

# GERENCIAMENTO DE PACOTES COM DPKG

Nesse documento, vou tentar passar um pouco da minha experiência com essa ferramenta. Se você desconhece com um pacote debiana funciona, sugiro a leitura <http://goo.gl/OsIaPN>

Instalação de um pacote .deb

Opção “-i” para instalação de um pacote .deb.

No mesmo diretório do pacote, rode o comando:

```
dpkg -i pacote.deb
```

Basicamente o pacote e seus arquivos serão extraídos, e os script de pós e pré instalação, fará configurações adicionais. **ATENÇÃO** o dpkg não resolve dependência, ou seja, se o pacote que você esta instalando depender de outro pacote para funcionar, ele vai emitir um aviso que o programa depende de A, B, ou C dependências geralmente, em um sistema bem configurado, o comando “**apt-get install -f**” resolve. **ATENÇÃO** nunca misture pacotes das versões **testing, unstable** e principalmente nunca instale **PPA** do Ubuntu no Debian, isso deixará seu sistema inutilizável.

---

Opção **-unpack** descompacta o pacote no sistema, mas não processa nenhum script de pré ou pós-instalação.

```
dpkg --unpack pacote
```

O pacote está no sistema, mas ele ainda não está instalado, para concluir a instalação, você deve usar o comando **-configure**, para que os scripts de pós instalação e gatilhos de outro pacote sejam lançados, para isso:

```
dpkg --configure pacote
```

---

Opção “-r” remove tudo do pacote, exceto os arquivos de configuração.

```
dpkg -r pacote
```

---

Opção “-P” purge, apaga tudo do pacote até seus arquivos de configuração.

```
dpkg -P pacote
```

---

Opção “-V” verifica a integridade do pacote, com essa opção você pode verificar se o pacote instalado no sistema está, por exemplo, com um arquivo de configuração faltando.

Vamos ao exemplo, e adivinha? Vou usar o pacote tree, esse carinha sofre sempre nos meus artigos xD. Esse pequeno programa mostra uma saída de arquivos e diretórios no formato de árvore genealógica.

Vou usar o `dpkg -L` para mostrar todos os arquivos desse pacote no sistema, mais a frente você terá informações sobre o “-L”, por hora ele serve para mostrar todos os arquivos no sistema de um determinado pacote. Com o `tree` instalado, eu vou usar o comando já citado.

```
dpkg -L tree
```

```
/usr  
/usr/bin  
/usr/bin/tree  
/usr/share  
/usr/share/doc  
/usr/share/doc/tree  
/usr/share/doc/tree/copyright
```

Eu suprimi a saída para o que interessa, veja esses são os arquivos do pacote `tree` dentro da árvore de diretórios do sistema, eu vou apagar último arquivo `/usr/share/doc/tree/copyright`, para demonstrar o comando `dpkg -V`.

```
dpkg -V
```

```
??5????? /usr/share/doc/tree/copyright
```

Aqui ele informa que o arquivo `copyright` está faltando no pacote `tree`, claro que esse arquivo não vai causar nenhum mal funcionamento, é só para demonstrar.

---

Opção “-C” usada para checar o sistema de pacotes, e mostrar se existe algum problema, muito útil.

Vou usar o `aptitude download tree` para baixar o pacote `tree.deb` no meu diretório corrente, e vou executar a instalação do pacote com `dpkg -i tree.deb`, quando chegar na metade da instalação eu vou abortar com `CONTROL+C` para deixar o pacote desconfigurado, após eu vou executar:

```
dpkg -C
```

Os pacotes a seguir estão apenas meio configurados, provavelmente devido a problemas ao configurá-los pela primeira vez. A configuração deve ser feita novamente usando `dpkg --configure <pacote>` ou a opção no menu do `dselect`:

```
tree          displays an indented directory tree, in color
```

Os pacotes a seguir foram disparados por gatilhos, mas o processamento de gatilhos ainda não terminou. O processamento de gatilhos pode ser executado usando `dselect` ou `dpkg --configure --pending` (ou `dpkg --triggers-only`):

```
man-db       on-line manual pager
```

Veja que ele me indicou o problema e também a solução, basta executar:

```
dpkg --configure tree
```

---

Opção `--get-selections` mostra uma lista de pacotes que estão instalados.

```
dpkg --get-selections
```

Você também pode enviar a saída para um arquivo.

```
dpkg --get-selections > pacotes.txt
```

---

Opção *--set-selections* aceita uma entrada padrão, para modificar o estado de pacote, você pode marcar o pacote como *deinstall* que significa desinstalado, e também como *hold* que é comumente feito quando você não quer que um determinado pacote atualize de versão.

Para marcar um pacote instalado como desinstalado, basta:

```
echo "pacote deinstall" | dpkg --set-selections
```

Para confirmar:

```
dpkg --get-selections | grep pacote
```

Para marcar um pacote como *hold* e impedir sua atualização:

```
echo "pacote hold" | dpkg --set-selections
```

Para confirmar:

```
dpkg --get-selections | grep pacote
```

**OBS** estou usando “pacote” aqui como exemplo, substitua pelo pacote que você deseja fazer a alteração.

---

Opção *--add-architecture* serve para você adicionar uma arquitetura diferente para um pacote, isso é comumente usado para instalar pacotes x86 em sistema amd64.

Adiciona a arquitetura i386, para instalação de um programa como o Skype por exemplo.

```
dpkg --add-architecture i386
```

**OBS** sempre que utilizar esse comando rode o comando *apt-get update* imediatamente, para que uma lista de pacotes da arquitetura seja baixada no sistema, caso contrário, você terá vários erros com dependências.

O parâmetro *--remove-architecture* serve para remover uma arquitetura adicionada.

```
dpkg --remove-architecture i386
```

---

Opção *--print-architecture* imprime na tela a arquitetura atual usada no dpkg.

```
dpkg --print-architecture
```

---

Opção “-c” mostra o conteúdo de um pacote deb, se você baixou um pacote e deseja ver qual o conteúdo do pacote basta:

```
dpkg -c pacote.deb
```

---

Opção “-I” você pode ver informações do pacote.

```
dpkg -I pacote.deb
```

---

Opção “-l” mostra todos os pacotes instalados, é muito útil para verificar se um programa está instalado, com o auxílio do grep.

Vou verificar se o programa tree está instalado na minha máquina

```
dpkg -l | grep tree  
ii tree                1.7.0-3
```

---

Opção “-s” mostra o estado pacote.

```
dpkg -s tree  
Package: tree  
Status: install ok installed  
Priority: optional  
Section: utils  
Installed-Size: 100  
Maintainer: Florian Ernst <florian@debian.org>  
Architecture: amd64  
Version: 1.7.0-3  
Depends: libc6 (>= 2.14)
```

---

Opção “-L” mostra cada arquivo do pacote e a onde ele se encontra no sistema, vou usar ele no programa tree para determinar todos os arquivos dele no sistema:

```
dpkg -L tree  
/.  
/usr  
/usr/bin  
/usr/bin/tree  
/usr/share  
/usr/share/doc  
/usr/share/doc/tree  
/usr/share/doc/tree/copyright  
/usr/share/doc/tree/TODO  
/usr/share/doc/tree/changelog.Debian.gz  
/usr/share/doc/tree/changelog.gz  
/usr/share/doc/tree/README.gz  
/usr/share/man  
/usr/share/man/man1
```

---

Opção “-S” é útil para determinar de que pacote determinado arquivo no sistema vem.

```
dpkg -S usr/bin/tree
```

```
tree: /usr/bin/tree
```

```
dpkg -S /bin/cat
```

```
coreutils: /bin/cat
```

Ele me informa que o comando tree vem do pacote tree e que o comando cat vem do pacote coreutils.

---

Opção *-force*, essa é grande e complexa, vou abordar só o que eu acho útil, mas informações você pode verificar a página de manual.

Opção *-force-all* vai fazer todas as opções de force, até encontrar uma que funciona, ou desmanchar o seu sistema inteiro rrsrsr xD

Às vezes é útil, quando você quer reinstalar um pacote muito problemático:

```
dpkg -i -force-all pacote.deb
```

Opção *downgrade*, essa opção vai instalar um pacote velho substituindo um novo.

Por exemplo tenho o tree na versão 2.0 instalado no sistema, mas quero instalar a versão 1.0, o dpkg já faz isso por padrão, sem reclamar (**CUIDADO**).

Opção *remove-reinstreq* é muito útil, suponhamos que você tem um pacote no sistema quebrado ao quadrado rrsrsrsr, e você não consegue remover e nem fazer nada no maldito, com esse comando você dá uma boa noite Cinderela no dpkg, ele vai esquecer que o pacote existe.

Isso é útil quando você está com um pacote muito problemático e você não está conseguindo nem instalar programas com o apt por causa dele, ele ainda vai permanecer no sistema, mas o dpkg não vai mais encher o saco.

Vou usar o tree, coitado ele nem tem tamanho para ser problemático, mas alguém tem que ser a cobaia xD

```
dpkg -r --force-remove-reinstreq tree
```

Recuperar arquivos de configuração apagados acidentalmente ou por qualquer motivo

```
dpkg --force-confmiss --force-confnew -i pacote
```

Você também pode remover pacotes essenciais do sistema o que não é uma boa ideia, também pode instalar pacotes com conflitos de versão, ou que quebram com um pacote já instalado, novamente isso não é uma boa ideia, geralmente o *-force-all* na maioria dos casos ajuda, mas se você precisa de umas dessas peculiaridades, consulte: `man dpkg`.

---

Opção `-no-act` simula uma instalação com o `dpkg`, ou seja, ele não vai instalar realmente o pacote.

```
dpkg -no-act -i pacoite.deb
```

---

Opção `-R` instalação recursiva de todos os pacotes, ele vai adentrar em todos os diretórios e instalar o que ele encontrar com `*.deb`

```
dpkg -iR diretório/
```

---

Opção `-G` por padrão o `dpkg` não nem aí, se você tiver um pacote da versão 5 instalado no sistema e mandar um ele instalar a versão 1, ele vai fazer isso e não vai emitir nada, com essa opção você pode instalar vários pacotes e garantir que eles não irão regredir de versão.

```
dpkg -iG pacote.deb  
dpkg -iG *.deb (instalar vários pacotes e garantir que nada vai regredir de versão)
```

---

Opção `-E` não instala o pacote se a mesma versão já estiver instalado.

```
dpkg -Ei pacote  
dpkg -Ei *.deb (útil para instalar vários pacotes)
```

---

Opção `-instdir=dir` você pode instalar o pacote em um diretório diferente;

```
Por exemplo vou instalar o pacote tree em uma pasta na minha home chama dir.  
dpkg -i -instdir=/home/fernando/dir tree_1.7.0-3_amd64.deb
```

---

Autor: Fernando Debian.

Comunidade Debian Brasil no facebook, acessem:

<https://goo.gl/qfNjkj> -- <https://goo.gl/9gv0zp>