

6. Overrides para o lintian

6.1 Para binários

Crie o arquivo `debian/<pacote>.lintian-overrides`. Coloque as mensagens do lintian (sem E:, I:, P:, W: ou X:) dentro desse arquivo.

6.2 Para sources

Crie o arquivo `debian/source/lintian-overrides`. Coloque as mensagens do lintian (sem E:, I:, P:, W: ou X:) dentro desse arquivo.

7. Patches com quilt

7.1 Criação de patch

- ▶ Estando no diretório do upstream, emita os comandos:

```
# quilt push -a
# quilt new nome_do_patch
# quilt add arquivo(s)_a_ser(em)_patchado(s)
```

- ▶ Edite normalmente o(s) arquivo(s) do upstream, envolvido(s) pelo patch. Depois, execute:

```
# quilt refresh
# quilt header -e
```

- ▶ Inserir como cabeçalho:

```
Description: descrever o que o patch faz.
Author: nome_do_autor_do_patch <e-mail>
Last-Update: YYYY-MM-DD
```

- ▶ Finalizar com:

```
# quilt pop -a
```

7.2 Modo alternativo de criação de patches

- ▶ Aplique os patches pré-existentes com `# quilt push -a`

- ▶ Edite normalmente o(s) arquivo(s) do upstream, envolvido(s) pelo patch. Depois, execute:

```
# dpkg-source --commit
```

- ▶ Desaplique todos os patches:

```
# quilt pop -a
```

7.3 Edição de patch preexistente

- ▶ No diretório `debian`:

```
# quilt push nome_do_patch
```

- ▶ Edite normalmente o(s) arquivo(s) envolvido(s) pelo patch.

Depois, execute:

```
# quilt refresh
# quilt pop -a
```

8. DHs

- ▶ Relação completa de DHs (há uma manpage para cada um): `# apropos dh_`
- ▶ Para ver os DHs atuando durante a construção do pacote, descomente `export DH_VERBOSE=1`, em `debian/rules`.

9. Ignorar alterações em arquivo

Criar o arquivo `debian/source/options`. Poderão ser utilizadas expressões regulares. Ex:

```
extend-diff-ignore = "^(test|src|ctrltag)$"
```

10. Bibliografia essencial

- ▶ Vídeaulas sobre empacotamento: <http://debianet.com.br>
- ▶ FHS: <https://refspecs.linuxfoundation.org/fhs.shtml>
- ▶ Manual de política Debian: <https://debian.org/doc/debian-policy>
- ▶ Guia de referência para desenvolvedores: <https://debian.org/doc/manuals/developers-reference>
- ▶ Guia do novo mantenedor: <https://debian.org/doc/manuals/debmake-doc/index.en.html>
- ▶ Canto dos desenvolvedores: <https://debian.org/devel>
- ▶ Debian FAQ (empacotamento): http://debian.org/doc/manuals/debian-faq/ch-pkg_basics.en.html
- ▶ WNPP: <https://debian.org/devel/wnpp>
- ▶ Listas Debian: <http://lists.debian.org>

- ▶ Lista de pacotes virtuais: <https://debian.org/doc/packaging-manuals/virtual-package-names-list.yaml>
- ▶ Lista de de pseudo-pacotes para bugs: <https://debian.org/Bugs/pseudo-packages>
- ▶ Times no Debian: <https://wiki.debian.org/Teams>
- ▶ Candidatura a mantenedor: <https://wiki.debian.org/DebianMaintainer>
- ▶ Candidatura a desenvolvedor: <https://nm.debian.org>

- ▶ Statistics to Choose a Debian Package to Help: https://people.debian.org/~eriberto/udd/help_a_package.html
- ▶ Links interessantes: <https://people.debian.org/~eriberto/#links>

EMPACOTAMENTO DE SOFTWARE NO DEBIAN

Versão 3.5 - 14 de setembro de 2020



debian

© 2007-2020 João Eriberto Mota Filho

<http://debianet.com.br>
eriberto@debian.org

4096R/04EBE9EF: 357D CB0E EC95 A01A EBA1 F0D2 DE63 B9C7 04EB E9EF
4096R/3A1FACFE: 88E6 D789 C7D4 09F8 5EE7 A7B8 C834 29CC 3A1F ACFE

1. Construção da jaula Sid (unstable)

1.1 Procedimentos iniciais

```
# debootstrap sid /jaula-sid http://deb.debian.org/debian
# chroot /jaula-sid
# echo proc /proc proc defaults 0 0 >> /etc/fstab
# mount /proc
# apt update
```

Obs: caso o sid (unstable) esteja quebrado e apresente problemas na instalação, utilize o stable e, depois, faça um dist-upgrade para sid.

1.2 Ajuste do /etc/bash.bashrc

Insira no final do arquivo /etc/bash.bashrc:

```
alias ls='ls --color=auto'
alias tree='tree -aC'
alias debuildsa='dpkg-buildpackage -sa -ksua_chave_gpg'
alias uscan-check='uscan --verbose --report'
alias debcheckout='debcheckout -a'
export DEBFULLNAME="seu_nome_completo_sem_acentos/cedilha"
export DEBEMAIL="seu_e-mail"
export EDITOR=mcedit
export LANG=C.UTF-8
export LANGUAGE=C.UTF-8
export LC_ALL=C.UTF-8
export QUILT_PATCHES=debian/patches
export PS1='JAULA-SID-\u@h:\w\$ '
mount /proc
```

Obs: txt pronto para uso em <http://bit.ly/bash-rc-txt>

A seguir, execute: # source /etc/bash.bashrc

1.3 Procedimentos complementares

```
# apt install autopkgtest blhc devscripts dh-make dput-ng
git-buildpackage how-can-i-help mc quilt spell tardiff tree
# dpkg-reconfigure tzdata
# apt clean
```

1.4 Criação do ~/.lintianrc

Crie o arquivo ~/.lintianrc (o mesmo que /root/.lintianrc) e insira as seguintes linhas:

```
display-info = yes
pedantic = yes
display-experimental = yes
color = auto
```

1.5 Ajuste do /etc/apt/sources.list

Edite o arquivo, duplique a linha existente e altere o início da segunda linha de deb para deb-src, como está abaixo. Execute # apt update.

```
deb http://deb.debian.org/debian sid main
deb-src http://deb.debian.org/debian sid main
```

1.6 Ajuste do git-buildpackage

Edite o arquivo /etc/git-buildpackage/gbp.conf, descomentando e alterando as seguintes linhas na seção [DEFAULT]:

```
debian-branch = debian/master
pristine-tar = True
```

1.7 Como sair da jaula

A jaula está pronta. Para sair da mesma, execute: # exit ou Ctrl D

1.8 Inserção das chaves GPG RSA 4096

Fora da jaula, exporte as suas chaves GPG (privada e pública):

```
$ gpg -a --export nr_da_chave > nr_da_chave.pub
$ gpg -a --export-secret-keys nr_da_chave > nr_da_chave.key
```

Mova os arquivos para dentro da jaula unstable, enjaula-se, importe as chaves e remova os arquivos. Para importar:

```
# gpg --list-keys
# echo "pinentry-mode loopback" >> ~/.gnupg/gpg.conf
# gpg --import chave.key chave.pub
```

1.9 Habilitação da chave para a assinatura de pacotes

Edite o arquivo /etc/devscripts.conf e insira o número COMPLETO da sua chave GPG na linha DEBSIGN_KEYID, descomentando-a. Veja:

```
DEBSIGN_KEYID=357DCB0EEC95A01AEBA1F0D2DE63B9C704EBE9EF
```

1.10 Backup da jaula para uso futuro

```
# tar -cvjf jaula-sid.tar.bz2 --exclude='*/proc/*' --
exclude='*/apt/lists/*dists*' /jaula-sid
```

2. Dicas de empacotamento

- ▶ Pacote novo? Inédito? Verifique se já existe alguma menção ao mesmo no Debian com os comandos apt search <nome> e wnpp-check <nome>.
- ▶ Verificar as licenças usadas pelo upstream:

```
# licensecheck -r *
# egrep -sriA25 '(public dom|copyright)' | less
```
- ▶ Debianização: # dh_make -f ../<fonte>.tar.gz -c <licença>

- ▶ Obter a relação dos mantenedores anteriores:

```
# egrep '@|\[)' debian/changelog
```
- ▶ Caso o lintian aponte problemas difíceis de entender, execute # lintian -i no diretório do upstream.
- ▶ Depois de executar o debuild, verifique com # tree debian/nome_do_pacote se tudo está certo.
- ▶ Para testar o debian/watch, use # uscan-check

3. Verificação final

3.1 Checklist final de empacotamento

Sempre siga o prescrito em <http://bit.ly/pkgcheck>

3.2 Teste de construção do pacote (cowbuilder)

Fora da jaula, para criar um ambiente de testes, execute:

```
# cowbuilder create
```

Para verificar um pacote, execute:

```
# cowbuilder build arquivo.dsc
```

Para atualizar a jaula cowbuilder, execute:

```
# cowbuilder update
```

Importante: caso não consiga criar a jaula (Sid quebrado) execute:

```
# cowbuilder --distribution stable create
```

Depois, edite o arq. /var/cache/pbuilder/base.cow/etc/apt/sources.list e altere o nome stable para unstable (ou sid). A seguir, execute:

```
# cowbuilder update
```

3.3 Teste de instalação do pacote

```
# piuparts arquivo.deb
```

4. Revisão Debian (novo pacote)

Dentro da jaula execute: # dch -i (Ou --qa, --team, --bpo, --nmu)

5. Nova versão do programa

Dentro da jaula, faça o download (com wget, por exemplo) da nova versão do programa desejado. Depois, execute:

```
# apt source pacote
# cd diretório_do_pacote
# uupdate ../programa-nova_versão.tar.gz
```

Obs: pacote refere-se à versão antiga do código-fonte. É possível usar # uscan para baixar a nova versão.