (Edgy Eft) e Debian (Woody), veja o padrão destas convenções na nomenclatura da distribuição, descubra em:

<https://wiki.ubuntu.com/DevelopmentCodeNames>

<http://www.debian.org/doc/manuals/project-history/ch-releases.en.html>

Esta é uma pequena introdução sobre o Ubuntu e Debian, claro que em um curso de linux, esta é a aula onde qualquer aluno queria fugir, pois são apenas históricos sem qualquer linha de comando, mas este histórico tem que ser apresentado para reforçar a estabilidade da ferramenta que iremos apresentar aqui.



ubuntu



debian



COMPARANDO UBUNTU LINUX COM OUTRAS DISTRIBUIÇÕES LINUX

COMPARANDO UBUNTU LINUX COM OUTRAS DISTRIBUIÇÕES LINUX:

Após o primeiro artigo da série Curso Linux, (começando com Ubuntu linux), vamos tratar uma questão que sempre está em alta. Antes de escolher a distribuição Ubuntu Linux para trabalhar no desktop ou no servidor. Qual é a diferença?

Se você fizer login em uma sessão de linha de comando num sistema Ubuntu e um Red Hat Enterprise ou Fedora Linux, verá poucas diferenças. Existem diretórios em comum assim como utilitários, com as mesmas funcionalidades.



Então, o que torna o Ubuntu Linux diferente de outras distribuições de Linux?
Uma das diferenças é a maneira de instalar o Ubuntu Linux.

A complexidade de inicialização e instalação de um sistema Linux foi reduzida a alguns cliques com o mouse na instalação do Ubuntu Linux , tornando alguns processos da instalação totalmente automáticos, onde qualquer usuário com conhecimento médio poderá realizar facilmente uma instalação Linux. Em sistemas com base no Red Hat, que é o caso do Fedora e CentOS por exemplo, o usuário tem um programa de instalação com muitas opções, tais como a criação de uma instalação para estação de trabalho ou servidor, assim como selecionar pacotes individualmente para serem instalados, e configurar opções administrativas.



Um dos motivos da popularização do Ubuntu Linux sem dúvida é a facilidade de fazer a instalação, e antes mesmo de instalar o usuário já saberá como o sistema será, pois o processo é realizado em um LIVE CD.

Outra grande diferença entre outras distribuições Linux e o Ubuntu está no gerenciador de pacotes, (trocando em “miúdos”, no gerenciador de instalação e atualização de novos softwares).

O gerenciador de pacotes das distribuições Ubuntu Linux e Debian Linux são praticamente o mesmo, porém as implementações feitas no Ubuntu são significativamente diferentes, do Debian e também é diferente da maioria de outros sistemas baseados em Debian, todas utilizam o APT (Advanced Package Tool)

Você pode usar o APT para instalar, remover, consultar atualizações de pacotes do tipo deb. Sistemas Linux baseados no Red Hat usam um sistema de pacotes chamado de RPM para lidar com as mesmas tarefas de instalar, remover, consultar atualização.

Vamos aprender a trabalhar com o APT, no Ubuntu Linux, e isto servirá para o Debian também, os comandos do APT são os mesmos.

Outra grande diferença é a forma como os sistemas trabalham no que diz respeito à inicialização, é a tela de login, desktop padrão, papel de parede, conjunto ícone e muito mais. Neste ponto há uma grande diferença, embora Red Hat e Ubuntu utilizem o mesmo ambiente no desktop (Window Manager) que é o GNOME, as ferramentas utilizadas e a localizações nos menus são totalmente diferentes.

O Ubuntu Linux tem uma abordagem muito simples, se você acabou de instalar um sistema com Ubuntu Linux, não verá um grande número de aplicativos disponíveis, porém tudo pode ser facilmente instalado, tudo está à altura de um comando do APT.

O ambiente desktop é bastante simples e elegante, uma mistura de alguns dos melhores e mais funcionais aplicativos disponíveis para o desktop Linux.

É característica do Ubuntu Linux esta simplicidade, que tem como objetivo não sobrecarregar o usuário.

Outra característica única de um sistema Ubuntu é a prática intencional do travamento da utilização do usuário root, utilizando o utilitário sudo (www.gratisoft.us/sudo/intro.html), que permite executar um comando como usuário root, para executar tarefas administrativas do sistema, ainda falaremos muito sobre o sudo aqui no [curso de linux](#).



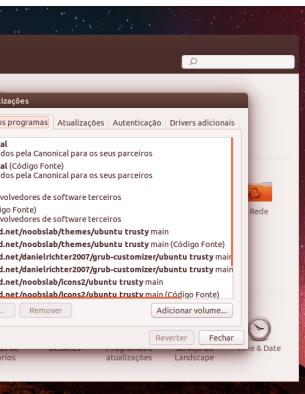
O login com usuário root entrega privilégios que permitem o acesso a qualquer componente do sistema. Seria fácil remover todo um sistema de arquivos como um usuário root, em um único comando, o Ubuntu Linux tenta limitar a utilização desta conta, liberando apenas em momentos prudentes.

A maioria das distribuições Linux exige que o usuário efetue login como root, ou use o comando su para executar tarefas de administração, no entanto um usuário do Ubuntu faz isso através do utilitário sudo com o seu próprio login.

Ubuntu Linux tem características únicas, com algumas vantagens e desvantagens, mas elas estão longe de ser limitadas. Ubuntu tem as ferramentas no lugar e permite que você as personalize e ganhe experiência. A ideia é ter um sistema de fácil manutenção e seguro. Isto torna o Ubuntu um sistema rápido, direto e fácil de manusear.

FERRAMENTAS PARA
INSTALAR SOFTWARES
ADICIONAIS NO
UBUNTU LINUX.





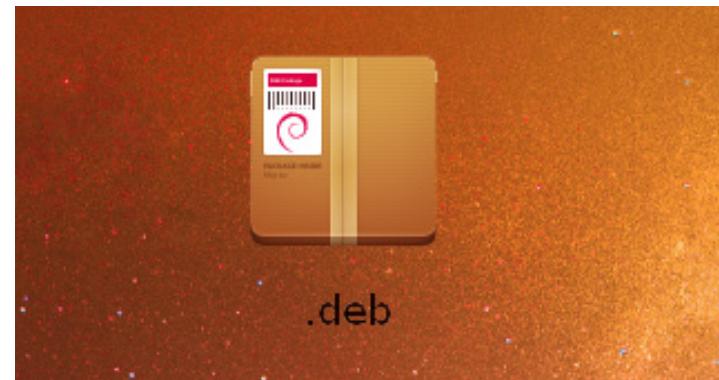
FERRAMENTAS PARA INSTALAR SOFTWARES ADICIONAIS NO UBUNTU LINUX.

Nesta parte do ebook de Linux, estamos nos preparando para iniciar a manutenção dos softwares de um sistema Ubuntu Linux, e também Debian Linux. Com as ferramentas APT, dpkg e Aptitude

Para trabalhar manutenção de instalação de softwares tem o utilitário aptitude, que é uma excelente opção de front-end para o APT em modo texto. O utilitário dselect é um front-end para trabalhar com o dpkg, todas estas opções também estão disponíveis na maioria dos sistemas Debian.

Ubuntu Linux utiliza os pacotes no formato do Debian, os famosos arquivos com .deb (com extensão .deb), o arquivo .deb (pacotes) podem não só conter arquivos do softwares, como também ter muitas informações sobre o conteúdo do pacote.

Um software Linux pode incluir metadados com descrições, dependências, arquitetura do computador (32 bits, 64 bits), tamanho, licenciamento e outras informações.

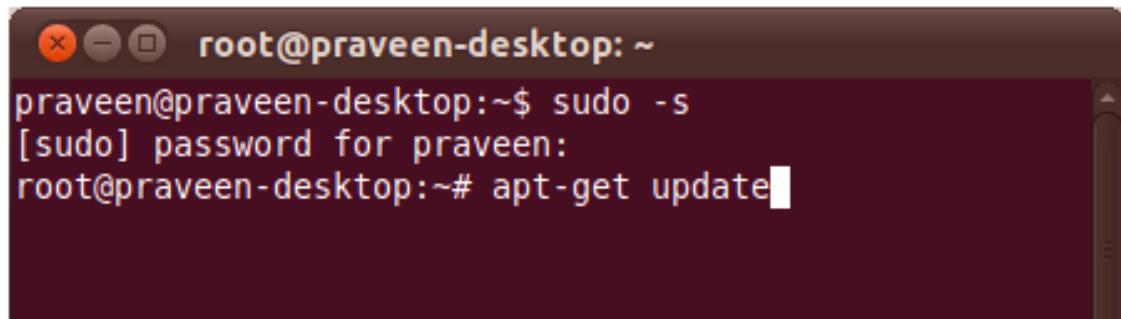


A ferramenta aptitude funciona muito bem para a maioria dos casos de instalação de software no Ubuntu Linux e no Debian, no entanto existem muitas outras ferramentas para gerenciar estes pacotes. Particularmente prefiro o APT, aqui no curso de linux nós vamos ver todas estas ferramentas.

Na próxima página um pequeno resumo das ferramentas disponíveis para manipular pacotes no Linux, em distribuições derivadas do Debian Linux, é claro. Já comentei comparando Ubuntu Linux com outras distribuições Linux, que sistemas baseados no RedHat utilizam o rpm, que não veremos agora.

APT:

Vamos utilizar o APT para baixar e instalar pacotes a partir de repositórios online. (Veremos como trabalhar com repositórios online, aqui no curso de linux). O APT pode ser chamado com os comandos apt-get, apt-cache e assim por diante. Pode ser usado para instalar pacotes localmente também, no entanto é normalmente utilizado para trabalhar com o software online, ou seja em repositórios online, como por exemplo o site o Ubuntu Linux



A screenshot of a terminal window titled "root@praveen-desktop: ~". The window shows the following command being run:
praveen@praveen-desktop:~\$ sudo -s
[sudo] password for praveen:
root@praveen-desktop:~# apt-get update

Dpkg:

Vamos utilizar o dpkg para fazer a instalação de pacotes .deb localmente, ou seja aqueles pacotes que você poderá baixar de um site ou utilizar a partir de um CD-ROM da distribuição.

Aptitude:

Vamos utilizar o comando aptitude para trabalhar com pacotes .deb em repositórios online. A aptitude é a ferramenta mais recomendada porque é “semi-automático”, ele cuida de algumas tarefas que você deveria fazer manualmente se estiver trabalhando com o dpkg ou o APT.

O Ubuntu Linux é instalado a partir de um único CD-ROM ou DVD. Após a instalação, você pode executar o comando apt-cache stats, este comando irá apresentar um relatório sobre o total de pacotes disponíveis:

```
$ apt-cache stats
Total de Nomes de Pacotes: 12985 (519k)
  Pacotes normais: 7643
  Pacotes puramente virtuais: 245
  Pacotes virtuais únicos: 980
  Pacotes virtuais misturados: 60
  Faltando: 4057
Total de versões distintas: 8396 (437k)
Total de descrições distintas: 10591 (254k)
Total de dependências: 64642 (1810k)
Total de relações ver/arquivo: 9536 (153k)
Total de relações Desc/Arquivo: 10591 (169k)
Total de mapeamentos "Provides": 2014 (40,3k)
Total de strings "globbed": 113 (1416)
Total de espaço de dependência de versão: 369k
Total de espaço frouxo: 75,6k
Total de espaço contabilizado para: 3036k
```



Como você pode ver, temos um estoque de pacotes para instalação no Ubuntu Linux, tem mais de 12.000 pacotes. A comunidade Debian / Ubuntu é muito cuidadosa, e apenas inclui software que são apropriados.

Embora a comunidade Debian acredite que apenas software livre são necessários para fazer o bom funcionamento de um sistemas Linux, há casos em que as pessoas querem ou precisam colocar software proprietário em suas máquinas.

O Debian suporta softwares proprietário, mas não inclui softwares proprietários na distribuição principal de um sistema Debian Linux. Estes softwares proprietários muitas vezes estão disponíveis no site FTP do Debian em repositórios não livres. Existem muitos pacotes, cujo único trabalho é instalar e utilizar o software proprietário, claro ninguém está autorizado a distribuir por contra própria estes softwares.

O grupo que cuida do Ubuntu Linux a Canonical, trabalhar de forma bem semelhante ao Debian. Veja o licenciamento do Ubuntu oferecendo quatro categorias de software: o main, restricted, universe, e multiverse.

Main:

Contém software que são distribuídos gratuitamente e apoiados pela equipe do Ubuntu. Grande parte deste software está disponível quando você instala o Ubuntu via CD-ROM.

Restricted:

Contém software que é comum a muitos sistemas Linux, também tem suporte da equipe do Ubuntu, mas não podem ser licenciados como software livre.

Universe:

São pacotes que não podem ser licenciados como software livre. Os softwares universe não têm garantia de segurança e correção.

Multiverse:

Contém software que não atende o conceito de software livre. Software multiverse não tem qualquer suporte, e cabe a você determinar a validade do licenciamento, suporte etc.

```
mysql-l...x86_64 0:5.1.71-1.el6 will be updated
mysql-l...x86_64 0:5.1.73-3.el6_5 will be an update
Dependency Resolution
Dependencies Resolved

=====
Package          Arch      Version       Repo
=====
installing:
mysql-server   x86_64  5.1.73-3.el6_5
installing mysql.x86_64 5.1.73-3.el6_5
mysql          x86_64  5.1.73-3.el6_5
perl-DBD-MYSQL x86_64  4.011-1.el6
perl-DBI        x86_64  1.614-1.el6
updating for dependencies:
mysql-l...     x86_64  5.1.73-3.el6_5
transaction Summary
=====
install      4 Package(s)
upgrade      1 Package(s)

total download size: 12 M
warning: this ok [y/N]: y
```



COMO MANIPULAR
REPOSITÓRIOS DO APT?

COMO MANIPULAR REPOSITÓRIOS DO APT?

Continuando o curso de Linux com ubuntu, vamos ver como manipular os repositórios APT, para entender como o Ubuntu Linux gerencia repositórios de pacotes deb.

Em versões anteriores do Ubuntu, os repositórios multiverse e universe, ([veja a aula sobre a diferença entre os grupos de pacotes do Ubuntu Linux](#)) não estavam ativos por padrão, agora eles já estão ativos.

Uma preocupação que você poderá ter: Licenças e patches podem não estar disponíveis nos repositórios multiverse e universe. O que para muitos usuários é um problema, se você está considerando que seu sistema deverá permanecer sempre atualizado e com a segurança 100%, cuidado ao utilizar pacotes multiverse e universe, eu nunca tive problemas, mas é sempre bom esclarecer.

Para desativar os repositórios multiverse e universe, abra o arquivo /etc/apt/sources.list

Exemplo:

```
$sudo vi /etc/apt/sources.list
```

Lembre da aula onde compararmos o Ubuntu Linux com outras distribuições, o uso do comando sudo é uma particularidade do Ubuntu.



Comente as linhas, como no exemplo abaixo, colocando o caracter # para ativar o comentário, aperte a tecla “insert” para ativar o modo de edição no editor “VI”

```
#deb http://us.archive.ubuntu.com/ubuntu/ feisty universe
#deb-src http://us.archive.ubuntu.com/ubuntu/ feisty universe
#deb http://us.archive.ubuntu.com/ubuntu/ feisty multiverse
#deb-src http://us.archive.ubuntu.com/ubuntu/ feisty multiverse
#deb http://security.ubuntu.com/ubuntu feisty-security universe
#deb-src http://security.ubuntu.com/ubuntu feisty-security universe
#deb http://security.ubuntu.com/ubuntu feisty-security multiverse
#deb-src http://security.ubuntu.com/ubuntu feisty-security multiverse
```

Este é um exemplo no ubuntu feisty, não muda muito para as outras versões.
Claro, agora devemos salvar o arquivo.

No VI, tecle ESC, depois digite: x! , e tecle ENTER, é claro!

Calma, nós vamos ter uma aula apenas de VI aqui no curso de Linux, VI é o melhor editor para modo texto no Linux.

Da mesma forma, se você quiser adicionar repositórios extras, simplesmente adicione a linha no arquivo /etc/apt/sources.list.

```
$sudo vi /etc/apt/sources.list
```

Simplesmente inserimos uma linha iniciando com deb (para pacotes pré-compilados) ou deb-src (pacotes com código fonte), em seguida a URL para o repositório, depois o nome da distribuição (como por exemplo o feisty acima), e descrições do grupo de componente dos pacotes (main, multiverse ou universe).



Arquivo: /etc/apt/sources.list

```
du/ubuntu/ oneiric main restricted
  u.com/ubuntu oneiric-security main
du/ubuntu/ oneiric-updates main
  du/ubuntu/ oneiric-backports main
  nical.com/ubuntu oneiric partner
  u.com/ubuntu oneiric main
ibuntu.org/ oneiric free non-free
```

Se você adicionar outros repositórios de terceiros, antes de modificar o seu sistema Linux certifique-se de analisar a autenticidade da entidade que oferece o software. É fácil adicionar falhas ou softwares maliciosos no seu sistema se você não tiver um cuidado. Seja prudente, embora esta prática seja pouco provável entre os usuários de sistemas Linux, muitos são administradores de servidores, de redes e sempre devemos ter este cuidado.

Só use o software a partir de fontes bem conhecidas, e tenha sempre um meio de verificar o software antes de instalar.

Para obter mais informações sobre software repositórios veja o Debian Repository HOWTO

Segue um exemplo de entrada para um /etc/apt/sources.list :

```
deb ftp://sunsite.cnlab-switch.ch/mirror/debian/ instável principais
  contrib non-free
```

APT GERENCIANDO PACOTES (SOFTWARES)

APT GERENCIANDO PACOTES (SOFTWARES)

Iremos ver agora como trabalhar com o comando APT, para gerenciar pacotes no Ubuntu Linux e no Debian Linux.

Embora o dpkg e o APT trabalhem em conjunto, a maior parte do tempo o APT será suficiente para instalar qualquer software que você precise. Também será possível fazer download de pacotes, atualização e realizar pesquisas.

Hoje nesta série do curso de Linux com Ubuntu, vamos fazer um pequeno resumo dos comandos do apt, este resumo será o nosso manual de referências do APT ao longo deste curso de administração de servidores linux com Ubuntu.

Lista dos principais comandos do APT no Ubuntu Linux ou Debian Linux:

```
$ sudo apt-get update
```

Consulta as entradas dos repositórios listado nos arquivos /etc/apt/sources.list e atualiza a base de dados de pacotes disponíveis. Sempre execute este comando para manter a base de dados de pacotes atualizada.

```
$ sudo apt-cache search
```

Pesquisa na base de dados dos repositórios APT, pela palavra chave. Este comando devolve o nome e a descrição do pacote encontrado.

```
$ sudo apt-get install
```

Este comando faz o download e instala um ou mais pacotes de um repositório APT. A partir da versão 0.6 do APT, este comando irá verificar automaticamente a autenticidade do pacote, através de chaves gpg . Vamos ver mais sobre como adicionar chaves gpg de repositórios APT:

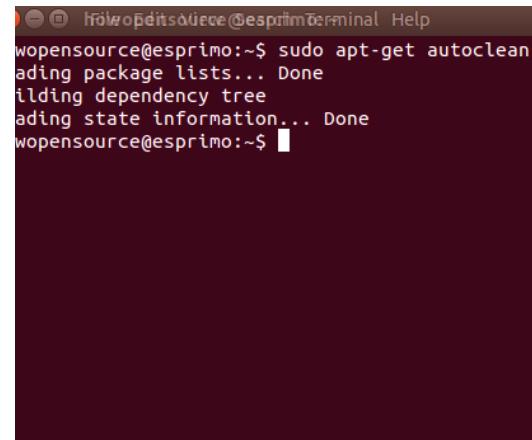
```
$ sudo apt-get install -d
```

Verifica atualizações de todos os pacotes de um sistema Ubuntu Linux, em seguida executa o download e instalação:

```
$ sudo apt-get dist-upgrade
```

Atualiza todo o sistema para uma nova versão da distribuição, mesmo que isso signifique eliminar pacotes. Esta não é a melhor opção para atualizar um sistema:

```
$ sudo apt-get autoclean
```



```
wopensource@esprimo:~$ sudo apt-get autoclean
  Reading package lists... Done
  Building dependency tree
  Reading state information... Done
wopensource@esprimo:~$ █
```



```
caeza@caezar:~$ sudo apt-get autoremove
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
caeza@caezar:~$ sudo apt-get clean
caeza@caezar:~$
```

Pode ser executado a qualquer momento para excluir pacotes parcialmente baixados

`sudo apt-get clean`

Limpa todos os pacotes de /var/cache/apt/archives, muito bom para libertar espaço em disco:

`$ sudo apt-get -purge remove < pacote >`

Remove todos os pacotes e seus arquivos de configuração. Para não remover os arquivos de configuração, apenas retire o parâmetro `-purge` :

`$ sudo apt-get -f install`

Remove todos os pacotes e seus arquivos de configuração. Para não remover os arquivos de configuração, apenas retire o parâmetro –purge :

```
$ sudo apt-get -f install
```

Verifica as dependências e tenta corrigir os problemas, no caso de uma instalação corrompida. apt-config -v

Mostra a versão do apt e de seus utilitários:

```
$ sudo apt-key list
```

Lista as chaves gpg, dos repositórios APT:

```
$ sudo apt-cache stats
```

Mostra as estatísticas de todos os pacotes instalados:

```
$ sudo apt-cache depends
```

Mostra as dependências de um pacote (se está ou não instalado):

```
$ sudo apt-cache search
```

Pesquisa por um pacote nos repositórios on-line.

Pronto, já temos nosso manual de referências do comando APT, na próxima parte vamos fazer alguns exercícios para fixar estes comandos. O apt-get install sem duvida será o mais utilizado no seu dia para instalar novos pacotes. E não se esqueça de executar o comando man apt-get





APT - COMO ADICIONAR REPOSITÓRIO DE TERCEIROS?

APT - COMO ADICIONAR REPOSITÓRIO DE TERCEIROS?

Vamos aprender como adicionar repositório de terceiros no Ubuntu Linux. É um bom exemplo, pois existem milhares de repositórios APT com milhares de softwares disponíveis.

Vamos fazer um pequeno exercício para demonstrar como o Ubuntu Linux trabalha como a instalação de pacotes. Iremos instalar o Google Picasa, que dispensa maiores apresentações.

Vamos instalar a partir de um software de terceiros, ou seja, não é um repositório padrão do Ubuntu Linux, sendo assim vamos precisar adicionar o repositório ao /etc/apt/sources.list, como já vimos na aula anterior. Como manipular repositórios do APT, no Ubuntu .

Adicionando o repositório APT

Para começar, vamos adicionar o repositório do Google, em /etc/apt/sources.list

Para isto, execute o comando:

```
$ sudo echo "deb http://dl.google.com/linux/deb/ stable non-free"  
>> /etc/apt/sources.list
```

Este comando vai adicionar a linha deb http://dl.google.com/linux/deb/ stable non-free no arquivo /etc/apt/sources.list , lembrando que no capítulo de como manipular repositórios do APT no Ubuntu nós comentamos detalhes sobre esta entrada no /etc/apt/sources.list

Agora vamos baixar e validar as chaves de autenticidade do repositório do google:

```
$ sudo wget https://dl-ssl.google.com/linux/linux_signing_key.pub  
-O /tmp/key.pub
```

Pronto, isto baixou o arquivo pgp para /tmp/key.pub

Hora de adicionar a chave pgp :

```
$ sudo apt-key add /tmp/key.pub
```

Pronto, assim importamos a chave do repositório do google, e já está validado em nosso sistema

Podemos listar a chave adicionada com o comando:

```
$ sudo apt-key list  
uid Google, Inc. Linux Package Signing Key  
linux-packages-keymaster@google.com  
sub 2048g/C07CB649 2007-03-08
```

Next, update the APT package cache to refresh the new repository.
This is done using sudo and running apt-get update. Make sure to check for the Google repository as it scrolls by:



```
udo apt-get update  
t:1 http://dl.google.com stable Release.gpg [189B]  
n http://dl.google.com stable/non-free Translation-en_US
```

Algo parecido com a saída anterior deverá ser retornado pelo comando sudo apt-key list.

Agora precisamos atualizar os repositórios disponíveis, para que os pacotes do repositório do google fiquem disponíveis para a instalação em nosso sistema Ubuntu Linux:

```
$sudo apt-get update
```

```
Get:1 http://dl.google.com stable Release.gpg [189B]  
Ign http://dl.google.com stable/non-free Translation-en_US  
Get:2 http://dl.google.com stable Release [1026B]
```

Para fazer um teste vamos pesquisar e ver se o picasa está disponível para instalação:

```
$ sudo apt-cache search picasa
```

```
picasa - Picasa is software that helps you instantly find, edit  
and share all the pictures on your PC.
```

Podemos verificar os dados do pacote, com o comando:

```
$ sudo apt-cache show picasa
Package: picasa
Status: install ok installed
Priority: optional
Section: non-free/graphics
Installed-Size: 106560
Maintainer: Picasa Linux Maintainer <picasa -linux-packager@google.com>
Architecture: i386
Version: 3.0.5744-02
Depends: libc6 (>= 2.2), libasound2, zlib1g, gconf2, libfreetype6 (>=
2.1.10)
```

Description: Image management application from Google

Picasa is software that helps you instantly find, edit and share all the pictures on your PC. Every time you open Picasa, it automatically locates all your pictures (even ones you forgot you had) and

sorts them into visual albums organized by date with folder names you will

recognize. You can drag and drop to arrange your albums and make labels to

create new groups. Picasa makes sure your pictures are always organized.

Picasa also makes advanced editing simple by putting one-click fixes and powerful effects at your fingertips. And Picasa makes it a snap to share your pictures - you can email, print photos, make gift CDs, post

pictures on your blog, and even share your photos online with Picasa Web</picasa>

Conseguimos também ver as dependências dos pacotes do picasa:

```
$ sudo apt-cache depends picasa
picasa
    Depende: libc6
    Depende: libasound2
    Depende: zlib1g
    Depende: gconf2
    Depende: libfreetype6
```

Enfim, vamos instalar picasa via comando apt-get

```
$sudo apt-get install picasa
```

O comando ap-get install vai fazer o download do arquivo do pacote .deb e suas dependências e já fazer a instalação, pronto simples assim.



Lembrando, que este exemplo só poderá ser executado em um Ubuntu Linux ou Debian Linux (ou derivado) com interface gráfica já instalada, caso contrário ao executar o comando `sudo apt-get install picasa` as dependências serão bem maiores.



COMO INSTALAR PACOTES (SOFTWARES) NO UBUNTU VIA APT?

COMO INSTALAR PACOTES (SOFTWARES) NO UBUNTU VIA APT?

Como utilizar o APT para instalar pacotes no Ubuntu Linux ou no Debian Linux. Veremos como são resolvidas as dependências de pacotes .deb

Agora você já pode instalar utilizando sudo apt-get install nome_do_pacote qualquer software disponível de autoria do Google no seu Ubuntu Linux, pois na aula anterior o APT no Ubuntu Linux – Como adicionar repositório de terceiros, aprendemos como adicionar repositório de terceiros ao nosso APT.

Com um exemplo simples, podemos instalar softwares com o picasa:

```
$ sudo apt-get install picasa
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  picasa
0 upgraded, 1 newly installed, 0 to remove and 115 not upgraded.
Need to get 21.7MB of archives.
After unpacking 82.3MB of additional disk space will be used.
Get:1 http://dl.google.com stable/non-free picasa 2.2.2820-5
[21.7MB]
Fetched 21.7MB in 1m3s (340kB/s)
Selecting previously deselected package picasa.
(Reading database ... 88015 files and directories currently installed.)
Unpacking picasa (from .../picasa_2.2.2820-5_i386.deb) ...
Setting up picasa (2.2.2820-5) ...
```



```
fabio@ubuntu-desktop:~$ sudo apt-get install filezilla
[sudo] password for fabio:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Os pacotes extra a seguir serão instalados:
  filezilla-common libtinyxml2.6.2 libwxbase2.8-0 libwxgtk2.8-0
Pacotes sugeridos:
  libgnomeprintui2.2-0
0 novos pacotes a seguir serão instalados:
```

Ficou muito simples instalar novos pacotes no seu Ubuntu Linux, faremos mais alguns testes como, por exemplo:

Instalar a linguagem de programação php, famosa e conhecida por desenvolvedores web.

```
$ sudo apt-get install php5
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Os pacotes extras a seguir serão instalados:
  apache2-mpm-prefork apache2-utils apache2.2-common libapache2-mod-php5
  php5-common
Pacotes sugeridos:
  apache2-doc apache2-suexec apache2-suexec-custom php-pear php5-timezone
  nedb
Os NOVOS pacotes a seguir serão instalados:
  apache2-mpm-prefork apache2-utils apache2.2-common libapache2-mod-php5
  php5
  php5-common
0s pacotes atualizados, 6 pacotes novos instalados, 0 a serem removidos e
18 não atualizados.
É preciso baixar 4005kB de arquivos.
Depois desta operação, 10,9MB adicionais de espaço em disco serão usados.
Você quer continuar [S/n]?
```

Veja que nesse exemplo, pedimos para instalar a linguagem de programação php5 em nosso sistema Linux, e o APT nos informou que ele também irá instalar os pacotes, apache2-mpm-prefork apache2-utils apache2.2-common libapache2-mod-php5 php5-common, pois são pacotes necessários para que o php5 funcione em nosso sistema Ubuntu Linux.

Mas como o APT sabe disto?

Na Aula APT Gerenciando pacotes (softwares) no Ubuntu, onde falamos sobre o gerenciamento de pacotes com APT, podemos lembrar que um pacote .deb, guarda suas dependências, ou seja, os pacotes que ele depende para funcionar em um sistema Ubuntu Linux ou Debian Linux.

```
$ sudo apt-cache depends php5
php5
|Depende: libapache2-mod-php5
|Depende: libapache2-mod-php5filter
Depende: php5-cgi
Depende: php5-common
```

O Comando acima lista as dependências dos pacotes do php5.

Se você executar:

```
sudo apt-cache depends libapache2-mod-php5
sudo apt-cache depends libapache2-mod-php5filter
sudo apt-cache depends php5-cgi
sudo apt-cache depends php5-common
```

Vai conseguir ver as dependências de todos os outros pacotes. O melhor de tudo isso é que você não precisa se preocupar com nada, pois o APT vai resolver todas estas dependências e instalar o pacote pronto para usar.



Claro que existem exceções, onde o APT não consegue resolver problemas de dependências, mas isso geralmente ocorre quando você acaba utilizando pacotes de versões diferentes do Ubuntu, coisa que eu não recomendo.

Nota: Caso você não tenha entendido o porquê da opção sudo no inicio de cada comando, pode dar uma olhada na aula Comparando Ubuntu Linux com outras distribuições Linux, para entender a diferença entre o Ubuntu Linux e outras distribuições.

Completando esta aula, já podemos garantir que sabemos instalar pacotes no Ubuntu Linux e no Debian Linux via o utilitário APT, na próxima parte deste ebook vamos aprender a remover pacote via APT.



COMO ATUALIZAR PACOTES (SOFTWARES) COM O APT?

COMO ATUALIZAR PACOTES (SOFTWARES) COM O APT?

Isso é bem mais simples do que parece, o APT facilita muito o trabalho de atualização do sistema.

Ao longo do tempo pacotes mudam e novas versões são lançadas, para adicionar novos recursos e corrigir problemas. Você pode usar o APT para atualizar seu sistema para novas versões de pacotes.

Em primeiro lugar, podemos verificar se existem atualizações para os pacotes o seu sistema Ubuntu, vamos utilizar o apt-get com a opção update, não se esqueça de ler o post APT Gerenciando pacotes (softwares) no Ubuntu:

```
$sudo apt-get update
```

Este comando procura por novas versões de pacotes nos repositórios disponíveis, e atualiza a lista de pacotes e versões em cache no seu sistema Ubuntu Linux.

Para atualizar os pacotes no seu sistema Ubuntu Linux usando a opção upgrade para o apt-get.

Como atualizar um sistema Ubuntu Linux completo?

```
$ sudo apt-get upgrade
```

Você deve sempre atualizar a lista de pacote antes de atualizar seu sistema, por isso é uma boa ideia executar dois comandos juntos, separando os comandos por ponto e vírgula, como no exemplo:

```
$ sudo apt-get update; sudo apt-get upgrade
Get:1 http://dl.google.com stable Release.gpg
[189B]
...
The following packages will be upgraded:
app-install-data app-install-data-commercial
apport apport-gtk bind9-host
capplets-data dbus dbus-1-utils
```



Lembrando que com este comando, sudo apt-get upgrade estaremos atualizando o sistema completo, ou seja, todos os pacotes do seu Ubuntu Linux.

Como atualizar um pacote individual no Ubuntu Linux?

O mais interessante neste caso é que para atualizar um pacote individualmente no Ubuntu Linux, nós vamos usar o mesmo comando utilizado para instalar um pacote via APT, ou seja o apt-get install

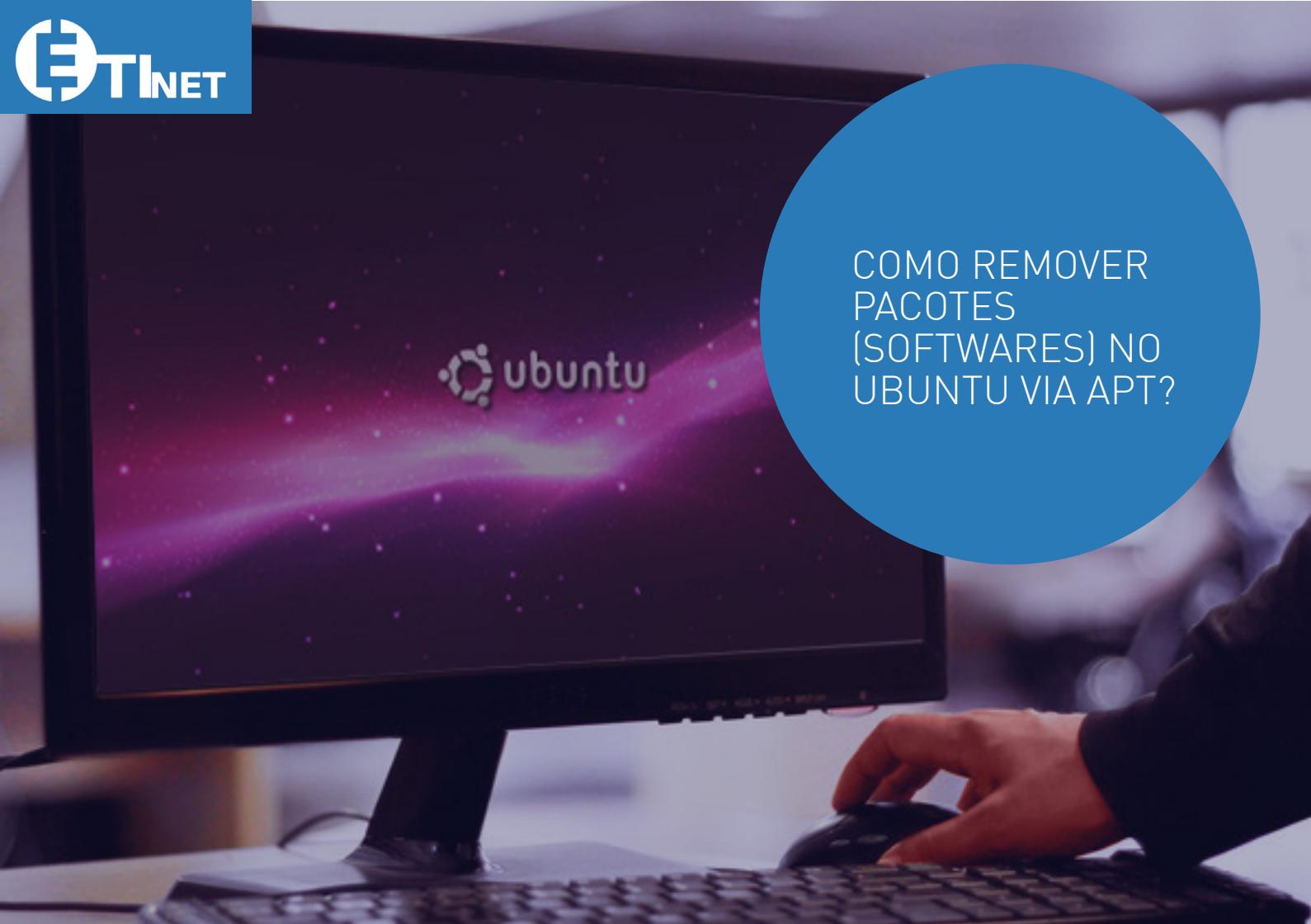
Relembrando do exemplo do post: Como instalar pacotes (softwares) no Ubuntu via APT onde instalamos o picasa, se o google resolver lançar uma nova versão do seu software de gerenciamento de imagem, para você atualizar seu Ubuntu Linux, basta digitar, apt-get install picasa:

```
$sudo apt-get install picasa
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
picasa já é a versão mais nova.
0 pacotes atualizados, 0 pacotes novos instalados, 0 a serem
removidos e 12 não atualizados.
```



No exemplo anterior, o APT verificou que já temos a última versão do pacote em nosso sistema, e não será necessário fazer a atualização. Caso uma nova versão estivesse disponível, o APT automaticamente iria atualizar o picasa.

Hoje, aprendemos como atualizar pacotes no Ubuntu Linux via APT, mais uma das facilidades do APT, já que podemos atualizar o sistema completo ou um pacote individual.

A photograph of a computer monitor displaying the Ubuntu desktop environment. The screen shows the classic purple space-themed wallpaper with the 'ubuntu' logo in the center. A person's hands are visible at the bottom, using a mouse over a keyboard.

COMO REMOVER PACOTES (SOFTWARES) NO UBUNTU VIA APT?



```
pauto@pauto-Latitude-E5430:~$ sudo apt-get remove gimp
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Os seguintes pacotes foram instalados automaticamente e já não são necessários:
  libbabl-0.0-0 libgegl-0.0-0 libilmbase6 libopenexr6 libgimp2.0
  libwebkitgtk-1.0-common libwebkitgtk-1.0-0 libjavascriptcoregtk-1.0-0
  gimp-data
Use 'apt-get autoremove' para removê-los.
Os pacotes a seguir serão REMOVIDOS:
  gimp
  gimp-data
  libbabl-0.0-0
  libgegl-0.0-0
  libilmbase6
  libopenexr6
  libgimp2.0
  libwebkitgtk-1.0-common
  libwebkitgtk-1.0-0
  libjavascriptcoregtk-1.0-0
  gimp-data
```

COMO REMOVER PACOTES (SOFTWARES) NO UBUNTU VIA APT?

Remover pacotes no Ubuntu Linux é tão fácil quanto instalar, talvez até mais fácil. Para remover pacotes via apt no Ubuntu ou no Debian vamos usar o comando apt-get remove

Parece simples e realmente é, mas vamos fazer algumas considerações.

Vamos remover o pacote do picasa, que instalamos no post Como instalar pacotes (softwares) no Ubuntu via APT

Removendo pacotes via APT:

```
$sudo apt-get remove picasa
Reading package lists ... Feito
Construindo árvore dependência
Lendo estado informações ... Feito
Os seguintes pacotes serão removidos:
  picasa
0 atualizados, 0 novos instalados, 1 a remover e 115 não actualiza-
dos.
Necessidade de obter 0B de arquivos.
Após descompactar 82.3MB de espaço em disco serão libertados.
Você deseja continuar [Y / n]? Y
```

Será solicitada uma configuração. Muitas vezes é preciso remover um pacote e todos seus arquivos de configuração, vamos pegar um exemplo do samba, os arquivos de configuração por padrão estão /etc/samba, neste diretório você tem o famoso arquivo smb.conf :

```
$sudo apt-get remove --purge samba
```

Pronto, qualquer configuração do samba será removida também.

Parada para reflexão: Tenha calma neste momento, vamos aprender a instalar e configurar o samba. Peguei o pacote do samba neste exemplo, pois é bem usual.

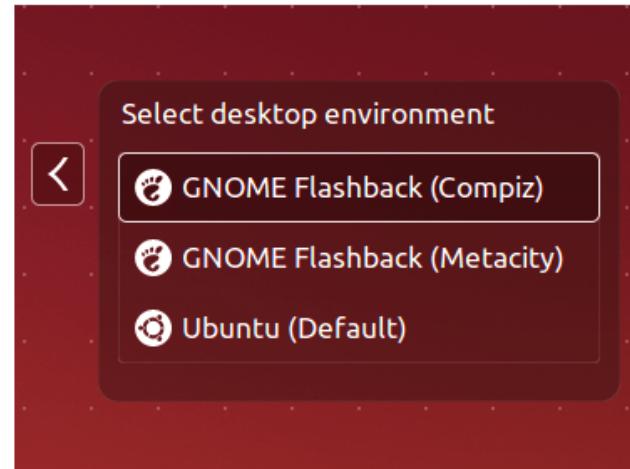
Outro exemplo interessante:

E se eu remover o gnome? É isso mesmo, vamos imaginar que você levantou revoltado com o mundo e resolveu remover o gnome do seu Ubuntu.

Sim é bem possível, tem gente que não gosta do gnome, ou que está querendo trocar para KDE.

Na verdade esse exemplo é bom para analisar que o gnome não tem apenas um pacote e sim vários, muitos pacotes mesmo, são 195 para ser mais exato na instalação do meu notebook por exemplo.

Podemos usar o comando com o operador “ * ”



```
$ sudo apt-get remove gnome*
```

Os seguintes pacotes foram automaticamente instalados e não são mais requeridos:

```
menu icon-tropical-theme emerald-tropical-theme vim-addon-manager  
libglade2-ruby1.8 libgconf2-ruby libinklevel4 libtext-glob-perl  
libatk1-ruby1.8 libdate-calc-perl m4 metacity-tropical-theme autoconf  
libchm1 python-chm libglade2-ruby libcarp-clan-perl  
emerald-ubuntu-sunrise-theme intltool libtool gdm-tropical-theme  
autotools-dev nautilus-script-manager libgconf2-ruby1.8 libglib2-ruby1.8  
libltdl7-dev gtk-ubuntu-sunrise-theme libcairo-ruby1.8  
libfile-find-rule-perl gdm-ubuntu-sunrise-theme libgdk-pixbuf2-ruby1.8  
automake wallpaper-ubuntu-sunrise-theme gtk-tropical-theme libxml++2.6-2  
wallpaper-tropical-theme gromit icon-ubuntu-sunrise-theme libart2-ru-  
by1.8
```

```
libgtk2-ruby1.8 imagemagick libconfig-tiny-perl libpangol-ruby1.8  
libnumber-compare-perl libbit-vector-perl
```

Use 'apt-get autoremove' para removê-los.

Os pacotes a seguir serão REMOVIDOS:

```
acroread-plugin-speech alacarte apturl assogiate bluez-gnome  
brdesktop-artwork-gnome cairo-dock-plug-ins camorama clamtk compiz  
compiz-gnome contact-lookup-applet deskbar-applet ekiga eog evince evo-
```

```
lution
 evolution-data-server evolution-exchange evolution-plugins evolution-
-webcal
 f-spot fast-user-switch-applet file-roller firefox-3.0-gnome-support
 firefox-gnome-support gconf-editor gdebi gdm gdm-guest-session gedit-
-plugins
 gksu gnochm gnome-about gnome-accessibility-themes gnome-app-install
 gnome-applets gnome-applets-data gnome-art gnome-cards-data gnome-com-
mon
 gnome-control-center gnome-desktop-data gnome-doc-utils gnome-games
 gnome-games-data gnome-icon-theme gnome-icon-theme-nuovo gnome-keyring
 gnome-mag gnome-media gnome-media-common gnome-menus gnome-mime-data
 gnome-mount gnome-netstatus-applet gnome-nettool gnome-orca gnome-pa-
nel
 gnome-panel-data gnome-pilot gnome-pilot-conduits gnome-pkg-tools
 gnome-power-manager gnome-screensaver gnome-session gnome-settings-da-
emon
 gnome-spell gnome-splashscreen-manager gnome-system-monitor
 gnome-system-tools gnome-terminal gnome-terminal-data gnome-themes
 gnome-themes-extras gnome-tropical-theme gnome-ubuntu-sunrise-theme
 gnome-user-guide gnome-utils gnome2-globalmenu-applet gparted
 gstreamer0.10-gnomevfs gufw gvfs gvfs-backends hwtest-gtk inkblot jo-
ckey-gtk
```

```
language-pack-gnome-pt language-pack-gnome-pt-base libbonoboui2-0
libcanberra-gnome libdeskbar-tracker libebook1.2-9 libecal1.2-7
libedata-book1.2-2 libedata-cal1.2-6 libedataserverui1.2-8 libeel2-2
libexchange-storage1.2-3 libgail-gnome-module libgksu2-0 libglobal-
menu-gnome
libgnome-desktop-2-7 libgnome-keyring0 libgnome-keyring1.0-cil libgno-
me-mag2
libgnome-media0 libgnome-menu2 libgnome-pilot2 libgnome-speech7
libgnome-vfs2.0-cil libgnome-vfsmm-2.6-1c2a libgnome-window-settings1
libgnome2-0 libgnome2-canvas-perl libgnome2-common libgnome2-perl
libgnome2-ruby libgnome2-ruby1.8 libgnome2-vfs-perl libgnome2.0-cil
libgnomecanvas2-0 libgnomecanvas2-common libgnomecanvas2-ruby1.8
libgnomecups1.0-1 libgnomekbd-common libgnomekbd3 libgnomekbdui3
libgnomenu0-2 libgnomeprint2.2-0 libgnomeprint2.2-data libgnomeprin-
tui2.2-0
libgnomeprintui2.2-common libgnomeui-0 libgnomeui-common libgnomev-
fs2-0
libgnomevfs2-bin libgnomevfs2-common libgnomevfs2-extra libgtkhtml-e-
ditor0
libgtkhtml3.14-19 libgtksourceview1.0-0 liblpint-bonobo0 libmbca0
```

```
libpam-gnome-keyring libpanel-applet2-0 libpolkit-gnome0 libxine1-gno-
me
mousetweaks nautilus nautilus-cd-burner nautilus-script-audio-convert
nautilus-share network-manager-gnome ooobasis3.0-gnome-integration
policykit-gnome python-gmenu python-gnome2 python-gnome2-desktop
python-gnomecanvas python-pyatspi rhythmbox seahorse seahorse-plugins
software-properties-gtk ssh-askpass-gnome startupmanager
system-config-printer-gnome tangerine-icon-theme tomboy totem
totem-gstreamer totem-mozilla totem-plugins totem-plugins-extra totem-
-xine
tracker-search-tool tropical-theme tsclient ubufox ubuntu-docs
ubuntu-sunrise-theme ubuntuperfeito update-manager update-notifier
usb-creator vim-full vim-gnome vim-rails vinagre vino
xulrunner-1.9-gnome-support yelp zenity
0 pacotes atualizados, 0 pacotes novos instalados, 195 a serem removidos
e 19 não atualizados.
Depois desta operação, 593MB de espaço em disco serão liberados.
Você quer continuar [S/n]? n
```

Por favor, se executar este comando apenas para teste não se esqueça de responder N (“Não”) para a pergunta: Você quer continuar?

Pronto, o apt vai remover seu gnome completo, isso poderia ser aplicado para qualquer pacote que tem um conjunto de pacotes.

O interessante também é que o APT já analisa os pacotes que não serão mais necessários, que não fazem parte do gnome e sugere que execute o comando apt-get autoremove para remover destes pacotes também.

Pronto o APT trabalhou por nós, e não vai deixar o nosso sistema Ubuntu Linux com pacotes desnecessários





ENCONTRANDO
PACOTES
(SOFTWARES)
NO UBUNTU COM APT

ENCONTRANDO PACOTES (SOFTWARES) NO UBUNTU COM APT:

Muitas vezes precisamos instalar pacotes no Ubuntu, mas não sabemos o nome completo. O APT disponibiliza uma ferramenta para encontrar pacotes nos seus repositórios.

Agora, já sabemos instalar pacotes com APT no ubuntu, e também remover estes pacotes.

Vamos ver agora como usar o comando apt-cache search, para encontrar pacotes:

```
$ apt-cache search <nome do pacotes>
```

Por exemplo:

```
$ apt-cache search picasa
kipi-plugins - image manipulation/handling plugins for KIPI aware
programs
python-gdata - Google Data Python client library
picasa - Image management application from Google
```

O comando apt-cache não necessita ser utilizado em conjunto com o comando sudo, assim como o apt-get install, mesmo porque qualquer usuário pode fazer uma pesquisa nos pacotes disponíveis.

Como encontrar pacotes, um pouquinho mais completos?

Às vezes você não sabe o nome completo de um pacote .deb. Digamos que você queira instalar uma versão específica do kernel.

Sim no Ubuntu Linux e no Debian você pode instalar várias versões do kernel com um simples comando.

No nosso exemplo eu quero instalar a versão do kernel 2.6.27-11. E agora? Qual é o nome do pacote .deb?

Vamos descobrir com o comando:

```
$ apt-cache search kernel | grep -i 2.6
```

Esse comando vai ao retorno de vários pacotes relacionados ao kernel 2.6, eu poderia usar um filtro mais específico:

```
$ apt-cache search kernel | grep -i 2.6.27
linux-headers-2.6.27-11-generic - Linux kernel headers for ver-
sion 2.6.27 on x86/x86_64
linux-headers-2.6.27-11 - Header files related to Linux kernel
version 2.6.27
linux-image-2.6.27-7-generic - Linux kernel image for version
2.6.27 on x86/x86_64
linux-restricted-modules-2.6.27-7-generic - Non-free Linux ker-
nel modules for version 2.6.27 on x86/x86_64
linux-restricted-modules-2.6.27-11-generic - Non-free Linux ker-
nel modules for version 2.6.27 on x86/x86_64
linux-image-2.6.27-11-generic - Linux kernel image for version
2.6.27 on x86/x86_64
```



Pronto, agora já sabemos que o nome do pacote .deb para o kernel 2.6.27 é linux-image-2.6.27-11-generic. Lembrando, isso é apenas um exemplo, talvez no seu ubuntu esteja disponível outra versão do kernel, emfim depende muito dos repositórios que você tem habilitados.

A photograph of a young man with dark hair, smiling, wearing a white button-down shirt and a bright blue tie. He is seated at a desk, facing a computer monitor. His hands are on a keyboard and mouse. The background is a plain, light-colored wall.

COMO LIMPAR OS PACOTES BAIXADOS COM O APT NO UBUNTU

COMO LIMPAR OS PACOTES BAIXADOS COM O APT NO UBUNTU:

Após a instalação inicial de uma versão do Ubuntu, o Ubuntu mantém os pacotes baixados em cache no seguinte diretório /var/cache/apt/ para acelerar a transferência se você precisar deles novamente.

Tudo bem que espaço em disco, não são nossos maiores problemas hoje, mas isto pode eventualmente usar uma grande quantidade de espaço do seu HD. Você pode remover este cache de pacotes, mas vai precisar baixar um pacote removido do cache novamente se for necessário para resolver alguma dependência de instalação por exemplo.

Para limpar todos os pacotes em cachê mantidos pelo APT vamos utilizar o comando apt-get clean.

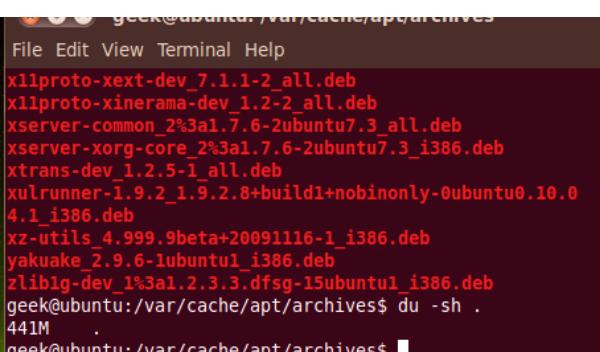
Se quiser ter certeza que tem algum pacote em cache, os conhecidos arquivos .deb, podem executar o comando find para verificar:

```
$ find /var/cache/apt/ -name \*.deb
/var/cache/apt/archives/ttf-farsiweb_0.4.dfsg-6_all.deb
/var/cache/apt/archives/lib soprano4_2.2.2+dfsg.1-1ubuntu1~intrepid1_i386.deb
/var/cache/apt/archives/libclucene0ldbl_0.9.20-3_i386.deb
/var/cache/apt/archives/libamrnb3_7.0.0.2-0.0medibuntu1_i386.deb
/var/cache/apt/archives/libqt4-script_4.4.3-0ubuntu1_i386.deb
/var/cache/apt/archives/libqt4-svg_4.4.3-0ubuntu1_i386.deb
/var/cache/apt/archives/kdelibs5_4%3a4.2.2-0ubuntu1~intrepid2_i386.deb
```

No meu caso isto retornou muita coisa, mas se eu precisar saber o tamanho deste cache do APT eu posso analisar isto com o comando du -h

```
$du -h /var/cache/apt/  
4,0M  /var/cache/apt/archives/partial  
70M   /var/cache/apt/archives  
79M   /var/cache/apt/
```

No meu caso, o cache do APT, esta ocupando apenas 79 GB, vamos combinar que nem compensa rodar sudo apt-get clean , mesmo porque eu posso precisar destes pacotes para resolver algumas dependências na instalação de outro software.



geek@ubuntu: /var/cache/apt/archives

```
File Edit View Terminal Help
x11proto-xext-dev_7.1.1-2_all.deb
x11proto-xinerama-dev_1.2-2_all.deb
xserver-common_2%3a1.7.6-2ubuntu7.3_all.deb
xserver-xorg-core_2%3a1.7.6-2ubuntu7.3_i386.deb
xtrans-dev_1.2.5-1_all.deb
xulrunner-1.9.2_1.9.2.8+build1+nobinonly-0ubuntu0.10.0
4.1_i386.deb
xz-utils_4.999.9beta+20091116-1_i386.deb
yakuake_2.9.6-1ubuntul_i386.deb
zlib1g-dev_1%3a1.2.3.3.dfsg-15ubuntul_i386.deb
geek@ubuntu:/var/cache/apt/archives$ du -sh .
441M .
geek@ubuntu:/var/cache/apt/archives$
```

Mas caso você seja louco por espaço:

```
$ sudo apt-get clean
$ find /var/cache/apt/ -name \*.deb
```

O diretório está vazio agora.

O comando, `find` e `du` que foi utilizado neste aula será revisto, pode ficar tranquilo, o `find` principalmente tem muitas opções.

COMO INSTALAR PACOTES (SOFTWARES) COM O DPKG NO UBUNTU LINUX



COMO INSTALAR PACOTES (SOFTWARES) COM O DPKG NO UBUNTU LINUX:

O utilitário dpkg trabalha em uma camada mais baixa do que os utilitários APT. O APT usa o dpkg para gerenciar software em seu sistema Ubuntu.

O trabalho do dpkg é semelhante a outros utilitários como, por exemplo, o yum e rpm do Red Hat em distribuições baseadas em Linux.

Habitualmente o APT tem como funcionalidade realizar praticamente qualquer coisa, mas existem momentos em que o dpkg irá ser necessário, como por exemplo, descobrir qual o pacote está associado com um determinado arquivo em seu sistema.

Os principais comandos do dpkg estão no glossário de linux, vamos utilizar todos esses comandos aqui no curso de linux com ubuntu.

Com o comando dpkg, qualquer usuário poderá executar consultas no banco de dados de pacote do sistema. Usar os comandos para instalar ou remover software do seu sistema, só poderá ser executado com os privilégios de root.

Instalando um pacote com dpkg:

O comando dpkg se concentra apenas em manipular pacotes, enquanto o apt-get vai cuidar da bagunça e de outros detalhes para descobrir quem hospeda um repositório de pacotes e baixar de lá.

Para simplificar: O exemplo a seguir usa o apt-get para baixar um pacote e depois dpkg para instalá-lo, só para dar um gostinho do comando dpkg.

Primeiro, faça o download do pacote .deb usando comando apt-get. A opção -d é para executar apenas o download em instalar o pacote:

```
sudo apt-get -d install minicom
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Pacotes sugeridos:
  lrzs
Os NOVOS pacotes a seguir serão instalados:
  minicom
0 pacotes atualizados, 1 pacotes novos instalados, 0 a serem re-
movidos e 1128 não atualizados.
É preciso baixar 175kB de arquivos.
Depois desta operação, 1188kB adicionais de espaço em disco se-
rão usados.
Obter:1 http://ftp.ubuntu.com jaunty/main minicom 2.3-1ubuntu1
[175kB]
Baixados 175kB em 5s (31,2kB/s)
Baixar completo e no modo somente baixar ("download only")
```

Lembrando que o arquivo foi baixado /var/cache/apt/archives/, onde se encontra o nosso cache local dos arquivos .deb baixados pelo APT, se você quiser limpar o cache do APT poderá executar isso depois, veja o post e falamos sobre o assunto:

```
$sudo dpkg -i /var/cache/apt/archives/minicom_2.3-1ubuntu1_i386.deb  
Selecionando pacote previamente não selecionado minicom.  
(Lendo banco de dados ... 200846 arquivos e diretórios atualmente instalados).  
Desempacotando minicom (de .../minicom_2.3-1ubuntu1_i386.deb)  
...  
Configurando minicom (2.3-1ubuntu1) ...
```



Lembrando que o pacote minicom_2.3-1ubuntu1_i386.deb foi baixado pela versão do meu ubuntu, na versão que você estiver utilizando provavelmente o pacote do minicom estará em outra versão. Sendo assim utilize a tecla [TAB] para completar o comando:

```
$sudo dpkg -i /var/cache/apt/archives/minicom [TAB]
```

Sempre comento pra quem está começando com Linux, a tecla [TAB] é uma ótima companheira.

Pronto é só, agora já sabemos utilizar o dpkg para instalar pacotes no ubuntu, lembrando que estes comandos também podem ser executados no Debian Linux.

COMO REMOVER PACOTES (SOFTWARES) COM DPKG NO UBUNTU?

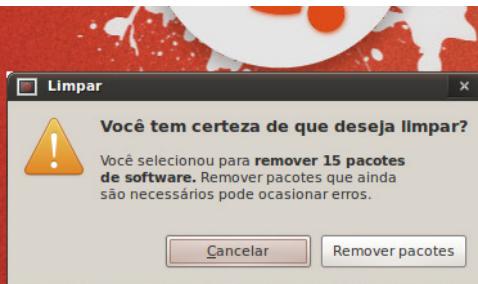
COMO REMOVER PACOTES (SOFTWARES) COM DPKG NO UBUNTU?

Remover pacotes do Ubuntu Linux com utilitário dpkg é muito simples, temos duas opções para remoção de pacotes no Ubuntu

No último post do curso de linux com ubuntu, mostrei como instalar pacotes utilizando o dpkg, hoje vamos remover o mesmo pacote, até porque o utilitário minicom provavelmente não será utilizado por você.

Para remover um pacote instalado com o dpkg, use a opção `-r` da seguinte forma:

```
$ sudo dpkg -r minicom  
(Reading database ... 89182 files and directories currently installed.)  
Removing minicom ...
```



Se você quiser remover o pacote e seus arquivos de configuração de uma só vez, use a opção -P

```
$ sudo dpkg -P minicom  
(Reading database ... 89126 files and directories currently installed.)  
Removing minicom ...  
Purging configuration files for minicom ...
```

Quando devemos usar a opção -r ou -P?

Pois bem, vamos pensar na remoção de um pacote que tenha um arquivo de configuração, como por exemplo, o pacote do samba, os arquivos de configuração estão em /etc/samba, o arquivo smb.conf contém todas as configurações deste famoso servidor de arquivos para redes windows, se você precisa remover o samba, mas ainda acha que vai precisar do seu smb.conf, então execute:

```
$ sudo dpkg -r samba
```





Mas se você ficou nervoso com a versão de seu samba, e não quer mais ver a “cor” do seu smb.conf, deve executar:

```
$ sudo dpkg -P samba
```

Simples assim, da mesma forma existe o apt-get remove samba -purge e o apt-get remove samba, veja o post da [aula de apt-get](#).

USANDO O DPKG PARA OBTER INFORMAÇÕES SOBRE PACOTES .DEB NO UBUNTU

USANDO O DPKG PARA OBTER INFORMAÇÕES SOBRE PACOTES .deb NO UBUNTU:

No Debian e Ubuntu os pacotes vêm empacotados em arquivos simples, com a extensão .deb. Cada arquivo deb contém um ou mais arquivos que compõem o pacote em si, arquivos de apoio, documentação e talvez código fonte.

Assim, um arquivo .deb é basicamente um repositório de arquivos que deseja instalar no seu computador, além de alguns cabeçalhos e controles de informações que identifica o software (descrições, checksums, informações, e assim por diante).

Você pode extrair uma série de informações sobre um pacote com o comando dpkg.

Como extrair arquivos de um pacote .deb com dpkg?

O exemplo abaixo extrai os arquivos do arquivo .deb e direciona para o diretório atual do pacote rsync no diretório em /tmp.

Vamos usar o apt-get para baixar o pacote do rsync para fazer um teste:

```
$ mkdir /tmp/rsync_conteudo  
$ sudo apt-get -d install rsync
```





O Comando dpkg -x , é utilizado para extrair os arquivos de um pacote:

```
$ sudo dpkg -x rsync_2.6.9-3ubuntu1.1_i386.  
deb /tmp/rsync_contents  
$ ls /tmp/rsync_contento  
etc usr
```

Você deve substituir o arquivo rsync_2.6.9-3ubuntu1.1_i386.deb no comando acima por qualquer arquivo .deb que você possa vir a baixar.

Consultando informações sobre os pacotes .deb

Como verificar a versão de um pacote deb, podemos utilizar o dpkg -p para verificar a versão de um pacote já instalado.

```
$ dpkg -p rsync  
Version: 2.6.9-3ubuntu1.1
```

Como verificar a versão que ainda não está instalada?

O exemplo a seguir mostra como consultar os pacotes instalados por um pacote chamado rsync e exibir informações e versão sobre esse pacote:

Use a opção -l para obter informações sobre um deb

```
$ dpkg -l rsync_2.6.9-3ubuntu1.1_i386.deb  
novo pacote debian, versão 2.0.
```

Como listar todos os pacotes instalados?

O comando dpkg -l lista de todos os pacotes instalados em seu sistema, digite o seguinte:

```
$ dpkg -l | less  
...  
ii  acpi 0,09-1  exibe informações sobre dispositivos ACPI  
...
```

Ou só para visualizar informações de um pacote específico utilizar a opção -l, com um nome de pacote:

```
$ dpkg -l rsync  
ii  rsync      3.0.5-1ubuntu2  fast remote file copy program  
(like rcp)
```

Como verificar o status de um pacote?

Verificar o status de um pacote, neste caso o pacote deve estar instalado também:

```
$ dpkg -s rsync
Package: rsync
Status: install ok installed
Priority: optional
Section: net
Installed-Size: 664
Maintainer: Ubuntu Core Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture: i386
Version: 3.0.5-1ubuntu2
Depends: libacl1 (>= 2.2.11-1), libc6 (>= 2.8), libpopt0 (>= 1.14), lsb-base (>= 3.2-14), base-files (>= 4.0.1)
Suggests: openssh-client, openssh-server
Conffiles:
```

```
/etc/default/rsync 126a5f9d12d57da27b95192fed269ad5
/etc/init.d/rsync 6e44406359668ee5d2fd0be20effddca
Description: fast remote file copy program (like rcp)
rsync is a program that allows files to be copied to and from remote
machines in much the same way as rcp. It has many more options than
rcp, and uses the rsync remote-update protocol to greatly speed up
file transfers when the destination file already exists.

...
```

Listar o conteúdo de um pacote .deb

O comando dpkg -c lista o conteúdo de um arquivo deb

```
$ dpkg -c rsync_2.6.9-3ubuntu1.1_i386.deb
drwxr-xr-x root / root 0 2007-08-17 20:48. /
drwxr-xr-x root / root 0 2007-08-17 20:48. /
usr/
drwxr-xr-x root / root 0 2007-08-17 20:48. /
usr/bin /
-rwxr-xr-x root / root 294864 2007-08-17
20:48. / usr / bin / rsync
drwxr-xr-x root / root 0 2007-08-17 20:48. /
usr / share /
drwxr-xr-x root / root 0 2007-08-17 20:48. /
usr / share / doc /
...
...
```



Como verificar os arquivos instalados por um pacote .deb?

Para ver os arquivos instalados de um pacote e o que seu sistema está utilizando, use a opção -L:

```
$ dpkg -L minicom
/ .
/ usr
/ usr / share
/ usr / share / man
/ usr/share/man/man1
/ usr/share/man/man1/minicom.1.gz
/ usr/share/man/man1/xminicom.1.gz
/ usr/share/man/man1/ascii-xfr.1.gz
...
...
```

Se o pacote não for completamente removido, você poderá ver alguns arquivos de configuração:

```
$ dpkg -L minicom  
/etc  
/etc/minicom  
/etc/minicom/minicom.users
```

Para remover pacote com o dpkg, verifique o post como remover pacotes com dpkg no ubuntu.

Estes exemplos são o básico para se trabalhar com o dpkg, mas nem por isso está é uma lista exaustiva. Outras opções disponíveis incluem aquelas para reconfiguração de pacotes (dpkg-reconfigure). Verifique a página de manual do dpkg para mais informações, com o comando \$ man dpkg.

GERENCIANDO PACOTES (SOFTWARE) NO UBUNTU COM APTITUDE

GERENCIANDO PACOTES (SOFTWARE) NO UBUNTU COM APTITUDE:

O dpkg e APT são ferramentas que já existem há muito tempo e funcionam bem, mas existem algumas situações onde ambas as ferramentas podem exigir um conhecimento muito elevado. O aptitude faz as coisas de maneira mais fáceis, automatizando algumas operações de pacotes (como por exemplo, executar o apt-get update antes de atualizar ou instalar um pacote)

Permitindo a flexibilidade suficiente para ser útil. Por esses motivos, recomendamos a utilização do aptitude em linha de comando do linux sempre que possível.

Abaixo está uma pequena lista dos comandos mais utilizados para começar os nossos trabalhos, exatamente da mesma forma que fizemos com o APT e com dpkg

sudo aptitude:

Inicia a interface do aptitude. Use Ctrl + T para acesso ao menu e a tecla q para sair.

aptitude help:

Listas de ajuda para o uso do aptitude.

aptitude search:

Listas de pacotes combinando com a palavra chave.

sudo aptitude update:

Atualiza os índices de pacotes disponíveis nas fontes APT.

sudo aptitude upgrade:

Atualiza todos os pacotes para suas últimas versões.

aptitude show:

Lista informações sobre o pacote, que esteja instalado ou não.

sudo aptitude download:

Faz o download do pacote, mas não instala.

sudo aptitude clean:

Remove todos os arquivos .deb do cache em /var/cache/apt/archives.

sudo aptitude autoclean:

Remove todos os pacotes .deb desatualizados do /var/cache/apt/archives.

sudo aptitude install:

Instala o pacote no seu sistema. Existem várias opções para a seleção de versões específicas e utilização de curingas.

sudo aptitude remove:

Remove um ou mais pacotes do sistema.

sudo aptitude dist-upgrade:

Atualiza todos os pacotes para sua versão mais recente, automatiza a remoção ou instalação de pacotes quando necessário.

Sendo assim temos uma lista dos principais comandos do aptitude, que será utilizado nos demais posts sobre este assunto.

COMO PESQUISAR
PACOTES
(SOFTWARES) COM O
APTITUDE NO
UBUNTU LINUX?

COMO PESQUISAR PACOTES (SOFTWARES) COM O APTITUDE NO UBUNTU LINUX?

Você pode pesquisar um pacote com o aptitude usando uma palavra-chave, ou o nome completo do pacote, assim como com usado no APT. Aqui está uma pesquisa usando a palavra “minic”, que retorna o pacote e Minicom.

```
$ aptitude search minic
p    minicom      - friendly menu driven serial communication
```

Um pouco mais simples que pesquisar pacote com o APT.

Lembrando que não precisamos usar o “sudo”, quando estivermos pesquisando pacote com o aptitude.

Outro ponto interessante é que por padrão, o aptitude irá sempre irá executar um apt-get update antes de instalar ou atualizar um pacote. Você pode, entretanto usar o comando para executar apenas a atualização, caso queira executar uma pesquisa e ter certeza que pesquisa e ter certeza que está buscando na última versão do repositório do APT.

```
$sudo aptitude update
Get:1 http://security.ubuntu.com feisty-security Release.gpg
[191B]
Ign http://security.ubuntu.com feisty-security/main Translation-
-en_US
Get:2 http://us.archive.ubuntu.com feisty Release.gpg [191B]
...
...
```

Se desejar atualizar todos os pacotes no sistema, você pode enviar junto à opção de atualização para o aptitude. Isto irá instalar quaisquer novos pacotes que estejam nos repositórios (neste exemplo, não houve novos pacotes):

```
$ sudo aptitude upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
Building tag database... Done
No packages will be installed, upgraded, or removed.
0 packages upgraded, 0 newly installed, 0 to remove and 0 not
upgraded.
Need to get 0B of archives. After unpacking 0B will be used.
```

Portanto para pesquisar pacotes com o aptitude é muito simples, se quiser ter certeza que está pesquisando a última atualização execute o comando sudo aptitude update antes do aptitude search “palavra chave do pacote”.

Parece muito óbvio para quem já trabalha com o APT para pesquisar pacotes, mas é bom lembrar que o aptitude é um sistema mais completo que o APT e merece algum estudo.

```
nick@Jolt: ~
nick@Jolt: ~$ sudo apt-get install cifs-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  keyutils
Suggested packages:
  winbind
The following NEW packages will be installed:
  cifs-utils keyutils
  upgraded, 2 newly installed, 0 to remove and 283 not upgraded.
Need to get 120 kB of archives.
After this operation, 410 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
% [Connecting to us.archive.ubuntu.com (2001:67c:1562::15)]
```

INSTALANDO PACOTES/SOFTWARES COM APTITUDE NO UBUNTU



INSTALANDO PACOTES/SOFTWARES COM APTITUDE NO UBUNTU:

Instalar softwares / pacotes no Ubuntu é muito simples, você já instalou um pacote usando o apt-get aqui neste curso de linux, fazer a mesma coisa com o aptitude vai parecer até repetição, mas não é. O aptitude é um pouco mais interativo.

Por exemplo, para fazer o download de um pacote .deb com o aptitude podemos usar o comando:

```
$ sudo aptitude download minicom
Get:1 http://us.archive.ubuntu.com feisty/main minicom
2.2-4build1 [168kB]
Fetched 1B in 0s (2B/s)
```



Se você quer apenas instalar o pacote do Minicom, você pode chamar o aptitude conforme abaixo:

```
$ sudo aptitude install minicom  
...  
Need to get 0B/265kB of archives. After unpacking 1401kB will be  
used.  
Do you want to continue? [Y/n/?] n
```

Lembrando que a atualização de um pacote assim como no APT é feita de forma automática, ou seja, caso o pacote minicom esteja instalado no seu sistema o aptitude simplesmente vai atualizar, caso o pacote não esteja instalado o aptitude fará a instalação normalmente.

Se há uma série de pacotes que deseja instalar, você pode usar um curinga com aptitude para selecionar vários pacotes a serem instalados de uma só vez. Aqui, vamos instalar todos os pacotes que contém a palavra minic.

Isso também seleciona todas as dependências de cada pacote usando o que chama de um combinador do aptitude:

```
$sudo aptitude install "~nminic"
...
The following NEW packages will be automatically installed:
  libxfce4mcs-client3 libxfce4mcs-manager3 libxfce4util4 libxfcegui4-4 lrzsx
  xfce4-panel s
The following NEW packages will be installed:
  libxfce4mcs-client3 libxfce4mcs-manager3 libxfce4util4 libxfcegui4-4 lrzsx
  minicom xfce4-minicmd-plugin
  xfce4-panel
0 packages upgraded, 8 newly installed, 0 to remove and 0 not upgraded.
Need to get 702kB/967kB of archives. After unpacking 4645kB will be used.
Do you want to continue? [Y/n/?]
```

Lembrando que o aptitude trabalha com os mesmos repositórios e caches do APT, então caso você utilize o apt-get install ou o aptitude install estará pagando os pacotes da mesma fonte, mas o aptitude é mais completo.



COMO REMOVER
PACOTES
(SOFTWARES) COM
APTITUDE NO
UBUNTU E LIBERAR
ESPAÇO DO SEU HD?

COMO REMOVER PACOTES (SOFTWARES) COM APTITUDE NO UBUNTU E LIBERAR ESPAÇO DO SEU HD?

Remover pacotes com o aptitude é tão fácil quanto instalá-los, basta usar a opção “remove”. Mas remover um pacote muitas vezes não é o bastante para liberar espaço no seu HD.

Para remover pacotes no ubuntu:

```
$ sudo aptitude remove minicom
...
The following packages are unused and will be REMOVED:
  lrzs
The following packages will be REMOVED:
  minicom
0 packages upgraded, 0 newly installed, 2 to remove and 0 not
upgraded.
Need to get 0B of archives. After unpacking 1401kB will be free-
ed.
Do you want to continue? [Y/n/?]
```

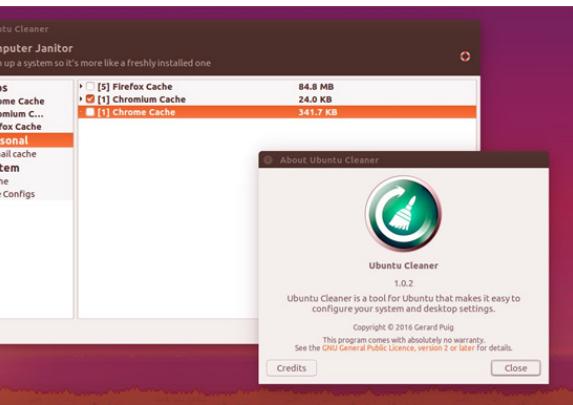
Como limpar o cachê do aptitude?

Quando instalamos pacotes com o aptitude, ele será sempre o download do arquivo .deb em cache local no diretório / var/cache/apt/archives. Com o tempo você vai querer limpar esse cache aptitude.



Se você olhar, poderá ver que existem alguns arquivos no cache:

```
$ ls /var/cache/apt/archives
...
mysql-common_5.1.30really5.0.75-0ubuntu10.2_all.deb
ndiswrapper-common_1.53-2ubuntu1_all.deb
ndiswrapper-utils-1.9_1.53-2ubuntu1_i386.deb
openjdk-6-jre_6b14-1.4.1-0ubuntu12_i386.deb
openjdk-6-jre-headless_6b14-1.4.1-0ubuntu12_i386.deb
openjdk-6-jre-lib_6b14-1.4.1-0ubuntu12_all.deb
openoffice.org-java-common_1%3a3.0.1-9ubuntu3.1_all.deb
partial
python-gnome2-extras_2.19.1-0ubuntu14_i386.deb
raptor-utils_1.4.18-2_i386.deb
rhino_1.7R1-2_all.deb
ttf-bengali-fonts_1%3a0.5.4ubuntu2_all.deb
ttf-kannada-fonts_1%3a0.5.4ubuntu2_all.deb
ttf-oriya-fonts_1%3a0.5.4ubuntu2_all.deb
ttf-telugu-fonts_1%3a0.5.4ubuntu2_all.deb
ttf-wqy-zenhei_0.8.34-cvs20081027-0ubuntu1_all.deb
tzdata_2009r~repack-0ubuntu9.04_all.deb
tzdata-java_2009r~repack-0ubuntu9.04_all.deb
unrar_1%3a3.8.5-1_i386.deb
```



Para remover estes arquivos em cache o aptitude requer usar apenas a opção clean ou autoclean:

```
$ sudo aptitude clean  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Reading extended state information  
Initializing package states... Done  
Building tag database... Done
```

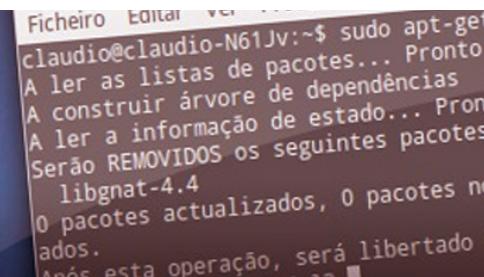
Verifique novamente o cache do aptitude:
\$ ls /var/cache/apt/archives

Vai mostrar que os pacotes foram removidos, se você tiver uma conexão lenta e demorou uma semana para baixar as últimas atualizações deve pensar duas vezes antes de usar a opção “clean” e optar em usar a opção “autoclean remove” que apenas remove os pacotes ultrapassados.

No meu caso é até besteira limpar o cache, só tenho 81 MB ocupado pelo cache do aptitude:

```
$ du -h /var/cache/apt/archives/
16K  /var/cache/apt/archives/partial
81M  /var/cache/apt/archives/
```





Ficheiro Editar View Help claudio@claudio-N61Jv:~\$ sudo apt-get update
A ler as listas de pacotes... Pronto
A construir árvore de dependências
A ler a informação de estado... Pronto
Serão REMOVIDOS os seguintes pacotes:
libgnat-4.4
0 pacotes actualizados, 0 pacotes novos.
Ades. Após esta operação, será libertado

Muitas vezes depois de atualizar uma distribuição completa, com os famosos comandos:

```
$sudo apt-get update  
$sudo apt-get upgrade  
$sudo apt-get dist-upgrade
```

É bom executar um \$ sudo aptitude clean, pra liberar espaço.

A blurred background image showing a person's hands on a computer keyboard. To the left of the hands, there is a large, semi-transparent blue circular overlay. Inside this circle, the text 'USANDO O SHELL DO UBUNTU LINUX' is centered in white capital letters.

USANDO O SHELL DO
UBUNTU LINUX

USANDO O SHELL DO UBUNTU LINUX:

O uso de um interpretador de comandos shell (normalmente chamado apenas shell) nos retorna aos primórdios dos primeiros sistemas Unix. O shell vai muito além da execução de comandos, característica tais como as variáveis de ambiente, análises e uma ampla variedade de funções para programação.

Aqui neste curso de linux, já estamos utilizando muitos comandos, agora vamos falar de shell, sem medo de ser feliz, e sem medo do modo texto.

Embora o shell mais utilizado no Linux sistemas seja o bash, temos muitos outros disponíveis como por exemplo o sh, csh ksh, bash etc.

No Ubuntu Linux, o sh é uma ligação simbólica para /bin/dash.

Shell na interface gráfica e Shell no modo texto:

A forma mais comum de acessar um shell a partir de uma interface gráfica do Linux é utilizando uma janela de terminal. A partir de uma interface gráfica, muitas vezes você pode acessar terminais virtuais para chegar a um shell. Sem a interface gráfica no modo texto, você vai cair diretamente em um shell após o login.

Usando o Shell na interface gráfica:

Para abrir uma janela do Terminal do GNOME (o desktop padrão do Ubuntu), selecione Aplicativos → Acessórios → Terminal. Isso abre um terminal gnome.

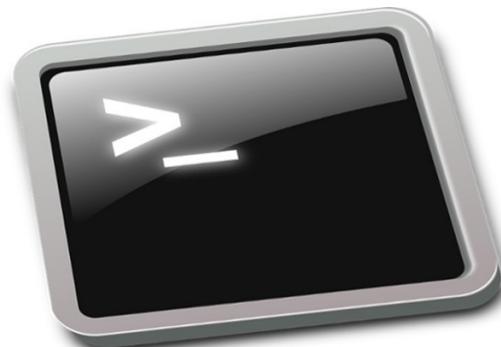
Comandos da Figura acima, ilustram que o shell atual é o bash (/bin/bash), o usuário atual é o usuário que iniciou o desktop (pedrod), e o diretório atual é o diretório home do usuário (/home/pedrod). O nome de usuário (pedrod) e host (pedrod-laptop) aparece na barra de título.

O gnome-terminal janela não só permite o acesso a um shell, como também tem a possibilidade de abrir várias sessões de terminal. Por exemplo, clique em Arquivo →Abrir nova aba, para abrir outro shell em uma guia diferente.

Você também pode usar o controle de sequências de teclas para trabalhar com uma janela de terminal.

Exemplo:

Abra um shell em uma nova guia digitando:
Ctrl + Shift + T,Abra uma nova janela de terminal com Ctrl + Shift + n,Para fechar uma guia do shell, use o Ctrl + Shift + w, e fechar uma janela de terminal com Ctrl + Shift + q.



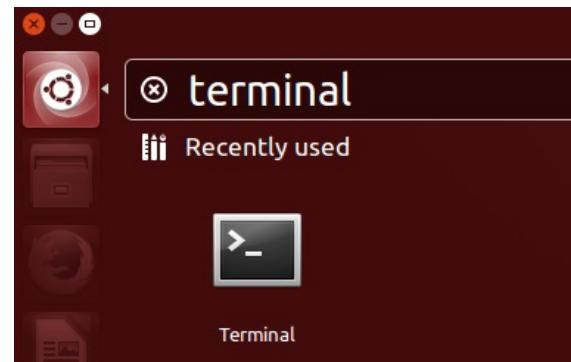
Lembrando, que algumas destas funcionalidades só estão disponíveis em um shell acessado via interface gráfica.

Além do gnome-terminal, há muitos outros terminais que podem ser utilizados na interface gráfica, como por exemplo:

-xterm (emulador de terminal de base que vem com o X Window System),

-aterm emulador (terminal emulador VT102 Afterstep XVT)

-konsole (emulador de terminal fornecido com o desktop KDE)



Usando Terminais Virtuais:

Aqui eu já vi muitas pessoas assustadas, tanto a ponto de desistirem de usar sistema Linux quando se deparam com a tela do console.

Quando usamos o runlevel 2, 3 ou 5, seis consoles virtuais (conhecido como tty1 até tty6) são criados no modo texto, já com o login em modo texto, claro.

Se um desktop X Window System está executando o X (X é o servidor de interface gráfica, isso mesmo aquela interface bonitinha do gnome é gerenciada por um servidor) provavelmente ele estará em execução no console virtual 7.

Você pode mudar para outra consola virtual com Ctrl + Alt + F1, Ctrl + Alt + F2, e assim por diante até 6, se estiver em uma interface gráfica.

Em modo texto, você pode alternar usando Alt + F1, Alt + F2, e assim por diante.



COMO FUNCIONA O AMBIENTE SHELL DO UBUNTU LINUX?

COMO FUNCIONA O AMBIENTE SHELL DO UBUNTU LINUX?

Depois de aprender a acessar um shell (a partir do modo texto ou uma janela de terminal), vamos ver aqui no curso Linux ubuntu, quais arquivos são responsáveis pela configuração do ambiente da shell bash.

As configurações do shell estão localizados em alguns arquivos. Você pode fazer suas próprias versões desses arquivos para substituir suas configurações.



Arquivos de inicialização:

O Bash executa os arquivos de inicialização que definem as configurações na hora do login.

O Bash procura os arquivos de inicialização em /etc/profile, este arquivo não precisa ter permissão execução.

Algumas distribuições do Linux armazem os de arquivos de inicialização em /etc/profile e /etc/profile.d

O Bash também procura por arquivos de inicialização em /etc/bash.bashrc , ou .bashrc para configurações personalizadas por usuário.

Quando uma sessão de usuário é fechada, são executados o comando do arquivo ~/bash_logout.

Esses arquivos são padrão para o Shell Bash. Há uma variedade de maneiras de alterar o seu ambiente shell.

Usando Bash History:

O bash é o shell usado por padrão pelos sistemas Linux mais modernos e mais um punhado de outros sistemas operacionais tais como o famoso Mac OS X, isso mesmo, o sistema da Apple também tem um Shell Bash.

O bash tem uma característica que permite rever o histórico e fazer reutilização de comandos que foram executados no passado. Isto pode ser muito útil, pois muitos comandos Linux são longos e complicados.

Quando o bash é iniciado, ele lê o arquivo `~/.bash_history` e carrega na memória o conteúdo. Este arquivo é definido pela variável de ambiente `$HISTFILE`, para verificar utilize o comando
`echo $HISTFILE`

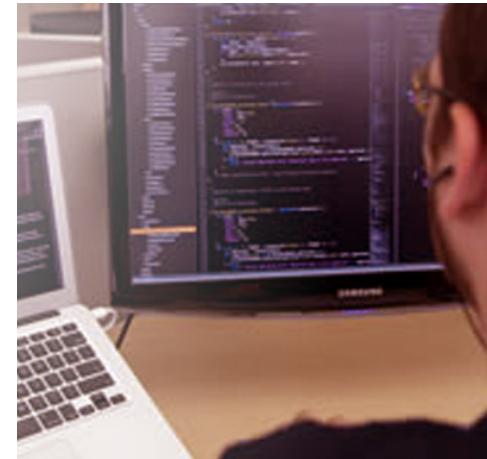
Observação: Vamos falar muito sobre variáveis de ambiente mais adiante no curso de Linux.

Durante uma sessão bash, os comandos são adicionados ao histórico ainda na memória, quando fazemos o logoff da sessão o histórico é gravado da memória de volta para o arquivo. `bash_history`.

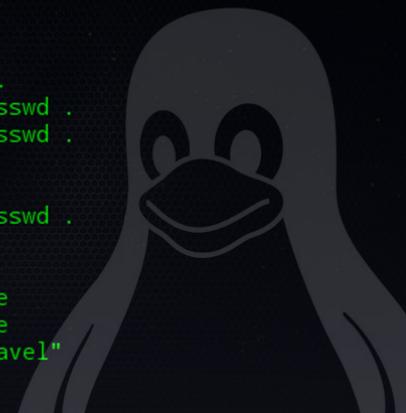
O número de comandos que o Linux pode guardar em memória está definido em `$HISTSIZE`, enquanto o número de comandos realmente armazenados no arquivo do histórico é definido por `$HISTFILESIZE`.

Utilize o comando abaixo, para verificar isso:

```
$ echo $HISTFILE $HISTSIZE $HISTFILESIZE  
/home/e-tinet/.bash_history 500 500
```



```
rabio@rabio-pc:~$ history
1 uname -a
2 echo $HISTSIZE
3 history
4 cd /
5 ls -la
6 cp passwd .
7 cp /erc/passwd .
8 cp /etc/passwd .
9 ls
10 cd ~
11 cp /etc/passwd .
12 ls -l
13 rm passwd
14 mkdir teste
15 rmdir teste
16 echo "Variavel"
17 clear
18 history
```



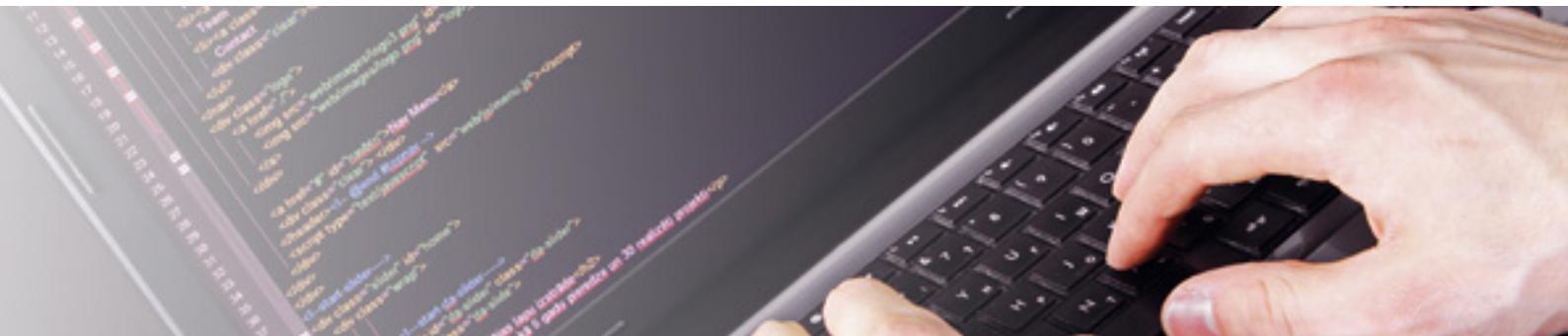
O comando `history`, ajuda a listar o conteúdo do histórico de comando do Linux:

Lista os 5 últimos comandos:

```
$ history 5
975 mkdir extras
976 mv *doc extras/
977 ls -CF
978 vi house.txt
979 history
```

SHELL DO LINUX, DICAS PARA LINHA DE COMANDO E REDIRECIONADORES





COMO FUNCIONA O AMBIENTE SHELL DO UBUNTU LINUX?

Trabalhar no Linux em modo texto é muito fácil, mas aprender alguns truques é essencial, pois muitos comandos são bastante longos. Neste Curso de Linux com Ubuntu eu quero passar também estas dicas para facilitar a vida na hora de usar o terminal do Linux.

Depois de aprender a acessar o shell via terminal virtual e também no modo gráfico, e entender um pouco mais sobre como o shell bash inicia o seu ambiente, vamos ver hoje dicas rápidas e valiosas.

A tecla TAB, e o seu poder de auto completar:

Você pode usar a tecla Tab para completar os diferentes tipos de informações na linha de comando. Aqui estão alguns exemplos onde você digita um nome parcial, seguido da tecla Tab, para tentar completar as informações que deseja na linha de comandos do shell.

```
$ cd /home/pe <Tab>
```

Depois de pressionada a tela o shell vai tentar completar o comando, no meu caso existe um diretório /home/pedro, o auto complete vai funcionar.

Outro exemplo, que tem o mesmo efeito, é usar o operador “~”

```
$ cd ~ pe<Tab>
```

Vamos acessar novamente o diretório home do usuário pedro, que fica claro em /home/pedro.

Ou pode ser utilizado em conjunto com outros comandos também.

```
$ echo $PATH <Tab>
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/
usr/games
```

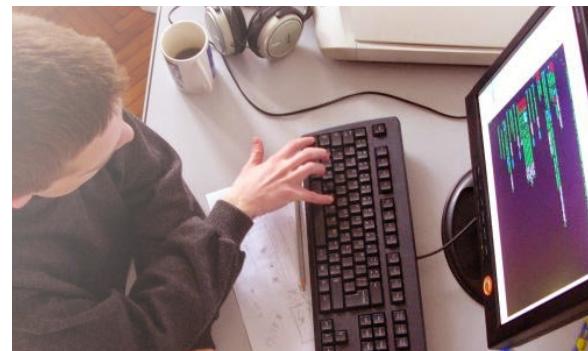
Vai retornar a variável de ambiente \$PATH, lembrando vamos falar de variável de ambiente mais pra frente.

O comando ping, que serve para enviar pacotes ICMP pela rede, pode ser utilizado com a tecla , neste caso o ping vai listar todos os hosts contidos no arquivo /etc/hosts, uma forma de facilitar e sugerir os hosts que podem ser utilizados com o comando.

```
$ ping <Tab>
e-tinet.example.com @ @ @ pedro.example.
com linux.example.com
```

Conclusão: a tecla é a minha preferida quando estou trabalhando no shell do Linux. Com o passar do tempo você acaba se acostumando a utilizar em qualquer situação.

Eu sempre falo: Se você não quiser usar a tecla tudo bem, mas vai sofrer de L.E.R antes do que eu.





The screenshot shows a web page with a title "Introdução à Programação C" and a subtitle "Exemplo de programa em C". The page contains code snippets in C, including declarations and comments. A note at the bottom states: "Um programa em C é composto por um conjunto de funções e qual o programa começa a ser executado chama-se main()."

```
"GENERATOR"
<title>Introdução à Programação C</title>

<center><big><font color="#FFB000"><b><b><strong>Introdução à Programação C</strong></b></b></font></big></big></p>
</p>
<h2>Exemplo de programa em C</h2>
<pre>// Exemplo de programa em C
uma linha de comentário
()
// declara a variável "x";
+ 2; // soma 3 com 2
>
>
<p>Um programa em C é composto por um conjunto de funções e qual o programa começa a ser executado chama-se main().</p>
```

Redirecionando saídas do shell:

Os comandos do shell são interativos, ou seja, você digita o comando e obtém o resultado na mesma hora, ou deveria ser assim.

Por exemplo, o comando ls

```
$ ls /
bin      dev      initrd.img      lost+found  opt
sbin      sys      var
boot     etc      initrd.img.old   media       proc  se-
linux    tmp      vmlinuz
cdrom    home     lib                  mnt       root  srv
usr      vmlinuz.old
```

Temos dois tipos de saídas, saída de comando normal e saída de erro, no exemplo acima o comando ls / resultou em uma saída de comando normal, mas o comando resultou em um erro?

```
$ ls /barra  
ls: impossível acessar /barra: Arquivo ou diretório inexistente
```

Sim, o comando acima resultou em um erro, pois o diretório /barra não existe.

Redirecionando a saída normal do shell:

Vamos usar o operador “ → ” para redirecionar a saída normal do shell:

```
$ ls / > log.txt
```

O comando acima não vai retornar nada no shell, pois sua saída foi redirecionada com o operador “ → ” para o arquivo log.txt

Você acabou de criar um arquivo contendo uma lista de diretórios e arquivos da raiz do seu sistema Linux. Se você estiver se perguntando: Mas onde eu utilizo este comando?

Quando estivermos trabalhando com shell script vamos poder usar estes redirecionadores para gerar arquivos de log e outras coisas.

Verifique o arquivo log.txt com o comando cat

```
$ cat log.txt
```

O comando cat vai listar o conteúdo do arquivo.

Redirecionando a saída de erro do shell:

O fluxo de erro padrão do shell pode ser redirecionado utilizando o “2→”. Aqui estão alguns exemplos:

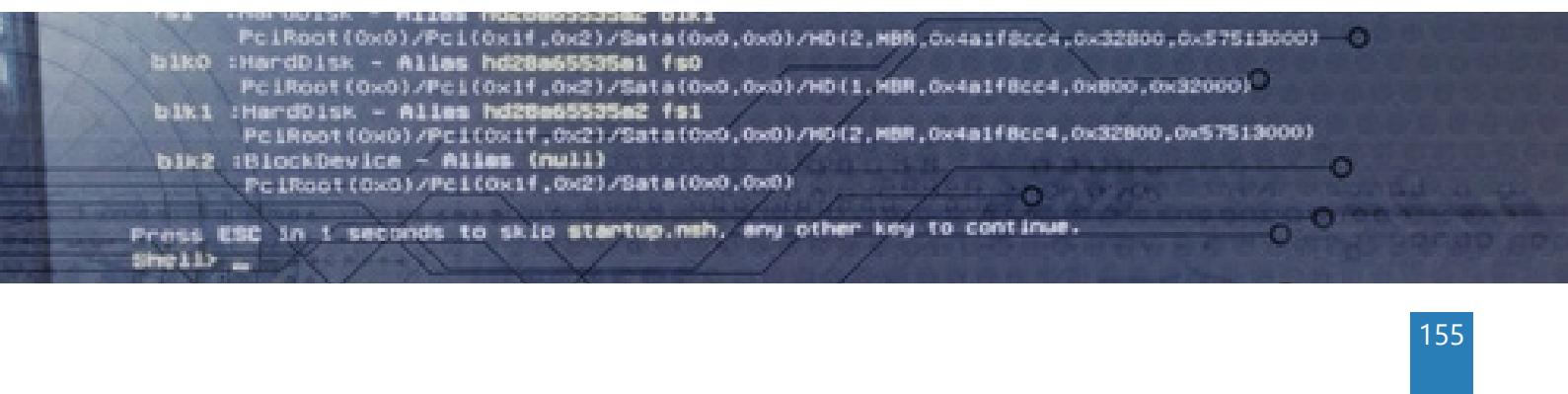
```
$ ls /barra 2> erro.txt
```

Opá! Agora o comando não listou nenhum erro na tela, sim, pois o erro foi redirecionado para o arquivo erro.txt

Use o cat novamente para verificar:

```
$ cat erro.txt
ls: impossível acessar /barra: Arquivo ou diretório inexistente
```

Agora parece besteira trabalhar com esses redirecionadores, mas pensando que você pode criar um script para executar vários comandos, será bem interessante.



```
hd0 :> PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0x4a1f8cc4,0x32800,0x57513000)
hd1K0 :> HardDisk - Alias hd2Baed5535e1 f80
PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(1,MBR,0x4a1f8cc4,0x800,0x32800)
hd1K1 :> HardDisk - Alias hd2Baed5535e2 f81
PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)/HD(2,MBR,0x4a1f8cc4,0x32800,0x57513000)
hd1K2 :> iBlockDevice - Alias (null)
PciRoot(0x0)/Pci(0x1f,0x2)/Sata(0x0,0x0)

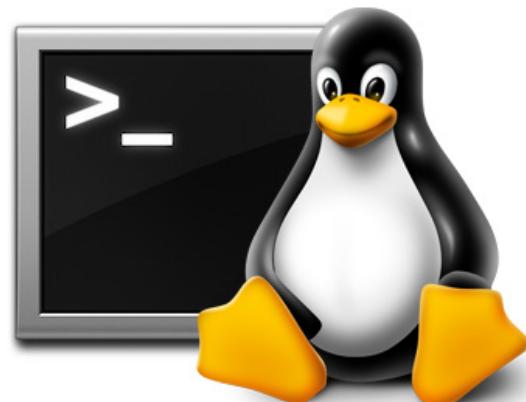
Press ESC in 7 seconds to skip startup.nsh, any other key to continue.
Shell> ~
```

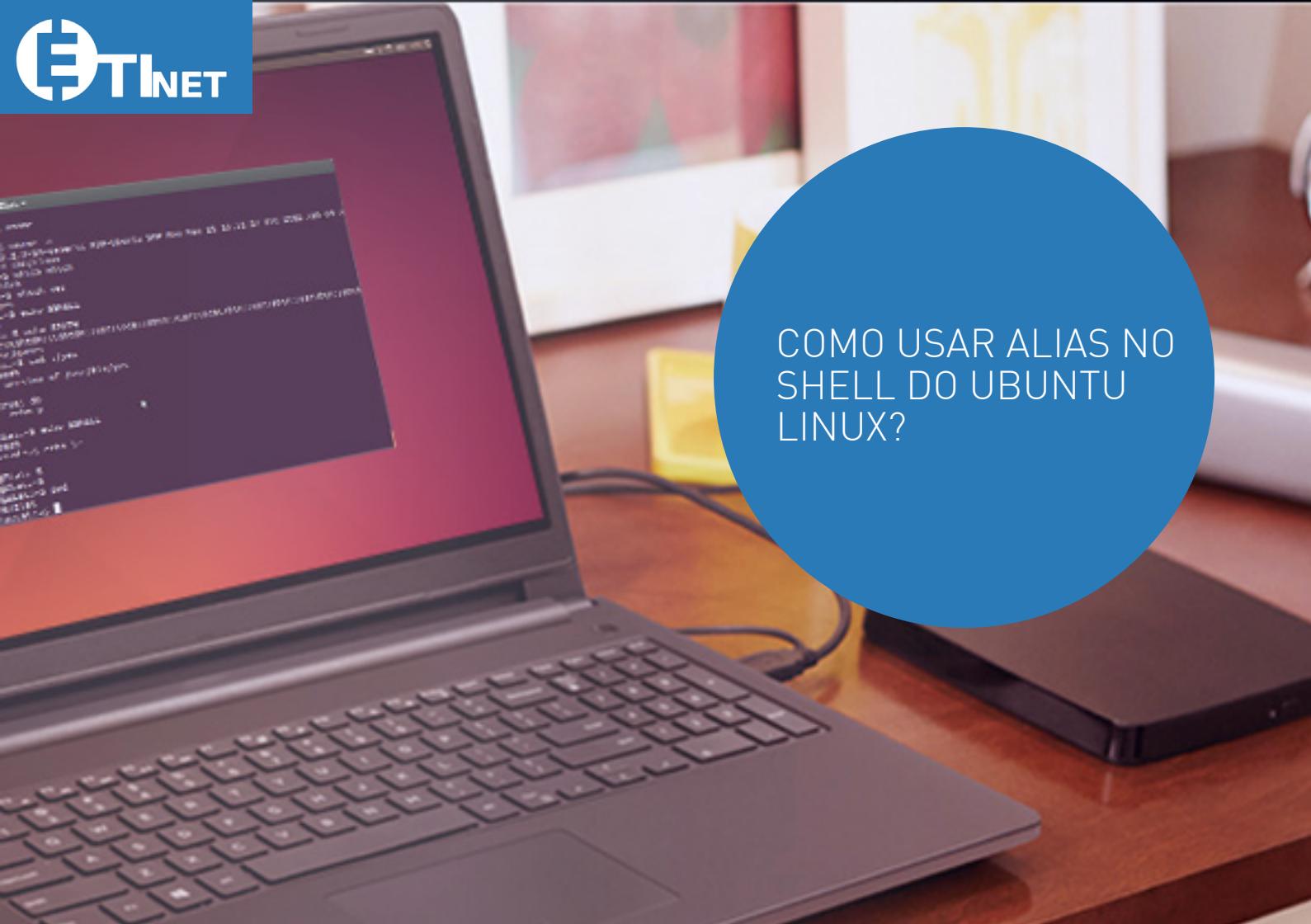
Mas, se pouco importa guardar o erro de um comando, você pode redirecionar a saída deste comando para /dev/null

```
$ ls /barra 2> /dev/null
```

Pronto, assim se o comando retornar um erro ele não será mostrado na tela e não será salvo em nenhum arquivo.

Agora é praticar um pouco no shell, lembrando que alguns comandos podem parecer sem utilidade agora, mas com certeza vamos utilizar todos para gerar scripts shell e automatizar tarefas.





COMO USAR ALIAS NO SHELL DO UBUNTU LINUX?

COMO USAR ALIAS NO SHELL DO UBUNTU LINUX?

O comando alias vai nos ajudar a definir APELIDOS para comando do sistema. Alguns APELIDOS já estão definidos ou podem ser especificados pelo usuário no arquivo de configuração de ambiente que discutimos anteriormente.

Como listar os APELIDOS do sistema Linux?

Para saber quais dos apelidos (alias) definidos no shell do Linux, vamos usar o comando alias, olha só que obvio.

```
$ alias
alias cp='cp -i'
alias ls='ls --color=auto'
alias mv='mv -i'
alias rm='rm -i'
```

Observe que alguns apelidos (aliases) são definidos simplesmente como uma forma de adicionar opções para um comando o padrão do sistema.

Por exemplo:

```
$ alias mv='mv -i'
```

Estamos criando um alias chamado mv, que vai executar o comando mv -i, assim quando o usuário executar o comando mv, será a mesma coisa que executar mv -i. A opção -i do comando mv, vai solicitar uma confirmação caso seja necessário sobreescriver algum arquivo.

O Ubuntu Linux vem apenas com o alias ls definido, alias ls='ls -color=auto' , que coloca um pouco de cor ao listar arquivos.

Como definir seus próprios aliases em uma sessão do shell bash
Podemos criar um alias na sessão atual do shell bash.

```
$ alias ls = 'ls -la'
```

Agora execute o comando ls

```
$ ls
drwxr-xr-x  21 root root  4096 2010-02-09 09:30 .
drwxr-xr-x  21 root root  4096 2010-02-09 09:30 ..
drwxr-xr-x   2 root root  4096 2010-02-09 09:29 bin
drwxr-xr-x   3 root root  4096 2010-02-09 09:30 boot
lrwxrwxrwx   1 root root    11 2009-11-04 19:03 cdrom -> media/cdrom
drwxr-xr-x  16 root root  3940 2010-02-25 19:21 dev
drwxr-xr-x 135 root root 12288 2010-02-25 19:21 etc
drwxr-xr-x   6 root root  4096 2009-04-13 06:33 home
lrwxrwxrwx   1 root root   33 2010-02-09 09:30 initrd.img -> boot/initrd.
```

```
img-2.6.28-18-generic
lrwxrwxrwx 1 root root 33 2009-12-10 09:43 initrd.img.old -> boot/initrd.img-2.6.28-17-generic
drwxr-xr-x 19 root root 12288 2010-02-09 09:29 lib
drwx----- 2 root root 16384 2009-11-04 19:03 lost+found
drwxr-xr-x 3 root root 4096 2010-02-25 19:21 media
drwxr-xr-x 2 root root 4096 2009-04-13 06:33 mnt
drwxr-xr-x 5 root root 4096 2010-01-17 00:23 opt
dr-xr-xr-x 142 root root 0 2010-02-25 19:20 proc
drwx----- 18 root root 4096 2010-02-14 19:08 root
drwxr-xr-x 2 root root 4096 2010-02-09 09:29 sbin
drwxr-xr-x 2 root root 4096 2009-03-06 13:21 selinux
drwxr-xr-x 2 root root 4096 2009-04-20 10:59 srv
drwxr-xr-x 12 root root 0 2010-02-25 19:20 sys
drwxrwxrwt 14 root root 4096 2010-02-25 19:23 tmp
drwxr-xr-x 11 root root 4096 2009-04-20 11:00 usr
drwxr-xr-x 15 root root 4096 2009-04-20 11:07 var
lrwxrwxrwx 1 root root 30 2010-02-09 09:30 vmlinuz -> boot/vmlinuz-2.6.28-18-generic
lrwxrwxrwx 1 root root 30 2009-12-10 09:43 vmlinuz.old -> boot/vmlinuz-2.6.28-17-generic
```

Como definir Alias definitivamente?

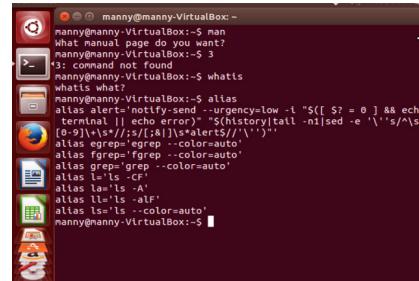
Claro que vamos precisar salvar os nossos APELIDOS (aliases) para lista adicione essa linha ao seu arquivo `~/.bashrc` para a definição ficar salva.

Vamos ver um exemplo:

```
$ vim .bashrc
```

Adicione as seguintes linhas.

```
alias cp='cp -i'  
alias ls='ls --color=auto'  
alias l='ls -laF'  
alias mv='mv -i'  
alias rm='rm -i'
```



A screenshot of a Linux terminal window titled "manny@manny-VirtualBox: ~". The window shows the command-line interface with several alias definitions. The terminal background is dark, and the text is white. The aliases include cp, ls, l, mv, and rm, each with a specific command substitution.

```
manny@manny-VirtualBox: ~  
what manual page do you want?  
manny@manny-VirtualBox: ~$ 3  
:3: command not found  
manny@manny-VirtualBox: ~$ whatis  
whatis: command not found  
manny@manny-VirtualBox: ~$ alias  
alias alert='notify-send --urgency=low -t "5([ $? = 0 ] && echo  
terminal || echo error)" "$(history|tail -n1|sed -e '\''$'/s*\n*[^ ]* /g'\';s/[;]{2}/;/g')|xterm -e'\"$*\""  
alias less='less -R --color=auto'  
alias grep='grep --color=auto'  
alias grep='grep --color=auto'  
alias l='ls -CF'  
alias ls='ls -A'  
alias l='ls -lF'  
alias ls='ls --color=auto'  
manny@manny-VirtualBox: ~
```

```
# .bashrc

alias sqlplus='rlwrap sqlplus'
alias rman='rlwrap rman'

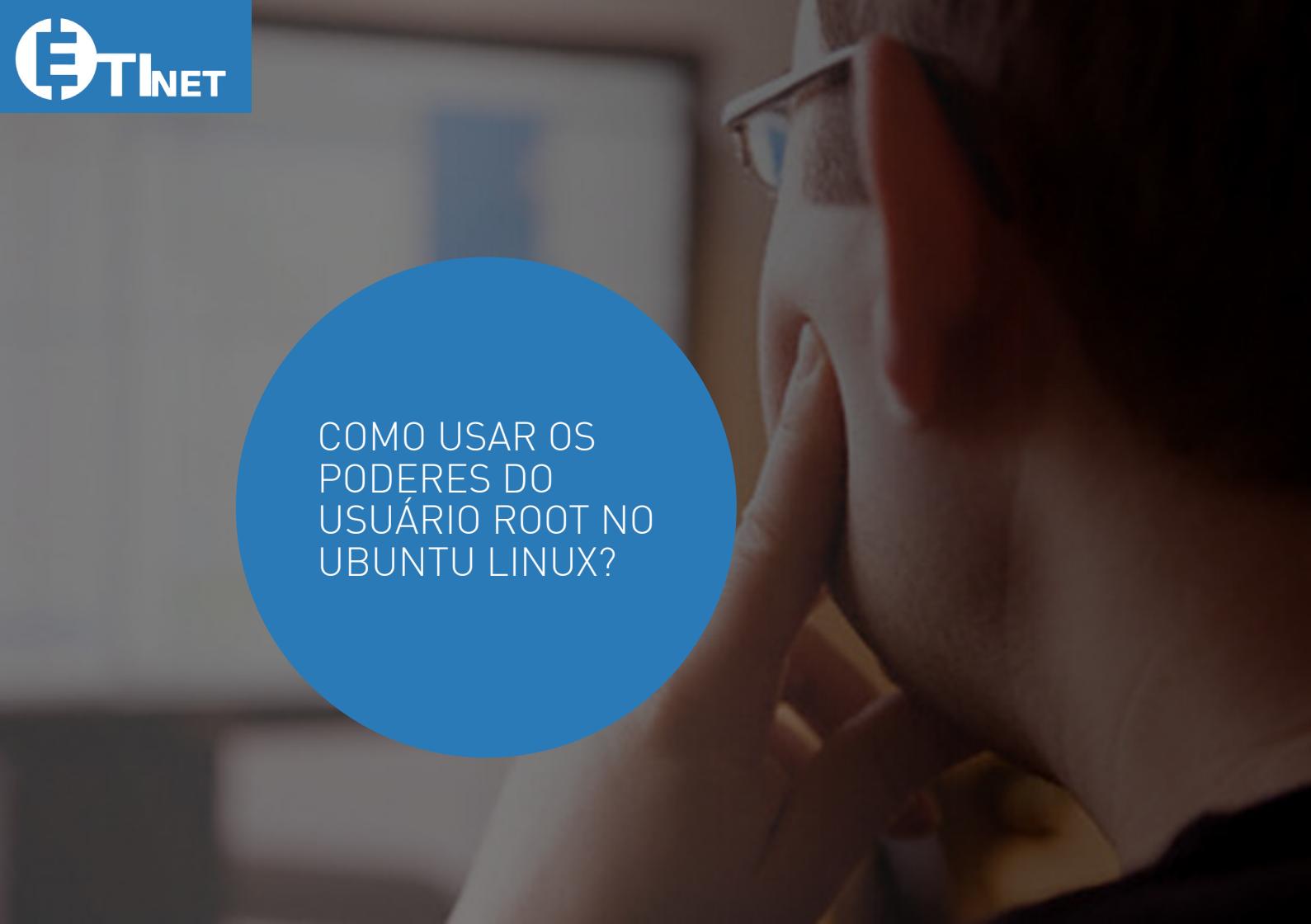
# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions
~
~
~
```

O arquivo `~/.bashrc` é mais um arquivo responsável pela configuração do ambiente shell, como já tratamos no post como funciona o ambiente Shell do Linux ? Toda vez que uma sessão do shell for iniciada, este arquivo vai ser executado, mesmo não tendo permissão de execução. (OBS: Vamos falar bastante sobre permissão de arquivos no Linux)

Como já estamos em um sessão do shell, podemos executar o comando abaixo, para que o bash leia novamente o `~/.bashrc`.

```
$ source ~/.bashrc
```

A dark, blurred background image showing a person from behind, wearing a light-colored shirt, sitting at a desk and working on a laptop computer.

COMO USAR OS PODERES DO USUÁRIO ROOT NO UBUNTU LINUX?

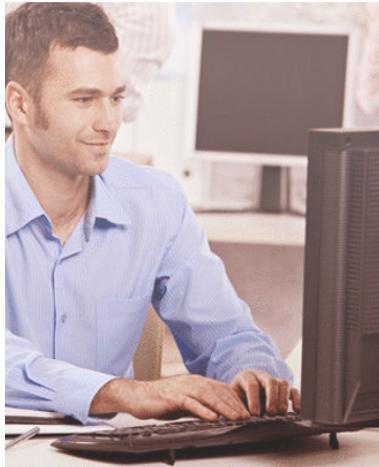
COMO USAR OS PODERES DO USUÁRIO ROOT NO UBUNTU LINUX?

Quando trabalhamos com Shell no Ubuntu Linux sempre surge uma dúvida:
Como executar comando que somente o usuário root tem permissão?

O comando sudo permite que um usuário comum no sistema execute comandos com o poder do usuário root. O sudo é uma grande ferramenta para a concessão de privilégios específicos.

O sudo é executado como root. O Ubuntu usa o comando sudo para executar comandos privilegiados, ao invés de usar o comando su .

O comando sudo é configurado em /etc/sudoers



ATENÇÃO!

Nunca editar este arquivo com seu editor de texto normal. Em vez disso, use sempre o comando visudo.

O arquivo /etc/sudoers é restrito, então você precisa usar o comando sudo para editar o arquivo. Por exemplo:

```
$ sudo visudo
```

O comando visudo inicia um editor por padrão, o editor nano. Se você olhar para o arquivo sudoers que acompanha a sua distribuição, você verá seções vazias delimitadas por comentários e uma sessão conforme abaixo:

```
ALL root = (ALL) ALL
```

Isso significa que o usuário root é permitido em todos os hosts, executar qualquer comando como qualquer usuário.

Para que todos os usuários que fazem parte do grupo admin possam adquirir privilégios de root:

```
% admin ALL = (ALL) ALL
```

No meu caso o parâmetro acima estava na última linha.

Quando você instalou o Ubuntu Linux, a conta de usuário que você criou foi automaticamente adicionado a este grupo, o admin. Você pode verificar isso com o comando.

```
$ id user1
uid=1000(user1) gid=1000(pedrod) grupos=1000(user1),4(adm),20(dialout),24(cdrom),46(plugdev),106(lpadmin),121(admin),122(sambashare)
```

Tudo bem, então a concluímos: Todo usuário que fizer parte do grupo admin, pode executar qualquer comando como usuário root usando o comando sudo. Muito cuidado ao executar o comando abaixo:

```
$ sudo addgroup user1 admin  
Adicionando o usuário `user1' ao grupo `admin' ...  
Adicionando usuário user1 ao grupo admin  
Concluído.
```

Agora o usuário chamado user1 poderá executar comandos com os poderes de root no sistema. Por exemplo:

```
$ sudo apt-get install nmap
```

Ou qualquer comando que execute ou delete arquivos, poderá também iniciar serviços do sistema, como por exemplo:

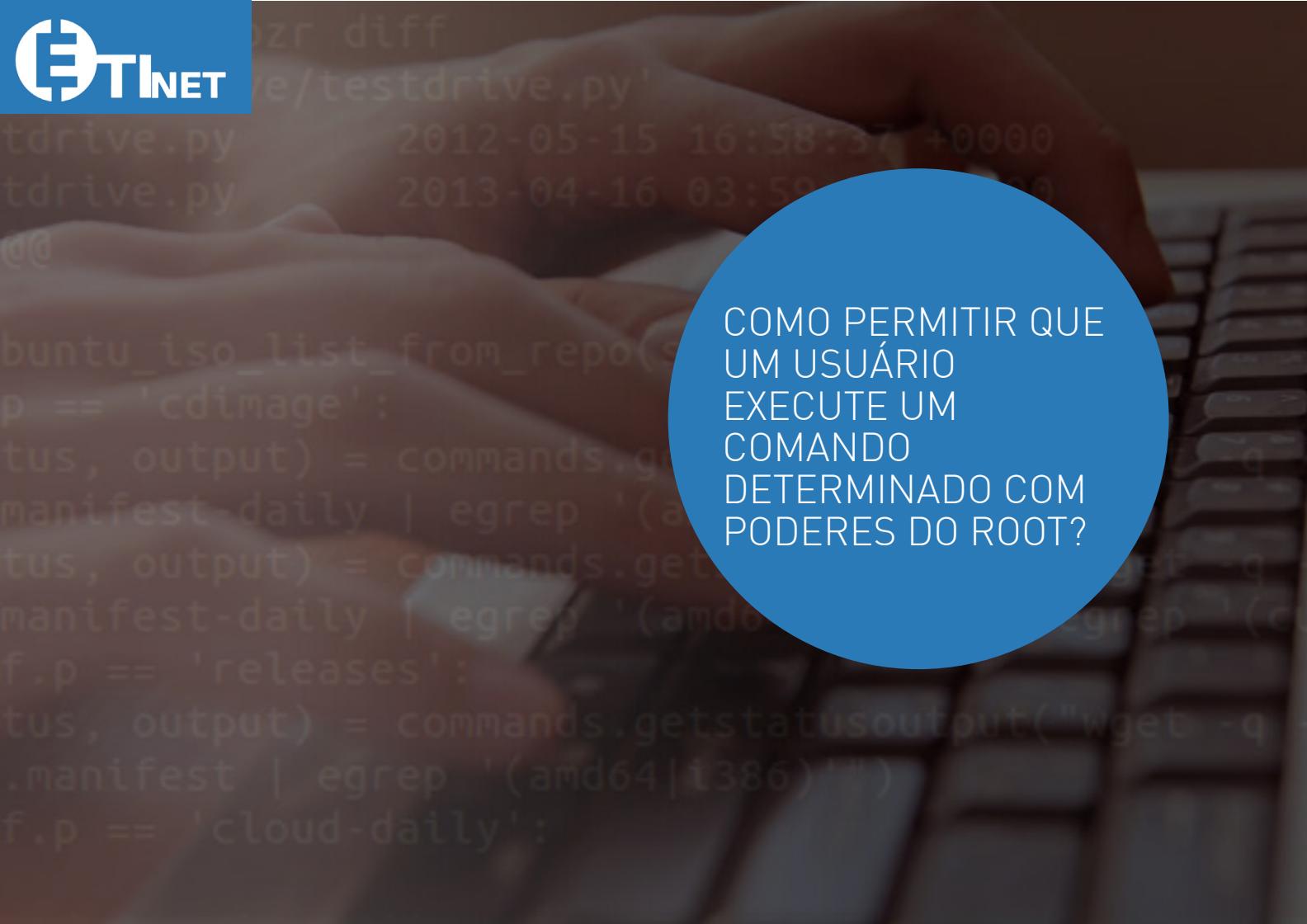
```
$ sudo /etc/init.d/networking restart
```

Ou seja, quem estiver no grupo admin, não é o root, mas é o cara no sistema.

Nós já usamos muito o comando sudo aqui no Curso Linux Ubuntu, eu já tinha avisado que na hora certa nós iríamos falar sobre o famoso sudo. Acabamos de discutir o básico sobre o uso e configuração deste comando.



```
tester@LinuxTST:~$ sudo add-apt-repository ppa:wine/wine-stable
You are about to add the following PPA:
  Welcome to the Wine Team PPA. Here you can find packages for every supported version of Ubuntu.
  This PPA was created by Maarten Lankhorst.
  More info: https://launchpad.net/~ubuntu-wine/+archive/stable
Press [ENTER] to continue or ctrl-c to cancel...
gpg: keyring `/tmp/tmpr3r98a/secring.gpg' created
gpg: keyring `/tmp/tmpr3r98a/pubring.gpg' created
gpg: requesting key F9CB8DB0 from hkp://keyserver.ubuntu.com:11371
gpg: /tmp/tmpr3r98a/trustdb.gpg: trustdb created
gpg: key F9CB8DB0: public key "Launchpad PPA <ppa:wine/wine-stable@lists.launchpad.net>"
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:                               imported: 1  (RSA: 1)
OK
```



COMO PERMITIR QUE
UM USUÁRIO
EXECUTE UM
COMANDO
DETERMINADO COM
PODERES DO ROOT?

COMO PERMITIR QUE UM USUÁRIO EXECUTE UM COMANDO DETERMINADO COM PODERES DO ROOT?

Algum dia você poderá precisar de um comando Linux que executado por um usuário comum do sistema, mas este comando somente o usuário root tem permissão. O SUDO revolve este pequeno problema.

No POST Como usar os poderes do usuário root no Ubuntu Linux da série Curso Linux Ubuntu, configuramos o SUDO através do comando visudo, de forma que um usuário execute comando com os poderes do ROOT, mas isso pode ser um problema, pois qualquer comando poderá ser executado.



Como configurar o SUDO?

No meu exemplo, vamos utilizar o usuário user1, este usuário não esta no grupo admin, para verificar que isso execute.

```
$ id user1  
uid=1001(user1) gid=1001(user1) grupos=1001(user1)
```

Agora vamos editar as configurações do SUDO:

```
$ sudo visudo
```

Vamos adicionar a linha abaixo, no final do arquivo:

```
user1 ALL= /usr/bin/less /var/log/messages
```

Pronto, com isso estamos dizendo para o SUDO que o comando /usr/bin/less /var/log/messages pode ser executado pelo usuário user1 .

Para testar, podemos nos transformar no usuário user1 com o comando abaixo:

```
$ su user1  
user1@laptop:~$
```

Agora podemos usar o comando less para testar:

```
user1@laptop:~$ /usr/bin/less /var/log/messages  
/var/log/messages: Permissão negada
```

Ok, permissão negada, mas se utilizar o sudo antes do comando, o sistema vai permitir que o comando seja executado.

```
user1@laptop:~$ sudo /usr/bin/less /var/log/messages
```

Outro teste interessante é tentar executar outro comando com o usuário user1.

```
user1@laptop:~$ sudo apt-get install nmap  
[sudo] password for user1:  
Sorry, user user1 is not allowed to execute '/usr/bin/apt-get  
install nmap' as root on laptop.
```



O comando apt-get install não pode ser executado pelo usuário user1.

O SUDO é uma solução muito boa para controlar os poderes de usuários no sistema, podemos configurar usuários com poderes de root, ou apenas liberar um comando específico para o usuário executar. Assim podemos controlar mais os usuário e suas permissões.

TRABALHANDO COM ARQUIVOS NO LINUX



TRABALHANDO COM ARQUIVOS NO LINUX:

Se você está querendo aprender a administrar sistemas Linux pode ter certeza, você vai trabalhar com arquivos, e tudo isso no modo texto. Sim, isso mesmo, no modo console, sem mouse, sem arrastar e soltar arquivos.

Estou começando esta série do curso Linux Ubuntu para trabalhar com arquivo, talvez esta seja a parte mais esperada deste curso Linux. Os demais posts deste curso, são especialmente para quem pretende trabalhar com o Ubuntu Linux, mas claro pode ser facilmente adaptado em outras distribuições.

Voltando ao assunto...

Tudo em um sistema de arquivos Linux pode ser visto como um arquivo. Isso inclui arquivos de dados, diretórios, dispositivos, pipes, links e outros tipos de arquivos.

Associado a cada arquivo é um conjunto de informação que determina quem pode acessar o arquivo e como eles podem acessá-lo. Será uma série de posts que abrange muitos comandos para explorar e trabalhar com arquivos.



Noções básicas sobre tipos de arquivos:

Há vários outros tipos de arquivos que você irá encontrar quando usar o Linux. Na linha de comando, existem muitas maneiras que você pode criar, localizar e listar diferentes tipos de arquivos.

Arquivos que fornecem acesso aos componentes de hardware no seu computador são referidos como arquivos do dispositivo. Há caracteres e dispositivos de bloco. Existem hard links e soft links, você pode usar para fazer o mesmo arquivo acessível a partir de locais diferentes (Pausa para pensar: Os softs links são muitos, mas muitos utilizados).

Os arquivos pipes e sockets são menos utilizados diretamente pelos usuários que fornecem pontos de acesso para os processos de comunicação uns com os outros.



Como descobrir um tipo de arquivo?

Você pode determinar o tipo de um arquivo usando o comando file. Eu sempre falo para meus alunos que no Linux, os arquivos não precisariam ter a extensão, como por arquivo.txt, claro fica muito mais fácil visualmente, porém para o Linux não faz diferença nenhuma.

Vamos fazer um exemplo, simples:

```
cd /usr/share/doc/  
  
$ file doc-base/install-docs.html  
doc-base/install-docs.html: XML 1.0 document text  
  
$ file doc-base/copyright  
doc-base/copyright: ASCII English text  
  
$ file doc-base/doc-base.html  
doc-base/doc-base.html/: directory  
  
$ file doc/doc-base/changelog.gz  
doc-base/changelog.gz: gzip compressed data, was "changelog", from  
Unix, last  
modified: Thu Feb 22 07:29:26 2007, max compression  
  
$ file shared-mime-info/shared-mime-info-spec.pdf  
shared-mime-info/shared-mime-info-spec.pdf: PDF document, version 1.4
```

O comando file foi executado para mostrar os arquivos da documentação do Ubuntu, em diferentes formatos. É simples determinar qual arquivo contém o texto, qual foi comprimido, PDF ou PostScript, ou HTML. Existe ainda um subdiretório, um nome estranho para um diretório (doc-base.html), mas tudo bem.

Criação de arquivos pode ser feita por qualquer aplicação que pode salvar seus dados. Se você apenas precisa criar alguns arquivos em branco para começar, há muitas maneiras de fazer isso. Aqui estão dois exemplos:

```
$ touch /tmp/meuarquivo.txt  
$ > /tmp/meuarquivo2.txt
```



E-TINET é um projeto pessoal de Pedro Delfino, profissional com mais de 14 anos de experiência em sistemas Linux. A E-TINET tem como objetivo treinar e capacitar os profissionais de tecnologia a trabalharem com o Linux profissionalmente.

[Veja aqui](#) como começar uma formação Linux profissional e domine, de uma vez por todas, esse sistema tão importante para a sua carreira.