

Aula 02 - Linux

Docupedia Export

Author:Goncalves Donathan (SO/OPM-TS21-BR)

Date:06-May-2024 17:56

Table of Contents

| | | |
|----------|--|-----------|
| 1 | GERENCIAMENTO DE PACOTES NO LINUX | 3 |
| 2 | GERENCIAMENTO DE APLICATIVOS | 7 |
| 3 | CRIAÇÃO E REMOÇÃO DE USUÁRIOS | 9 |
| 4 | GRUPOS NO LINUX | 10 |
| 5 | Instalando o VSCode pelo terminal | 11 |
| 5.1 | Processos no Linux | 15 |

1

GERENCIAMENTO DE PACOTES NO LINUX

São repositórios que disponibilizam pacotes para serem instalados pelo usuário.
Pacotes são qualquer tipo de programa ou ferramenta que possa ser utilizado por um sistema.



Package Management

Exemplos de Gerenciadores de Pacotes já trabalhados:
pip do Python, npm para web



Um software/tecnologia não se restringe a apenas um gerenciador, pois cada um pode possuir diferentes pacotes ou diferentes maneiras de gerenciar/installar um pacote em comum.

Por exemplo: foi usado o **npm**, mas também existe o **yarn** que poderia ser utilizado.

Tudo pode depender das preferências do programador, pois sempre existem suas vantagens e desvantagens.

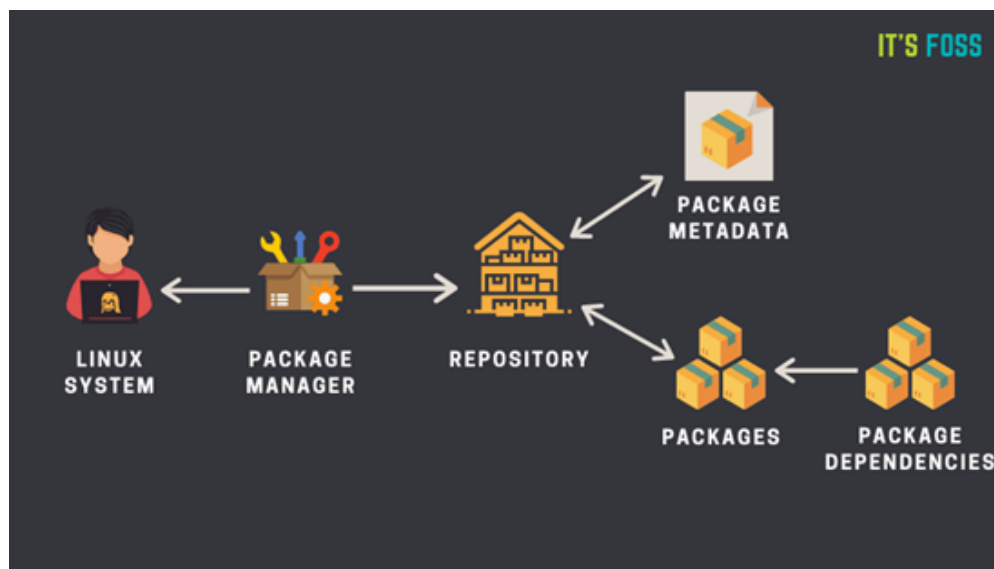


Quando se trata do Linux, existem vários gerenciadores diferentes, mas a princípio iremos trabalhar apenas com dois deles.

APT-GET e o SNAPPY



Funcionamento do gerenciador:



O Package Manager será chamado pelo sistema operacional através de comando utilizados pelo usuário.

O software irá consultar o repositório em algum servidor online.

Irá conferir todos os metadados do que deverá ser instalado e como deverá ser instalado, além de conferir se o software irá depender de algum outro pacote.

APT e SNAPPY no UBUNTU

Por padrão, os dois gerenciadores já vêm instalado na distribuição Ubuntu.

APT-GET é um pouco mais antigo, nasceu com a premissa de ser utilizado nas distribuições baseadas do Debian.

SNAPPY é mais recente, onde foi desenvolvido e mantido pela Canonical, a criadora do Ubuntu.

Observação: Muitos usuários preferem a utilização de apenas um ou dois gerenciadores, afinal, manter vários no sistema irá ocupar espaço de armazenamento.

2

GERENCIAMENTO DE APLICATIVOS

Trabalhar com pacotes/aplicativos que disponibilizam novos recursos ao nosso sistema.
Esse gerenciamento inclui: instalar, remover e atualizar.

Para baixar os pacotes mais recentes, precisa atualizar primeiro o gerenciador de pacotes:

apt-get update

```
donathan@donathan-VirtualBox:~$ apt-get update
Reading package lists... Done
E: Could not open lock file /var/lib/apt/lists/lock - open (13: Permission denied)
E: Unable to lock directory /var/lib/apt/lists/
W: Problem unlinking the file /var/cache/apt/pkgcache.bin - RemoveCaches (13: Permission denied)
W: Problem unlinking the file /var/cache/apt/srcpkgcache.bin - RemoveCaches (13: Permission denied)
donathan@donathan-VirtualBox:~$
```

Mas ao executar, haverá um erro de permissão do usuário.

Assim como no Windows, o Linux também possui um sistema de acesso administrativo.

Para isso, basta utilizar o comando **SUDO**.

Obs: Ao digitar, a senha não irá aparecer para manter a segurança, inclusive da quantidade de dígitos.

```
donathan@donathan-VirtualBox:~$ sudo apt-get update
[sudo] password for donathan:
Ign:1 http://archive.ubuntu.com/ubuntu noble InRelease
Ign:2 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Ign:4 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Ign:2 http://security.ubuntu.com/ubuntu noble-security InRelease
Ign:1 http://archive.ubuntu.com/ubuntu noble InRelease
Ign:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Ign:4 http://archive.ubuntu.com/ubuntu noble-backports InRelease
```

USUÁRIO ROOT

Quando se executa o **sudo**, estamos dando uma permissão como usuário root.
O comando sudo pode ser bloqueado de acordo com o usuário que está utilizando.

sudo passwd root (Alterar a senha do root).

Digitar a senha do root duas vezes (Ele pede que seja maior que 8 caracteres)

Comando **su** (Alterar para o usuário root).

Comando **exit** ou **Ctrl + D** (Voltar ao usuário padrão).

sudo apt-get install tree

Aplicação simples que mostra de maneira organizada os diretórios e os subdiretórios
Podendo executar o comando **tree** ou por exemplo **tree /etc** (pasta com arquivos de configuração).

sudo apt-get purge tree (deletando pacotes)

sudo apt-get dist-upgrade (atualização do ubuntu)

Esse comando pode ser utilizado para não precisar estar baixando um novo arquivo ISO para atualização do Ubuntu.

3

CRIAÇÃO E REMOÇÃO DE USUÁRIOS

Usuários de login, assim como estamos utilizando nas aulas

Código para criação de um novo usuário

Com isso ele cria um novo usuário, um novo grupo e uma nova pasta.

```
sudo adduser daniel
```

Através do código "**ls ..**" na pasta raiz (~), é possível ver que existe uma nova pasta criada.

Digitando o seguinte comando, é possível **verificar todos os usuários** que existem no sistema.

```
getent passwd
```

Todos esses usuários que aparecer são criados para as finalidades do sistema, além de aparecer os dois usuários criados por nós.

Além de poder deslogar para entrar no novo login.

Remover um usuário

```
sudo userdel --remove daniel
```

4

GRUPOS NO LINUX

Os grupos conseguem gerenciar as permissões que os usuário possuem.
Cada grupo pode possuir inúmeros usuários.

Verificar todos os grupos do sistema.

```
getent group
```

Criação de um novo grupo.

```
sudo groupadd devs
```

Deletando um grupo.

```
sudo groupdel devs
```

Definindo usuário em um grupo.

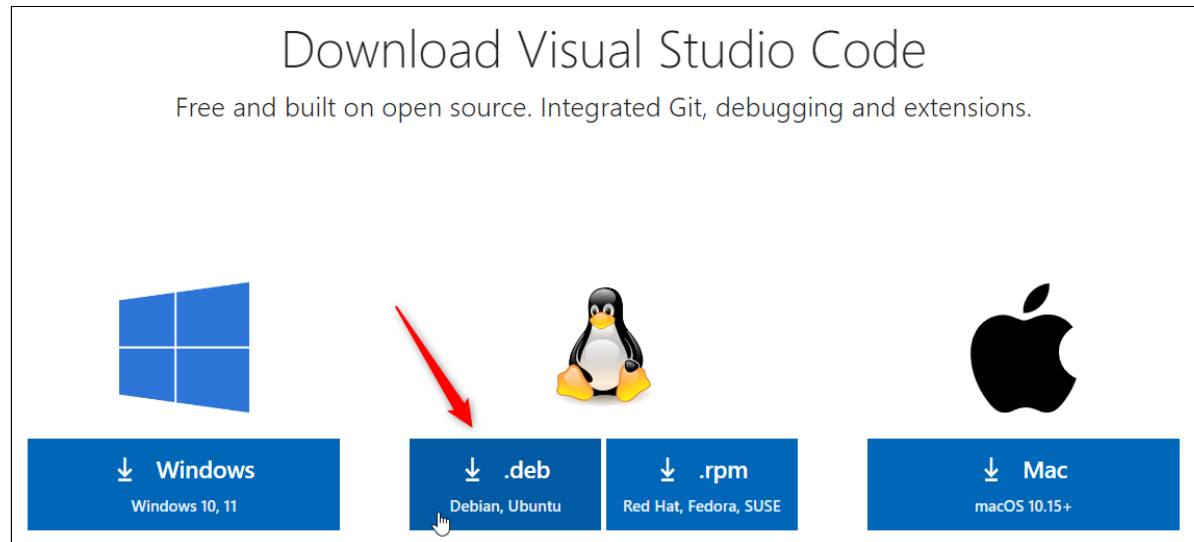
```
sudo usermod -a -G devs daniel
```

5

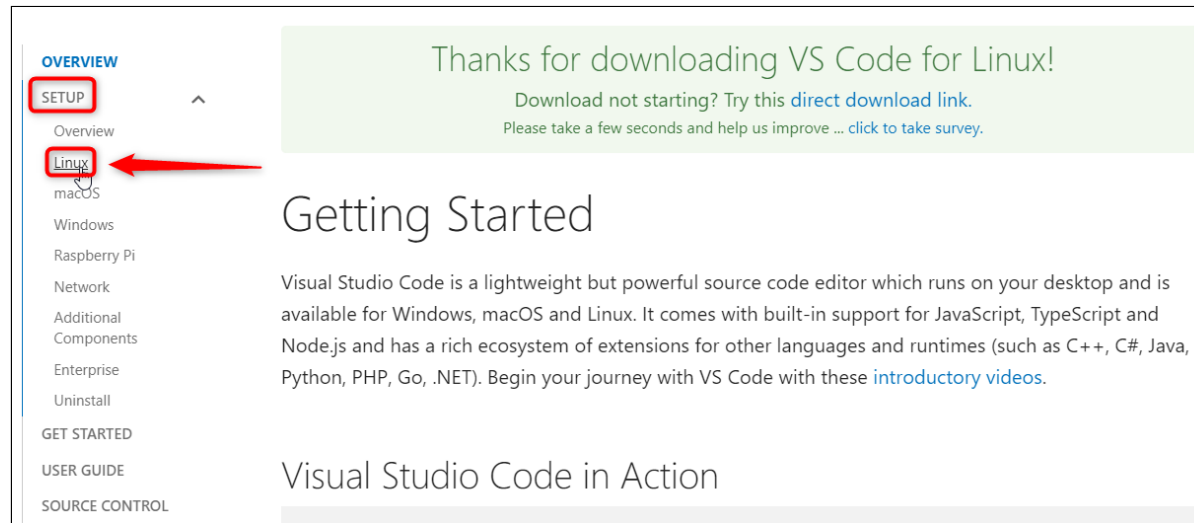
Instalando o VSCode pelo terminal

O primeiro passo é pesquisar no próprio site ou pelo google a melhor maneira de fazer isso.

Download normal no site oficial.



Instruções para instalação no Linux.



Comando para instalar o Linux pelo terminal após baixar arquivo .deb

Visual Studio Code on Linux

[Edit](#)

Installation

See the [Download Visual Studio Code](#) page for a complete list of available installation options.

By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy statement](#).

Debian and Ubuntu based distributions

The easiest way to install Visual Studio Code for Debian/Ubuntu based distributions is to download and install the [.deb package \(64-bit\)](#), either through the graphical software center if it's available, or through the command line with:

```
sudo apt install ./<file>.deb
```

```
# If you're on an older Linux distribution, you will need to run this instead:
```

```
# sudo dpkg -i <file>.deb
```

```
# sudo apt-get install -f # Install dependencies
```

SNAP

Pelo snap é mais simplificado essa instalação, porque ele tenta cuidar das complexidades internamente, então basta digitar o seguinte código no terminal.

```
sudo snap install --classic code
```

Estrutura de instalação:

--classic é utilizado para permitir que o VSCode acesse e modifique o sistema de arquivos local.

code é como o VSCode é chamado.

REPOSITÓRIOS OFICIAIS ou REPOSITÓRIOS DE TERCEIROS.

Repositório oficial: Vem configurado com a instalação do SO (Sistema operacional).

Repositórios de terceiros: São adicionados pelos usuários, isso é comum para softwares que requerem atualizações mais frequentes ou versões mais recentes que as disponíveis nos repositórios oficiais.

Obs: Para a instalação do VSCode pelo apt, é necessário adicionar um repositório de terceiro.

APT-GET

Para instalar pelo apt, será um pouco mais manual, sendo mais detalhado na hora da instalação. Faremos através dos seguintes comandos:

Comando utilizado para garantir que todos as dependências, segurança, versão esteja tudo atualizado.

```
sudo apt update
```

Instalando as dependências necessárias, ou seja, ferramentas que o VSCode irá precisar para funcionar normalmente.

```
sudo apt install software-properties-common apt-transport-https wget
```

Importando a chave GPG da Microsoft.

O GPG (GNU Privacy Guard) é usado para assinar digitalmente os repositórios e pacotes, ou seja, verificar a integridade e a autenticidade dos pacotes baixados, garantindo que eles são seguros para instalação.

```
wget -q https://packages.microsoft.com/keys/microsoft.asc -O- | sudo apt-key add -
```

Adicionando o repositório do VSCode, pois virá de um repositório terceiro.

```
sudo add-apt-repository "deb [arch=amd64] https://packages.microsoft.com/repos/vscode stable main"
```

Atualizando novamente o cache do pacote

```
sudo apt update
```

Instalando o VSCode

```
sudo apt install code
```

5.1

Processos no Linux

Funciona como se fosse o gerenciador de tarefas do Windows.

Mostra todos os processos rodando no SO (Sistema Operacional).

```
top
```

```
top - 23:17:30 up 32 min,  1 user,  load average: 0,22, 0,14, 0,10
Tasks: 242 total,  1 running, 196 sleeping,  0 stopped,  0 zombie
%Cpu(s):  0,3 us,  0,1 sy,  0,0 ni, 99,5 id,  0,1 wa,  0,0 hi,  0,0 si,  0,0 st
KiB Mem : 4956248 total, 2337340 free, 1248832 used, 1370076 buff/cache
KiB Swap: 483800 total, 483800 free,  0 used. 3411132 avail Mem
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ | COMMAND |
|------|---------|----|----|---------|--------|--------|---|------|------|---------|-----------------|
| 2071 | matheus | 20 | 0 | 3754940 | 275180 | 108088 | S | 0,9 | 5,6 | 0:44.67 | gnome-shell |
| 3234 | matheus | 20 | 0 | 49024 | 3928 | 3184 | R | 0,6 | 0,1 | 0:00.56 | top |
| 7 | root | 20 | 0 | 0 | 0 | 0 | I | 0,3 | 0,0 | 0:00.11 | kworker/u8:0-ev |
| 1870 | matheus | 20 | 0 | 696448 | 95232 | 53236 | S | 0,3 | 1,9 | 0:41.06 | Xorg |
| 2422 | matheus | 20 | 0 | 803952 | 38008 | 28324 | S | 0,3 | 0,8 | 0:07.53 | gnome-terminal- |
| 3370 | matheus | 20 | 0 | 2592124 | 103400 | 79928 | S | 0,3 | 2,1 | 0:01.09 | WebExtensions |
| 1 | root | 20 | 0 | 225480 | 9128 | 6636 | S | 0,0 | 0,2 | 0:04.16 | systemd |
| 2 | root | 20 | 0 | 0 | 0 | 0 | S | 0,0 | 0,0 | 0:00.01 | kthreadd |

É possível instalar alguns pacotes que mostram um pouco diferentes essa visualização.

```
sudo apt-get install htop
```

```
htop
```



```

 1 [||||||| 18.7%] Tasks: 148, 462 thr; 3 running
 2 [||||||| 17.5%] Load average: 0.21 0.16 0.11
 3 [ 0.0%] Uptime: 00:34:00
 4 [ 0.6%]
Mem[||||||| 1.23G/4.73G]
Swp[ 0K/472M]

```

| PID | USER | PRI | NI | VIRT | RES | SHR | S | CPU% | MEM% | TIME+ | Command |
|------|---------|-----|----|-------|-------|-------|---|------|------|---------|--|
| 2071 | matheus | 20 | 0 | 3808M | 269M | 105M | S | 20.3 | 5.6 | 0:51.20 | /usr/bin/gnome-shell |
| 1870 | matheus | 20 | 0 | 680M | 95496 | 53236 | R | 16.3 | 1.9 | 0:43.87 | /usr/lib/xorg/Xorg vt2 -displayfd 3 -aut |
| 2422 | matheus | 20 | 0 | 785M | 38008 | 28324 | S | 8.1 | 0.8 | 0:09.45 | /usr/lib/gnome-terminal/gnome-terminal-s |
| 2080 | matheus | 20 | 0 | 3808M | 269M | 105M | S | 4.1 | 5.6 | 0:02.58 | /usr/bin/gnome-shell |
| 1879 | matheus | 20 | 0 | 680M | 95496 | 53236 | S | 2.7 | 1.9 | 0:02.53 | /usr/lib/xorg/Xorg vt2 -displayfd 3 -aut |
| 4163 | matheus | 20 | 0 | 34404 | 4484 | 3492 | R | 2.0 | 0.1 | 0:01.38 | htop |
| 2077 | matheus | 20 | 0 | 3808M | 269M | 105M | S | 2.0 | 5.6 | 0:02.59 | /usr/bin/gnome-shell |
| 2078 | matheus | 20 | 0 | 3808M | 269M | 105M | S | 1.4 | 5.6 | 0:02.27 | /usr/bin/gnome-shell |
| 2079 | matheus | 20 | 0 | 3808M | 269M | 105M | S | 1.4 | 5.6 | 0:02.47 | /usr/bin/gnome-shell |
| 2012 | matheus | 20 | 0 | 123M | 2248 | 1880 | S | 0.7 | 0.0 | 0:02.26 | /usr/bin/VBoxClient --draganddrop |
| 2006 | matheus | 20 | 0 | 123M | 2248 | 1880 | S | 0.7 | 0.0 | 0:02.26 | /usr/bin/VBoxClient --draganddrop |

Para encerrar o processo de algum software, é possível fazer de duas maneiras.

```
kill [número do processo]
```

```
killall [nome do processo]
```