

The Reduction Method of ISO image for Operating System of Open Source Based on Linux

Metode Reduksi ISO Image Sistem Operasi Open Source Berbasis Linux

Nana Suryana ^{*,1}, Didi Rosiyadi¹

¹ Pusat Penelitian Informatika
Lembaga Ilmu Pengetahuan Indonesia
Gedung 20 Lt 3, Jalan Sangkuriang 154 D, Bandung 40135
Indonesia

Abstract

In this paper, the Reduction Method of ISO image for Operating System of Linux Distro (Open Source Based on Linux) is proposed. IGOS Nusantara as a Linux Distro is used on this study. The reduction method of ISO image for IGOS Nusantara consists of modules elimination and usage of an optimal compression technique. Using the reduction method optimally on the rpm files of the application packages of IGOS Nusantara will decrease their sizes. Experiment results demonstrate that some of the rpm files can be reduced from 40 MB to 29 MB. All reduced rpm files and other modules are processed into ISO image. By this method, the original size 890 MB of ISO image can be shrunked into 685 MB. The small and compact size of ISO image are very useful to save the internet bandwidth in downloading IGOS Nusantara and also to make it more responsive when it is used by users.

Key Words: Linux Distro, IGOS Nusantara, ISO image, rpm files, operating system, Open Source

Abstrak

Tulisan ini menjelaskan tentang strategi dalam pengembangan Distro Linux agar memiliki ukuran kecil. Distro Linux yang dikembangkan adalah IGOS Nusantara. Distro berukuran kecil ditandai dengan ukuran berkas ISO *image* yang juga kecil. Metode reduksi ukuran ISO *image* meliputi eliminasi modul-modul yang tidak diperlukan dan menerapkan teknik kompresi yang optimal. Pemakaian metode reduksi pada berkas rpm dari paket-paket aplikasi IGOS Nusantara, terbukti dapat memperkecil ukurannya. Beberapa berkas rpm yang semula berukuran 40MB, dapat tereduksi menjadi 29MB. Seluruh berkas rpm tereduksi dan beragam modul lainnya kemudian diproses menjadi sebuah ISO *image*. Setelah memakai metode reduksi, ISO *image* yang semula berukuran 890MB, kini telah berkurang menjadi 685MB. Ukuran yang kecil dan ringkas pada akhirnya akan menghemat penggunaan *bandwidth* internet yang dipakai saat mengunduh IGOS Nusantara dan membuatnya lebih responsif saat dipakai oleh pengguna.

Kata kunci: Distro Linux, IGOS Nusantara, ISO *image*, berkas rpm, sistem operasi, *Open Source*

1. PENDAHULUAN

Penggunaan Sistem Operasi *Open Source* berbasis Linux atau Distro Linux mengalami pertumbuhan yang tinggi. Pertumbuhan ini disebabkan oleh adanya kebebasan dalam hal lisensi. Lisensi *Open Source* mengizinkan setiap pengembang perangkat lunak melakukan modifikasi dan distribusi. Kebebasan tersebut mendorong pengembang *Open Source* untuk membuat Distro Linux yang unggul

atau lebih baik dibandingkan dengan sistem operasi lain [1], [2]. Pengembangan distro linux mempunyai tantangan antara lain (1) ukuran aplikasi semakin bertambah besar, dan (2) adanya kebutuhan untuk memasukkan sebanyak mungkin aplikasi ke dalam satu distro Linux. Peningkatan penggunaan *Open Source* juga disebabkan oleh kualitas yang baik dari perangkat lunaknya. Ini terbukti dengan penggunaan perangkat lunak *Open Source* seperti *Web Server Apache* yang dipakai sebesar 60% di seluruh dunia [3], [4].

Tantangan-tantangan yang muncul dalam pengembangan distro tersebut berpengaruh terhadap besarnya ISO *image* yang dihasilkan. Ukuran ISO *image* pada akhirnya berpengaruh terhadap

*Corresponding Author. Tel: +6222-2504711

Email: ns@informatika.lipi.go.id

Received: 7 Sep 2012; revised: 4 Oct 2012; accepted: 21 Oct 2012

Published online: 26 Nov 2012

© 2012 INKOM 2012/13-NO193

responsifitas sistem operasi pada saat dipakai oleh pengguna. Ukuran ISO *image* yang besar akan memerlukan koneksi internet yang lebih besar pada saat diunduh. Sistem operasi yang tidak responsif akan mengurangi peluang distro Linux dipakai oleh pengguna. Terkait hal ini, beberapa pengembang Distro Linux berupaya untuk mengembangkan distro dengan banyak aplikasi dan tetap responsif.

Penelitian pembuatan atau pengembangan Distro Linux di Indonesia diantaranya telah dilakukan oleh Ahmad Sofyan [5]. Pengembangan Distro Linux yang telah dilakukan oleh Ahmad Sofyan dipakai sebagai pembanding besarnya ISO *image* yang dihasilkan dan fitur yang ada di distro Linux. Penelitian lain yang terkait adalah *benchmark* terhadap beberapa metode kompresi [6]. *benchmark* menghasilkan rasio kompresi yang paling optimal. Hasil *benchmark* kemudian dipakai untuk mendapatkan metoda yang cocok dalam penelitian ini.

2. LANDASAN TEORI

2.1 Kompresi Berkas

Kompresi merupakan pengurangan ukuran suatu berkas menjadi ukuran yang lebih kecil dari aslinya. Kompresi berkas akan memiliki rasio tinggi saat terdapat suatu berkas yang berukuran besar dan data di dalamnya mengandung banyak pengulangan karakter. Teknik kompresi adalah mengganti karakter yang berulang-ulang tersebut dengan suatu pola tertentu sehingga berkas memiliki ukuran lebih kecil.

Ada dua jenis kompresi data, yaitu kompresi tanpa kehilangan (*lossless data compression*) dan kompresi berkehilangan (*lossy data compression*). *Lossy compression* menyebabkan adanya perubahan data dibandingkan sebelum dilakukan proses kompresi. Sebagai gantinya, *lossy compression* memberikan tingkat kompresi yang lebih tinggi. Tipe ini cocok untuk kompresi file suara digital dan gambar digital. File suara dan gambar secara alamiah masih bisa digunakan walaupun tidak berada pada kondisi yang sama sebelum dilakukannya kompresi.

Sebaliknya, *Lossless compression* memiliki derajat kompresi yang lebih rendah tetapi dengan akurasi data yang terjaga antara sebelum dan sesudah proses kompresi. Kompresi ini cocok untuk basis data, dokumen atau spreadsheet. Pada *lossless compression* ini tidak diijinkan adanya bit yang hilang dari data pada proses kompresi.

Saat ini terdapat berbagai tipe algoritma kompresi, antara lain: Huffman, LIFO, ZHUF, LZ77 dan variannya (LZ78, LZW, GZIP), Dynamic Markov Compression (DMC), Block-Sorting Lossless, Run- Length, Shannon-Fano, Arithmetic, PPM

(Prediction by Partial Matching), Burrows-Wheeler Block Sorting, dan Half Byte.

Untuk mengkompresi data, ada beberapa format file yang digunakan, antara lain: bzip2 (.bz2), gzip (.gz), lzma (.lzma), lzo (.lzo), pack (.z), compress (.Z). Perbedaan masing-masing format kompresi ini adalah algoritma yang digunakan. Seperti misalnya bzip2 yang menggunakan Burrows-Wheeler transform diikuti dengan move-to-front transform dan terakhir Huffman coding. Format gzip yang menggunakan algoritma DEFLATE untuk kompresi data, lzma menggunakan algoritma 7-zip, dan lzo menggunakan algoritma LZ0. Beberapa dari format kompresi data ini terkadang digunakan bersama-sama ketika meng-archive file. Seperti misalnya Tape Archiver (.tar) yang digunakan bersama dengan bzip2 (ekstensi file menjadi .tar.bz2), atau dengan gzip (ekstensi file menjadi .tar.gz) atau dengan compress (ekstensi file menjadi .tar.Z) [7].

2.2 Paket Aplikasi

Paket Aplikasi adalah perangkat lunak berikut metadata, yaitu nama lengkap perangkat lunak yang bersangkutan, keterangan mengenai kegunaannya, nomor versi, pemasok (vendor), checksum, dan daftar dependensi yang diperlukan untuk menjalankan perangkat lunak tersebut dengan benar. Setelah instalasi, metadata disimpan dalam database paket lokal. Untuk menambah atau menghapus aplikasi di distro Linux, pengguna menggunakan package manager, yaitu program untuk memasang dan menghapus perangkat lunak, dan mencari komponen yang dibutuhkan oleh setiap perangkat lunak.

Secara umum ada tiga jenis paket yang dapat dipasang, yaitu (1) Source tarball (.tgz, .bz, .bz2), yaitu kode sumber yang harus dipasang secara manual dan terdiri dari tahap configure, make, dan make install. (2) Binary (.deb, .rpm, .pkg), yaitu paket instalasi yang telah dibuat dan dikhususkan untuk distribusi Linux tertentu. (3) Repository, yaitu bundle paket dan plugins yang dikumpulkan dan diakses melalui package manager.

2.3 Repositori

Ada ribuan program yang tersedia untuk dipasang di distro Linux. Program ini disimpan dalam bentuk arsip perangkat lunak (repositori) dan tersedia bebas untuk dipasang melalui Internet. Repositori mempermudah memasang program baru dalam distro Linux. Pemasangan program dari repositori resmi membuat sistem lebih aman, karena setiap program yang dipasang telah dibangun khusus untuk distro Linux dan diperiksa sebelum program tersebut diinstalasi. Untuk mengelola perangkat

lunak, repositori dikelompokkan ke dalam empat grup, yaitu *all*, *extra*, *other*, dan *update*.

Dasar pemikiran yang digunakan untuk menentukan perangkat lunak mana yang akan masuk dalam kategori mana adalah berdasarkan pada tingkat dukungan dari tim pengembangan perangkat lunak yang tersedia untuk suatu program, dan penilaian terhadap terpenuhi atau tidaknya program pada Filosofi Perangkat Lunak Bebas.

2.4 RPM Package Manager

Distro Linux yang dikembangkan dan dibahas dalam penelitian ini adalah IGOS Nusantara yang disingkat IGN. Aplikasi yang ada di IGN sebagian besar dikemas dalam bentuk berkas rpm. RPM Package Manager (awalnya bernama RedHat Package Manager, disingkat RPM) adalah sebuah sistem manajemen paket. Nama RPM mengacu kepada dua hal yaitu format berkas paket perangkat lunak, dan sebuah alat bantu perangkat lunak bebas yang dipakai untuk melakukan instalasi, deinstalasi, verifikasi dan kueri paket perangkat lunak dalam format ini. Berkas RPM memakai format dasar yang sesuai dengan standar dari Linux Standard Base [8].

RPM ditulis pada tahun 1997 oleh Erik Troan dan Marc Ewing untuk Perusahaan Red Hat. RPM pertama kali dipakai di Red Hat Linux. Saat ini format rpm digunakan oleh banyak distribusi Linux. RPM juga telah diadaptasi ke sistem operasi lain seperti Novell Netware (versi 6.5 SP3) dan IBM AIX versi 5.

2.5 Modul dalam RPM

Berkas rpm dapat berisi modul hasil kompilasi kode program (source code). Berkas rpm juga dapat berisi modul atau file dokumentasi, file grafik, file audio, file video, file terjemahan dari tiap negara dan lainnya [8]. Berkas rpm untuk dipakai pengguna global atau pengguna di banyak negara akan memasukkan berkas-berkas yang spesifik untuk tiap negara. Satu berkas rpm dapat memiliki puluhan berkas terjemahan bahasa untuk puluhan negara.

Berkas-berkas yang dikemas dalam rpm untuk kondisi tertentu tidaklah diperlukan. Distro Linux seperti IGOS Nusantara yang ditujukan untuk pengguna di Indonesia hanya memerlukan berkas tertentu saja. Pengguna Distro Linux di Indonesia sebegini besar hanya memerlukan berkas berbahasa Indonesia dan satu berkas berbahasa Inggris.

Besar atau kecilnya hasil pengurangan berkas sangat bervariasi. Beberapa berkas terjemahan hanya berukuran beberapa KB, sedangkan beberapa berkas terjemahan lainnya berukuran puluhan MB. Dokumentasi aplikasi yang menyertakan tampilan program dalam bentuk grafik akan berukuran lebih

besar dibanding dokumentasi dalam bentuk teks saja.

Hasil kompilasi kode program dan atau berkas lainnya selanjut dikemas dan dikompresi agar rpm berukuran kecil. Pada awalnya rpm memakai metode kompresi *gzip*. Sejak tahun 2009 dimulai upaya memakai metode kompresi yang lebih baik, yaitu memakai kompresi *xz*. Kompresi *xz* adalah pengembangan dari kompresi *lzma*.

2.6 Proses *build ISO image*

Berkas rpm dan berkas lainnya yang sudah dikumpulkan, selanjutnya diproses untuk pembuatan *ISO image*. *ISO image* adalah jenis file arsip yang berisi salinan seluruh isi sistem operasi yang dihasilkan oleh proses *build*. Saat proses berlangsung, berkas-berkas rpm dipasang atau diinstalasi ke dalam satu lingkungan khusus atau *chroot*. Setelah sistem operasi lengkap terbentuk dalam *chroot*, selanjutnya dilakukan penggabungan seluruh berkas yang ada di *chroot* menjadi satu berkas berukuran besar. Berkas berukuran besar inilah yang disebut file *ISO image*.

Pada Gambar 1 terlihat *ISO image* dihasilkan dengan menggunakan program *livecd-creator*. Program *livecd-creator* memakai atau menerapkan konfigurasi yang ada di berkas *kickstart* dengan nama *ign8.ks*.

3. METODE REDUKSI

Metode reduksi *ISO image* yang dilakukan dalam penelitian ini terdiri dari 5 tahapan yaitu: analisis aplikasi minimal, pemilihan metode kompresi, eliminasi modul yang tidak diperlukan, konfigurasi skrip dan ujicoba.

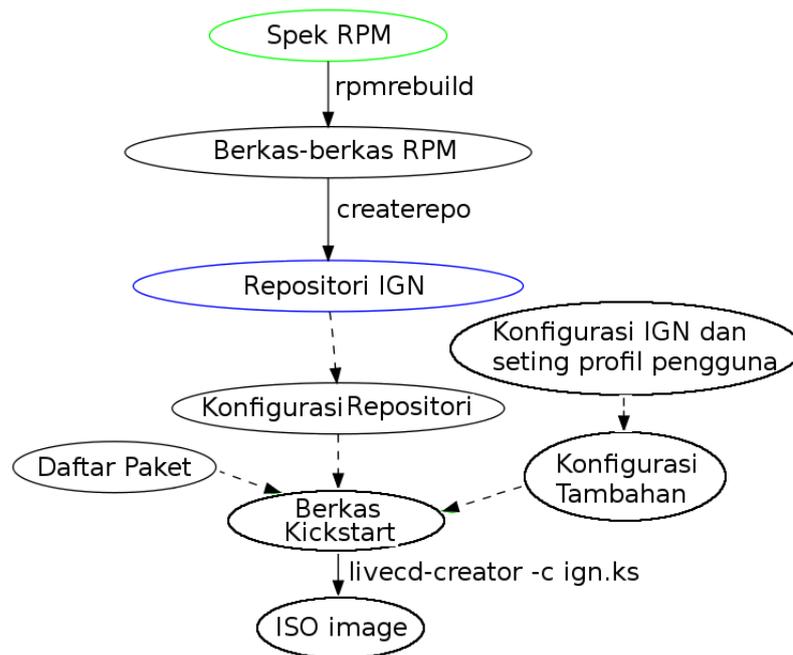
3.1 Aplikasi-aplikasi Minimal

Aplikasi minimal adalah program-program yang diperlukan oleh sistem operasi untuk masuk ke terminal. Aplikasi minimal diperoleh dengan cara memilih dan memasukkannya ke daftar paket, lalu disertakan pada berkas *kickstart*. Hasil yang diperoleh untuk kebutuhan minimal sistem operasi adalah sebanyak 220 berkas rpm.

3.2 Pemilihan Metode Kompresi

Pemilihan metode kompresi yang akan digunakan pada penelitian ini, didasarkan pada evaluasi terhadap hasil perbandingan atau *benchmark* metode kompresi *gzip*, *bzip2* dan *lzma*. Hasil kompresi terhadap kode program OpenOffice versi 1.1 yang berukuran 212.664.320 byte atau 203 MB terdapat pada Tabel I.

Angka satu sampai sembilan adalah tingkat kompresi yang dipakai. Terlihat bahwa metode kompresi *lzma* dengan tingkat kompresi sembilan



Gambar 1. Proses *build ISO image*

Tabel I. Hasil Kompresi OpenOffice dalam Byte

	gzip	bzip2	lzma
1	86322815	76147880	67456213
2	84858575	74320824	62085798
3	83561997	73467586	59547691
4	81312776	73044026	58245872
5	79798262	72762041	56694215
6	79179298	72540199	56182079
7	78995264	72512833	55535273
8	78816280	72314472	54678948
9	78768334	72223858	54068819

Sumber: [2]

menghasilkan ukuran yang paling kecil, yaitu 54.068.819 byte [2].

Rasio kompresi diperoleh dengan cara menghitung: Ukuran kompresi / Ukuran tanpa kompresi * 100%. Dari data pada Tabel I dapat diketahui bahwa rasio kompresi terbaik dengan memakai lzma pada tingkat sembilan adalah sebesar 25.4

Perbandingan atau *benchmark* kedua diamati dari hasil kompresi terhadap berkas *ISO image* yang aslinya berukuran 668 MB.

Pada Tabel II terlihat bahwa hasil maksimal diperoleh dengan menggunakan metode kompresi lzma dengan pemakaian block sebesar 1024KB, yaitu menghasilkan 162MB atau 25% [9].

Tabel II. Rasio Kompresi *ISO image*

method	block size	Slax data size	percent
uncompressed	-	668 MB	100%
mksquashfs+gzip	64 KB	227 MB	34%
mksquashfs+gzip	1024 KB	222 MB	33%
mksquashfs+ lzma	64 KB	191 MB	28%
mksquashfs+ lzma	128 KB	184 MB	27%
mksquashfs+ lzma	512 KB	172 MB	26%
mksquashfs+ lzma	1024 KB	162 MB	25%

Sumber: [9]

3.3 Eliminasi Modul

Eliminasi modul yang tidak diperlukan dapat dilakukan pada saat kompilasi/*build* aplikasi. Eliminasi dapat juga dilakukan melalui konfigurasi di berkas kickstart. Eliminasi dilakukan dengan membuang modul atau berkas terjemahan yang tidak dibutuhkan. Berkas yang dipakai hanyalah berkas terjemahan bahasa Indonesia dan bahasa Inggris.

3.4 Konfigurasi Skrip

Konfigurasi untuk melakukan reduksi modul yang tidak diperlukan disimpan dalam sebuah skrip. Berkas kickstart ini berisi skrip-skrip lainnya untuk melakukan proses *build* sesuai konfigurasi yang telah ditentukan. Skrip dibuat dalam bentuk skrip kickstart [10]. Skrip dalam bentuk bash memakai beberapa program, antara lain memakai grep [11] [12]. Skrip dimasukkan dalam blok `%post` sampai

dengan %end. Berikut ini skrip yang terkait untuk reduksi modul yang tidak dipakai di IGOS Nusantara:

```
%post
rpm -e --nodeps meld
rpm -e --nodeps lovelock-backgrounds-single
rpm -e --nodeps lovelock-backgrounds-gnome
rm -fR /usr/share/themes/Fedora
rm -fR /usr/share/themes/Glossy
rm -fR /usr/share/backgrounds/\textit{image}s
rm -fR /usr/share/backgrounds/tiles
rm -f /var/lib/rpm/__.db*
rm -f /var/lib/rpm/__.db*
rm -f /boot/initramfs*
localedef --list-archive |
  grep -v -i "id_ID.utf8|en_US.utf8" |
  xargs localedef --delete-from-archive
%end
```

Baris yang berisi instruksi localedef akan menampilkan semua berkas terjemahan yang ada, kemudian dengan perintah grep hanya akan mempertahankan berkas terjemahan bahasa Indonesia, yaitu id_ID.utf8 dan berkas terjemahan bahasa Inggris, yaitu en_US.utf8. Skrip untuk menghapus berkas terjemahan yang tidak terpakai selanjutnya digabung dengan skrip lain. Kemudian aplikasi livecd-creator dijalankan dengan memakai parameter atau konfigurasi gabungan. Hasil dari livecd-creator adalah berkas ISO image.

3.5 Ujicoba

Selanjutnya dilakukan uji coba terhadap ISO image yang telah dihasilkan. Uji ini dimaksudkan untuk memeriksa fungsi dari distro Linux dan untuk mendapatkan informasi tentang waktu yang diperlukan sejak komputer dinyalakan sampai muncul layar Desktop. Pengujian fungsi adalah untuk memastikan distro dapat dijalankan di beragam jenis komputer, yang meliputi beragam jenis motherboard, beragam prosesor, beragam ukuran RAM, beragam kartu VGA, chipset Wifi, GSM/CDMA modem dan lainnya. Pengujian tahap awal dilakukan dengan memakai Virtual Machine untuk mendapatkan hasil pengujian secara cepat, murah dan mudah. Setelah pengujian memakai Virtual Machine sukses, kemudian dilanjutkan dengan pengujian di Personal Computer (PC), Netbook, Laptop dan komputer lainnya yang tersedia di tempat penelitian. Pengujian lanjutan adalah memakai media yang berbeda, yaitu CD/DVD dan USB Flash Disk.

4. HASIL DAN PEMBAHASAN

Eliminasi berkas terjemahan (selain bahasa Indonesia dan bahasa Inggris) dari berkas rpm serta pemakaian metode kompresi lzma/xz terhadap

Google Chrome memberikan hasil sebagai berikut: Panjang berkas Google Chrome dari Google Inc. adalah 40 MB, sedangkan panjang berkas Google Chrome pada IGN8 adalah 29 MB. Terlihat bahwa metode kompresi xz menghasilkan rasio kompresi sebesar 72.5%, atau lebih dari seperempat panjang file Google Chrome dapat tereduksi.

Eliminasi melalui proses kompilasi berkas src.rpm menjadi rpm untuk Firefox hasilnya sebagai berikut: Panjang berkas Firefox dari Fedora 17 adalah 25MB, sedangkan berkas Firefox pada IGN8 adalah 4MB. Rasio kompresi untuk berkas rpm Firefox ini adalah 16%, atau lebih dari empat perlima panjang berkas telah tereduksi.

Pengujian memakai media CD/DVD hasilnya lebih lambat dibandingkan dengan memakai USB Flash Disk. Ini adalah hal yang logis, karena kecepatan pembacaan media CD/DVD melibatkan komponen mekanis yang tentunya lebih lambat.

Pengujian yang dilakukan di Personal Computer dan Notebook dengan prosesor setara Intel Core 2 Duo, RAM 2 GB, serta memakai CD/DVD, ternyata memerlukan waktu 3-5 menit bagi distro Linux untuk menyelesaikan proses boot sampai menampilkan Desktop.

Ketika melakukan proses build, banyak berkas rpm tereduksi menjadi ISO image LiveCD atau LiveDVD, sehingga ukurannya juga menjadi lebih kecil/ringkas. ISO image LiveDVD jika tanpa reduksi berukuran lebih dari 1014 MB. Pemakaian metode reduksi berhasil membuat LiveDVD tersebut menjadi berukuran 889 MB, atau terdapat pengurangan sebesar 125MB (12.32%).

Pengujian ISO image dilakukan dua tahap, yaitu pengujian sebelum memakai metode reduksi, dan pengujian setelah memakai metode reduksi. Pengujian ISO image dilakukan dengan memakai Virtual Machine (VirtualBox versi 4.2) dan menggunakan USB Flash Disk (UFD). Pengujian dengan VirtualBox 4.2 memakai konfigurasi default yang sudah disediakan oleh VirtualBox. Hasil pengujian memakai VirtualBox dan UFD memberikan waktu yang hampir sama, hanya memiliki perbedaan 1 hingga 3 detik.

Waktu yang diperlukan untuk booting sistem operasi sebelum memakai metode reduksi adalah 72 detik. Sedangkan setelah memakai metode reduksi diperlukan waktu 45 detik. Hasil ini memperlihatkan bahwa proses booting menjadi 62.5% dari waktu booting ISO image tanpa reduksi. Dengan kata lain, waktu booting ISO image tereduksi menjadi lebih dari sepertiga kali lebih cepat dari booting ISO image tanpa reduksi.

5. KESIMPULAN

Metode reduksi telah menghasilkan paket rpm berukuran lebih kecil. Ukuran lebih kecil membuat sistem lebih responsif, karena waktu *booting* menjadi lebih cepat sekitar sepertiga kali dari waktu *booting* ISO image tanpa reduksi.

Ucapan Terima Kasih

Ucapan terima kasih penulis sampaikan kepada Pusat Penelitian Informatika Lembaga Ilmu Pengetahuan Indonesia atas dukungan dana penelitian melalui Daftar Isian Pelaksanaan Anggaran (DIPA) 2012.

Daftar Pustaka

- [1] E. Siever, S. Figgins, R. Love, and A. Robbins, *Linux in a Nutshell*, 6th ed. O'Reilly Media, 2009.
- [2] G. Moody, *Rebel Code: Linux and the Open Source Revolution*. Perseus Publishing, 2001.
- [3] C. Ruiz and W. N. Robinson, "Measuring open source quality: A literature reviews," *International Journal of Open Source Software and Processes (IJOSSP)*, vol. 3, no. 3, pp. 48–65, 2011.
- [4] A. Capiluppi, K.-J. Stol, and C. Boldyreff, "Software reuse in open source: A case study," *International Journal of Open Source Software and Processes (IJOSSP)*, vol. 3, no. 3, pp. 10–35, 2011.
- [5] A. Sofyan, *Membuat Distro Linux Sendiri*. Dian Rakyat, 2006.
- [6] L. Collin, "A quick benchmark: Gzip vs. bzip2 vs. lzma," Web site: <http://tukaani.org/lzma/benchmarks.html> [Last accessed: 2 October 2012].
- [7] J. Novy and B. Nottingham, "Features xz rpm payloads," Web site: <http://fedoraproject.org/wiki/Features/XZRpmPayloads> [Last accessed: 2 October 2012].
- [8] E. C. Bailey, *Maximum RPM*. Sams Publisher, 1997.
- [9] T. M., "Compression benchmarks," Web site: <http://www.squashfs-lzma.org/> [Last accessed: 2 October 2012].
- [10] E. McCallum, *Managing RPM-Based Systems with Kickstart and Yum*. O'Reilly Media, 2007.
- [11] A. Robbins, *Bash Pocket Reference*. O'Reilly Media, 2010.
- [12] J. Bambenek and A. Klus, *grep Pocket Reference*. O'Reilly Media, 2009.