

**TUTORIAL  
PENGENALAN DAN PRAKTEK  
MEMBUAT PAKET DEBIAN**

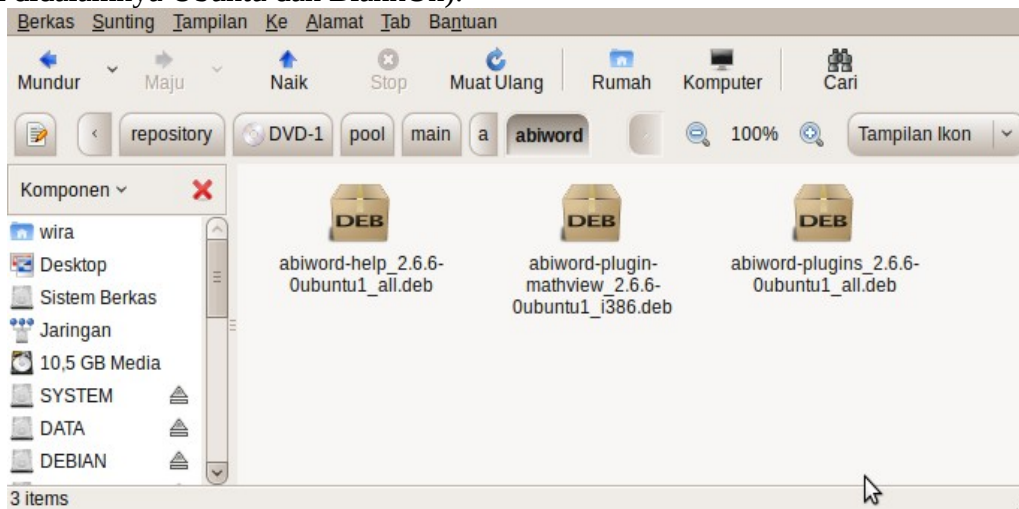


**OLEH  
PUTU WIRAMASWARA WIDYA  
TIM PENGEMBANG BLANKON**

**Disampaikan dalam acara Blankonf ke-I  
di Universitas Pakuan Bogor  
Tanggal 20-21 Juni 2009**

## PENDAHULUAN

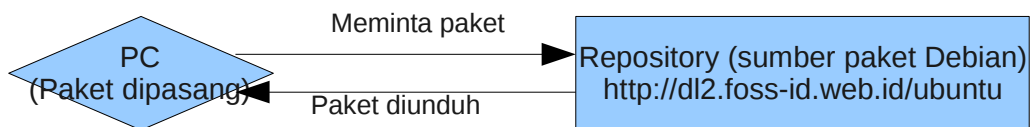
Paket Debian merupakan jenis berkas dengan ekstensi \*.deb yang berisikan berkas-berkas aplikasi yang siap di-install ke dalam sistem operasi Debian beserta distribusi (distro) turunannya (termasuk didalamnya Ubuntu dan BlankOn).



**Contoh berkas paket Debian**

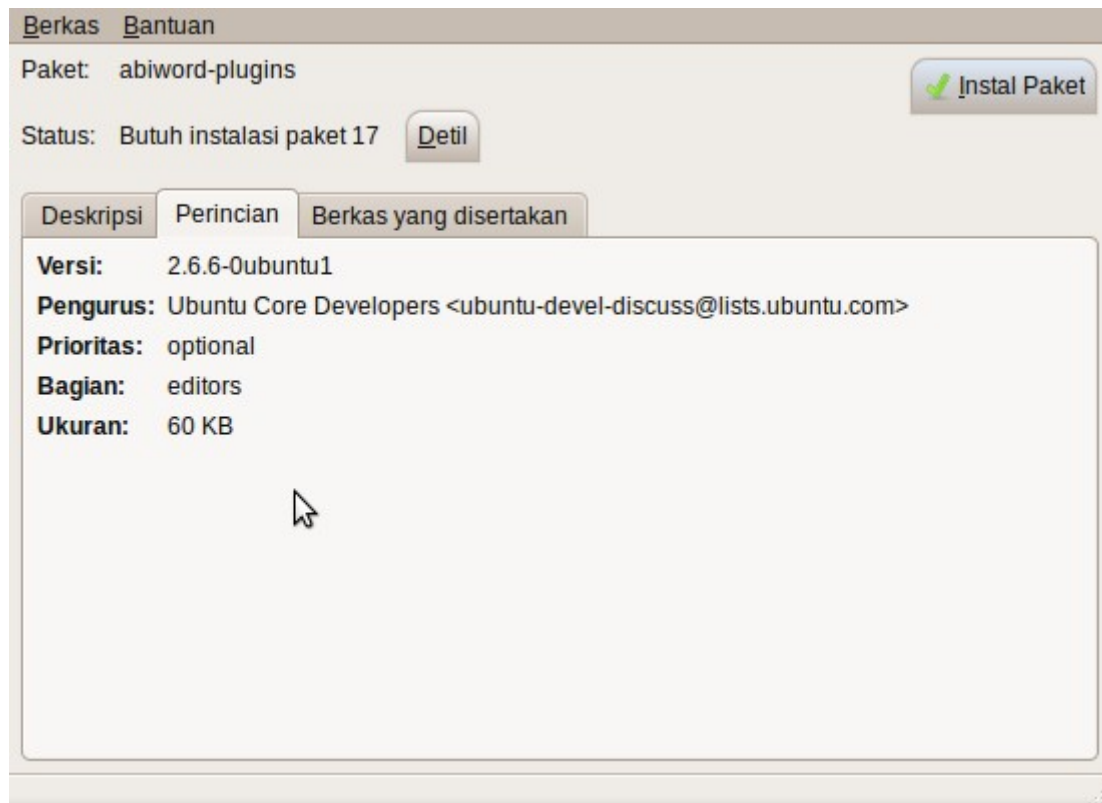
Untuk melakukan instalasi suatu paket Debian pada sistem, digunakan program bantu yang bernama dpkg yang akan langsung berinteraksi dengan paket dan langsung meng-install atau menghapusnya sesuai dengan keperluan. Namun, program bantu dpkg ini tidak ikut meng-installkan paket ketergantungan dari paket yang dipasang, maka dari itu dibuatlah sebuah alat bantu untuk dpkg yang dikenal dengan APT yang membantu dalam mengunduh, meng-install paket debian, menghapus paket yang ter-install dan melakukan pemutakhiran/update dari suatu paket Debian.

Dalam melakukan instalasi software menggunakan APT, paket-paket dari software yang tersedia disimpan dalam *repository* atau gudang paket dalam bentuk paket Debian yang nantinya akan diunduh sendiri oleh APT, kemudian APT melakukan instalasi terhadap paket software yang diminta untuk instalasi melalui dpkg. Misalnya, jika anda meng-install aplikasi Firefox entah itu melalui perintah “apt-get install firefox” atau melalui aplikasi bantu Synaptic Package Manager, paket Firefox beserta paket ketergantungannya (*dependency*) yang terdapat di dalam *repository* akan diunduh ke sistem dan kemudian di-install melalui APT.



**Proses pemasangan paket**

Selain berisikan berkas-berkas yang akan dipasang, sebuah paket debian juga berisikan informasi dari paket itu sendiri, paket yang dibutuhkan oleh paket tersebut atau *dependency*-nya, serta skrip-skrip yang akan dijalankan sebelum atau pasca instalasi.



### ***Informasi dari suatu paket Debian***

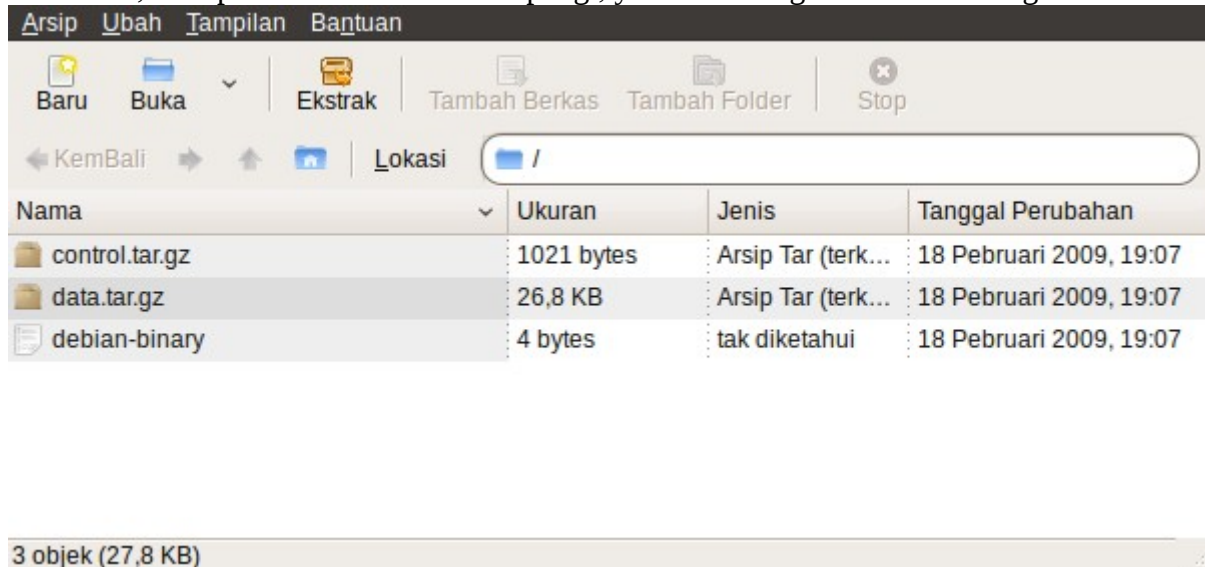
Tutorial yang akan disampaikan dalam kesempatan ini adalah bagaimana cara untuk membuat paket Debian dari suatu kode sumber dari suatu software. Adapun tujuan dan maksud dari pembuatan paket Debian antara lain :

- Memaketkan software untuk dipasang oleh khayalak umum. Paket-paket software yang ada pada *repository* Debian atau Ubuntu yang ada saat ini bukanlah merupakan buatan dari pengembang software tersebut, melainkan oleh tim pemaket atau *maintainer* dari distribusi Linux masing-masing (Di Ubuntu dikenal dengan istilah MOTU). Seorang *Maintainer* harus aktif untuk memutakhirkan paket softwarentya jika pengembang merilis versi terbaru dari software tersebut.
- Memaketkan software untuk membuat distribusi Linux turunan. Untuk membuat distro turunan, tentunya harus dilakukan pembangunan ulang atau modifikasi dari beberapa paket tertentu (misalnya branding, artwork, dll). Seperti pada distro BlankOn, yang membutuhkan modifikasi dari paket-paket tertentu dari distro Ubuntu.
- Membuat modifikasi dari paket software sebelumnya yang berisikan penambahan fitur serta penambalan dari sisi keamanan. Hasil dari modifikasi bisa diberikan kepada maintainer resmi dari distribusi tertentu atau bisa diunggah secara tidak resmi ke Launchpad PPA. Sementara manfaat yang diharapkan anda dapatkan setelah mengikuti tutorial ini adalah :
- Mengerti dan memahami konsep atau cara untuk membangun suatu paket Debian serta bagaimana konsep publikasinya.
- Menjadi landasan teori untuk menjadi seorang *Maintainer* atau Pemaket dari suatu distribusi Linux.

## MENGENAL LEBIH DALAM PAKET DEBIAN

### *Apa saja yang ada didalam suatu paket Debian?*

Paket Debian merupakan berkas arsip dengan format AR. Isi dalam suatu Paket Debian bisa anda lihat menggunakan aplikasi manajer arsip yang biasa anda gunakan di desktop. Di dalam suatu paket Debian, terdapat dua buah berkas arsip lagi, yaitu `data.tar.gz` dan `control.tar.gz`.



### *Isi dari suatu paket Debian*

Berkas-berkas software yang akan dipasang ke sistem diletakkan dalam berkas arsip `data.tar.gz`. Sedangkan informasi-informasi dari paket seperti nama paket, versi, paket ketergantungan, dan lain-lain disimpan dalam berkas `control.tar.gz`. Selain itu, skrip-skrip yang perlu dijalankan sebelum dan setelah instalasi suatu paket juga disimpan di dalam berkas `control.tar.gz`.

### *Bagaimana penamaan suatu Paket Debian?*

Penamaan dari paket Debian tidaklah sembarangan. Penamaan harus sesuai dengan aturan yang sudah baku dari suatu distribusi Linux. Contoh dari penamaannya adalah sebagai berikut :

**hello\_2.4.0ubuntu1\_i386.deb**

Ada tiga bagian dari nama tersebut dengan pemisah tanda underscore “\_”, yaitu (dari kiri ke kanan :

- Nama paket, nama ini menjadi identitas dari paket tersebut. Jika anda ingin memasang suatu paket Debian menggunakan fitur APT, maka nama inilah yang harus disebut.
- Versi paket, berisikan nomor versi dari paket software tersebut. Di akhir nomor versi terdapat beberapa embel-embel kata yang ketentuannya diatur oleh kesepakatan pemaket suatu distro.
- Arsitektur, berisikan nama arsitektur komputer yang bisa dipakai untuk memasang paket tersebut. Untuk paket yang tidak bergantung pada arsitektur (seperti paket non-binary), arsitektur yang ditulis adalah “all”.

### *Apa saja jenis-jenis paket Debian?*

Ada beberapa jenis paket Debian berdasarkan isi dari paket tersebut. Setiap jenis paket memiliki embel-embel tertentu. Berikut adalah jenis-jenis paket yang umum :

- Paket binari, merupakan jenis paket yang berisi program utama yang *executable* dari suatu software. Penamaan paket ini tidak memiliki embel-embel apa. Misalnya, “hello”.
- Paket pustaka atau *library*, berisikan kumpulan-kumpulan pustaka yang dibutuhkan dari

software tersebut. Nama dari paket pustaka diawali dengan kata “lib” dan diakhiri dengan nomor yang disebut dengan SONAME. Misalnya, “libhello0”.

- Paket header, berisikan beberapa kumpulan skrip yang ditujukan untuk membangun suatu software lain yang memiliki ketergantungan terhadap paket tersebut. Nama dari paket ini diawali dengan kata “lib” dan diakhiri dengan “SONAME-dev”. Misalnya, “libhello0-dev”.
- Paket debugger, berisikan berbagai berkas simbol-simbol yang ditujukan untuk keperluan *debugging* untuk penelusuran kesalahan dari software. Nama dari paket debugger diakhiri dengan kata “-dbg”.
- Paket data, berisikan berbagai data tambahan yang diperlukan oleh suatu software. Biasanya berisikan artwork-artwork yang digunakan oleh software tersebut. Biasanya, paket data diakhiri dengan kata “-data”, namun kadang-kadang bisa berbeda karena tidak ada aturan yang baku.
- Metapackage, merupakan paket yang merujuk ke beberapa paket. Nama paket ini tidak ada aturan bakunya. Contoh dari paket ini yang bisa ditemui pada *repository* Ubuntu adalah “ubuntu-desktop” yang merujuk ke semua paket esensial yang dibutuhkan untuk menjalankan desktop Ubuntu berbasis GNOME.

### **Bagaimana membangun Paket Debian?**

Cara yang dilakukan untuk membuat paket Debian siap install adalah dengan membangun paket Debian dari kode sumber atau *source code* dari suatu aplikasi. Cara ini adalah cara umum yang digunakan oleh para pemaket, karena paket bisa langsung dibuat setelah software tersebut di-*compile* dari *source code*. Untuk melakukan hal ini, anda perlu melakukan konversi paket kode sumber menjadi paket kode sumber Debian. Proses konversi ini mungkin sangat rumit, namun hal ini bisa dibantu dengan tool-tool bantu yang sudah tersedia.

Proses konversi akan menghasilkan sebuah folder baru di dalam folder kode sumber bernama “debian”. Di dalam folder ini, terdapat beberapa berkas-berkas yang memiliki fungsi masing-masing dan sangat esensial dalam pembuatan paket Debian. Beberapa berkas-berkas tersebut antara lain : copyright (berisi keterangan hak cipta program), changelog (berisi daftar perubahan software), control (berisi keterangan paket serta nama paket ketergantungan), rules (berisi skrip untuk melakukan pembangunan paket Debian dari kode sumber software tersebut), dan sebagainya. Hal yang paling sulit untuk diedit adalah berkas rules, karena anda harus menuliskan skrip-skrip pekerjaan yang dilakukan dalam pembangunan paket. Namun, hal ini bisa dipermudah dengan menggunakan kumpulan skrip bernama CDBS (Common Debian Build System) yang merupakan sekumpulan skrip siap pakai yang bisa dimanfaatkan dalam berkas rules ini.

Setelah proses konversi, maka dilakukanlah proses pembangunan paket Debian dengan menggunakan perintah “dpkg-buildpackage”. Proses ini diawali dengan proses kompilasi dari software dari kode sumber menjadi kode biner yang bisa dijalankan oleh mesin, kemudian proses pemaketan dari software tersebut menjadi berkas paket Debian sehingga nantinya siap di-install ke komputer.

### **Bagaimana mempublikasikan suatu Paket Debian?**

Untuk mempublikasikan paket Debian yang sudah anda bangun, ada beberapa cara yang anda lakukan. Jika paket anda ingin dimasukkan ke dalam *repository* resmi dari distro Debian (dan juga di Ubuntu), anda bisa mendaftarkan diri terlebih dahulu menjadi seorang *maintainer* Debian. Silahkan baca <http://www.debian.org/devel/join/newmaint> untuk lebih lanjut.

Jika anda ingin mengunggah paket-paket buatan anda sendiri secara tidak resmi (misalnya paket berisi beberapa fitur tambahan menarik), anda bisa mengunggah melalui layanan PPA (Persona Package Archive) Launchpad (<http://launchpad.net>) yang merupakan *website* untuk berkolaborasi membangun berbagai software FOSS.

## PEMBUATAN PAKET DEBIAN DARI KODE SUMBER

Tutorial ini berisi langkah-langkah untuk membuat Paket Debian dari kode sumber software. Seperti yang dijelaskan sebelumnya, cara ini merupakan cara umum yang digunakan oleh para *maintainer* untuk membuat paket Debian.

Dalam tutorial ini, akan digunakan contoh software bernama GNU Hello yang merupakan software yang berjalan pada terminal untuk menampilkan teks “Hello, World!”. Cara yang dilakukan untuk software lainnya mungkin saja berbeda dengan cara yang diberikan, terutama untuk software yang memerlukan pustaka yang banyak serta sangat kompleks (seperti aplikasi berbasis GUI). Mungkin saja ada beberapa berkas yang memerlukan proses modifikasi agar proses kompilasi menjadi lebih lancar. Namun, alur kerja untuk membangun paket Debian dari kode sumber tetaplah sama untuk semua jenis software.

### Pasang paket-paket yang diperlukan

Langkah pertama dalam membangun paket Debian adalah memasang beberapa software yang dibutuhkan dalam membangun paket Debian. Semua software yang dibutuhkan sudah tersedia pada *repository* Debian, Ubuntu maupun BlankOn. Berikut adalah daftar software yang dibutuhkan beserta fungsinya :

- `build-essential`, merupakan metapackage yang merujuk ke berbagai paket-paket yang umumnya digunakan dalam melakukan kompilasi suatu software.
- `cdbs`, berisikan kumpulan-kumpulan skrip siap pakai untuk mempermudah anda dalam membangun suatu paket Debian dari kode sumber dan menyederhanakan penggunaan `dehelper` dalam penulisan aturan-aturan pembangunan paket Debian yang tersimpan pada berkas “`debian/rules`”.
- `devscripts`, berisikan berbagai perintah-perintah yang sangat membantu tim pemaket, namun tidak begitu penting.
- `fakeroot`, mensimulasikan pekerjaan sebagai root atau administrator. Proses kompilasi tentunya membutuhkan kewenangan sebagai root yang bisa disimulasikan melalui software ini.
- `gnupg`, berfungsi untuk membuat kunci GNUPG yang nantinya akan digunakan sebagai identifikasi diri anda sebagai pemaket.
- `dehelper`, berfungsi untuk mempermudah pembangunan paket Debian dari kode sumber dari software yang aturan-aturannya sudah ditulis pada berkas “`debian/rules`”.

Untuk memasang software yang disebutkan di atas, ketik perintah berikut pada terminal :

```
$ sudo apt-get install build-essential cdbs devscripts fakeroot gnupg dehelper
```

### Membuat identitas sebagai Pemaket

Langkah kedua dalam melakukan pembuatan paket Debian adalah menuliskan nama lengkap dan E-Mail anda sebagai informasi bahwa anda yang melakukan pemaketan atau *maintaining* dari suatu software. Untuk melakukan hal ini, buka berkas `.bashrc` pada folder home melalui terminal dengan perintah berikut :

```
$ nano .bashrc
```

Kemudian pada akhir dari isi berkas tersebut, isilah keterangan berikut :

```
export DEBFULLNAME="Nama lengkap pemaket"
export DEBEMAIL="email@pemaket.com"
```

Simpan berkas tersebut, kemudian ketik perintah berikut pada terminal :

```
$ source .bashrc
```

Selain itu, anda juga perlu membuat kunci GPG untuk mengidentifikasi bahwa paket yang anda buat beserta perubahan-perubahan yang dilakukan merupakan hasil kerja anda. Untuk



menaruhnya di dalam folder Home jika memang tidak memiliki hak akses ke root. Setelah proses tersebut, maka anda akan mendapatkan folder baru dengan format nama : namapirantilunak-nomorversi. Misalnya, hello-4.5 untuk GNU Hello versi 4.5.

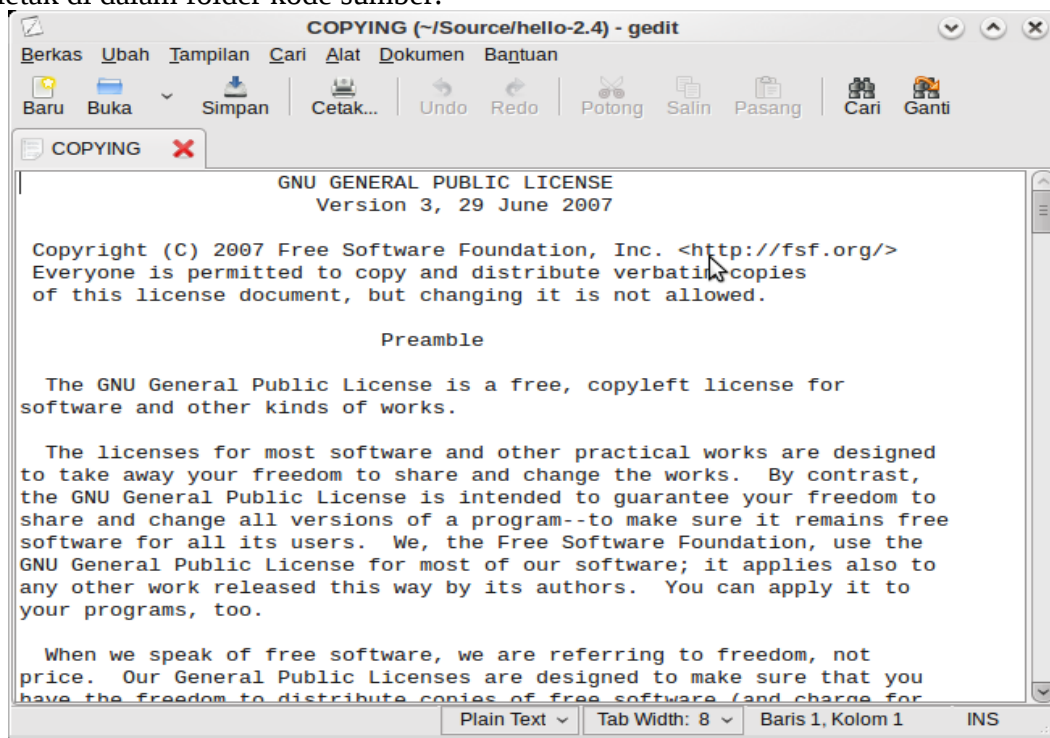
### Modifikasi software (Jika perlu)

Jika anda ingin melakukan modifikasi dari software yang tersebut, anda bisa memodifikasinya sesuai dengan kebutuhan anda. Modifikasi bisa dilakukan pada kode program, bahasa, artwork, dll. Untuk melakukan hal tersebut, tentunya anda memerlukan kemampuan tertentu, tergantung pada apa yang ingin anda modifikasi. Misalnya jika anda ingin menyunting program utama, anda harus menguasai bahasa pemrograman yang digunakan untuk menulis program bersangkutan.

### Mengkonversi paket source menjadi paket source Debian

Paket kode sumber (source code) terlebih dahulu harus dikonversi menjadi paket sumber Debian agar nantinya paket software dapat dibangun secara mudah.

Sebelum melakukan konversi, anda terlebih dahulu harus mencatat lisensi apa yang digunakan pada kode sumber tersebut. Untuk melakukan hal ini, anda bisa baca berkas COPYING yang terletak di dalam folder kode sumber.



#### *Lisensi dari suatu software*

Pada contoh ini, software “Hello” menggunakan lisensi GPL (GNU General Public License) versi 3.0, lisensi yang umum digunakan oleh software *open source*.

Kemudian, anda bisa melakukan konversi melalui terminal dengan masuk terlebih dahulu ke folder kode sumber menggunakan perintah “cd nama-folder” seperti berikut :

```
$ cd hello-4.5/
```

Lalu, ketik perintah “dh\_make -c lisensi --createorig” untuk melakukan konversi. Opsi “--create-orig” berfungsi agar dh\_make membuat berkas kode sumber asli, yang belum dimodifikasi menjadi paket source Debian. Berkas kode sumber asli akan diakhiri dengan ekstensi \*orig.tar.gz

```
$ dh_make -c gpl --createorig
```

DH\_MAKE akan menanyakan jenis paket yang akan dikonversi. Karena kita akan menggunakan kumpulan skrip CDBS (Common Debian Build Systems) untuk mempermudah dalam membuat skrip “rules” yang berisikan aturan-aturan dalam pembuatan paket Debian, ketik



huruf “b” dan tekan tombol Enter. Anda boleh saja memilih opsi “Single binary”, namun prosesnya sedikit agak rumit karena memerlukan penyuntingan skrip “rules” secara manual.

```
wira@wira-laptop:~/Source/hello-2.4$ dh_make --createorig -c gpl

Type of package: single binary, multiple binary, library, kernel module,
kernel patch or cdb?
[s/m/l/k/n/b] b

Maintainer name : Putu Wiramaswara Widya
Email-Address   : wira@unknown
Date            : Sun, 14 Jun 2009 09:44:59 +0800
Package Name    : hello
Version         : 2.4
License         : gpl
Using dpatch    : no
Using quilt     : no
Type of Package : cdb?
Hit <enter> to confirm:
Skipping creating ../hello_2.4.orig.tar.gz because it already exists
Done. Please edit the files in the debian/ subdirectory now. hello
uses a configure script, so you probably don't have to edit the Makefiles.
wira@wira-laptop:~/Source/hello-2.4$
```

Program `dh_make` berhasil dijalankan. Sekarang, anda akan melihat folder baru bernama “debian”. Anda bisa mencoba mengetik perintah “ls” untuk menampilkan isi berkas dan folder yang terdapat pada folder kode sumber.

```
$ ls
ABOUT-NLS  ChangeLog.O  configure    doc           Makefile.in  src
aclocal.m4  config.h     configure.ac  gnulib       man           stamp-h1
AUTHORS     config.in    contrib      INSTALL      NEWS         tests
build-aux   config.log   COPYING      Makefile     po           THANKS
ChangeLog   config.status debian       Makefile.am  README      TODO
```

Sekarang, masuk ke folder “debian” dengan mengetik perintah “cd debian”

```
$ cd debian
$ ls
changelog  docs          info          menu.ex      rules
compat     emacsen-install.ex  init.d.ex    postinst.ex  watch.ex
control    emacsen-remove.ex  init.d.lsb.ex  postrm.ex
copyright  emacsen-startup.ex  manpage.1.ex  preinst.ex
cron.d.ex  hello.default.ex   manpage.sgml.ex  prerm.ex
dirs       hello.doc-base.EX  manpage.xml.ex  README.Debian
```

Pada folder tersebut terdapat beberapa berkas yang diperlukan untuk membangun paket Debian dari software “hello”. Berikut adalah penjelasannya :

- changelog, berisikan daftar perubahan yang dilakukan pada
- control, berisikan berbagai informasi dari paket yang akan dibangun nantinya, seperti nama paket, versi, deskripsi serta ketergantungan paket.
- copyright, berisikan informasi hak cipta dari software.
- README.Debian, berisikan informasi-informasi yang ditulis oleh maintainer, misalnya berisikan perubahan apa saja yang ada serta apa saja masalah yang mungkin akan dialami.
- rules, merupakan skrip Makefile yang berisikan aturan-aturan dalam melakukan pembangunan suatu paket Debian. Penyuntingan berkas ini adalah yang paling sulit dalam pembangunan paket Debian, namun dengan menggunakan skrip dari CDBS, pembangunan paket Debian menjadi lebih mudah.

```
#!/usr/bin/make -f
include /usr/share/cdb/1/rules/debhelper.mk
include /usr/share/cdb/1/class/autotools.mk

# Add here any variable or target overrides you need.
```

### **Berkas rules yang sudah diisi oleh skrip dari CDBS**

- dirs, berikan daftar folder yang akan dibuat pada saat proses pemasangan paket debian nantinya, anda cukup sebutkan folder-folder yang secara default tidak ada di sistem.
- docs, berisi daftar berkas dokumentasi yang akan dimasukkan dalam paket Debian, seperti daftar BUG, TODO List, README, dsb.

Selain berkas-berkas tersebut, terdapat juga berkas lainnya yang berekstensi \*.ex atau \*.EX yang akan berfungsi jika ekstensi \*.ex tersebut dihilangkan. Karena program hello tidak membutuhkan skrip-skrip tersebut, kita bisa menghapusnya dengan mengetik perintah berikut :

```
$ rm *.ex *.EX
```

### **Mengubah changelog**

Apa saja perubahan yang dilakukan pada software harus dicatat pada berkas yang bernama changelog. Untuk mengubahnya, silahkan ketik perintah berikut (ganti perintah nano dengan editor teks favorit anda) :

```
$ nano changelog
```

Isi dari changelog kira-kira seperti berikut :

```
hello (2.4-1) unstable; urgency=low
```

```
* Initial release (Closes: #nnnn) <nnnn is the bug number of your ITP>
```

```
-- Putu Wiramaswara Widya <wira@unknown> Sun, 14 Jun 2009 19:58:10 +0800
```

Setiap perubahan yang kita lakukan kita tulis sebagai daftar (yang diawali dengan tanda bintang “\*”). Anda juga perlu menaikkan nomor versi jika memang ada perubahan dari kode sumber. Sesuaikan nomor versi dengan aturan dari distribusi bersangkutan. Berikut adalah contoh berkas changelog setelah diubah :

```
hello (2.4-1) unstable; urgency=low
```

```
* Initial release
```

```
* Converted from tar.gz tarball file, no change
```

```
-- Putu Wiramaswara Widya <wira@unknown> Sun, 14 Jun 2009 19:58:10 +0800
```

Setelah selesai menyunting changelog, anda bisa langsung menyimpannya.

### **Mengubah berkas control**

Berkas control memuat nama paket, versi, deskripsi, paket ketergantungan (*dependency*) serta informasi penting lainnya dari suatu paket. Untuk mengubah isi berkas control, ketik perintah

berikut (Ganti perintah nano dengan editor teks favorit anda) :

```
$ nano control
```

Isi dari berkas control kira-kira seperti berikut :

```
Source: hello  
Section: unknown  
Priority: extra  
Maintainer: Putu Wiramaswara Widya <wira@unknown>  
Build-Depends: cdb, debhelper (>= 7), autotools-dev  
Standards-Version: 3.8.0  
Homepage: <insert the upstream URL, if relevant>  
  
Package: hello  
Architecture: any  
Depends: ${shlibs:Depends}, ${misc:Depends}  
Description: <insert up to 60 chars description>  
<insert long description, indented with spaces>
```

Bagian pertama (yang dicetak tebal) pada berkas control menyebutkan informasi paket kode sumber yang akan dikompilasi. Disini anda bisa mengubah nama paket source (semestinya tidak perlu), section (bagian software), priority (prioritas software), Maintainer (nama pemaket serta e-mail), Build-Depends (paket-paket yang diperlukan dalam pembangunan).

Bagian kedua dan selanjutnya (yang diharis bawah) berisikan informasi dari paket software siap instal yang akan dibangun. Karena paket yang akan dibangun hanya paket binernya saja, maka hanya ada satu bagian saja yang dibuat. Pada bagian ini, kita bisa tentukan Package (nama paket), Architecture (software akan dibangun pada arsitektur mesin komputer apa saja), Depends (mengatur paket-paket software yang diperlukan agar software yang akan dikompilasi berjalan lancar), dan Description (berisikan deskripsi dari software tersebut).

Anda bisa mengubah bagian description dengan merujuk pada berkas README dari kode sumber. Contoh dari bagian description sebagai berikut :

```
Description: Hello prints a friendly greeting.  
Hello prints a friendly greeting It also serves as a sample GNU  
package, showing practices that may be useful for GNU projects.
```

## Mengubah berkas copyright

Copyright berisikan informasi-informasi hak cipta dari software. Untuk mengedit berkas ini, anda bisa ketik perintah berikut (Ganti perintah nano dengan editor teks favorit anda) :

```
$ nano copyright
```

Isi dari contoh berkas copyright adalah sebagai berikut :

```
This package was debianized by Putu Wiramaswara Widya <wira@unknown> on  
Sun, 14 Jun 2009 19:58:10 +0800.
```

```
It was downloaded from <url://example.com>
```

```
Upstream Author(s):
```

```
<put author's name and email here>  
<likewise for another author>
```

```
Copyright:
```

```
<Copyright (C) YYYY Name OfAuthor>  
<likewise for another author>
```

```
License:
```

```
<Put the license of the package here indented by 4 spaces>
```

```
The Debian packaging is copyright 2009, Putu Wiramaswara Widya  
<wira@unknown> and  
is licensed under the GPL, see `/usr/share/common-licenses/GPL'.
```

```
# Please also look if there are files or directories which have a  
# different copyright/license attached and list them here.
```

Anda perlu mengubah beberapa bagian yang ada pada berkas copyright tersebut. Beberapa bagian yang perlu anda ubah adalah sbb :

- “It was downloaded from <url://example.com>”, Ganti url://example.com dengan situs dimana anda mendapatkan kode sumber software.
- “Upstream author:”, isikan dengan nama penulis software. Anda bisa melihatnya melalui berkas README atau AUTHORS dari kode sumber.
- “Copyright:”, isikan dengan pemilih hak cipta dari software. Anda bisa melihatnya melalui berkas README atau AUTHORS dari kode sumber.
- “License:”, isikan dengan deskripsi lisensi dari software. Anda bisa melihatnya melalui berkas README atau melalui berkas kode sumber utama dari software.

Berikut adalah contoh modifikasi dari berkas copyright :

```
This package was debianized by Putu Wiramaswara Widya <wira@unknown> on  
Sun, 14 Jun 2009 19:58:10 +0800.
```

```
It was downloaded from <http://www.gnu.org>
```

```
Upstream Author(s):
```

```
Mike Haertel  
David MacKenzie  
Jan Brittonson  
Roland McGrath  
Charles Hannum  
Bruce Korb  
Karl Eichwalder  
Karl Berry  
The King
```

```
Copyright:
```

```
Copyright (C) 1999, 2005, 2006 Free Software Foundation, Inc.
```

```
License:
```

```
This program is free software: you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation, either version 3 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
The Debian packaging is copyright 2009, Putu Wiramaswara Widya  
<wira@unknown> and  
is licensed under the GPL, see `/usr/share/common-licenses/GPL'.
```

```
# Please also look if there are files or directories which have a
# different copyright/license attached and list them here.
```

Setelah selesai, anda bisa menyimpan berkas copyright tersebut.

### Mengubah berkas rules

Berkas rules berisikan aturan-aturan yang akan dilakukan dalam pembangun paket paket Debian dari suatu software. Karena sebelumnya sudah menggunakan opsi CDBS pada saat proses konversi, anda hanya akan melihat beberapa baris saja pada berkas rules yang intinya akan memuat skrip-skrip CDBS untuk membantu proses pembangunan. Anda boleh saja mengabaikan perubahan berkas ini.

Namun jika anda ingin memerlukan opsi khusus dalam pembangunan paket Debian, anda boleh mengubahnya. Tata cara dalam perubahan berkas rules bisa anda lihat pada dokumentasi Debian New Maintainer Guide atau dokumentasi CDBS.

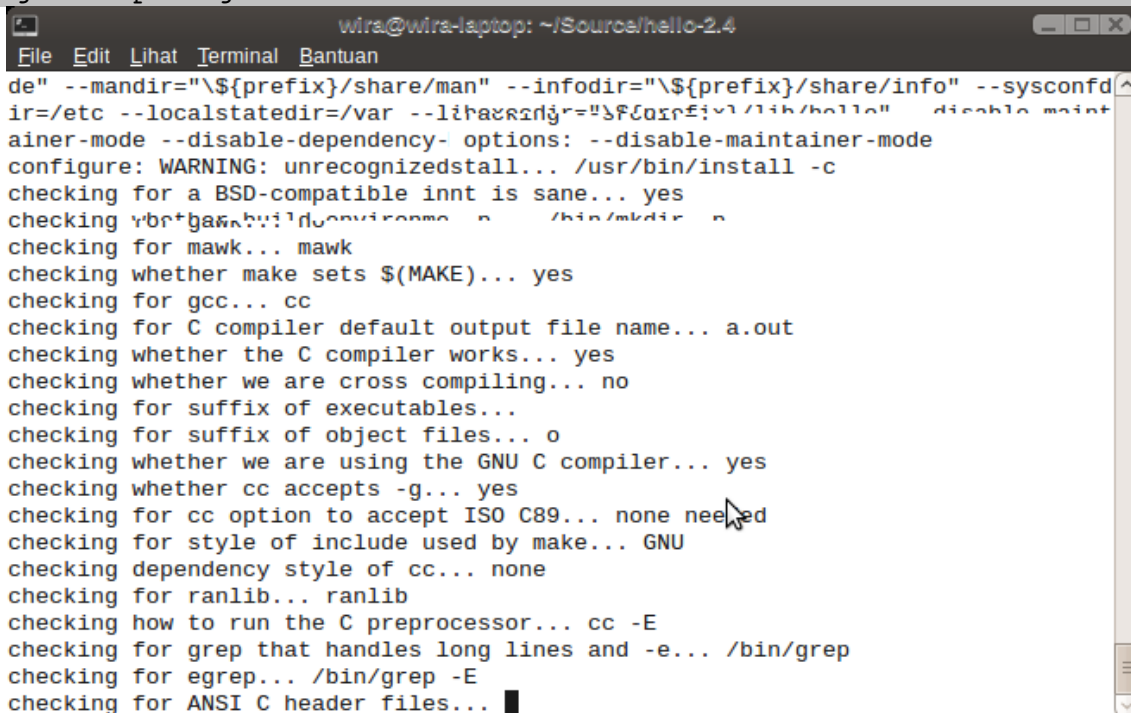
### Membangun paket Debian

Langkah terakhir yang dilakukan adalah membangun paket Debian dari paket kode sumber yang sudah dikonversi sebelumnya menjadi paket kode sumber Debian. Untuk melakukan hal ini, naik ke folder paling atas dari folder kode sumber dengan perintah berikut :

```
$ cd ..
```

Kemudian, anda bisa ketik perintah berikut untuk memulai membangun Paket Debian :

```
$ dpkg-buildpackage -rfakeroot
```



```
wira@wira-laptop: ~/Source/hello-2.4
File Edit Lihat Terminal Bantuan
de" --mandir="\${prefix}/share/man" --infodir="\${prefix}/share/info" --sysconfd
ir=/etc --localstatedir=/var --libexecdir="\${prefix}/lib/hello" --disable-maint
ainer-mode --disable-dependency- options: --disable-maintainer-mode
configure: WARNING: unrecognizedstall... /usr/bin/install -c
checking for a BSD-compatible innt is sane... yes
checking for mawk... mawk
checking whether make sets $(MAKE)... yes
checking for gcc... cc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether cc accepts -g... yes
checking for cc option to accept ISO C89... none needed
checking for style of include used by make... GNU
checking dependency style of cc... none
checking for ranlib... ranlib
checking how to run the C preprocessor... cc -E
checking for grep that handles long lines and -e... /bin/grep
checking for egrep... /bin/grep -E
checking for ANSI C header files... █
```

### *Proses kompilasi dan pembangunan paket Debian*

Proses kompilasi software dan pembangun Paket Debian akan berjalan, tunggu sebentar hingga proses kompilasi berjalan dengan sukses. Setelah proses pembangunan selesai, maka akan dihasilkan beberapa berkas yang terletak pada folder atas. Berikut adalah berkas-berkas yang dimaksud :

- *hello\_2.4.orig.tar.gz*, merupakan berkas kode sumber asli yang masih belum dikonversi menjadi berkas kode sumber Debian.
- *hello\_2.4-1.diff.gz*, berisikan patch yang akan mengubah berkas kode sumber asli menjadi berkas kode sumber Debian.

- *hello\_2.4-1.dsc*, berisikan ringkasan dari paket kode sumber yang dikompilasi dan dibangun menjadi paket Debian. Berkas ini sudah dikunci dengan kunci GPG yang sebelumnya dibuat.
- *hello\_2.4-1.changes*, berisikan perubahan-perubahan yang dilakukan setiap membangun paket Debian dari software. Berkas ini juga dikunci dengan kunci GPG
- *hello\_2.4-1\_i386.deb*, merupakan paket Debian yang sudah dibuat dan siap untuk dipasang.

Jika anda menjadi seorang *maintainer* atau pemaket, anda hanya mengunggah berkas *orig.tar.gz*, *diff.gz*, *dsc* dan *changes* yang sudah dibangun ke server penyedia distribusi Linux bersangkutan (e.q. Ubuntu, Debian, BlankOn dll), karena nantinya berkas “deb” akan diolah oleh server tersebut untuk semua jenis arsitektur yang didukung. Namun sebelum diunggah, tidak ada salahnya untuk dicoba terlebih dahulu menggunakan software bantu seperti : *pbuilder*, *lintian*, *gdebi*, dll.

## PENUTUP

Demikianlah penjelasan mengenai pengenalan paket Debian dan tutorial tentang cara membangun paket Debian dari kode sumber suatu software. Semoga semuanya bermanfaat bagi anda untuk sekedar menambah ilmu atau ingin serius menjadi seorang *maintainer* atau pemaket.

Jika anda ingin menjadi seorang pemaket yang handal, anda bisa membaca beberapa referensi berikut untuk menambah ilmu anda dalam menjadi pemaket :

- Debian New Maintainer Guide, merupakan panduan yang sangat cocok untuk menjadi seorang maintainer. Anda bisa memasang paket “maint-guide” melalui APT untuk membaca dokumentasi ini.
- Debian Policy, merupakan dokumentasi yang berisikan aturan-aturan dalam pemaketan Debian.
- CDBS Documentation, berisikan panduan-panduan dalam menggunakan skrip CDBS (Common Debian Build System) dalam membantu membangun suatu paket Debian. Silahkan cari di Google dengan kata kunci “CDBS Documentation”.
- Ubuntu MOTU School, merupakan sekolah bagi anda ingin menjadi tim MOTU atau pemaket di Ubuntu, silahkan buka website <http://wiki.ubuntu.com/MOTU/School> untuk membukanya.
- Dokumentasi pemaket BlankOn, berisikan berbagai informasi untuk menjadi pemaket di BlankOn, silahkan buka website <http://dev.blankonlinux.or.id/wiki/Pemaket> untuk membuka dokumentasi ini.

Semoga anda bisa bergabung menjadi tim pemaket atau *maintainer* pada distribusi BlankOn, Ubuntu, Debian, dsb. Jika anda memiliki masalah, anda bisa meminta bantuan kepada para pemaket yang ahli.