Slackware Software Packaging: How and Where To Start

How many of you are aware of the steps in creating Slackware packages

Pritvi Jheengut

August 2016

This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Outline

Table of Contents

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

Shell

- Some basic shell programming skills
- GNU Coreutils

Example

GNU Coreutils

- mkdir
- cat
- ls

Shell

- Some basic shell programming skills
- GNU Coreutils

Example

GNU Coreutils

- mkdir
- cat
- ls

Shell

- Some basic shell programming skills
- GNU Coreutils

Example

GNU Coreutils

- mkdir
- cat
- ls

Shell

- Some basic shell programming skills
- GNU Coreutils

Example

GNU Coreutils

- mkdir
- cat
- ls

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Unix

- tar
- cp/mv/install
- wget/curl
- chmod/chown
- mktemp/mkdir
- strip
- autotools/make/cmake/python/perl/ruby/
- find/xargs/grep

Compression Utility
gzip
bzip2
xz/lzma

mime
tgz
tbz
txz

Compression Utility
gzip
bzip2
xz/lzma

mime
tgz
tbz
txz

Compression Utility
gzip
bzip2
xz/lzma

mime
tgz
tbz
txz

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- `http://tukaani.org/pkgtools/` an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- http://tukaani.org/pkgtools/ an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Slackware

- pkgtool - The Package Manager
- `http://tukaani.org/pkgtools/` an old forked version
- installpkg
- removepkg
- upgradepkg
- explodepkg
- makepkg - The Package Creator/Maker
- slackbuild scripting

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

A Package

Definition: An archive format with/out instructions used for the installation of files on a system

Example

- t?z
- deb
- rpm

mime

- Slackware
- Debian
- Red Hat
- OpenSUSE

A Package

Definition: An archive format with/out instructions used for the installation of files on a system

Example

- t?z
- deb
- rpm

mime

- Slackware
- Debian
- Red Hat
- OpenSUSE

A Package

Definition: An archive format with/out instructions used for the installation of files on a system

Example

- t?z
- deb
- rpm

mime

- Slackware
- Debian
- Red Hat
- OpenSUSE

Definition of a package

A Slacker's goal to world domination

A Package

Definition: An archive format with/out instructions used for the installation of files on a system

Example

- t?z
- deb
- rpm

mime

- Slackware
- Debian
- Red Hat
- OpenSUSE

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Package Manager

A package manager is a collection of software tools that

- install
- upgrade
- configure
- remove
- performs other related jobs on

software packages for a Linux Distribution.

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

mktmp

mktmp

Create temporary files/directories that are predictable, safe and to avoid possible race conditions.

```
cd /tmp/$"Temporary Directory"
```

/tmp

The /tmp directory is assumed to cleaned regularly, sometimes assumed a tmpfs is mount over tmp
The source code is often extracted into /tmp directly.

mktmp

mktmp

Create temporary files/directories that are predictable, safe and to avoid possible race conditions.

```
cd /tmp/$"Temporary Directory"
```

/tmp

The /tmp directory is assumed to cleaned regularly, sometimes assumed a tmpfs is mount over tmp
The source code is often extracted into /tmp directly.

Copy The Source Code

Copy the source code archive into the tmp dir

```
cp ${old path}/${source code archive} .
```

Download The Source Code

Or Download the source code archive into the tmp dir

```
wget/curl
protocol://url/${source code archive}
```

Copy The Source Code

Copy the source code archive into the tmp dir

```
cp ${old path}/${source code archive} .
```

Download The Source Code

Or Download the source code archive into the tmp dir

```
wget/curl
protocol://url/${source code archive}
```

Copy The Source Code

Copy the source code archive into the tmp dir

```
cp ${old path}/${source code archive} .
```

Download The Source Code

Or Download the source code archive into the tmp dir

```
wget/curl
protocol://url/${source code archive}
```

Copy The Source Code

Copy the source code archive into the tmp dir

```
cp ${old path}/${source code archive} .
```

Download The Source Code

Or Download the source code archive into the tmp dir

```
wget/curl
protocol://url/${source code archive}
```

Extracting The Source Code

This step is optional

```
mkdir build
cd build
```

Extracting The Source Code

This step is optional

```
mkdir build
cd build
```

Extracting The Source Code

This step is optional

```
mkdir build
cd build
```

Extracting The Source Code

Extract files into the ‘mktmp‘/tmp directory

```
tar -xvvf ../"${source code archive}"
```

$$/tmp \xrightarrow[D]{} \$PKG \xrightarrow[D]{} build$$

Extracting The Source Code

Extract files into the ‘mktmp‘/tmp directory

```
tar -xvvf ../"${source code archive}"
```

$$/tmp \xrightarrow[D]{} \$PKG \xrightarrow[D]{} build$$

Extracting The Source Code

Extract files into the ‘mktmp‘/tmp directory

```
tar -xvvf ../"${source code archive}"
```

$$/tmp \xrightarrow[D]{\gg} \$PKG \xrightarrow[D]{\gg} build$$

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

configuration

Most of the times configuration is needed.

Example

autotools

- aclocal
- autoconf
- automake

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Configuration of the source code

Pre-Configuration and Configuration

pre-configuration

Rarely is pre-configuration performed.

Example

autotools

- aclocal
- autoconf
- automake

configuration

Most of the times configuration is needed.

Example

- ./configure
- cmake
- python setup.py
- gem
- qmake

Building the Source Code

After configuration, the code has to be compiled into machine code.

Example

- make
- perl Makefile.pl
- gem
- python setup.py build

Building the Source Code

After configuration, the code has to be compiled into machine code.

Example

- make
- perl Makefile.pl
- gem
- python setup.py build

Building the Source Code

After configuration, the code has to be compiled into machine code.

Example

- make
- perl Makefile.pl
- gem
- python setup.py build

Building the Source Code

After configuration, the code has to be compiled into machine code.

Example

- make
- perl Makefile.pl
- gem
- python setup.py build

Building the Source Code

Building the Source Code

After configuration, the code has to be compiled into machine code.

Example

- make
- perl Makefile.pl
- gem
- python setup.py build

A Summary

A summary of the job done

$$Pre - Configuration \longrightarrow\!\!\!\!\!\rightarrow Configuration \longrightarrow\!\!\!\!\!\rightarrow Building$$

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

Installation of Package Files in tmp directory

The package files are installed in a directory using the $PKG = \$TMP/package - \$PRGNAM$ shell path parameter.

```
make install DESTDIR=$PKG
python setup.py install --root=$PKG
gem install --install-dir $PKG/$DESTDIR
```

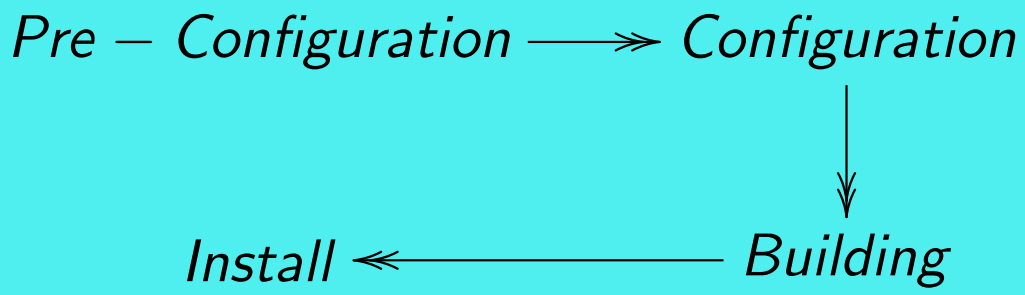
Installation of Package Files in tmp directory

The package files are installed in a directory using the $PKG = \$TMP/package - \$PRGNAM$ shell path parameter.

```
make install DESTDIR=$PKG
python setup.py install --root=$PKG
gem install --install-dir $PKG/$DESTDIR
```

Installation of Package Files in tmp directory

The package files are installed in a directory using the $PKG = \$TMP/package - \$PRGNAM$ shell path parameter.



stripping of binaries

After installation of object files/binaries, they need to be stripped off of unnecessary symbols.

```
find $PKG -print0 | xargs -0 file \
| grep -e "executable" -e "shared object" \
| grep ELF \
| cut -f 1 -d : \
| xargs strip --strip-unneeded \
2> /dev/null || true
```

stripping of binaries

After installation of object files/binaries, they need to be stripped off of unnecessary symbols.

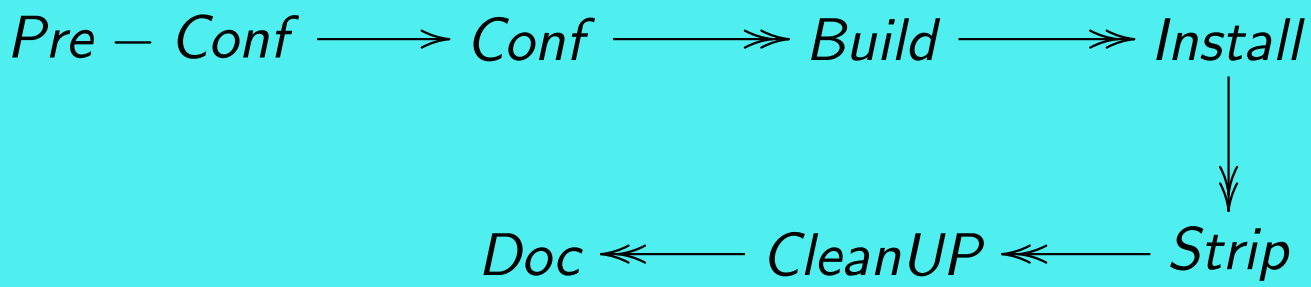
```
find $PKG -print0 | xargs -0 file \
| grep -e "executable" -e "shared␣object" \
| grep ELF \
| cut -f 1 -d : \
| xargs strip --strip-unneeded \
2> /dev/null || true
```

strip and the rest

stripping is one of a few more steps in making a more acceptable package, others include compressing the man and info pages, removing excess data and adding documentation

strip and the rest

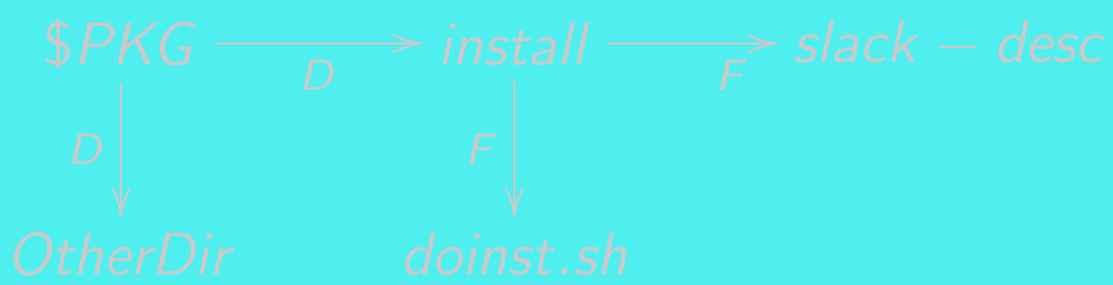
stripping is one of a few more steps in making a more acceptable package, others include compressing the man and info pages, removing excess data and adding documentation



$PKG/install directory

An extra install directory exist in all Slackware Packages. This folder normally contains two files.

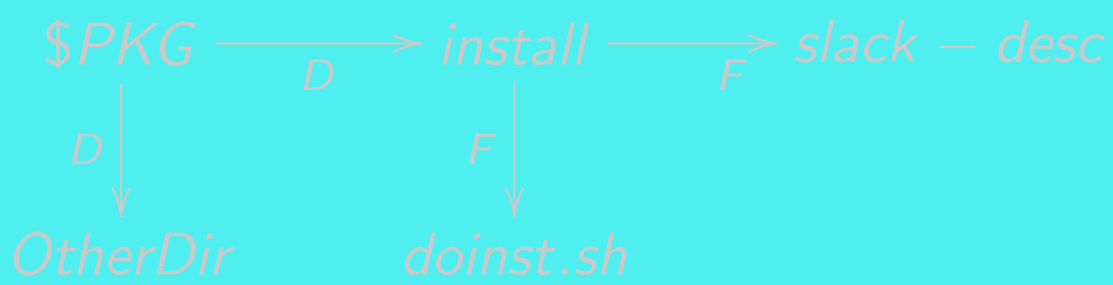
- slack-desc : This contains a summary and full description of the package.
- doinst.sh : contains instructions for the post installation stage, most of the time linking of files.



$PKG/install directory

An extra install directory exist in all Slackware Packages. This folder normally contains two files.

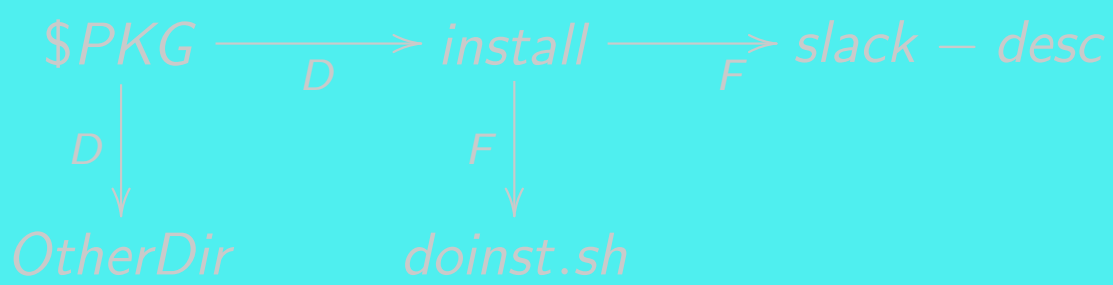
- slack-desc : This contains a summary and full description of the package.
- doinst.sh : contains instructions for the post installation stage, most of the time linking of files.



$PKG/install directory

An extra install directory exist in all Slackware Packages. This folder normally contains two files.

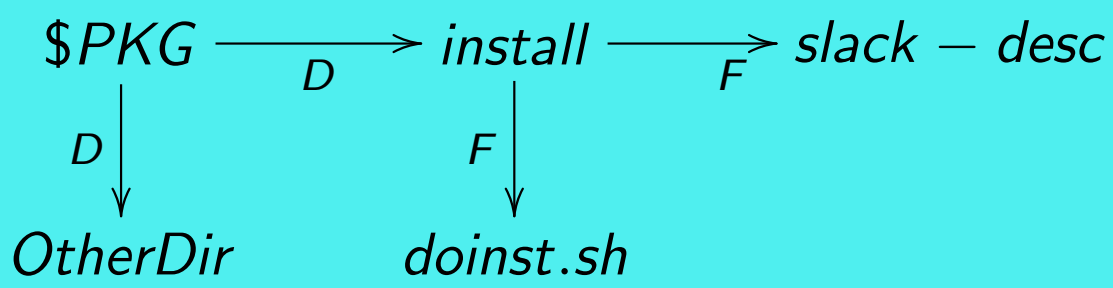
- slack-desc : This contains a summary and full description of the package.
- doinst.sh : contains instructions for the post installation stage, most of the time linking of files.



$PKG/install directory

An extra install directory exist in all Slackware Packages. This folder normally contains two files.

- slack-desc : This contains a summary and full description of the package.
- doinst.sh : contains instructions for the post installation stage, most of the time linking of files.



$PKG/install directory

Copy the slack-desc and a custom doinst.sh if necessary into ./install

```
mkdir -p $PKG/install

cat $CWD/slack-desc > $PKG/install/slack-desc

cat $CWD/doinst.sh > $PKG/install/doinst.sh
```

$PKG/install directory

Copy the slack-desc and a custom doinst.sh if necessary into ./install

```
mkdir -p $PKG/install

cat $CWD/slack-desc > $PKG/install/slack-desc

cat $CWD/doinst.sh > $PKG/install/doinst.sh
```

Makepkg

makepkg creates a Slackware compatible package. The package is constructed using the contents of the current directory and all sub directories.

```
cd $PKG
makepkg $PRGNAM.tgz
```

Makepkg

makepkg creates a Slackware compatible package. The package is constructed using the contents of the current directory and all sub directories.

```
cd $PKG
makepkg $PRGNAM.tgz
```

Example

makepkg options

- option -l :: add any symbolic links found to the install script (doinst.sh) and delete them.
- option -c :: makepkg will reset all directory permissions to 755 and ownership to root:root

Example

makepkg options

- option -l :: add any symbolic links found to the install script (doinst.sh) and delete them.
- option -c :: makepkg will reset all directory permissions to 755 and ownership to root:root

Makepkg

with makepkg full options

```
/sbin/makepkg -l y -c n \
/tmp/$PRGNAM-$VERSION-$ARCH-$BUILD$TAG.tgz
```

Package

This is it, your package is ready and waiting for you in /tmp.

Makepkg

with makepkg full options

```
/sbin/makepkg -l y -c n \
/tmp/$PRGNAM-$VERSION-$ARCH-$BUILD$TAG.tgz
```

Package

This is it, your package is ready and waiting for you in /tmp.

Slackbuilds

Want to Automate your package building, then the only sane way is maintaining a script that we commonly call slackbuild script. All the steps above are entered into a script and tada, its done.

```
#!/bin/bash

PKG=/tmp/package-$PRGNAM

rm -rf $PKG
mkdir -p $TMP $PKG
cd /tmp
rm -rf $PRGNAM-$VERSION
```

Slackbuilds

Want to Automate your package building, then the only sane way is maintaining a script that we commonly call slackbuild script. All the steps above are entered into a script and tada, its done.

```
#!/bin/bash

PKG=/tmp/package-$PRGNAM

rm -rf $PKG
mkdir -p $TMP $PKG
cd /tmp
rm -rf $PRGNAM-$VERSION
```

```
tar xvf $CWD/$PRGNAM-$VERSION.tar.gz
cd $PRGNAM-$VERSION

chown -R root:root .

./configure

make
make install DESTDIR=$PKG
```

```
find ... | xargs strip ...
mkdir -p $PKG/install

cat $CWD/slack-desc > $PKG/install/slack-desc
cat $CWD/doinst.sh > $PKG/install/doinst.sh

cd $PKG

/sbin/makepkg /tmp/$PRGNAM.tgz
```

slackbuilds.org

Templates can be found at http://slackbuilds.org/templates

Outline

1 Introduction

2 What is a Package?

3 Preparation For Building

4 Building The Source Code

5 Creating The Package

6 The End

Goal

- slack the hard way
- sip some beer
- Take over the world
- Relax...

Goal

- slack the hard way
- sip some beer
- Take over the world
- Relax...

Goal

- slack the hard way
- sip some beer
- Take over the world
- Relax...

Goal

- slack the hard way
- sip some beer
- Take over the world
- Relax...

- LaTeX Beamer
  http://latex-beamer.sourceforge.net/
- KDE Presentations
  http://www.kde.org/kdeslides/
- My Prezz at Github
  https://github.com/Bluetailedgecko/Slackware-Software-Packaging-Presentation

Slackware GNU/Linux

Questions ?

Pritvi Jheengut
z.coldplayer@gmail.com